# K55. SUCTION CUP PHONE PICKUP

It is often very convenient for other people in the room to listen to both sides of a phone conversation. The amplifier in this it has been designed for this use. A specially designed magnetic pickup with a suction cup attaches onto the earpiece of your phone at one end and into the amplifier at the other. The design is low cost and there is no direct electrical connection to the phone system.

The kit is constructed on single-sided printed circuit boards. Protel Autotrax & Schematic were used.

**ASSEMBLY INSTRUCTIONS**

Check off the components against the component listing. Make sure you identify every component. It is generally easiest if you solder the lowest height components first, the resistors then the capacitors & IC sockets. Make sure to get the electrolytic capacitors around the correct way. Note there is one link to make on the board next to the power supply pads. Use an offcut from a resistor to make the link.

**CIRCUIT DESCRIPTION**

The telephone pickup is really a magnetic field fluctuation detector. It picks up the oscillating magnetic field from the receiver of your telephone when someone is speaking to you. But it will also collect any other oscillating magnetic fields which happen to be floating
around in the air. For example, low & high frequency noise from your TV set or computer monitor or the characteristic mains humm from power lines. High frequency filters have been built into the circuit to reduce some of this unwanted noise.

The telephone pickup circuit consists of two high gain preamplifier stages in the LM358 followed by a power amplifier to drive the speaker. The IC's are low cost and easily available.

Both preamps in the LM358 are biased to half the supply voltage by R4 & R5. This allows maximum voltage swing at the outputs before hitting either supply rail. C6 bypasses any AC signal to ground, stabalizing the DC bias voltage. R3 is necessary to couple the DC bias voltage to IC1:B while also providing a high impedence to the input signal.

The RC feedback circuit on both preamps will, like any RC circuit, have a cutoff frequency. Or to think of it another way, the capacitance starts to act as a short circuit as the frequency increases and the gain decreases. The cutoff frequency is given by the formula:

$$f = 1 / (6.28 \times RC)$$

This gives a cutoff frequency of 2.7KHz which has the effect of limiting the amount of high frequency noise, or 'hiss'. This high frequency roll-off does not greatly affect voice frequency signals since voice frequency is nominally in the range 300Hz to 3.0KHz.

The output of the second preamp is fed to the power amplifier stage via C7 & P1. C7 removes any DC component from the amplified signal while P1 acts as a volume control. The LM386 is very easy to use and requires a minimum of external components. C8 provides filternig & bypassing for the internal bias network. C9 removes any DC component from the output signal. The gain of the LM386 may be set according to the combination of resistors & capacitors across pins 1 & 8. With no components the gain is 20. Finally use some wire to connect the speaker to the terminal block output.

Place the suction cup near the receiver on the handset. Keep the speaker away from the handset to stop any feedback. The pickup will be affected by strong magnetic fields - mains wiring, a computer monitor and TV set. Low roll-off filters (to reduce the 50Hz hum) could have been included in the circuit just as the high roll-off filters have been. However, since the pickup responds to low frequency this allows the unit to be used to trace mains wiring behind your walls or under the floor.

**WHAT TO DO IF IT DOES NOT WORK**

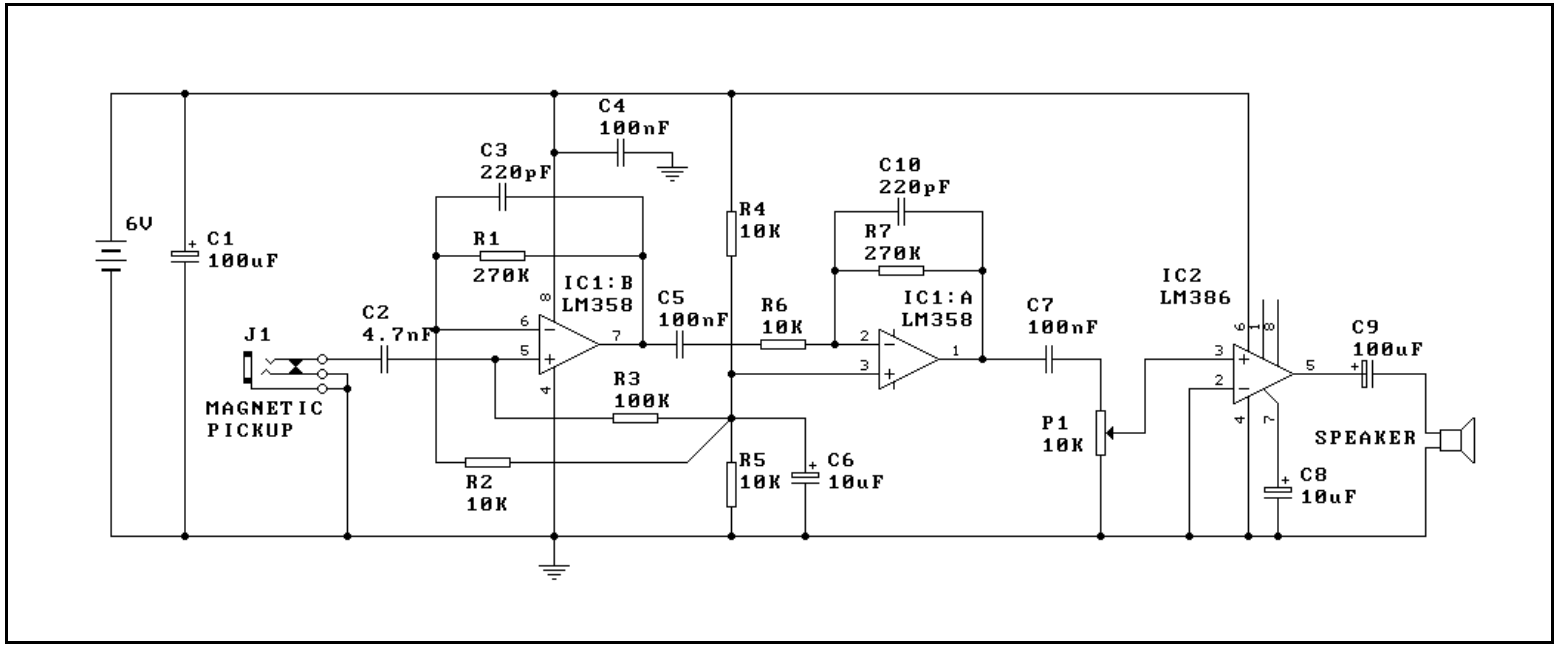
Poor soldering is the most likely reason. Check all solder joints carefully under a good light. Next check that all components are in their correct position on the PCB. Did you add the single link. Are the IC's in their correct places. Are the electrolytic capacitors around the correct way.

Visit our web site **http://kitsrus.com**

**COMPONENTS**

| | |
|---|---|
| Resistors 1/4w, 5%: | |
| 10K R2 R4 R5 R6 brown black orange | 4 |
| 270K R1 R7 red violet yellow | 2 |
| 100K R3 brown black yellow | 1 |
| Koa Trimpot 10K 103 | 1 |
| Monoblock capacitors: | |
| 100nF 104 C4 C5 C7 | 3 |
| Electrolytic capacitors: | |
| 10uF mini C6 C8 | 2 |
| 100uF mini C1 C9 | 2 |
| 220pF ceramic capacitors C3 C10 | 2 |
| 4n7 MPE box capacitor C2 | 1 |
| LM358 IC1 | 1 |
| LM386 IC2 | 1 |
| 8 pin IC socket | 2 |
| K55 PCB | 1 |
| Audio jack | 1 |
| 6V battery snap | 1 |
| 8 ohm speaker | 1 |
| Magnetic pickup with suction cap | 1 |
| 2 pole terminal block | 1 |

# K55. SUCTION CUP PHONE PICKUP



# KIT 56. DTMF DECODING & DISPLAY

This kit connects to the telephone line and records the number dialled (assuming tone dialling) and the duration of the call. The kit can also be used to record any DTMF tones detected via the built in microphone input. The kit will not detect numbers dialled using pulse dialling. The source code for the K1 micro-controller is provided on the floppy disk for study.

**Assembly**
Check the components supplied in the kit against the Component Listing. There are several points to note:

- make sure Q3 does not poke up too high.
- pins 15 & 16 on the LCD display are not used they are for backlighting (which this model of LCD does not have.) So the male 14-pin single in-line header goes in from pins 1 to 14.

Solder the lowest height components first; the resistors and capacitors. Solder the two crystals quickly as they are easily damaged by excess heat. The tuning fork crystal can be soldered down by the two pads provided. Put the 9V battery snap through the hole next to the pads to provide strain relief.

**Operation via telephone line**
Ensure that switches SW2 and SW3 are in the "TEL" position. Connect the decoder to the telephone line and switch on the power. When connected to the telephone line the decoder should be left switched on. The decoder monitors the line condition and turns itself off when the line is not looped (phone is hung up!). Current consumption is less than 50uA when off.

The decoder is activated whenever the phone is picked up ('the line is looped' in telephone jargon.) The LCD display is turned on and the last number dialed, together with the call duration, is displayed. When using the decoder for the first time (or after switching off), the number dialed is blank and the call duration is set to "00:00:00". Hanging up without dialing any digits will not affect the current information.

Dialing any digit will clear the last number dialed and re-start the call duration timer. The first digit dialed and any subsequent digits will be saved automatically. The decoder will save up to a maximum of 16 digits. Any further digits will be ignored.

In order to prolong battery life, the DTMF decoder chip is disabled sixty seconds after the phone is picked up. All dialing must be completed within sixty seconds of picking up the phone. Any digits dialed after sixty seconds are not detected and therefore ignored.

Call progress detection is not implemented on this unit. This means that the decoder cannot detect when the called party answers. Therefore the call duration time is the elapsed time from the first digit dialed to the phone being hung up.

**Operation via microphone input**
The microphone input can take a variety of different devices; a dynamic or ceramic microphone as well as a suction cup pickup coil. Be sure to insert the link next to the microphone jack if using a ceramic type microphone, otherwise leave open.

Disconnect the decoder from the telephone line, plug in a microphone or pickup coil and move switches SW2 and SW3 to the "MIC" position. The "MIC" position disables line monitoring and the decoder is unable to power down. Therefore the unit should be switched off when not in use.

Position the microphone close to the DTMF source. If using a pickup coil as input, attach the suction cup to the telephone handset next to the earpiece. Switch on the power. The last number detected will be blank and the call duration set to "00:00:00".

As each DTMF digit is detected it is displayed. If the decoder fails to respond to any DTMF tones, try positioning the microphone closer to the source or increasing the volume of the tones. Excessive volume may distort the signal presented to the DTMF decoder and cause the tones to be ignored. If using the pickup coil, try moving the coil to a different position relative to the earpiece (front, back or side). To reset the call duration time and clear the last number dialed, turn the power off then on again.

**Power Consumption**
The amount of current consumption varies depending on the mode of operation and the time elapsed since the decoder was activated.

TEL Mode: With the phone hung up, the decoder draws approximately 50uA from the battery. The current jumps to approximately 12mA for the first 60 seconds after picking up the handset. After 60 seconds the DTMF decoder chip and microphone circuitry are disabled and the current consumption drops to approximately 6mA. It will remain at 6mA until the unit is switched off or the phone is hung up.

MIC Mode: The decoder draws approximately 12mA from the battery while switched on.

---

# KIT 56. DTMF DECODING & DISPLAY

**Components - Kit 56**

| | | |
|---|---|---|
| 220K 1/2W resistors | R1 R2 | 2 |
| Resistors 1/4W, 5% | | |
| 1K | R11 | 1 |
| 10K | R7 12 13 14 15 18 22 | 7 |
| 22K | R24 | 1 |
| 56K | R3 | 1 |
| 68K | R4 R20 | 2 |
| 100K | R10 R16 R19 R21 | 4 |
| 180K | R9 | 1 |
| 220K | R5 | 1 |
| 270K | R6 | 1 |
| 390K | R17 R23 | 2 |
| 3M3 | R8 | 1 |
| 10K 103 Koa trimpot | VR1 | 1 |
| | | |
| 1N4148 | D2 | 1 |
| WO2 bridge rectifier | D1 | 1 |
| 2-pin header | LK1 | 1 |
| Jumper | | 1 |
| 3.579MHz crystal | Y1 | 1 |
| 4.096mhz crystal | Y2 | 1 |
| 10nF/63V box capacitor | C1 C2 | 2 |
| 10uF mini electrolytic | C8 C9 | 2 |
| 100nF 104 monoblok | C3 C4 C5 C10 C11 | 5 |
| 0.47uF 474 monoblok | C12 | 1 |
| 27pF ceramic | C6 C7 | 2 |
| BC557 transistor | Q1 Q3 | 2 |
| BC547 | Q2 | 1 |
| SPDT switch | SW1 SW2 | 2 |
| DPDT switch | SW3 | 1 |
| 2.5mm audio jack | J1 | 1 |
| Telephone jack | X1 | 1 |
| LCD 2x16 | | 1 |
| 14-pin male single-in-line header | | 1 |
| 14-pin female single-in-line header | | 1 |
| LM358 IC | U5 | 1 |
| MAX666 IC | U4 | 1 |
| Programmed 68HC705K1P | U2 | 1 |
| 8870 decoder IC | U1 | 1 |
| 8 pin IC socket | | 2 |
| 16 pin IC socket | | 1 |
| 18 pin IC socket | | 1 |
| 9V battery snap | | 1 |
| Plastic box & 4 screws | | 1 |
| K56 PCB | | 1 |

The source code for this Kit and a a general include file can be downloaded from the software download page of our website at

**http://kitsrus.com/soft.html**

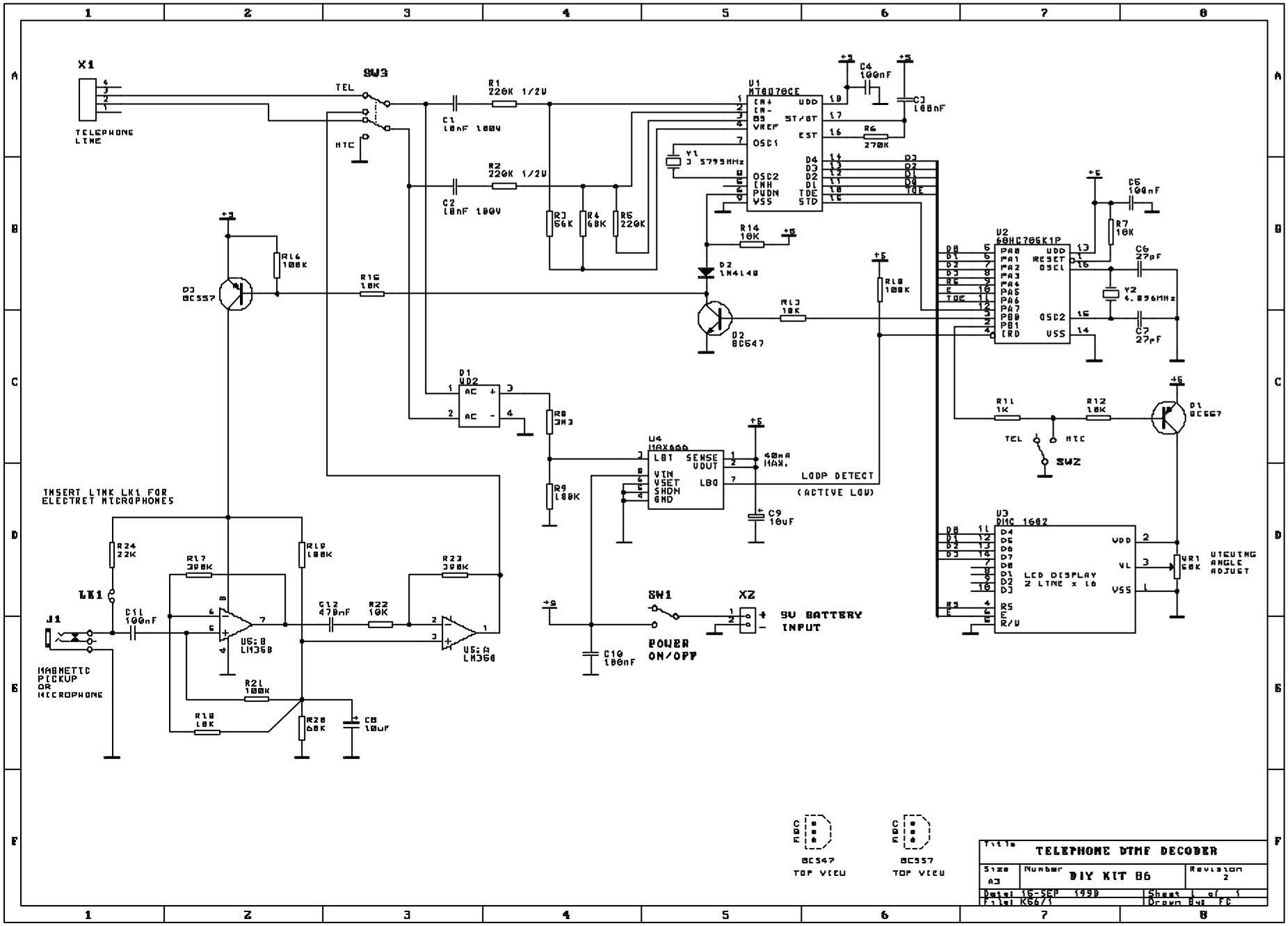
For our full range of kits see our website at

**http://kitsrus.com**

You may email the designer of this kit at

**frank@ozitronics.com**

# KIT 56. DTMF DECODING & DISPLAY



# KIT 56. DTMF DECODING & DISPLAY

# DIY Electronics

# Southern Cross II

# 8031 Single Board Computer

# Hardware Manual


## DIY Electronics (HK) Ltd

**PO Box 88458**
**Sham Shui Po**
**Hong Kong**

**Fax: (852) 2725-0610**
**Voice: (852) 2720-0255**
**Email: sales@kitsrus.com**
**http://kitsrus.com**



**November, 1998.**

# Introduction

The Southern Cross II single board computer is based around the 8032/52 8-bit microcontroller. Adding a power supply and connecting to any PC capable of running an 8051 cross-assembler enables the user to implement an 8031 software development system. Programming in BASIC is also possible using one of the 8052-BASIC chips currently available. If your board is fitted with one of these BASIC chips then switching from one software environment to the other is simply a matter of inserting or removing a link.

The Southern Cross II, when used in conjunction with the MVS-31 debugging monitor, implements a complete 8031-based assembly language software development system. The extra resources provided by an 8032/52 over an 8031,(extra timer and an additional 128 bytes of internal RAM memory), together with an independent external UART allows the user complete access to all the 8031's internal ports and timers. Software debugging tools such as breakpoints, watchpoints and single stepping are provided by the MVS-31 monitor program. Refer to the "MVS-31 User's manual" for further details.

# Getting Started

**Running in 8031 mode**

Insert the MVS-31 monitor EPROM into socket IC4 and ensure that jumper JP2 is set to the correct EPROM type. An 8032/52 microcontroller chip must be used in position IC1.

1. Insert a link on jumper JP1.
2. Move switch SW2 to the OFF position.
3. Connect a 9 - 12 volt DC plug pack via power connector J7.
4. Connect the supplied serial cable to connector **J5 ("HOST" port)**. Connect a terminal or PC (running terminal emulation software) to the D25 connector on the cable. Configure the terminal or PC serial port to 8 data bits, 1 stop bit and no parity. The baud rate should be set to 1200, 2400, 4800, 9600 or 19200 bps.
5. Turn on the power via switch SW3. LED D2 should light.
6. Press the Carriage Return ( Enter↵ ) key to trigger the MVS-31 automatic baud rate detection. A sign on banner will appear followed by a " > " prompt. Commands may now be entered.

**Running in BASIC mode**

This mode requires an 8052-BASIC microcontroller chip in position IC1. A 27256 EPROM is also necessary in position IC4 if BASIC programs are to be saved into EPROM. Ensure that jumper JP2 is set to the correct EPROM type.

1. Remove the link from jumper JP1.
2. Move switch SW2 to the OFF position.
3. Connect a 9 - 12 volt DC plug pack via power connector J7.
4. Connect the supplied serial cable to connector **J6 ("USER" port)**. Connect a terminal or PC (running terminal emulation software) to the D25 connector on the cable. Configure the terminal or PC serial port to 8 data bits, 1 stop bit and no parity. The baud rate should be set to 1200, 2400, 4800, 9600 or 19200 bps.
5. Turn on the power via switch SW3. LED D2 should light.
6. Press the space bar to trigger the BASIC-52 automatic baud rate detection. A sign on message will appear followed by a "READY" prompt. Commands may now be entered.

Please read the section on "BASIC-52 Programming" before attempting to save any programs into EPROM.

## EPROM types

The Southern Cross II can use three different EPROM types, 8K (2764), 16K (27128) and 32K (27256). The EPROM type is selected by setting jumper JP2 to the correct position as marked on the circuit board.

Note that in order to save BASIC programs in EPROM, a 27256 EPROM must be used.

## Southern Cross II Memory Map

*This manual does not aim to provide a detailed explanation of the architecture of the 8051 family of microcontrollers. However a short description is necessary so that the memory decoding scheme can be explained and understood.*

The 8051 family of microcontrollers implement the Harvard memory architecture. This means that these devices have **separate** address spaces for Program and Data memory, each up to 64K in size. Internal circuitry in the microcontroller accesses the correct memory based on the instruction being executed.

As well as having separate Program and Data spaces, these microcontrollers also have internal and external Program and Data memory. The 8051/52 series of microcontrollers can have up to 8K of internal program memory. Thus any program address in the range 0000h to 1FFFh is directed to internal Program memory. Any other program address is directed externally. It is possible to direct all program access externally by disabling internal program memory. This is done by connecting the /EA pin to ground. The 8031/32 series have no internal program memory so all program access must be directed externally. The microcontrollers running BASIC are 8052 series chips with the BASIC interpreter stored in internal program memory.

Program memory can only be read, not written to, The /PSEN signal is used as a read strobe for any **external** program memory. Data memory occupies a separate address space from Program memory. The /RD and /WR signals are used to access **external** Data memory as needed. Only Data memory can be written to and programs can only run from Program memory!

It is possible to combine all or part of the Program and Data memory into one address space. This is done by logically ORing the /PSEN and /RD signals with the decoded memory range required and using the resultant output as a read strobe. Combining Program and Data memory is necessary on the Southern Cross II so that users can download and run programs.

The Southern Cross II provides 16K of RAM memory that is accessible as both Program and Data memory. This is the area of memory where user programs are downloaded into and executed from. The other 16K of RAM is accessible as Data memory only.

The EPROM can be an 8K, 16K or 32K type. The EPROM address space is split, with 16K located at 0000h and the other 16K located at 8000h. The 16K block at 8000h is used to store BASIC programs when the Southern Cross II is used as a BASIC computer! Refer to the "BASIC-52 Programming" reference manuals for more information.

All external I/O devices such as the SPI and PPI chips are located in external data memory. Remember that only data memory can be written to!

Figure 1 gives an overall picture of the Southern Cross II memory map.

Memory and I/O decoding is provided via two programmable logic devices (GALs). Using GALs for address mapping adds great flexibility to the circuit design as well as significantly reducing chip count. Other memory and I/O address mapping schemes are possible by re-programming the GALs.
Contact DIY Electronics for further information.

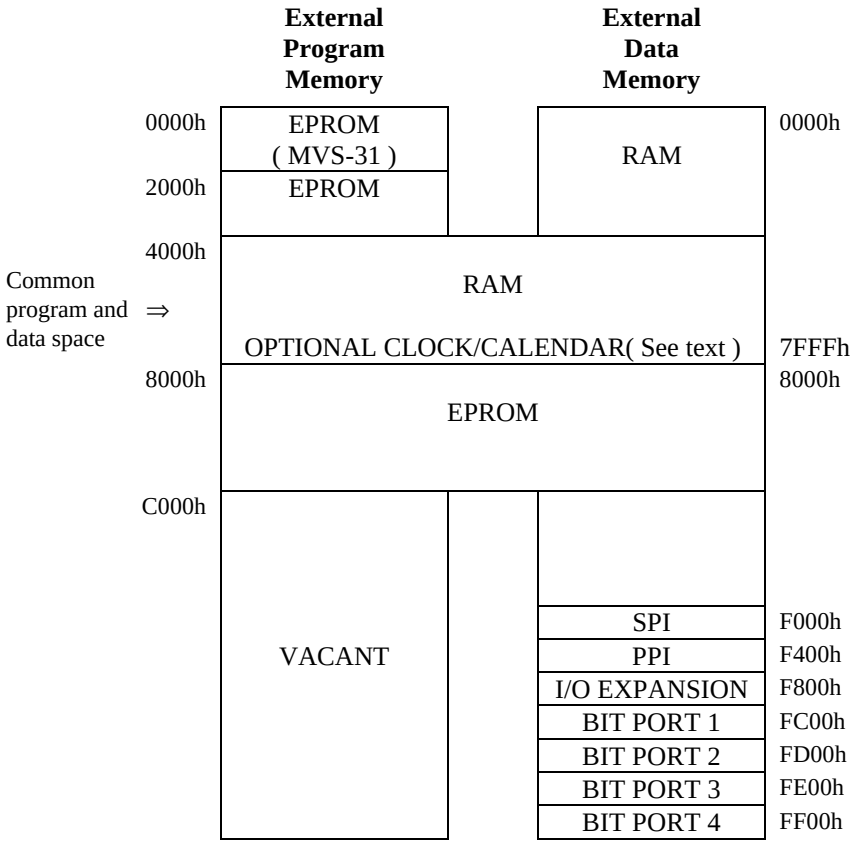



Fig 1

## Serial Ports

These are implemented as simple "3-wire" interfaces, using just TxD, RxD and GND. The modem control lines "CTS, DSR and DCD" are held at the RS-232 "ON" level and do not provide any hardware handshaking. No software handshaking protocol is used either.

The Southern Cross II has two serial ports.

1. Uses the internal serial port of the 8032/52 chip and is referred to as the "USER" port. Connection is via the 10-pin header labelled J6.
2. Uses an external SC2691 UART chip from Philips Components and is referred to as the "HOST" port. Connection is via the 10-pin header labelled J5

The RS-232 signals are brought out to standard D9 connectors by using a 10-way flat ribbon cable. One end is terminated with a 10-pin IDC plug and the other end with a crimp-style female D9 connector. By connecting pin 1 on the 10-pin plug with pin 1 on the D9 connectors, the signals will be in the correct positions to allow a "straight-through" connection to a D9-type PC COM port.

## RS-232 Interface - J5, J6

| Signal | J5, J6 pin | D9 pin |
|---|---|---|
| TxD | 3 | 2 |
| RxD | 5 | 3 |
| CTS | 6 | 8 |
| DSR | 2 | 6 |
| GND | 9 | 5 |
| DCD | 1 | 1 |

# Parallel I/O Ports

A number of parallel I/O ports are provided. Some are "static" in operation, meaning that any data written to the port is latched. The other ports are "dynamic" in nature. These ports do not latch data written to them. Output data is only valid while the port is being addressed. These ports are referred to as "bit ports".

## 8031 I/O ports - J1

The 8051 family of microcontrollers have four inbuilt 8-bit parallel ports, labelled P0-3. Two of these ports, P0 and P2, are used as address/data busses when external program and data memory access is required. This is the mode in which the Southern Cross II operates. Port bits P3.6 and P3.7 are used as /RD and /WR signals and therefore are not available for use as I/O lines. This leaves 13 I/O lines available for use, all of port 1 and bits 0-5 of port 3. Note that port bits P3.0 - P3.5 all have secondary functions eg. P3.0 and P3.1 are the Rx and Tx lines for the 8031 serial port.

These 13 I/O lines are brought out to a 16-pin header labelled J1.

| J1 pin | Signal | J1 pin | Signal |
|---|---|---|---|
| 1 | P1.0 | 2 | P1.1 |
| 3 | P1.2 | 4 | P1.3 |
| 5 | P1.4 | 6 | P1.5 |
| 7 | P1.6 | 8 | P1.7 |
| 9 | P3.0 / RxD | 10 | P3.1 / TxD |
| 11 | P3.2 / INT0 | 12 | P3.3 / INT1 |
| 13 | P3.4 / T0 | 14 | P3.5 / T1 |
| 15 | GND | 16 | GND |

Port bits P1.3, P1.4 ans P1.5 also have a secondary function. They are connected via switch SW2 to the memory decoder GAL, IC9, and used by the 8052-BASIC chips as EPROM programming control pins. These pins are only needed by BASIC-52 when programs are to be saved into EPROM. Refer to the section on "BASIC-52 Programming" for further details.

## 8255 I/O Ports - J2

Three 8-bit parallel ports are implemented using the 8255 Programmable Peripheral Interface (PPI) chip (IC6). This chip is mapped in external data memory at address F400h. These 24 I/O lines are brought out to a 26-pin header labelled J2.

| J2 pin | Signal | J2 pin | Signal |
|---|---|---|---|
| 1 | PA0 | 2 | PA1 |
| 3 | PA2 | 4 | PA3 |
| 5 | PA4 | 6 | PA5 |
| 7 | PA6 | 8 | PA7 |
| 9 | PB0 | 10 | PB1 |
| 11 | PB2 | 12 | PB3 |
| 13 | PB4 | 14 | PB5 |
| 15 | PB6 | 16 | PB7 |
| 17 | PC0 | 18 | PC1 |
| 19 | PC2 | 20 | PC3 |
| 21 | PC4 | 22 | PC5 |
| 23 | PC6 | 24 | PC7 |
| 25 | GND | 26 | GND |

## Bit Ports - J4

As mentioned previously, these are "dynamic" not static ports. With referrence to sheet 2 of the schematic diagram, the 8 I/O lines are actually driven by the data bus (BAD0 - 7). The ports are formed by decoding a 256 byte memory block and using the resulting output as an active low strobe, similar to the chip enable signals for the EPROM and RAM chips. These 4 bit ports are mapped in external data memory at the following address locations:

| | |
|---|---|
| Bit port 1 | FC00h |
| Bit port 2 | FD00h |
| Bit port 3 | FE00h |
| Bit port 4 | FF00h |

Although each port is mapped into a 256 byte block, it is only one address "wide". Therefore bit port 1 can be accessed at address FC00h, FC01h, FC02h ... etc right through to FCFFh. These ports are both input and output ports, depending on the instruction used to access them.

These ports are pin for pin compatible to the CN1 port on the Z80 based "Southern Cross" computer, also available from DIY Electronics. It is provided on the Southern Cross II to take advantage of a number of add-on boards already available for the "Southern Cross" computer.

The bit ports are brought out to a 16-pin header labelled J4.

| J4 pin | Signal | J4 pin | Signal |
|---|---|---|---|
| 1 | +5V | 2 | /RESET |
| 3 | /BIT PORT 1 | 4 | /BIT PORT 2 |
| 5 | /BIT PORT 3 | 6 | /BIT PORT 4 |
| 7 | GND | 8 | GND |
| 9 | BAD4 | 10 | BAD3 |
| 11 | BAD5 | 12 | BAD2 |
| 13 | BAD6 | 14 | BAD1 |
| 15 | BAD7 | 16 | BAD0 |

## I/O Expansion Interface - J3

A maximum of 256 bytes of I/O expansion is available via connector J3. This interface allows for further devices to be connected to the microcontroller. The interface is mapped in external data memory beginning at address location F800h.

| J3 pin | Signal | J3 pin | Signal |
|---|---|---|---|
| 1 | A0 | 2 | A1 |
| 3 | A2 | 4 | A3 |
| 5 | A4 | 6 | A5 |
| 7 | A6 | 8 | A7 |
| 9 | AD0 | 10 | AD1 |
| 11 | AD2 | 12 | AD3 |
| 13 | AD4 | 14 | AD5 |
| 15 | AD6 | 16 | AD7 |
| 17 | /RD | 18 | /WR |
| 19 | RESET | 20 | ALE |
| 21 | /IOEXP | 22 | |
| 23 | +5V | 24 | +5V |
| 25 | GND | 26 | GND |

Signals preceeded by " / " are active low, all other signals are active high.

The signal " /IOEXP " is the block decode signal for the expansion interface and should be used in conjunction with address lines A0-A7 to provide further address decoding.

Address lines A0-A7 have already been de-mulitplexed on the Southern Cross II board ie. they are actually address lines. This allows devices with separate address and data lines to be connected to the expansion interface directly. For those devices with a multiplexed address/data bus use signals AD0-AD7 and ALE.

## Clock/Calendar

An optional clock/calendar device is available for the Southern Cross II, a Dallas Semiconductor DS1216C Intelligent Socket. Access to this clock/calendar is provided in the MVS-31 debugging monitor.

The DS1216C contains an internal lithium battery. This keeps the clock going during power down and also makes the CMOS RAM non-volatile ie. no loss of data when power to the Southern Cross II is removed.

Battery life is over 10 years. No circuit modifications are necessary to use the DS1216C, just plug it in between the RAM chip and the IC socket (or solder directly to the PCB if purchased with the Southern Cross II kit).

## BASIC-52 Programming

As mentioned, the Southern Cross II board is capable of being programmed in BASIC. A number of suppliers, such as Intel and Micromint Inc., have available an 8052 microcontroller with a BASIC interpreter programmed into the chip's internal EPROM. The Southern Cross II needs one of these chips installed as IC1 in order run the BASIC-52 language.

BASIC-52 has the ability to store its programs into an external memory device such as EPROM. Programming EPROMs requires a voltage level of between 12.5 and 25 volts ( depending on the EPROM ). The Southern Cross II generates this programming voltage using a DC-DC converter chip, IC12.

The Southern Cross II is designed to use a 27256 EPROM for storing BASIC programs. These EPROMs require a programming voltage of 12.5 volts. Before attempting to save any programs in EPROM, the programming voltage must be set to this level. This is done by measuring the voltage on test pin TP1 and adjusting trimpot VR1 for 12.5 volts. This procedure is only required once. However it is worth checking this voltage periodically or whenever any problems occur with saving programs to EPROM.

The Southern Cross II only uses 27256 EPROMs for saving BASIC programs. Remove the MVS-31 monitor EPROM and install a 27256 type in its place. The MVS-31 monitor can reside on the same 27256 EPROM used to save BASIC programs. The monitor would reside in the bottom 8K starting at address 0000h whereas BASIC programs are saved at address locations 8000h upwards, which is in the upper 16K of the EPROM. The memory decoder GAL, IC9, splits the EPROM into two halves, with one half starting at 0000h and the other half starting at 8000h. Remember to set jumper JP2 to the correct EPROM type. A number of BASIC-52 programming and startup options are also available. Refer to the BASIC-52 reference manual for further details.

BASIC-52 uses port bits P1.3, P1,4 and P1.5 as EPROM programming control lines. The function of these lines is:

| | |
|---|---|
| P1.3 | ALE Disable |
| P1.4 | Program pulse width |
| P1.5 | Enable program voltage |

These signals are not connected directly to the EPROM or associated programming circuitry. Instead they are connected via switch SW2 and the memory decoder GAL, IC9. When the GAL detects a low on its pin 5, it uses these port bits to control the output signals " /ROM_CE, SYS_ALE and /VPP_CONT ". Moving SW2 to the " ON " position connects the port bits to the GAL and pulls pin 5 low, ready for EPROM programming.

These port bits can also be used as general purpose I/O lines and are brought out on connector J1. These pins MUST be isolated from any external voltage levels. This would be the case if any of these pins were inputs

being "driven" from a device connected to J1. Therefore, it is advisable before saving programs to EPROM to disconnect any device from J1.

To summarize, the following steps must be carried out before issuing a BASIC-52 programming command.

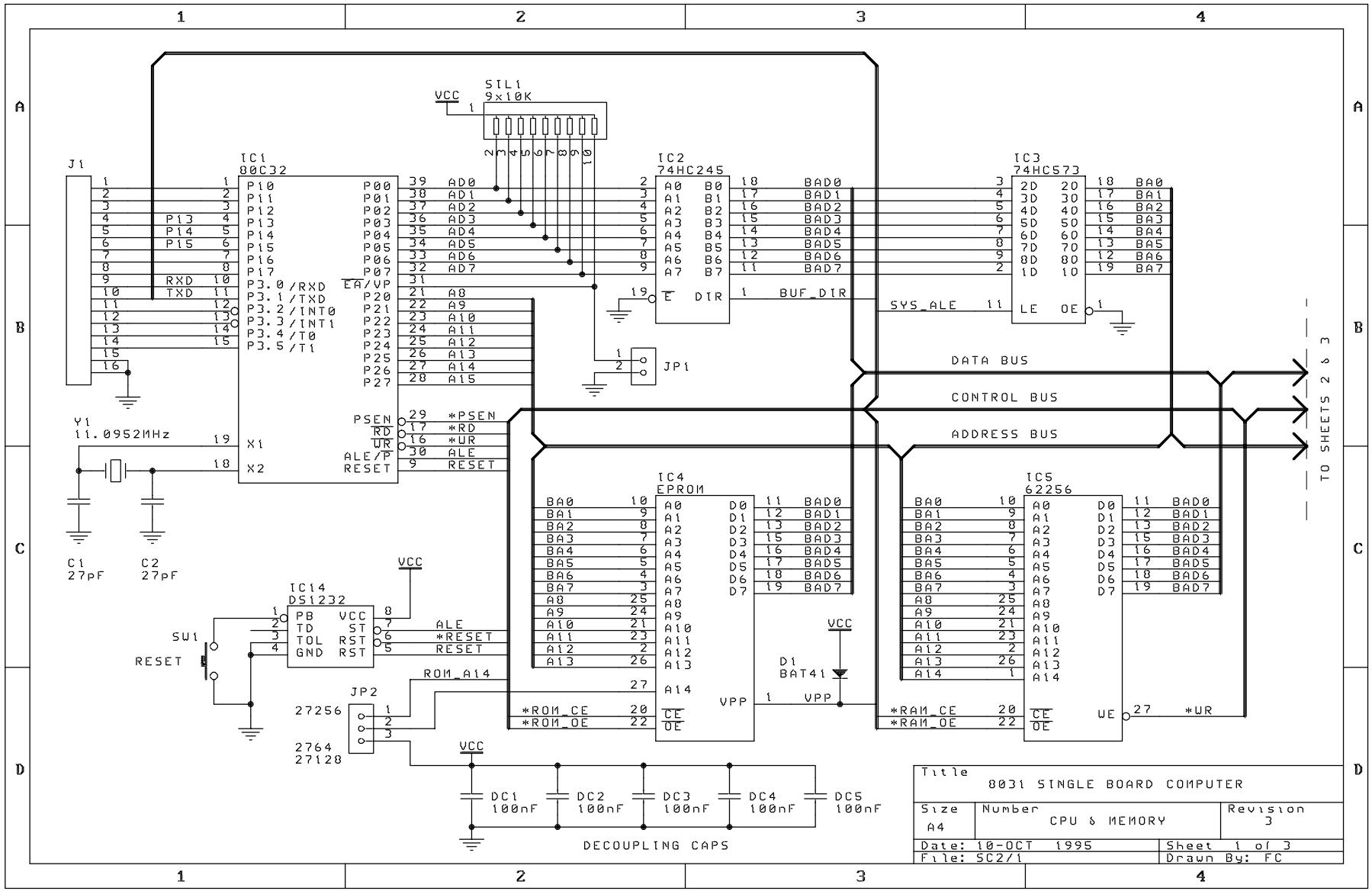
1. Disconnect any device attached to connector J1.
2. Move switch SW2 to the " ON " position.

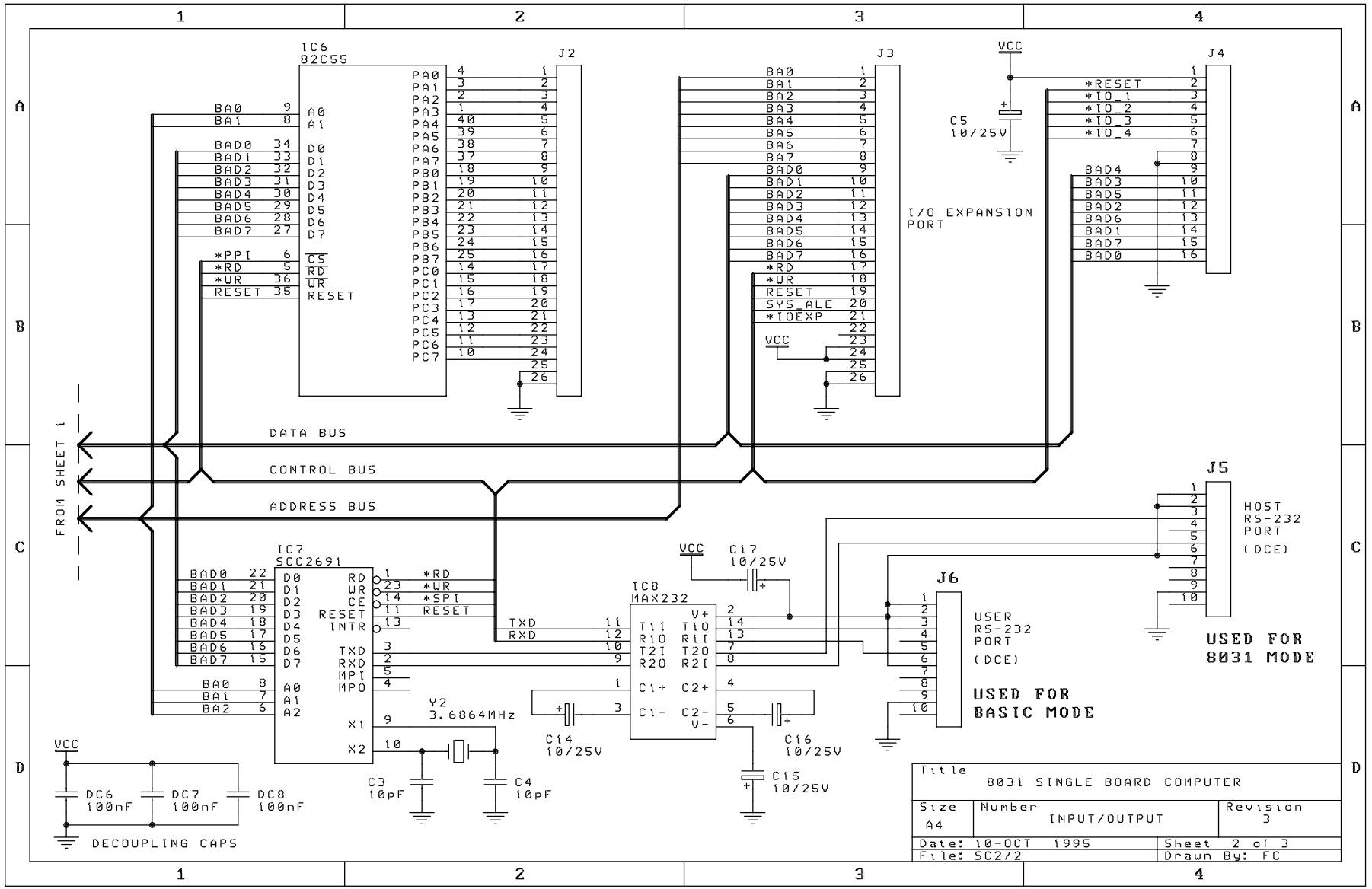
Whilst the MVS-31 monitor program and BASIC programs can be stored on the same 27256 EPROM it is advisable to use two separate EPROMs. If you wish to keep them together then make a copy of the MVS-31 EPROM and use the copy instead. The MVS-31 binary file is on the disk supplied with the Southern Cross II computer.

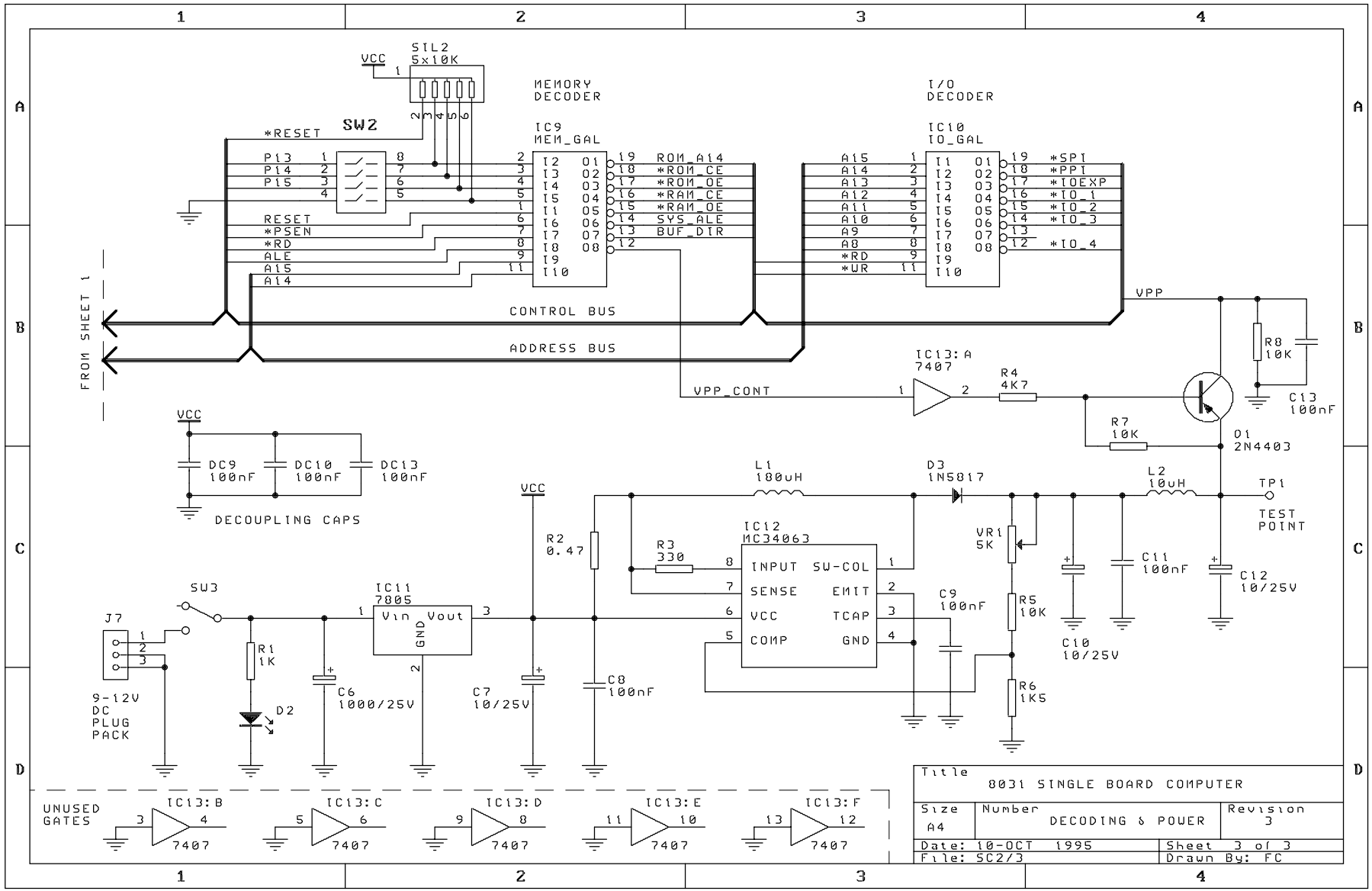
**References**

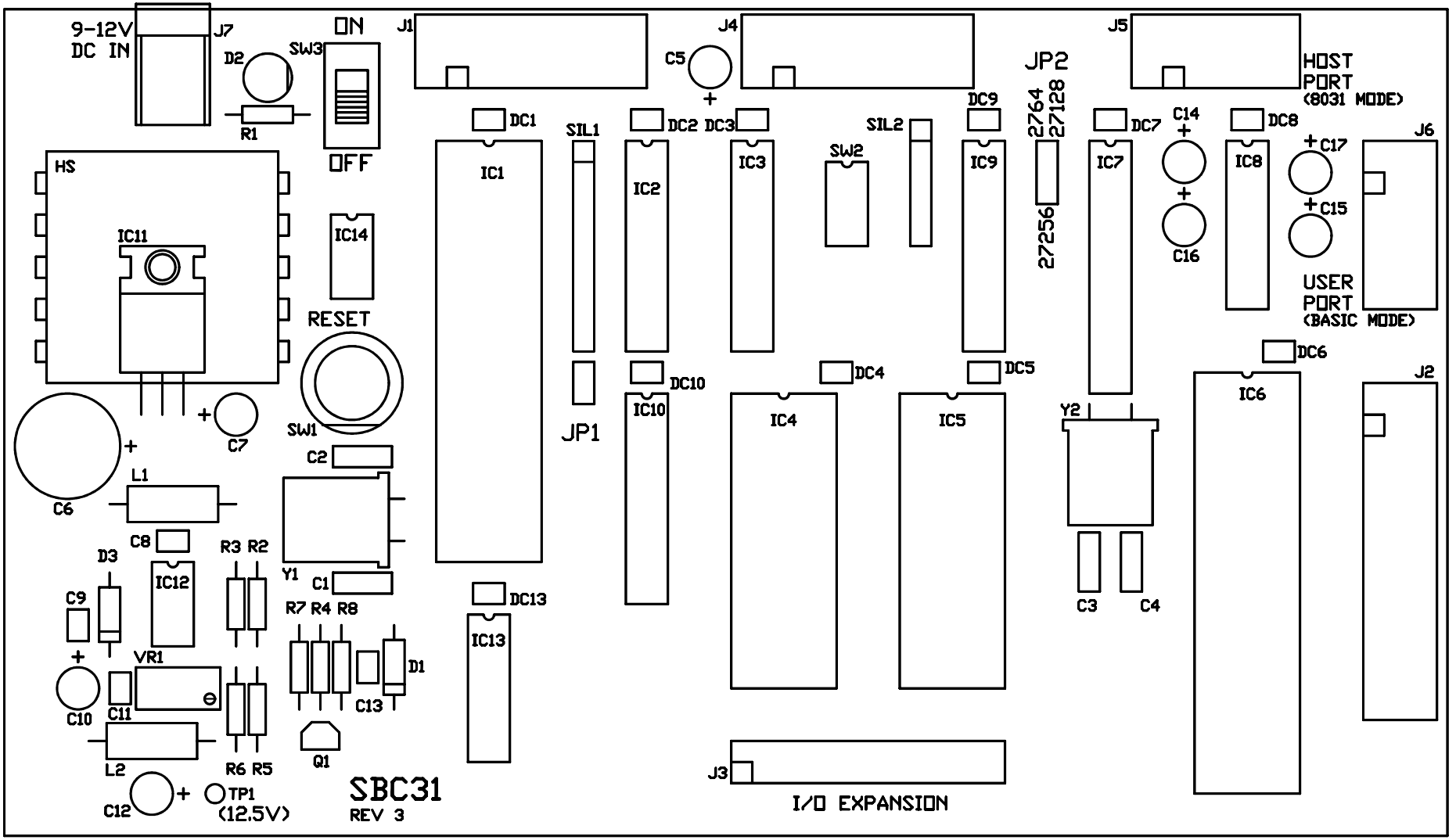
The hardware & software for the Southern Cross 2 were developed by Frank Crivelli for DIY Electronics. Frank can be contacted at **ozitronics@c031.aone.net.au**

The following web sites contain 80xx code & material
**http://www.lvr.com**
**http://livewire.ee.latrobe.edu/~sjm**
**http://cera2.com/gatox805.htm**
**http://www.atmel.com**


SIL1
9x10K
VCC
J1
IC1
80C32
P10
P11
P12
P13
P14
P15
P16
P17
P3.0/RXD
P3.1/TXD
P3.2/INT0
P3.3/INT1
P3.4/T0
P3.5/T1
P00
P01
P02
P03
P04
P05
P06
P07
EA/VP
P20
P21
P22
P23
P24
P25
P26
P27
PSEN
RD
WR
ALE/P
RESET
AD0
AD1
AD2
AD3
AD4
AD5
AD6
AD7
A8
A9
A10
A11
A12
A13
A14
A15
*PSEN
*RD
*WR
ALE
RESET
RXD
TXD
IC2
74HC245
A0
A1
A2
A3
A4
A5
A6
A7
B0
B1
B2
B3
B4
B5
B6
B7
E
DIR
BUF_DIR
BAD0
BAD1
BAD2
BAD3
BAD4
BAD5
BAD6
BAD7
JP1
IC3
74HC573
1D
2D
3D
4D
5D
6D
7D
8D
1Q
2Q
3Q
4Q
5Q
6Q
7Q
8Q
LE
OE
SYS_ALE
BA0
BA1
BA2
BA3
BA4
BA5
BA6
BA7
DATA BUS
CONTROL BUS
ADDRESS BUS
TO SHEETS 2 & 3
Y1
11.0952MHz
X1
X2
C1
27pF
C2
27pF
IC4
EPROM
VPP
*ROM_CE
*ROM_OE
ROM_A14
IC5
62256
*RAM_CE
*RAM_OE
WE
*WR
D1
BAT41
IC14
DS1232
PB
TD
TOL
GND
VCC
ST
RST
SU1
RESET
JP2
27256
2764
27128
DC1
100nF
DC2
100nF
DC3
100nF
DC4
100nF
DC5
100nF
DECOUPLING CAPS
Title
8031 SINGLE BOARD COMPUTER
Size
A4
Number
CPU & MEMORY
Revision
3
Date: 10-OCT 1995
Sheet 1 of 3
File: SC2/1
Drawn By: FC


IC6
82C55
J2
J3
J4
VCC
C5
10/25V
I/O EXPANSION
PORT
*RESET
*IO_1
*IO_2
*IO_3
*IO_4
BA0
BA1
BA2
BA3
BA4
BA5
BA6
BA7
BAD0
BAD1
BAD2
BAD3
BAD4
BAD5
BAD6
BAD7
*PPI
*RD
*WR
RESET
SYS_ALE
*IOEXP
PA0
PA1
PA2
PA3
PA4
PA5
PA6
PA7
PB0
PB1
PB2
PB3
PB4
PB5
PB6
PB7
PC0
PC1
PC2
PC3
PC4
PC5
PC6
PC7
A0
A1
D0
D1
D2
D3
D4
D5
D6
D7
CS
RD
WR
RESET
FROM SHEET 1
DATA BUS
CONTROL BUS
ADDRESS BUS
IC7
SCC2691
IC8
MAX232
J5
J6
HOST
RS-232
PORT
(DCE)
USER
RS-232
PORT
(DCE)
USED FOR
8031 MODE
USED FOR
BASIC MODE
D0
D1
D2
D3
D4
D5
D6
D7
RD
WR
CE
RESET
INTR
TXD
RXD
MPI
MPO
X1
X2
*SPI
T1I
R1O
T2I
R2O
T1O
R1I
T2O
R2I
V+
C1+
C1-
C2+
C2-
V-
Y2
3.6864MHz
C3
10pF
C4
10pF
C14
10/25V
C15
10/25V
C16
10/25V
C17
10/25V
DC6
100nF
DC7
100nF
DC8
100nF
DECOUPLING CAPS
Title
8031 SINGLE BOARD COMPUTER
Size
A4
Number
INPUT/OUTPUT
Revision
3
Date: 10-OCT 1995
Sheet 2 of 3
File: SC2/2
Drawn By: FC


SIL2
5x10K
VCC
SW2
*RESET
RESET
*PSEN
*RD
ALE
A15
A14
P13
P14
P15
MEMORY
DECODER
IC9
MEM_GAL
ROM_A14
*ROM_CE
*ROM_OE
*RAM_CE
*RAM_OE
SYS_ALE
BUF_DIR
I/O
DECODER
IC10
IO_GAL
*SPI
*PPI
*IOEXP
*IO_1
*IO_2
*IO_3
*IO_4
*WR
FROM SHEET 1
CONTROL BUS
ADDRESS BUS
VPP
VPP_CONT
IC13:A
7407
R4
4K7
R8
10K
C13
100nF
R7
10K
Q1
2N4403
DC9
100nF
DC10
100nF
DC13
100nF
DECOUPLING CAPS
L1
180uH
D3
1N5817
L2
10uH
TP1
TEST
POINT
R2
0.47
R3
330
IC12
MC34063
INPUT
SENSE
VCC
COMP
SW-COL
EMIT
TCAP
GND
VR1
5K
C9
100nF
R5
10K
R6
1K5
C10
10/25V
C11
100nF
C12
10/25V
SW3
J7
9-12V
DC
PLUG
PACK
R1
1K
D2
IC11
7805
Vin
Vout
GND
C6
1000/25V
C7
10/25V
C8
100nF
UNUSED
GATES
IC13:B
IC13:C
IC13:D
IC13:E
IC13:F
7407
Title
8031 SINGLE BOARD COMPUTER
Size
A4
Number
DECODING & POWER
Revision
3
Date: 10-OCT 1995
Sheet 3 of 3
File: SC2/3
Drawn By: FC


9-12V
DC IN
J7
D2
SW3
ON
OFF
R1
J1
J4
C5
J5
HOST
PORT
(8031 MODE)
JP2
2764
27128
27256
DC1
DC2
DC3
DC9
DC7
DC8
C14
C17
C15
C16
J6
USER
PORT
(BASIC MODE)
HS
IC11
IC14
IC1
SIL1
IC2
IC3
SW2
SIL2
IC9
IC7
IC8
RESET
SW1
JP1
DC10
DC4
DC5
DC6
IC10
IC4
IC5
IC6
J2
C6
C7
C2
L1
C8
D3
R3 R2
IC12
Y1
C1
Y2
C3
C4
C9
R7 R4 R8
DC13
IC13
VR1
D1
C10
C11
C13
L2
R6 R5
Q1
J3
I/O EXPANSION
C12
TP1
(12.5V)
SBC31
REV 3

# Southern Cross II - REGISTRATION

## Question: Why Register?

**Answer:** So that you can be notified when new hardware and software becomes available. The Southern Cross II will be supported in the future with new hardware, software and documentation.

Please fill in the form below and return to:

DIY ELECTRONICS (HK) LTD
PO BOX 88458
SHAM SHUI PO
HONG KONG

or fax to (852) 2725-0610.

Your comments about the Southern Cross II are most welcome.

| | | | | | | |
|---|---|---|---|---|---|---|
| **Name** | | | | | | |
| **Address** | | | | | | |
| | | | | | | |
| **Zip** | | **Country** | | | | |
| | | | | | | |
| **MVS-31** | | **Disk size for** | 360K | 1.2M | 720K | 1.44M |
| **Version & Date** | | **software updates?** | | | | |
| **Comments:** | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |
| | | | | | | |

# MULTI-SWITCH DOORBELL WITH INDICATORS


**T.K. HAREENDRAN**

Here's the circuit of a multi-switch input musical doorbell (shown in Fig.1). The circuit is built around the popular and less expensive quad D-latch CD4042B (IC1). When switch S6 is pushed to 'on' condition, the circuit gets +9V and the four data inputs (D1 through D4) of ICI are in low state because these are tied to ground via resistors R1 through R4. Polarity input (POL) pin 6 of IC1 is also pulled down by resistor R5. Clock input (pin 5) of the quad D-latch is wired in normally low mode and hence all the four outputs (Q0 through Q3) have the same states as their corresponding data inputs. As a result, LED1 through LED4 are in off condition.

There are four switches fitted at four different doors/gates outside the home and a monitoring panel (as shown in Fig. 2) in the common room of the home. If any switch is pressed by a visitor (for example, switch S1 at door 1), pins 2 and 4 of IC1 go high.

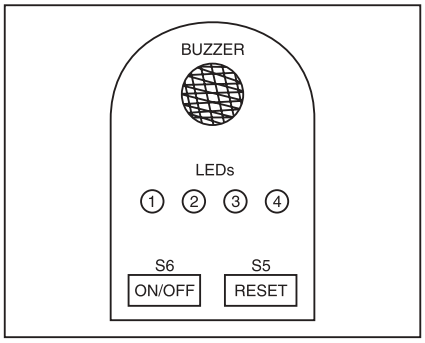



*Fig. 2: Suggested panel layout of musical doorbell*

Simultaneously, pin 3 to IC1 (Q0 output) goes low and LED1 starts glowing to indicate that switch S1 is pressed by someone.

Next, output pin 13 of the dual 4-input NOR gate (IC2, here wired as a single 4-input OR gate) goes high to forward bias buzzer-driver transistor T1 via resistor R10.

The final result is a soft and pleasing musical bell, which lasts until reset switch S5 is pressed by the owner. For this latching arrangement , output pin 13 of IC2 from the NOR gate is fed back to the clock input of IC1.

The circuit costs around Rs 100.

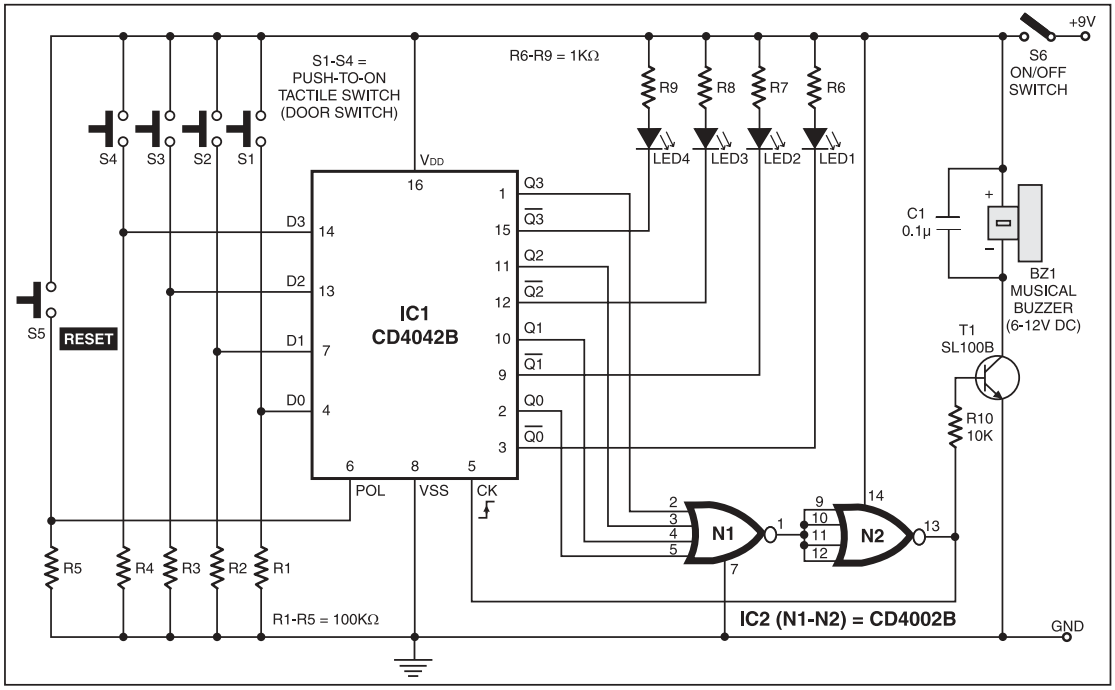



*Fig. 1: Multi-switch doorbell with indicators*

# SONG NUMBER DISPLAY


**PRABHASH K.P.**

Here's a circuit to display the song number in an audio system for quick reference to songs. It also serves the purpose of an extra visual indicator in modern audio systems.

When the power is switched on, the power-on-reset circuit comprising 3.3k resistor R20 and 1µF, 25V capacitor C6 resets the counters, showing '00' in the display. One can also reset the display to zero at any time by pressing reset switch S1.

When the first song starts playing, the output pins of IC1 (KA2281) go low and capacitor C5 starts charging. This forward biases transistor T1 and hence the input to IC3 at pin 1 goes to high state. As a result, the output of the counter goes to the next state, showing 01 on the display. The counter remains in this state until the song is completed.

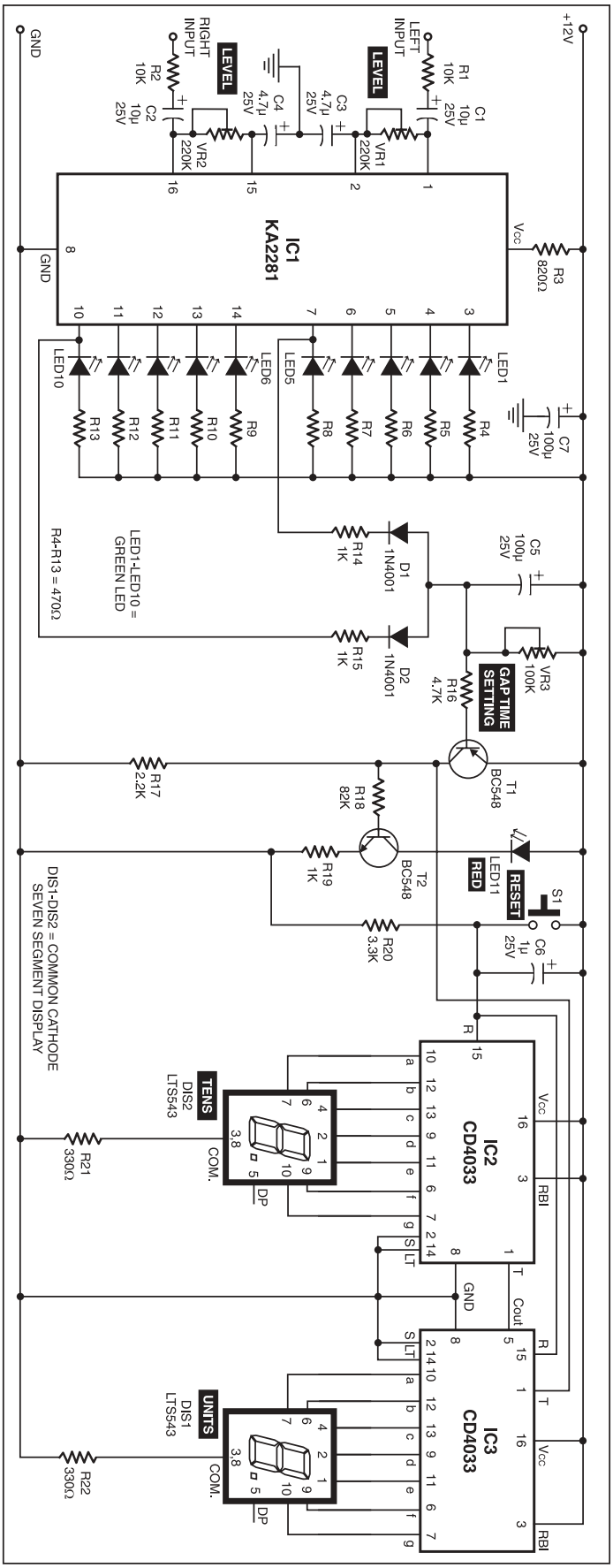
During the time gap before the next song starts playing, capacitor C5 discharges. After discharging of capacitor C5, the input to IC3 becomes low again. When the song starts, the process described above is repeated and the display shows 02. You can adjust VR3 to change the time gap setting. This must be set such that the circuit doesn't respond to short gaps, if any, within a song and responds only to long gaps between different songs.

Transistor T2 helps in gap-delay adjustment. The intensity of LED11 diminishes when a song is completed and the counter is ready to accept the next pulse.

Connect the input to the preamp output or equaliser output of the audio system. Adjust VR1 and VR2 to get the correct audio-level indication. If you are already using KA2281 for audio-level indication, just connect diodes D1 and D2 as shown in this circuit.

Note that the counter counts the songs by detecting the gaps. Therefore any long gap within a song may cause false triggering and the display will also be incremented. However, as this is very unlikely to happen, the circuit shows the correct song number almost all the time.

The circuit costs around Rs 100.



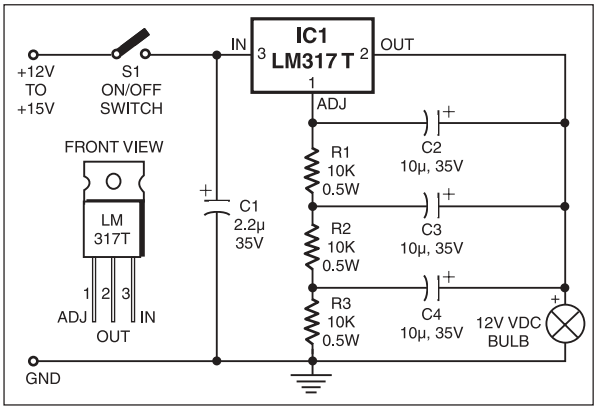
# FLASHING BEACON


**ASHOK K. DOCTOR**

A flashing beacon has many uses. It can be employed as a distress signal on highways or as a direction pointer for parking lots, hospitals, hotels, etc. Here we present a flashing beacon that uses well-known regulator IC LM317T. As LM317T regulator can deliver more than 1 amp. A small 12V, 10W bulb with a high-quality reflector can serve as a good visible blinker.

A 12-15V, 1A DC supply is connected to the input pin of the IC. A 12V, 10W bulb and a combination of resistors and capacitors are connected between the output pin and ADJ pin of the IC as shown in the figure. The IC is provided with an aluminium heat-sink to dissipate the heat generated while delivering full current. Since the IC has an inbuilt switch-on current limiter, it extends the bulb life.

For the shown values of resistors and capacitors, the bulb flashes at approximately 4 cycles per second. The number of flashes depends on the charge-discharge time of the capacitors. Different values of resistors and capacitors can be used to increase or decrease the number of flashes.




This circuit costs around Rs 50.

# INTRUDER RADIO ALERT SYSTEM


**DAVID NASH PIOUS**

Consider a situation where a burglar has entered your house and snapped the telephone wires, leaving you with no means of communication with the outside world. In such an emergency, you will find this intruder alarm to be very handy. It transmits a prerecorded emergency message repeatedly for reception by an FM receiver.

The message containing address, geographical location, name, etc is recorded onto a chip. The prerecorded message can then be transmitted repeatedly with the help of an FM transmitter, in the hope that some noble soul will hear it and inform the police about the incident.

The circuit comprises a sound recording-and-playback chip (UM5506BH). This chip consists of a 96kbit SRAM and can record up to six seconds of audio. (For details, refer 'Mini Voice Processor' circuit published in April 2000 issue of EFY.) After the required message has been recorded, it is passed to a low-power, VHF FM transmitter wired around BC547 and 2N2369 transistors. The range of this transmitter is 60 to 100 metres using a 40-70cm long wire as an antenna.

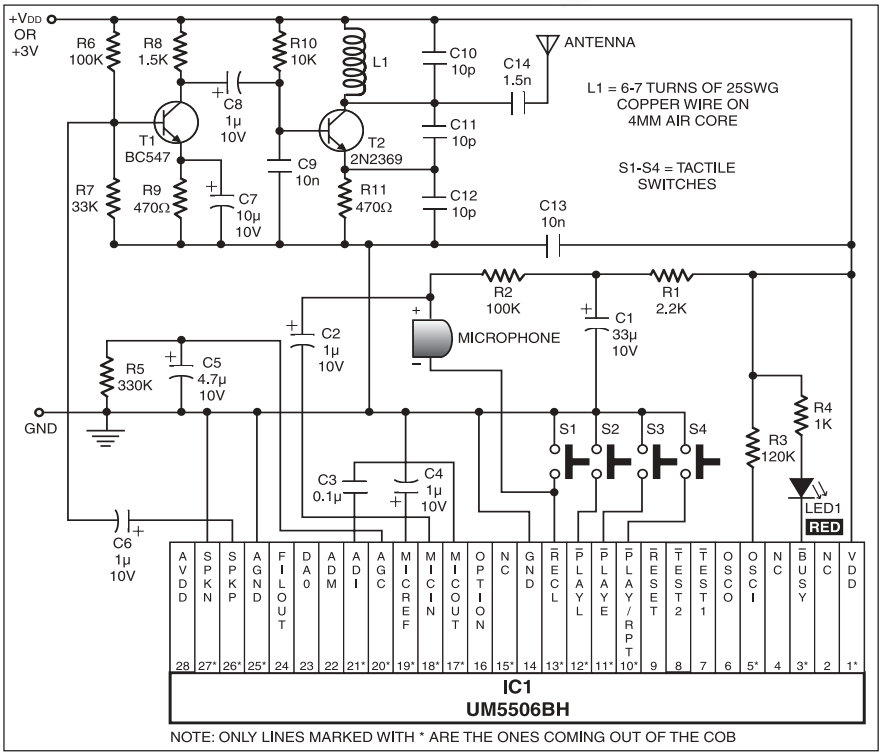



*Fig. 2: Circuit diagram of intruder radio alert system*

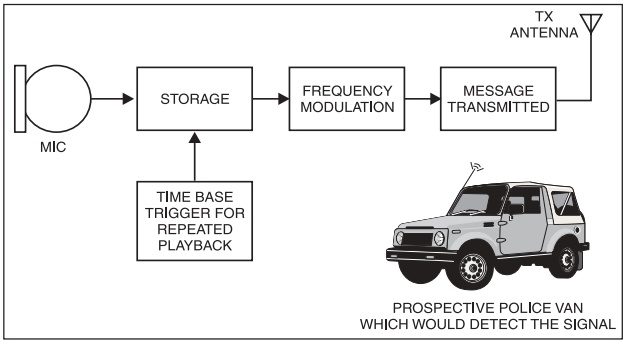



*Fig. 1: Block diagram of the intruder radio alert system*

The major advantage of this circuit is its low power consumption. The author operated it on 3V button cells (Maxell CR 2032, CR 2025, etc used in digital diaries). To transmit the prerecorded message, the play button is pressed repeatedly. The transmitted message can be heard over the FM receiver.

A possible modification, though it has legal complications, is to vary the coil inductance such that the transmission is on police band, thus alerting the police for quick help. Even the need of repeatedly pressing play button can be obviated by configuring an astable multiviberator (using IC 555 timer) to trigger IC UM5506BH every six seconds so that the message is played repeatedly.

This circuit costs around Rs 200.

# AUTOMATED TRAFFIC SIGNAL CONTROLLER


**VIKRAM BANERJEE**
**MRINAL KANTI MANDAL**
**DR ANIRUDHA GHOSAL**

This automated traffic signal controller can be made by suitably programming a GAL device. (For GAL programming you may refer to the construction project published on page 52 in EFY's September issue.) Its main features are:

1. The controller assumes equal traffic density on all the roads.

2. In most automated traffic signals the free left-turn condition is provided throughout the entire signal period, which poses difficulties to the pedestrians in crossing the road, especially when the traffic density is high. This controller allows the pedestrians to safely cross the road during certain periods.

3. The controller uses digital logic, which can be easily implemented by using logic gates.

4. The controller is a generalised one and can be used for different roads with slight modification.

5. The control can also be exercised manually when desired.

The time period for which green, yellow, and red traffic signals remain 'on' (and then repeat) for the straight moving traffic is divided into eight units of 8 seconds (or multiples thereof) each. Fig. 1 shows the flow of traffic in all permissible directions during the eight time units of 8 seconds each. For the left- and right-turning traffic and pedestrians crossing from north to south, south to north, east to west, and west to east, only green and red signals are used.

Table I shows the simultaneous states of the signals for all the traffic. Each row represents the status of a signal for 8 seconds. As can be observed from the table, the ratio of green, yellow, and red signals is 16:8:40 ( = 2:1:5) for the straight moving traffic. For the turning traffic the ratio of green and red signals is 8:56 ( = 1:7), while for pedestrians crossing the road the ratio of green and red signals is 16:48 ( = 2:6).

In Table II (as well as Table I) X, Y, and Z are used as binary variables to

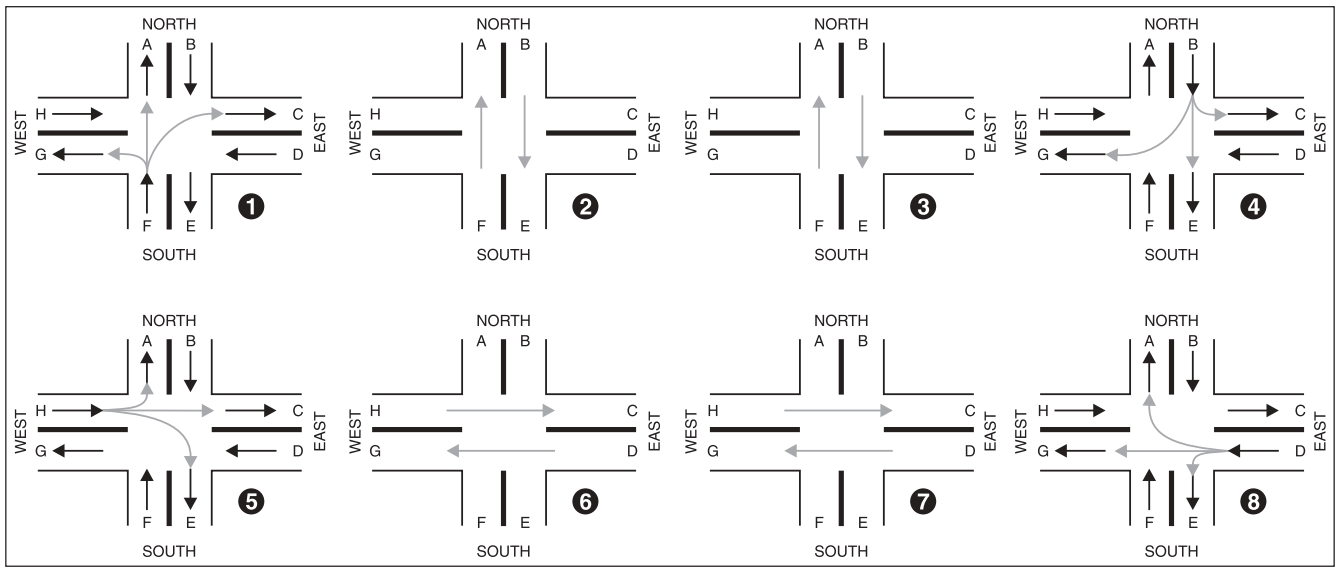



*Fig. 1: Flow of traffic in all possible directions*

**TABLE I**
**Simultaneous States of Signals for All the Traffic**

| X | Y | Z | B-C/B-G Lt/Rt | B-E St | D-E/D-A Lt/Rt | D-G St | F-G/F-C Lt/Rt | F-A St | H-A/H-E Lt/Rt | HC St | WALK (N-S)/(S-N) | WALK (E-W)/(W-E) |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | R | R | R | R | G | G | R | R | R | R |
| 0 | 0 | 1 | R | G | R | R | R | G | R | R | G | R |
| 0 | 1 | 0 | R | G | R | R | R | Y | R | R | G | R |
| 0 | 1 | 1 | G | Y | R | R | R | R | R | R | R | R |
| 1 | 0 | 0 | R | R | R | R | R | R | G | G | R | R |
| 1 | 0 | 1 | R | R | R | G | R | R | R | G | R | G |
| 1 | 1 | 0 | R | R | R | G | R | R | R | Y | R | G |
| 1 | 1 | 1 | R | R | G | Y | R | R | R | R | R | R |

**TABLE II**
**Boolean Functions for All the Signal Conditions**

| Signal | Reference | Boolean functions |
|---|---|---|
| Green | B-C(Lt)/B-G (Rt) | X'YZ |
| Green | B-E (St) | XYZ' + X'Y'Z |
| Red | B-E (St) | X + Y'Y'Z' |
| Yellow | B-E (St) | X'YZ |
| Green | D-E (Lt)/D-A (Rt) | XYZ |
| Green | D-G (St) | XYZ' + XY'Z |
| Red | D-G (St) | X' + XY'Z' |
| Yellow | D-G (St) | XYZ |
| Green | F-G(Lt)/F-C (Rt) | X'Y'Z' |
| Green | F-A (St) | X'Y' |
| Red | F-A (St) | X + X'YZ |
| Yellow | F-A (St) | X'YZ' |
| Green | H-A (Lt)/H-E (Rt) | XY'Z' |
| Green | H-C (St) | XY' |
| Red | H-C (St) | X' + XYZ |
| Yellow | H-C (St) | XYZ' |
| Green | Walk (N-S/S-N) | X'YZ' + X'Y'Z |
| Green | Walk (E-W/W-E) | XYZ' + XY'Z |

***Note**. X', Y', and Z' denote complements of variables X, Y, and Z, respectively.*

depict the eight states of 8 seconds each. Letters A through H indicate the left and right halves of the roads in four directions as shown in Fig. 1. Two letters with a dash in between indicate the direction of permissible movement from a road. Straight direction is indicated by St, while left and right turns are indicated by Lt and Rt, respectively.

The Boolean functions for all the signal conditions are shown in Table II. The left- and the right-turn signals for the traffic have the same state, i.e. both are red or green for the same duration, so their Boolean functions are identical and they should be connected to the same control output.

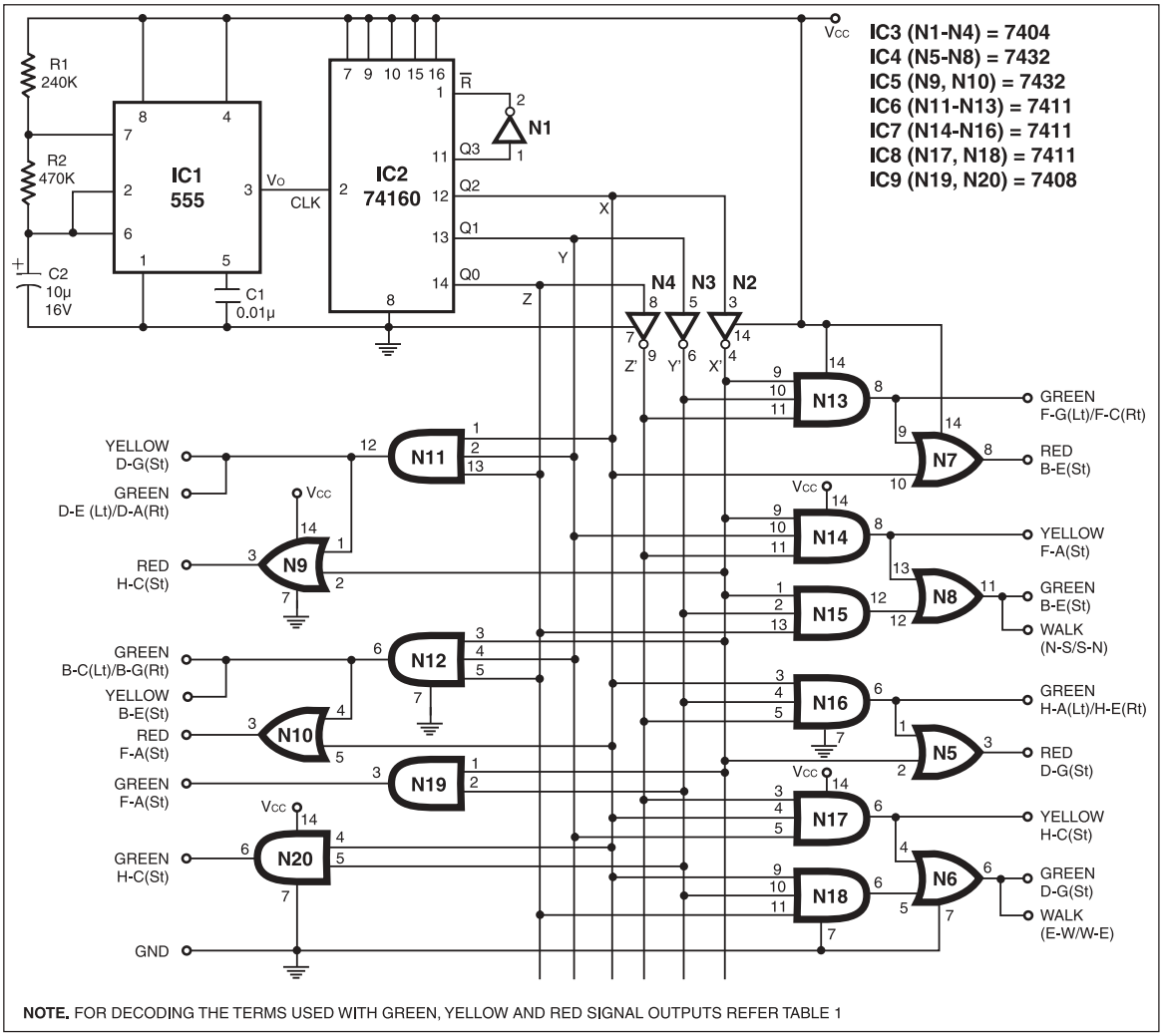
The circuit diagram for realising these Boolean functions is shown in Fig. 2. Timer 555 (IC1) is wired as an astable multivibrator to generate clock signal for the 4-bit counter 74160 (IC2). The time duration of IC1 can be adjusted by varying the value of resistor R1, resistor R2, or capacitor C2 of the clock circuit. The 'on' time duration T is given by the following relationship:

T = 0.695C2(R1 + R2)

IC2 is wired as a 3-bit binary counter by connecting its Q3 output to reset pin 1 via inverter N1. Binary outputs Q2, Q1, and Q0 form variables X, Y, and Z, respectively. These outputs, along with their complimentary outputs X', Y', and Z', respectively, are used as inputs to the rest of the logic circuit to realise various outputs satisfying Table I.

You can simulate various traffic lights using green, yellow, and red LEDs and feed the outputs of the circuit to respective LEDs via current-limiting resistors of 470 ohms each to check the working of the circuit. Here, for turning traffic and pedestrians crossing the road, only green signal is made available. It means that for the remaining period these signals have to be treated as 'red'.

In practice, the outputs of Fig. 2 should be connected to solidstate relays to operate high-power bulbs. Further, if a particular signal condition (such as turning signal) is not applicable to a given road, the output of that signal condition should be




**NOTE.** FOR DECODING THE TERMS USED WITH GREEN, YELLOW AND RED SIGNAL OUTPUTS REFER TABLE 1

*Fig. 2: The circuit diagram for traffic light signalling*

**Table III**
**Execution Results of Software Program**

| SIG-B | SIG-D | SIF-F | SIG-H | WALK(N-S) | WALK(E-W) |
|---|---|---|---|---|---|
| G G R Y | G G R Y | G G R Y | G G R Y | G R | G R |
| 0 0 1 0 | 0 1 0 0 | 1 1 0 0 | 0 0 1 0 | 0 1 | 0 1 |
| 0 1 0 0 | 0 1 0 0 | 0 1 0 0 | 0 0 1 0 | 1 0 | 0 1 |
| 0 1 0 0 | 0 1 0 0 | 0 0 0 1 | 0 0 1 0 | 1 0 | 0 1 |
| 1 0 0 1 | 0 1 0 0 | 0 0 1 0 | 0 0 1 0 | 0 1 | 0 1 |
| 0 0 1 0 | 0 1 0 0 | 0 0 1 0 | 1 1 0 0 | 0 1 | 0 1 |
| 0 0 1 0 | 1 0 0 0 | 0 0 1 0 | 0 1 0 0 | 0 1 | 1 0 |
| 0 0 1 0 | 1 0 0 0 | 0 0 1 0 | 0 0 0 1 | 0 1 | 1 0 |
| 0 0 1 0 | 0 0 1 1 | 0 0 1 0 | 0 0 1 0 | 0 1 | 0 1 |

***Note.*** *The first column under G (green) in each group of four signals indicates the turn signal, while the next three columns under GRY indicate signal for the straight traffic.*

connected to green signal of the next state (refer Table I).

The traffic signals can also be controlled manually, if desired. Any signal state can be established by entering the binary value corresponding to that particular state into the parallel input pins of the 3-bit counter. Similarly, the signal can be reset at any time by providing logic 0 at the reset pin (pin 1) of the counter using an external switch.

A software program to verify the functioning of the circuit using a PC is given below. (Source code and executable file will be provided in the next month's EFY-CD.) When executing the program, keep pressing Enter key to get the next row of results. The test results on execution of the program is shown in Table III.

This circuit costs around Rs 125.

## TRAFFIC.C

```
#include < stdio.h >
#include < conio.h >
#define TRUE 1
#define False 0

int not(int x);
int or2(int x,int y);
int or3(int x,int y,int z);
int and2(int x,int y);
int and3(int x,int y,int z);
int main(void)
{
int a,b,c;
int seq,green_bl,green_bs,red_bs,yellow_bs;
int green_dl,green_ds,red_ds,yellow_ds;
int green_fl,green_fs,red_fs,yellow_fs;
int green_hl,green_hs,red_hs,yellow_hs;
int walk_ns,stop_ns;
int walk_ew,stop_ew;

clrscr();
printf(" SIG-B SIG-D SIF-F SIG-H
WALK(N-S) WALK(E-W)\n");
printf("G G R Y G G R Y G G R Y G G R Y
G R G R\n");

for(seq = 0;seq < 8;seq + + )
{
c = (seq&1);b = (seq&2) > > 1;a = (seq&4) > > 2;
green_bl = and3(not(a),b,c);
green_bs = or2(and3(not(a),b,not(c)),and3(not(a),not(b),c));
red_bs = or2(a,and3(not(a),not(b),not(c)));
yellow_bs = and3(not(a),b,c);
green_dl = and3(a,b,c);
green_ds = or2(and3(a,b,not(c)),and3(a,not(b),c));
red_ds = or2(not(a),and3(a,not(b),not(c)));
yellow_ds = and3(a,b,c);
green_fl = and3(not(a),not(b),not(c));
green_fs = and2(not(a),not(b));
red_fs = or2(a,and3(not(a),b,c));
yellow_fs = and3(not(a),b,not(c));
green_hl = and3(a,not(b),not(c));
green_hs = and2(a,not(b));
red_hs = or2(not(a),and3(a,b,c));
yellow_hs = and3(a,b,not(c));
walk_ns = green_bs;
stop_ns = or3(and3(not(a),not(b),not(c)),and3(not(a),b,c),a);
walk_ew = green_ds;
stop_ew = or3(not(a),and3(a,b,c),and3(a,not(b),not(c)));
printf("%d %d %d %d %d %d %d %d
%d %d %d %d %d %d %d %d %d %d
%d %d\n",
    green_bl,green_bs,red_bs,yellow_bs,
    green_dl,green_ds,red_ds,yellow_ds,
    green_fl,green_fs,red_fs,yellow_fs,
    green_hl,green_hs,red_hs,yellow_hs,
    walk_ns,stop_ns,
    walk_ew,stop_ew);
    getch();
    }
    return;
    }
    int and2(int x,int y)
    {
    return(x && y);
    }
    int and3(int x,int y,int z)
    {
    return(x && y && z);
    }
    int or2(int x,int y)
    {
    return(x || y);
    }
    int or3(int x,int y,int z)
    {
    return(x || y || z);
    }
    int not(int x)
    {
    return(!x);
    }
```

# INFRARED TOY CAR MOTOR CONTROLLER

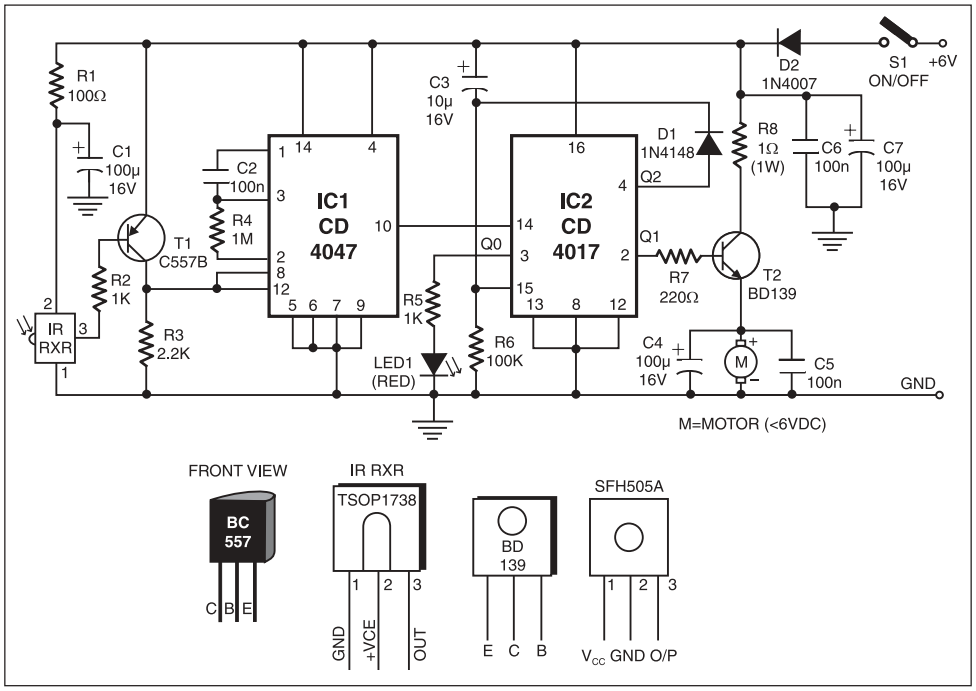

T.K. HAREENDRAN

This add-on circuit enables remote switching on/off of battery-operated toy cars with the help of a TV/video remote control handset operating at 30–40 kHz.

When the circuit is energised from a 6V battery, the decade counter CD4017 (IC2), which is configured as a toggle flip-flop, is immediately reset by the power-on-reset combination of capacitor C3 and resistor R6. LED1 connected to pin 3 (Q0) of IC2 via resistor R5 glows to indicate the standby condition. In standby condition, data output pin of the integrated infrared receiver/demodulator (SFH505A or TSOP1738) is at a high level (about 5 volts) and transistor T1 is 'off' (reverse biased). The monostable wired around IC1 is inactive in this condition.

When any key on the remote control handset is depressed, the output of the IR receiver momentarily transits through low state and transistor T1 conducts. As a result, the monostable is triggered and a short pulse is applied to the clock input (pin 14) of IC2, which takes Q1 output (pin 2) of IC2 high to switch on motor driver transistor T2 via base bias resistor R7 and the motor starts rotating continously (car starts running). Resistor R8 limits the starting current.

When any key on the handset is depressed again, the monostable is retriggered to reset decade counter IC2 and the motor is switched off. Standby LED1 glows again.




This circuit can be easily fabricated on a general-purpose printed board. After construction, enclose it inside the toy car and connect the supply wires to the battery of the toy car with right polarity. Rewire the DC motor connections and fix the IR receiver module in a suitable location, for example, behind the front glass, and connect its wires to the circuit board using a short 3-core ribbon cable/shielded wire.

***Note.*** Since the circuit uses modulated infrared beam for control function, ambient light reflections will not affect the circuit operation. However, fluorescent tubelights with electronic ballasts and CFL lamps may cause malfunctioning of the circuit.

# ANTI-THEFT SECURITY FOR CAR AUDIOS


T.K. HAREENDRAN

This small circuit, based on popular CMOS NAND chip CD4093, can be effectively used for protecting your expensive car audio system against theft.

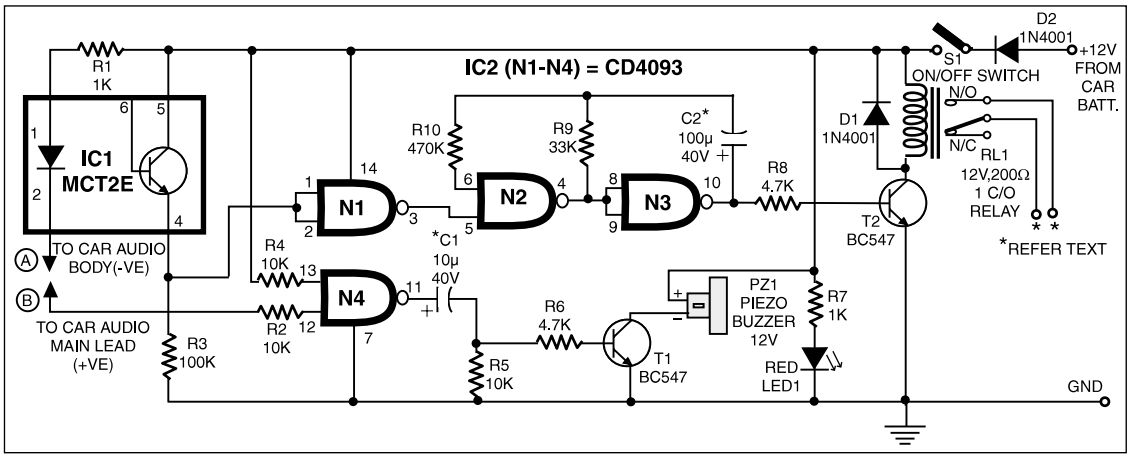
When 12V DC from the car battery is applied to the gadget (as indicated by LED1) through switch S1, the circuit goes into standby mode. LED inside optocoupler IC1 is lit as its cathode terminal is grounded via the car audio (amplifier) body. As a result, the output at pin 3 of gate N1 goes low and disables the rest of the circuit.

Whenever an attempt is made to remove the car audio from its mounting by cutting its connecting wires, the optocoupler immediately turns off, as its LED cathode terminal is hanging. As a result, the oscillator circuit built around gates N2 and N3 is enabled and it controls the 'on'/'off' timings of the relay via transistor T2. (Relay contacts can be used to energise an emergency beeper, indicator, car horns, etc, as desired.)

Different values of capacitor C2 give different 'on'/'off' timings for relay RL1 to be 'on'/'off'. With 100µF we get approximately 5 seconds as 'on' and 5 seconds as 'off' time.

Gate N4, with its associated components, forms a self-testing circuit. Normally, both of its inputs are in 'high' state. However, when one switches off the ignition key, the supply to the car audio is also disconnected. Thus the output of gate N4 jumps to a 'high' state and it provides a differentiated short pulse to forward bias transistor T1 for a short duration. (The combination of capacitor C1 and resistor R5 acts as the differentiating circuit.)

As a result, buzzer in the collector terminal of T1 beeps for a short duration to announce that the security circuit is intact. This 'on' period of buzzer can be varied by changing the values of capacitor C1 and/or resistor R5.



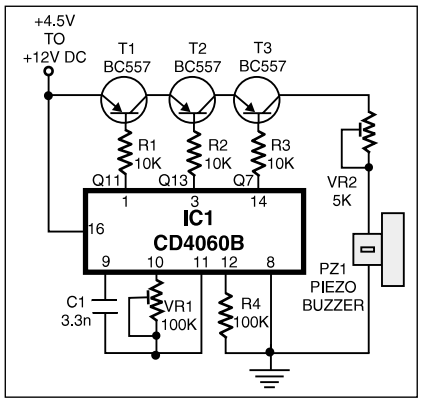

After construction, fix the LED and buzzer in dashboard as per your requirement and hide switch S1 in a suitable location. Then connect lead A to the body of car stereo (not to the body of vehicle) and lead B to its positive lead terminal. Take power supply for the circuit from the car battery directly.

***Caution.*** This design is meant for car audios with negative ground only. ❑

# SIMPLE TELEPHONE RING TONE GENERATOR

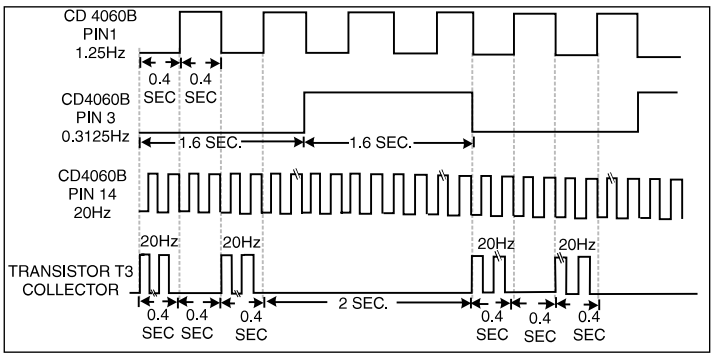

K. UDHAYA KUMARAN, VU3GTH

Here is a simple telephone ring tone generator circuit designed using only a few components. It produces simulated telephone ring tone and needs only DC voltage (4.5V DC to 12V DC). One may use this circuit in ordinary intercom or phone-type intercom.




The sound is quite loud when this circuit is operated on +12V DC power supply. However, the volume of ring sound is adjustable.

The commonly available 14-stage binary ripple counter with built-in oscillator (CMOS IC CD4060B) is used to generate three types of pulses, which are available from pin 1 ($O_{11}$), pin 3 ($O_{13}$), and pin 14 ($O_7$), respectively. Preset VR1 is adjusted to obtain 0.3125Hz pulses (1.6-second 'low' followed by 1.6-second 'high') at pin 3 of IC1. At the same time, pulses available from pin 1 will be of 1.25 Hz (0.4-second 'low', 0.4-second 'high') and 20 Hz at pin 14. The three output pins of IC1 are connected to base terminals of transistors T1, T2, and T3 through resistors R1, R2, and R3, respectively.




Transistors T1 through T3 are cascaded in such a way that the positive voltage available at the emitter of transistor T1 is extended to the collector of transistor T3 when the outputs of all the three stages are low. As a result, transistors T1 through T3 are forward biased for 0.4, 1.6, and 0.025 seconds, respectively and reverse biased for similar durations.

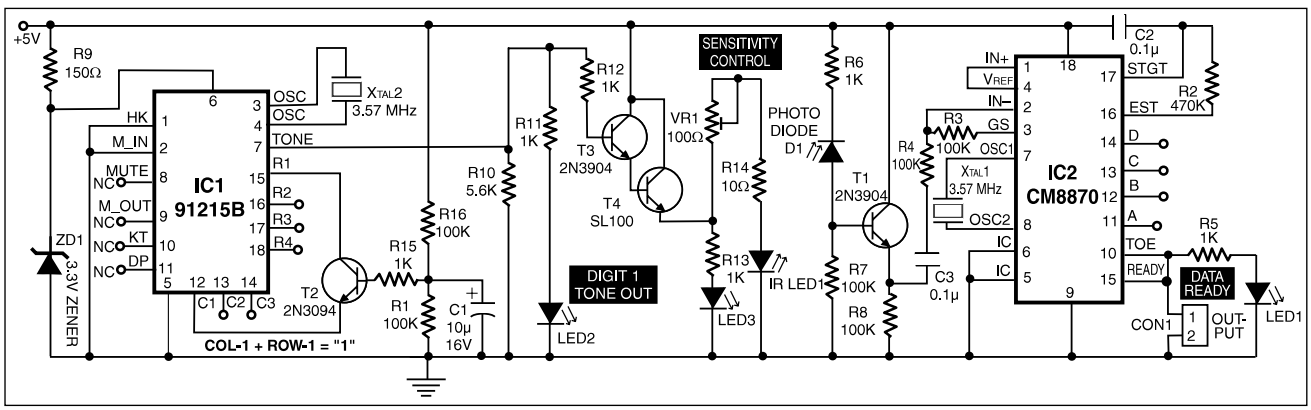
Using a built-in oscillator-type piezo-buzzer produces around 1kHz tone. In this circuit, the piezo-buzzer is turned 'on' and 'off' at 20 Hz for ring tone sound by transistor T3. 20Hz pulses are available at the collector of transistor T3 for 0.4-second duration. After a time interval of 0.4 second, 20Hz pulses become again available for another 0.4-second duration. This is followed by two seconds of no-sound interval. Thereafter the pulse pattern repeats itself.

Refer the figure that indicates waveforms available at various points including the collector of transistor T3. Preset VR2 can be used for adjusting the amplitude of the ring tone. ❑

# DTMF PROXIMITY DETECTOR

K.S. SANKAR

A DTMF-based IR transmitter and receiver pair can be used to realise a proximity detector. The circuit presented here enables you to detect any object capable of reflecting the IR beam and moving in front of the IR LED photodetector pair up to a distance of about 12 cm from it.




The circuit uses the commonly available telephony ICs such as dial-tone generator 91214B/91215B (IC1) and DTMF decoder CM8870 (IC2) in conjunction with infrared LED (IR LED1), photodiode D1, and other components as shown in the figure. A properly regulated 5V DC power supply is required for operation of the circuit.

The transmitter part is configured around dialer IC1. Its row 1 (pin 15) and column 1 (pin 12) get connected together via transistor T2 after a power-on delay (determined by capacitor C1 and resistors R1 and R16 in the base circuit of the transistor) to generate DTMF tone (combination of 697 Hz and 1209 Hz) corresponding to keypad digit "1" continuously.

LED 2 is used to indicate the tone output from IC3. This tone output is amplified by Darlington transistor pair of T3 and T4 to drive IR LED1 via variable resistor VR1 in series with fixed 10-ohm resistor R14. Thus IR LED1 produces tone-modulated IR light. Variable resistor VR1 controls the emission level to vary the transmission range. LED 3 indicates that transmission is taking place.

A part of modulated IR light signal transmitted by IR LED1, after reflection from an object, falls on photodetector diode D1. (The photodetector is to be shielded from direct IR light transmission path of IR LED1 by using any opaque partition so that it receives only the reflected IR light.) On detection of the signal by photodetector, it is coupled to DTMF decoder IC2 through emitter-follower transistor T1.

When the valid tone pair is detected by the decoder, its StD pin 15 (shorted to TOE pin 10) goes 'high'. The detection of the object in proximity of IR transmitter-receiver combination is indicated by LED1. The active-high logic output pulse (terminated at connector CON1, in the figure) can be used to switch on/off any device (such as a siren via a latch and relay driver) or it can be used to clock a counter, etc.

This DTMF proximity detector finds applications in burglar alarms, object counter and tachometers, etc. ❑

# UNDER-/OVER-VOLTAGE BEEP FOR MANUAL STABILISER


K. UDHAYA KUMARAN

Manual stabilisers are still popular because of their simple construction, low cost, and high reliability due to the absence of any relays while covering a wide range of mains AC voltages compared to that handled by automatic voltage stabilisers. These are used mostly in homes and in business centres for loads such as lighting, TV, and fridge, and in certain areas where the mains AC voltage fluctuates between very low (during peak hours) and abnormally high (during non-peak hours).

Some manual stabilisers available in the market incorporate the high-voltage auto-cut-off facility to turn off the load when the output voltage of manual stabiliser exceeds a certain preset high voltage limit. The output voltage may become high due to the rise in AC mains voltage or due to improper selection by the rotary switch on manual stabiliser.

One of the major disadvantage of using a manual stabiliser in areas with a wide range of voltage fluctuations is that one has to keep a watch on the manual stabiliser's output voltage that is displayed on a voltmeter and keep changing the same using its rotary switch. Or else, the output voltage may reach the preset auto-cut-off limit to switch off the load without the user's knowledge. To turn on the load again, one has to readjust the stabiliser voltage using its rotary switch. Such operation is very irritating and inconvenient for the user.

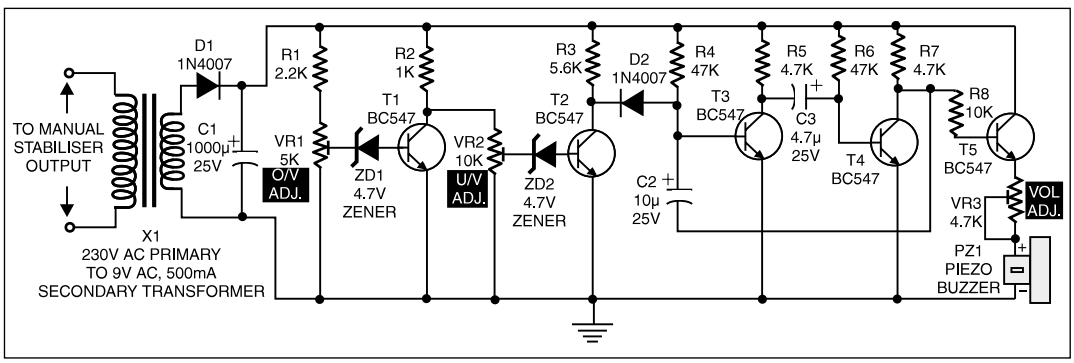
This under-/over-voltage audio alarm circuit designed as an add-on circuit for the existing manual stabilisers overcomes the above problem. Whenever the stabiliser's output voltage falls below a preset low-level voltage or rises above a preset high-level voltage, it produces different beep sounds for 'high' and 'low' voltage levels-short-duration beeps with short intervals between successive beeps for 'high' voltage level and slightly longer-duration beeps with longer interval between successive beeps for 'low' voltage level. By using these two different types of beep sounds one can readily readjust the stabiliser's AC voltage output with the help of the rotary switch. There is no need of frequently checking voltmeter reading.

It is advisable to preset the high-level voltage 10V to 20V less than the required high-voltage limit for auto-cut-off operation. Similarly, for low level one may preset low-level AC voltage 20V to 30V above minimum operating voltage for a given load.

The primary winding terminals of step-down transformer X1 are connected to the output terminals of the manual stabiliser. Thus, 9V DC available across capacitor C1 will vary in accordance with the voltage available at the output terminals of the manual stabiliser, which is used to sense high or low voltage in this circuit.

Transistor T1 in conjunction with zener diode ZD1 and preset VR1 is used to sense and adjust the high-voltage level for beep indication. Similarly, transistor T2 along with zener ZD2 and preset VR2 is used to sense and adjust low voltage level for beep indication.

When the DC voltage across capacitor C1 rises above the preset high-level voltage or falls below the preset low-level voltage, the collector of transistor T2 becomes high due to non-conduction of transistor T2, in either case. However, if the DC voltage sampled across C1 is within the preset high- and low-level voltage, transistor T2 conducts and its collector voltage gets pulled to the ground level. These changes in the collector voltage of transistor T2 are used to start or stop oscillations in the astable multivibrator circuit that is built around transistors T3 and T4. The collector of transistor T4 is connected to the base of buzzer driver transistor T5 through resistor R8. Thus when the collector voltage of transistor T4 goes high, the buzzer sounds. Preset VR3 is used to control the volume of buzzer sound.




In normal condition, the DC voltage sampled across capacitor C1 is within the permissible window voltage zone. The base of transistor T3 is pulled low due to conduction of diode D2 and transistor T2. As a result, capacitor C2 is discharged. The astable multivibrator stops oscillating and transistor T4 starts conducting because transistor T3 is in cut-off state. No beep sound is heard in the buzzer due to conduction of transistor T4 and non-conduction of transistor T5.

When the DC voltage across capacitor C1 goes above or below the window voltage level, transistor T2 is cut off. Its collector voltage goes high and diode D2 stops conducting. Thus there is no discharge path for capacitor C2 through diode D2. The astable multivibrator starts

oscillating. The time period for which the beep is heard and the time interval between two successive beeps are achieved with the help of the DC supply voltage, which is low during low-level voltage sampling and high during high-level voltage sampling. The time taken for charging capacitors C2 and C3 is less when the DC voltage is high and slightly greater when the DC voltage is low for astable multivibrator operation. Thus during low-level voltage sensing the buzzer beeps for longer duration with longer interval between successive beeps compared to that during high-voltage level sensing.

This circuit can be added to any existing stabiliser (automatic or manual) or UPS to monitor its performance. ❑

# 15-STEP DIGITAL POWER SUPPLY

NAVEEN THARIYAN


Here is a simple circuit to obtain variable DC voltage from 1.25V to 15.19V in reasonably small steps as shown in the table. The input voltage may lie anywhere between 20V and 35V.

The first section of the circuit comprises a digital up-down counter built around IC1- a quad 2-input NAND schmitt trigger (4093), followed by IC2- a binary up-down counter (4029). Two gates of IC 4093 are used to generate up-down logic using push buttons S1 and S2, respectively, while the other two gates form an oscillator to provide clock pulses to IC2 (4029). The frequency of oscillations can be varied by changing the value of capacitor C1 or preset VR1.

IC2 receives clock pulses from the oscillator and produces a sequential binary output. As long as its pin 5 is low, the counter continues to count at the rising edge of each clock pulse, but stops counting as soon as its pin 5 is brought to logic 1.

Logic 1 at pin 10 makes the counter to count upwards, while logic 0 makes it count downwards. Therefore the counter counts up by closing switch S1 and counts down by closing switch S2.

The output of counter IC2 is used to realise a digitally variable resistor. This section consists of four N/O reed relays that need just about 5mA current for their operation. (***EFY lab note***. *The original circuit containing quad bilateral switch IC 4066 has been replaced by reed relays operated by transistorised switches because of unreliable operation of the former*.) The switching action is performed using BC548 transistors. External resistors are connected in parallel with the reed relay contacts. If particular relay contacts are opened by the control input at the base of a transistor, the corresponding resistor across the relay contacts gets connected to the circuit.

The table shows the theoretical output for various digital input combinations. The measured output is nearly equal to the theoretically calculated output across regulator IC3 (LM317). The output voltage is governed by the following relationship as long as the input-to-output differential is greater than or equal to 2.5V:

Vout = 1.25(1+R2'/R1')

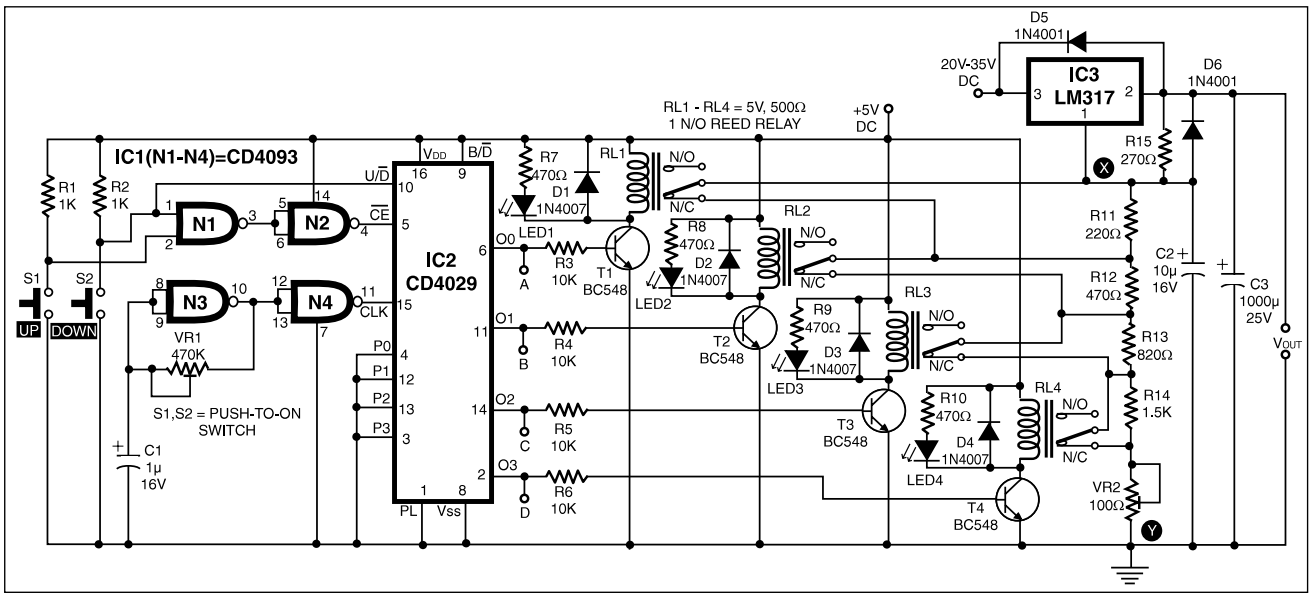
Where, R1' = R15 = 270 ohms (fixed)

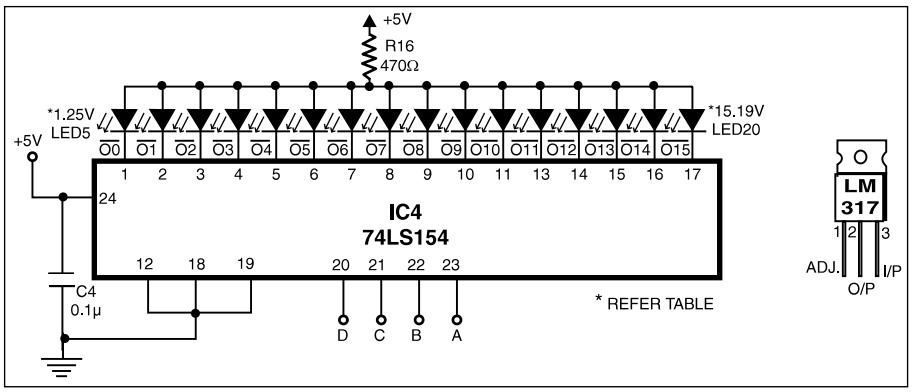
and R2' = R11 + R12 + R13 + R14

= 220 + 470 + 820 +1500 ohms

= 3,010 ohms (with all relays energised)

One can use either the binary weighted LED display as indicated by LED1 through LED4 in the circuit or a 74LS154 IC in conjunction with LED5 through LED20 to indicate one of the 16 selected voltage steps of Table I. The input for IC4 is to be tapped from points



**TABLE**

| Binary output | Equivalent dec no. | LED4 R14 (W) | LED3 R13 (W) | LED2 R12 (W) | LED1 R11 (W) | R2' (W) | Vout (V) |
|---|---|---|---|---|---|---|---|
| 0000 | 0 | Shorted | Shorted | Shorted | Shorted | 0 | 1.25 |
| 0001 | 1 | Shorted | Shorted | Shorted | 220 | 220 | 2.27 |
| 0010 | 2 | Shorted | Shorted | 470 | Shorted | 470 | 3.43 |
| 0011 | 3 | Shorted | Shorted | 470 | 220 | 690 | 4.44 |
| 0100 | 4 | Shorted | 820 | Shorted | Shorted | 820 | 5.05 |
| 0101 | 5 | Shorted | 820 | Shorted | 220 | 1040 | 6.06 |
| 0110 | 6 | Shorted | 820 | 470 | Shorted | 1290 | 7.22 |
| 0111 | 7 | Shorted | 820 | 470 | 220 | 1510 | 8.24 |
| 1000 | 8 | 1500 | Shorted | Shorted | Shorted | 1500 | 8.19 |
| 1001 | 9 | 1500 | Shorted | Shorted | 220 | 1720 | 9.21 |
| 1010 | 10 | 1500 | Shorted | 470 | Shorted | 1970 | 10.37 |
| 1011 | 11 | 1500 | Shorted | 470 | 220 | 2190 | 11.39 |
| 1100 | 12 | 1500 | 820 | Shorted | Shorted | 2390 | 11.99 |
| 1101 | 13 | 1500 | 820 | Shorted | 220 | 2540 | 13.01 |
| 1110 | 14 | 1500 | 820 | 470 | Shorted | 2790 | 14.17 |
| 1111 | 15 | 1500 | 820 | 470 | 220 | 3010 | 15.19 |

marked 'A' through 'D' in the figure. This arrangement can be used to replace the LED arrangement at points A, B, C, and D. This 74LS154 IC is a decoder/demultiplexer that senses the output of IC2 and accordingly activates only one of its 16 outputs in accordance with the count value. LEDs at the output of this IC can be arranged in a circular way along side the corresponding voltages.

## Working

When the power is switched on, IC2 resets itself, and hence the output at pins 6, 11, 14, and 12 is equivalent to binary zero, i.e. '0000'. The corresponding DC output of the circuit is minimum (1.25V). As count-up switch S1 is pressed, the binary count of IC2 increases and the output starts increasing too. At the highest count output of 1111, the output voltage is 15.19V (assuming the in-circuit resistance of preset VR2 as zero). Preset VR2 can be used for trimming the output voltage as desired. To decrease the output voltage within the range of 1.25V to 15.2V, count-down switch S2 is to be depressed.

***Notes.*** 1. When relay contacts across a particular resistor are opened, the corresponding LED glows.

2. The output voltages are shown assuming the in-circuit resistance of preset VR2 as zero. Thus when the in-circuit resistance of preset VR2 is not zero, the output voltage will be higher than that indicated here. ❑

# GENERATION OF 1-SEC. PULSES SPACED 5-SEC. APART


PRAVEEN SHANKER

This circuit using a dual-timer NE556 can produce 1Hz pulses spaced 5 seconds apart, either manually or automatically. IC NE556 comprises two independent NE555 timers in a single package. It is used to produce two separate pulses of different pulse widths, where one pulse initiates the activation of the second pulse.

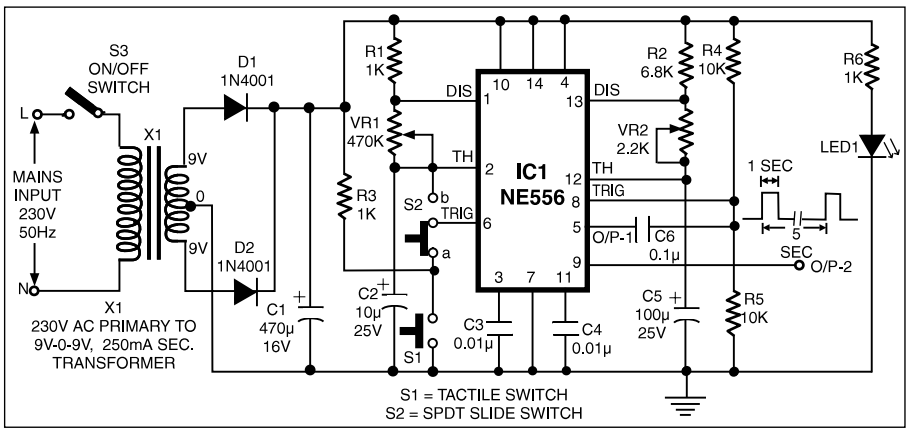
The first half of the NE556 is wired for 5-second pulse output. When slide switch S2 is in position 'a', the first timer is set for manual operation, i.e. by pressing switch S1 momentarily you can generate a single pulse of 5-second duration. When switch S2 is kept in 'b' position, i.e. pins 6 and 2 are shorted, timer 1 in NE556 triggers by itself.

The output of the first timer is connected to trigger pin 8 of second timer, which, in turn, is connected to a potential divider comprising resistors R4 and R5. Resistor R1, preset VR1, resistor R2, preset VR2, and capacitors C2 and C5 are the components determining time period. Presets VR1 and VR2 permit trimming of the 5-second and 1-second pulse width of respective sections.

When switch S2 is in position 'a' and switch S1 is pressed momentarily, the output at pin 5 goes high for about 5 seconds. The trailing (falling) edge of this 5-second pulse is used to trigger the second timer via 0.1µF capacitor C6. This action results in momentarily pulling down of pin 8 towards the ground potential, i.e. 'low'. (Otherwise pin 8 is at 1/2 Vcc and triggers at/below 1/3 Vcc level.) When the second timer is triggered at the trailing edge of 5-second pulse, it generates a 1-second wide pulse.

When switch S2 is on position 'b', switch S1 is disconnected, while pin 6 is connected to pin 2. When capacitor C is charged, it is discharged through pin 2 until it reaches 1/3Vcc potential, at which it is retriggered since trigger pin 6 is also connected here. Thus timer 1 is retriggered after every 5-second period (corresponding to 0.2Hz frequency). The second timer is triggered as before to produce a 1-second pulse in synchronism with the trailing edge of 5-second pulse.

This circuit is important wherever a pulse is needed at regular intervals; for instance, in 'Versatile Digital Frequency Counter Cum Clock' construction project published in EFY Oct. '97, one may use this circuit in place of CD4060-based circuit. For the digital clock function, however, pin 8 and 12 are to be shorted after removal of 0.1µF capacitor and 10-kilo-ohm resistors R4 and R5. ❑



# AUTOMATIC HEAT DETECTOR

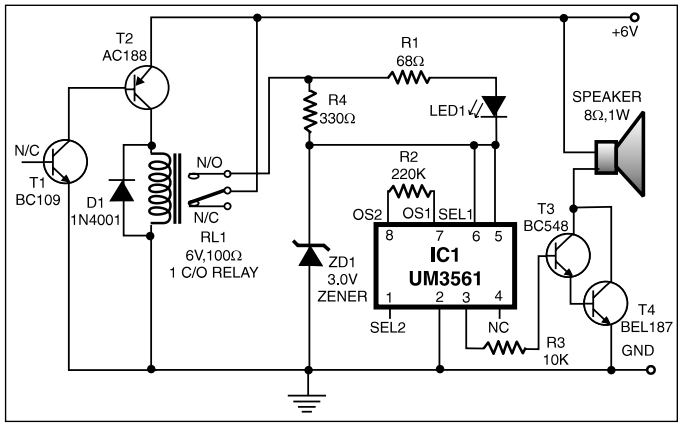

SUKANT KUMAR BEHARA

This circuit uses a complementary pair comprising npn metallic transistor T1 (BC109) and pnp germanium transistor T2 (AC188) to detect heat (due to outbreak of fire, etc) in the vicinity and energise a siren. The collector of transistor T1 is connected to the base of transistor T2, while the collector of transistor T2 is connected to relay RL1.

The second part of the circuit comprises popular IC UM3561 (a siren and machine-gun sound generator IC), which can produce the sound of a fire-brigade siren. Pin numbers 5 and 6 of the IC are connected to the +3V supply when the relay is in energised state, whereas pin 2 is grounded. A resistor (R2) connected across pins 7 and 8 is used to fix the frequency of the inbuilt oscillator. The output is available from pin 3.

Two transistors BC147 (T3) and BEL187 (T4) are connected in Darlington configuration to amplify the sound from UM3561. Resistor R4 in series with a 3V zener is used to provide the 3V supply to UM3561 when the relay is in energised state. LED1, connected in series with 68-ohm resistor R1 across resistor R4, glows when the siren is on.

| Pin Designation | | Sound Effect |
|---|---|---|
| SEL1 | SEL2 | |
| No Connection | No Connection | Police Siren |
| +3V | No Connection | Fire Engine Siren |
| Ground | No Connection | Ambulance Siren |
| Do not care | +3V | Machine Gun |




To test the working of the circuit, bring a burning matchstick close to transistor T1 (BC109), which causes the resistance of its emitter-collector junction to go low due to a rise in temperature and it starts conducting. Simultaneously, transistor T2 also conducts because its base is connected to the collector of transistor T1. As a result, relay RL1 energises and switches on the siren circuit to produce loud sound of a fire-brigade siren.

***Lab note***. We have added a table to enable readers to obtain all possible sound effects by returning pins 1 and 2 as suggested in the table. ❑

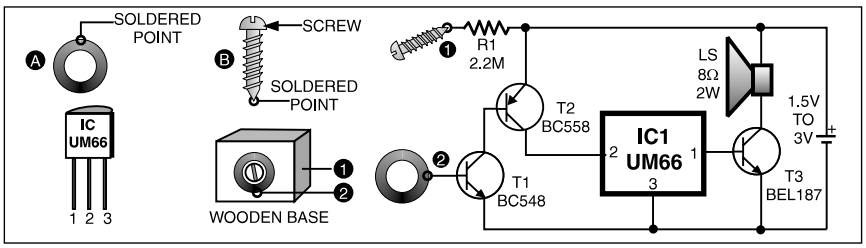
# MUSICAL 'TOUCH' BELL

SUKANT KUMAR BEHARA

Here is a musical call bell that can be operated by just bridging the gap between the touch-plates with one's fingertips. Thus there is no need for a mechanical 'on'/'off' switch because the touch-plates act as a switch. Other features include low cost and low power consumption. The bell can work on 1.5V or 3V, using one or two pencil cells, and can be used in homes and offices.

Two transistors are used for sensing the finger touch and switching on a melody IC. Transistor BC148 is npn type while transistor BC558 is pnp type.

The emitter of transistor BC148 is shorted to the ground, while that of transistor BC558 is connected to the positive terminal. The collector of transistor BC148 is connected to the base of BC558. The base of BC148 is connected to the washer (as shown in the figure). The collector of BC558 is connected to pin 2 of musical IC UM66, and pin 3 of IC UM66 is shorted to the ground. The output from pin 1 is connected to a transistor amplifier comprising BEL187 transistor for feeding the loudspeaker. One end of 2.2-mega-ohm resistor R1 is connected to the positive rail and the other to a screw (as shown in the figure). The complete circuit is connected to a single pencil cell of 1.5V.




When the touch-plate gap is bridged with a finger, the emitter-collector junction of transistor BC148 starts conducting. Simultaneously, the emitter-baser junction of transistor BC558 also starts conducting. As a result, the collector of transistor BC558 is pulled towards the positive rail, which thus activates melody generator IC1 (UM66). The output of IC1 is amplified by transistor BEL187 and fed to the speaker. So we hear a musical note just by touching the touch points.

The washer's inner diameter should be 1 to 2 mm greater than that of the screwhead. The washer could be fixed in the position by using an adhesive, while the screw can be easily driven in a wooden piece used for mounting the touch-plate. The use of brass washer and screw is recommended for easy solderability. ❑

# PRECISION AMPLIFIER WITH DIGITAL CONTROL


ANANTHA NARAYAN

This circuit is similar to the preceding circuit of the attenuator. Gain of up to 100 can be achieved in this configuration, which is useful for signal conditioning of low output of transducers in millivolt range.

The gain selection resistors R3 to R6 can be selected by the user and can be anywhere from 1 kilo-ohm to 1 meg-ohm. Trimpots can be used for obtaining any value of gain required by the user. The resistor values shown in the circuit are for decade gains suitable for an autoranging DPM.

Resistor R1 and capacitor C1 reduce ripple in the input and also snub transients. Zeners Z1 and Z2 limit the input to ±4.7V, while the input current is limited by resistor R1. Capacitors C2 and C3 are the power supply decoupling capacitors.

Op-amp IC1 is used to increase the input impedance so that very low inputs are not loaded on measurement. The user can terminate the inputs with resistance of his choice (such as 10 meg-ohm or 1 meg-ohm) to avoid floating of the inputs when no measurement is being made.

IC5 is used as an inverting buffer to restore polarity of the input while IC4 is used as buffer at the output of CD4052, because loading it by resistance of value less than 1 meg-ohm will cause an error. An alternative is to make R7=R8=1 meg-ohm and do away with IC4, though this may not be an ideal method.

**Truth Table (Control Input vs Gain)**

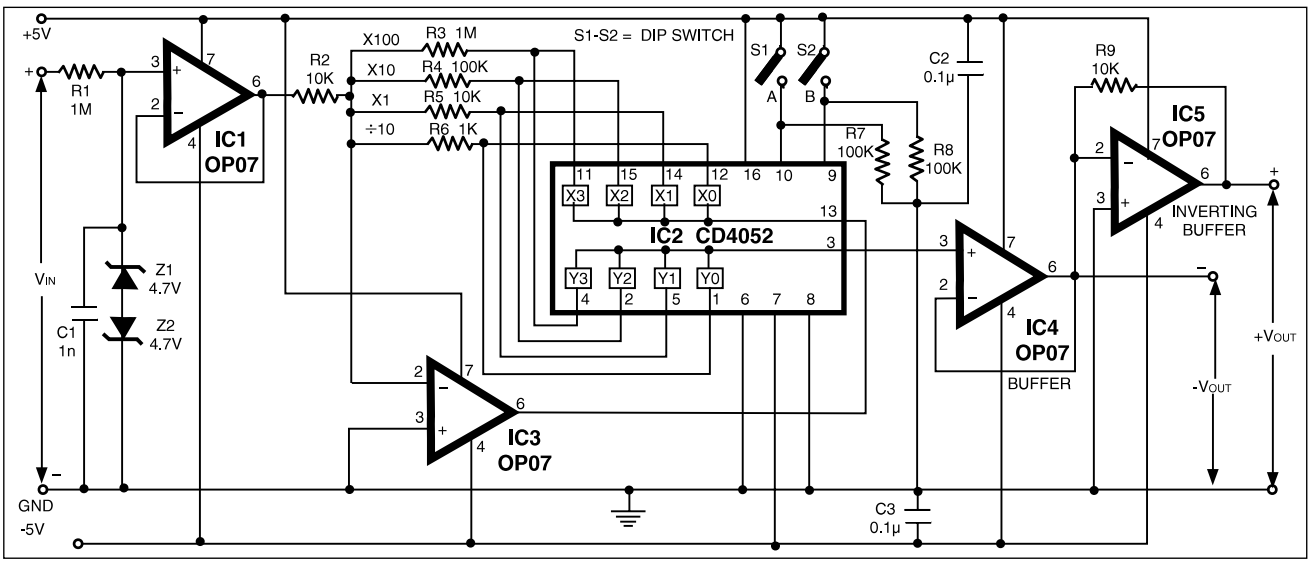
| X,Y (On-switch Pair) | (2) B | (1) A | Gain (Av.) |
|---|---|---|---|
| X0,Y0 | 0 | 0 | 1/10 |
| X1,Y1 | 0 | 1 | 1 |
| X2,Y2 | 1 | 0 | 10 |
| X3,Y3 | 1 | 1 | 100 |

Gains greater than 100 may not be practical because even at gain value of 100 itself, a 100µV offset will work out to be around 10 mV at the output (100µV x 100). This can be trimmed using the offset null option in the OP07, connecting a trimpot between pins 1 and 8, and connecting wiper to +5V supply rails. For better performance, use ICL7650 (not pin-compatible) in place of OP07 and use ±7.5V instead of ±5V supply.

Eight steps for gain or attenuation can be added by using two CD4051 and pin 6 inhibit on CD4051/52. More steps can be added by cascading many CD4051, or CD4052, or CD4053 ICs, as pin 6 works like a chip select.

Some extended applications of this circuit are given below.

1. Error correction in transducer amplifiers by correcting gain.
2. Autoranging in DMM.
3. Sensor selection or input type selection in process control.
4. Digitally preset power supplies or electronic loads.
5. Programmable precision mV or mA sources.
6. PC or microcontroller or microprocessor based instruments.
7. Data loggers and scanners. ❑



# ADD-ON STEREO CHANNEL SELECTOR



PRABHASH K.P.

The add-on circuit presented here is useful for stereo systems. This circuit has provision for connecting stereo outputs from four different sources/channels as inputs and only one of them is selected/ connected to the output at any one time.

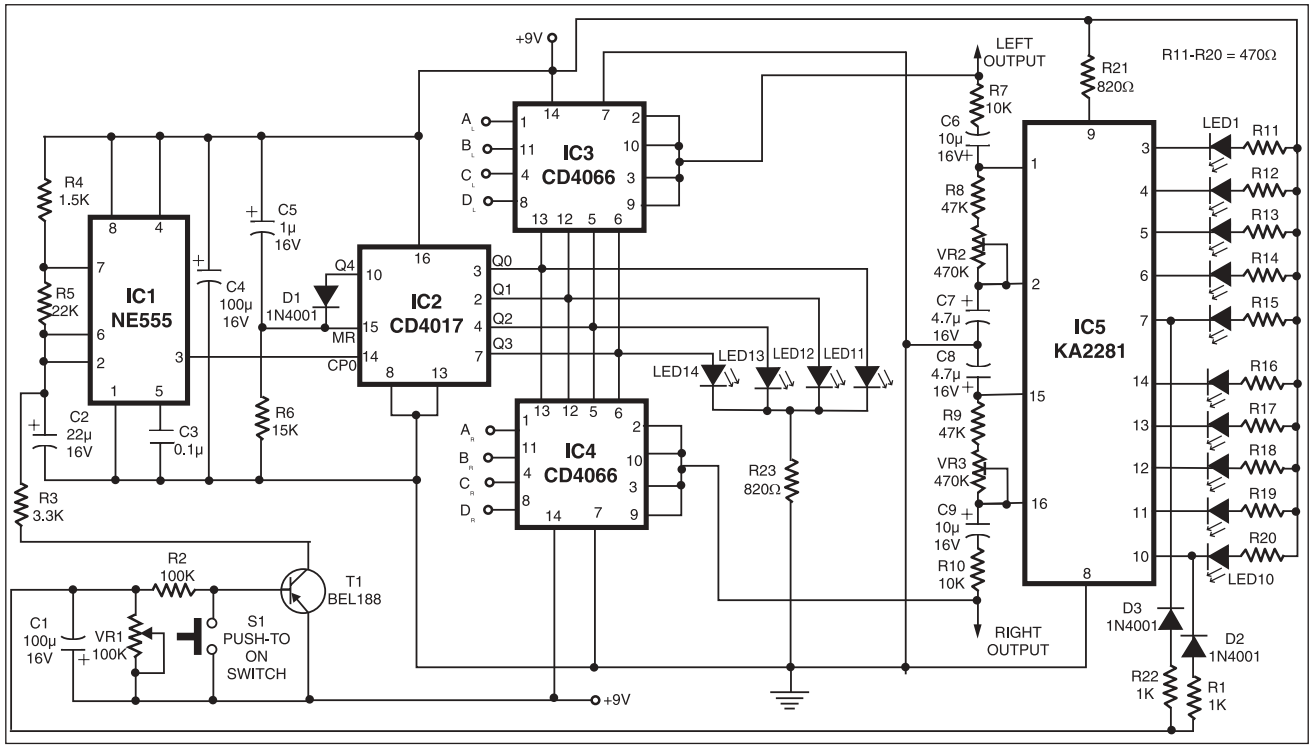
When power supply is turned 'on', channel A (A2 and A1) is selected. If no audio is present in channel A, the circuit waits for some time and then selects the next channel (channel B), This search operation continues until it detects audio signal in one of the channels. The inter-channel wait or delay time can be adjusted with the help of preset VR1. If still longer time is needed, one may replace capacitor C1 with a capacitor of higher value.

Suppose channel A is connected to a tape recorder and channel B is connected to a radio receiver. If initially channel A is selected, the audio from the tape recorder will be present at the output. After the tape is played completely, or if there is sufficient pause between consecutive recordings, the circuit automatically switches over to the output from the radio receiver. To manually skip over from one (selected) active channel, simply push the skip switch (S1) momentarily once or more, until the desired channel inputs gets selected. The selected channel (A, B, C, or D) is indicated by the glowing of corresponding LED (LED11, LED12, LED13, or LED14 respectively).

IC CD4066 contains four analogue switches. These switches are connected to four separate channels. For stereo operation, two similar CD4066 ICs are used as shown in the circuit. These analogue switches are controlled by IC CD4017 outputs. CD4017 is a 10-bit ring counter IC. Since only one of its outputs is high at any instant, only one switch will be closed at a time. IC CD4017 is configured as a 4-bit ring counter by connecting the fifth output Q4 (pin 10) to the reset pin. Capacitor C5 in conjunction with resistor R6 forms a power-on-reset circuit for IC2, so that on initial switching 'on' of the power supply, output Q0 (pin 3) is always 'high'. The clock signal to CD4017 is provided by IC1 (NE555) which acts as an astable multivibrator when transistor T1 is in cut-off state.

IC5 (KA2281) is used here for not only indicating the audio levels of the selected stereo channel, but also for forward biasing transistor T1. As soon as a specific threshold audio level is detected in a selected channel, pin 7 and/ or pin 10 of IC5 goes 'low'. This low level is coupled to the base of transistor T1, through diode-resistor combination of D2-R1/D3-R22. As a result, transistor T1 conducts and causes output of IC1 to remain 'low' (disabled) as long as the selected channel output exceeds the preset audio threshold level.

Presets VR2 and VR3 have been included for adjustment of individual audio threshold levels of left stereo channels, as desired. Once the multivibrator action of IC1 is disabled, output of IC2 does not change further. Hence, search-



ing through the channels continues until it receives an audio signal exceeding the preset threshold value. The skip switch S1 is used to skip a channel even if audio is present in the selected channel. The number of channels can be easily extended up to ten, by using additional 4066 ICs.

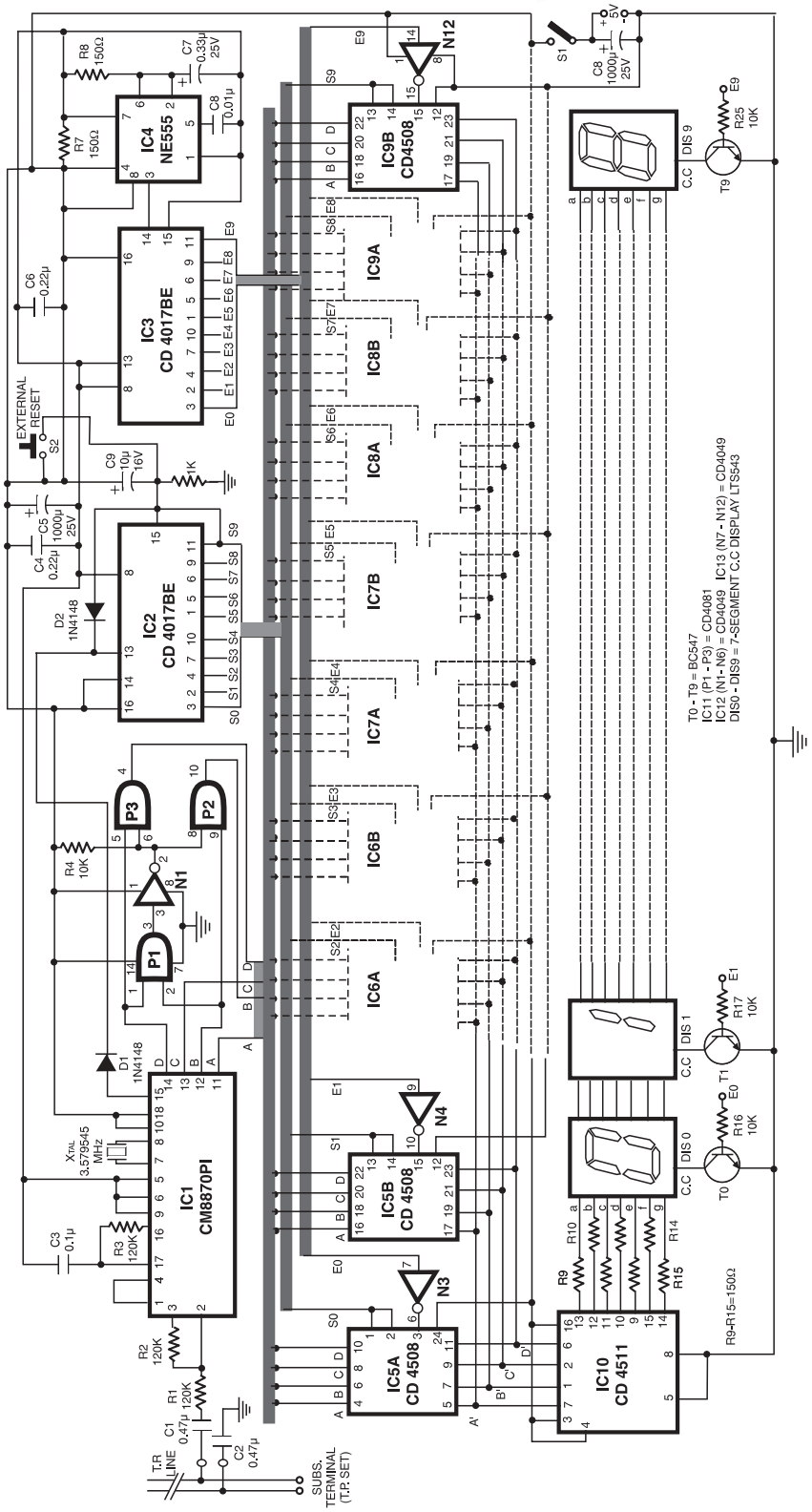
# Telephone Number Display

BHASKAR BANERJEE

The given circuit, when connected in parallel to a telephone, displays the number dialled from the telephone set using the DTMF mode. This circuit can also show the number dialled from the phone of the called party. This is particularly helpful for receiving any number over the phone lines.

The DTMF signal—generated by the phone on dialling a number—is decoded by DTMF decoder CM8870P1 (IC1), which converts the received DTMF signal into its equivalent BCD number that corresponds to the dialled number. This binary number is stored sequentially in 10 latches each time a number is dialled from the phone. The first number is stored in IC5A (1/2 of CD4508) while the second number is stored in IC5B and so on. The binary output from IC1 for digit '0' as decoded by IC1 is $1010_2$ ($=10_{10}$), and this cannot be displayed by the seven-segment decoder, IC10. Therefore the binary output of IC1 is passed through a logic-circuit which converts an input of '$1010_2$' into '$0000_2$' without affecting the inputs '1' through '9'. This is accomplished by gates N13 through N15 (IC11) and N1 (IC12).

The storing of numbers in respective latches is done by IC2 (4017). The data valid output from pin 15 of IC1 is used to clock IC2. The ten outputs of IC2 are sequentially connected to the store and clear inputs of all the latches, except the last one, where the clear input is tied to ground. When an output pin of IC2 is high, the corresponding latch is cleared of previous data and kept ready for storing new data. Then, on clocking IC2, the same pin becomes low and the data present at the inputs of that latch at that instant gets stored and the next latch is cleared and kept ready. The similar input and output pins of all latches are connected together to



form two separate input and output buses.

There is only one 7-segment decoder/driver IC10 for all the ten displays. This not only reduces size and cost but reduces power requirement too. The output from a latch is available only when its disable pins (3 and 15) are brought low. This is done by IC3, IC12 and IC13. IC3 is clocked by an astable multivibrator IC4 (555). IC3 also drives the displays by switching corresponding transistors. When a latch is enabled, its corresponding display is turned on and the content of that latch, after decoding by IC10, gets displayed in the corresponding display. For instance, contents of IC5A are displayed on display 'DIS1,' that of IC5B on 'DIS2' and so on. The system should be connected to the telephone lines via a DPDT switch (not shown) for manual switching, otherwise any circuit capable of sensing handset's off-hook condition and thereby switching relays, etc. can be used for automatic switching. The power-supply switch can also be replaced then. Such circuits, under different captions, can be found in EFY's back issues. Though this circuit is capable of showing a maximum of ten digits, one can reduce the display digits as required. For doing this, connect the reset pin of IC2, say, for a 7-digit display, with S6 output at pin 5.

The present circuit can be built on a veroboard and housed in a suitable box. The displays are common-cathode type. To make the system compact, small, 7-segment displays can be used but with some extra cost. Also, different colour displays can be used for the first three or four digits to separate the exchange code/STD code, etc. The circuit can be suitably adopted for calling-line display.

# LUGGAGE SECURITY SYSTEM

DHURJATI SINHA

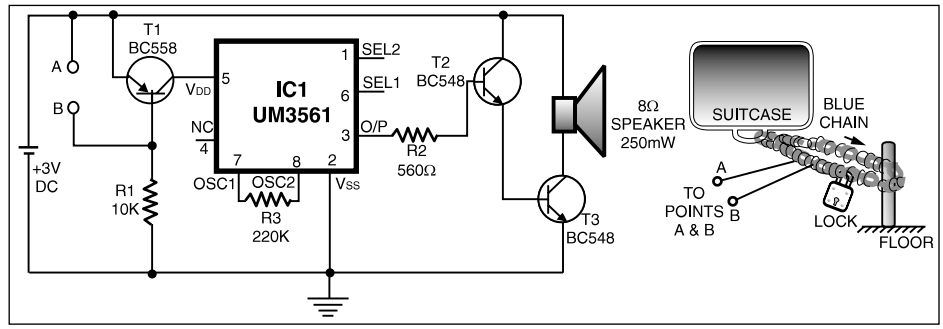

While travelling by a train or bus, we generally lock our luggage using a chain-and-lock arrangement. But, still we are under tension, apprehending that somebody may cut the chain and steal our luggage. Here is a simple circuit to alarm you when somebody tries to cut the chain.

Transistor T1 enables supply to the sound generator chip when the base current starts flowing through it. When the wire (thin enameled copper wire of 30 to 40 SWG, used for winding transformers) loop around the chain is broken by somebody, the base of transistor T1, which was earlier tied to positive rail, gets opened. As a result, transistor T1 gets forward biased to extend the positive supply to the alarm circuit. In idle mode, the power consumption of the circuit is minimum and thus it can be used for hundreds of travel hours.

To enable generation of different alarm sounds, connections to pin 1 and 6 may be made as per the table.

| Select 1 (Pin6) | Select 2 (Pin1) | Sound effect |
|---|---|---|
| X | X | Police siren |
| $V_{DD}$ | X | Fire-engine siren |
| $V_{SS}$ | X | Ambulance siren |
| "-" | $V_{DD}$ | Machine-gun sound |

*Note:* X = no connection; "-" = do not care

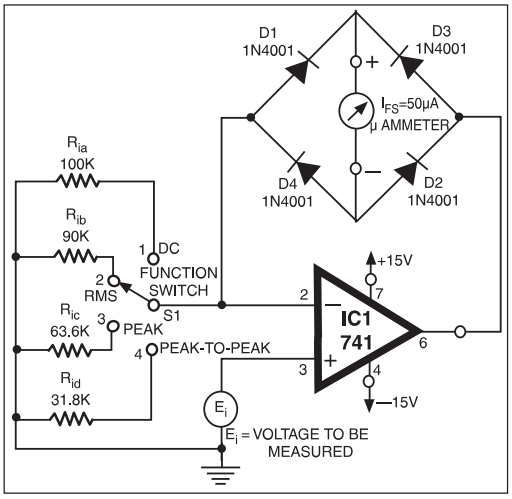


# UNIVERSAL HIGH-RESISTANCE VOLTMETER


YOGESH KATARIA

The full-scale deflection of the universal high-input-resistance voltmeter circuit shown in the figure depends on the function switch position as follows:

(a) 5V DC on position 1
(b) 5V AC rms in position 2
(c) 5V peak AC in position 3
(d) 5V AC peak-to-peak in position 4




The circuit is basically a voltage-to-current converter. The design procedure is as follows:

Calculate $R_I$ according to the application from one of the following equations:

(a) DC voltmeter: $R_{IA}$ = full-scale $E_{DC}/I_{FS}$

(b) RMS AC voltmeter (sine wave only): $R_{IB}$ = 0.9 full-scale $E_{RMS}/ I_{FS}$

(c) Peak reading voltmeter (sine wave only): $R_{IC}$ = 0.636 full-scale $E_{PK}/I_{FS}$

(d) Peak-to-peak AC voltmeter (sine wave only): $R_{ID}$ = 0.318 full-scale $E_{PK\text{-}TO\text{-}PK} / I_{FS}$

The term $I_{FS}$ in the above equations refers to meter's full-scale deflection current rating in amperes.

It must be noted that neither meter resistance nor diode voltage drops affects meter current.

*Note: The results obtained during practical testing of the circuit in EFY lab are tabulated in Tables I through IV.*

A high-input-resistance op-amp, a bridge rectifier, a microammeter, and a few other discrete components are all that are required to realise this versatile circuit. This circuit can be used for measurement of DC, AC RMS, AC peak, or AC peak-to-peak voltage by simply changing the value of the resistor connected between the inverting input terminal of the op-amp and ground. The voltage to be measured is connected to non-inverting input of the op-amp.

**TABLE I**
Position 1 of Function Switch

| $E_{dc}$ input | Meter Current |
|---|---|
| 5.00V | 44 μA |
| 4.00V | 34 μA |
| 3.00V | 24 μA |
| 2.00V | 14 μA |
| 1.00V | 4 μA |

**TABLE II**
Position 2 of Function Switch

| $E_{rms}$ input | Meter Current |
|---|---|
| 5V | 46 μA |
| 4V | 36 μA |
| 3V | 26 μA |
| 2V | 18 μA |
| 1V | 10 μA |

**TABLE III**
Position 3 of Function Switch

| $E_{pk}$ input | Meter Current |
|---|---|
| 5V peak | 46 μA |
| 4V peak | 36 μA |
| 3V peak | 26 μA |
| 2V peak | 16 μA |
| 1V peak | 6 μA |

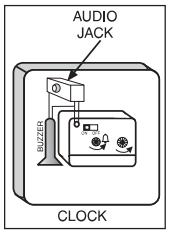
**TABLE IV**
Position 4 of Function Switch

| $E_{Pk\text{-}To\text{-}Pk}$ | Meter Current |
|---|---|
| 5V peak to peak | 46 μA |
| 4V peak to peak | 36 μA |
| 3V peak to peak | 26 μA |
| 2V peak to peak | 16 μA |
| 1V peak to peak | 7 μA |

# Time Switch

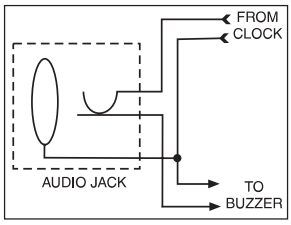

AVNISH PUNDIR

This circuit is especially designed for those who often need to wake up early in the morning. Ordinary alarms in electronic watches are not loud enough and very often they fail to wake up. The switch circuit described here will come handy; it can be used to switch on a TV, radio or tape recorder etc, which will not allow even the laziest amongst us to ignore their sound for too long. Besides, this time switch can also be used to switch on/off any other electric or electronic gadget at any time. What you need is a simple analogue electronic clock with alarm facility and a small circuit to implement the time switch.




This time switch has two modes. One is 'time-on' mode and the other is 'time-off' mode. In time-on mode, you set up the alarm in your clock as per normal procedure and at the set time this switch turns on the gadget connected at the output socket-1. In time-off mode, it turns your gadget off at the set time. The optional output socket-2 is wired in such a way that when you use this socket, the mode changes without having to flip the mode switch (i.e. mode switch can be omitted).

Please refer to the back panel diagram of a typical analogue clock and the audio jack, to see how the existing buzzer of the clock is required to be wired to the audio output from the clock. This will ensure that when plug is inserted in the audio jack, the clock's buzzer will remain off and not consume any power unnecessarily.

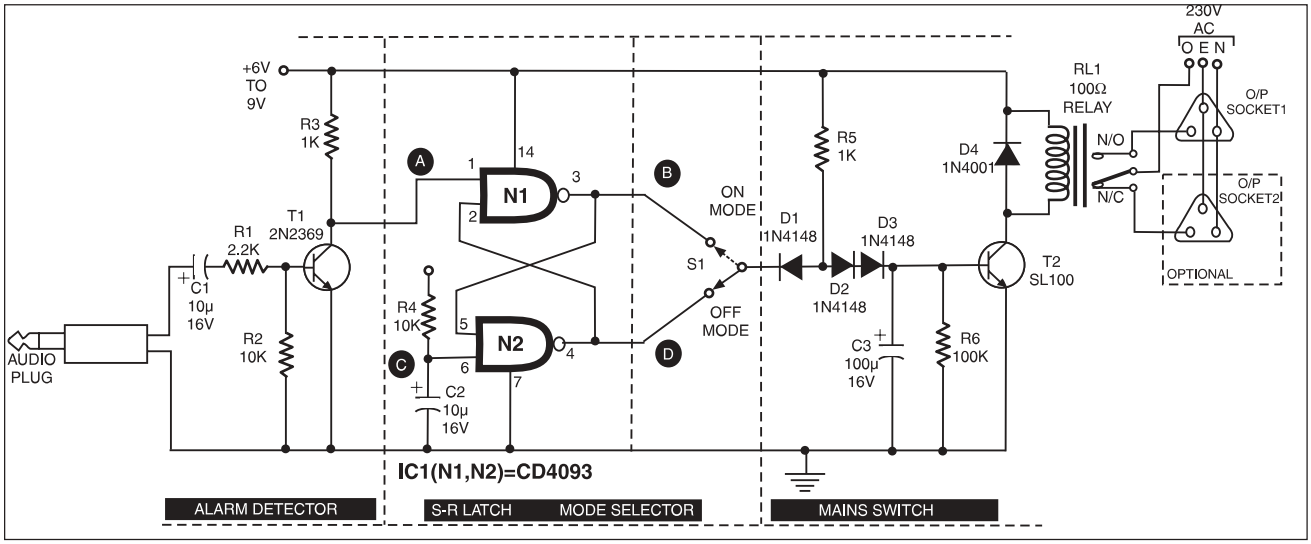



The audio alarm output from the clock is coupled to the AF detector built around low-power switching transistor T1. During alarm, the collector of transistor T1 will fluctuate around ground level and Vcc. During absence of audio alarm input, the collector of transistor T1 is held at Vcc potential.

The next stage consists of an S-R latch built around NAND gates N1 and N2. Capacitor C2 and resistor R4 are used for power-on-reset. On switching the power supply, gate N2 output will acquire logic 1 and that of gate N1 logic 0. This is the initial state, irrespective of the position of mode switch. At the time of alarm, when point A connected to collector of transistor T1 passes through logic 0 state, the output logic state of both the gates will toggle.

Assuming that mode switch is flipped to 'Mode Off' position at power-on-reset (when point D is at logic 1), initially diode D1 would be in blocking state and transistor T2 would be forward biased via resistor R5 and diodes D2 and D3. As a result, the relay is in energised state, which makes output power available at output socket-1 and cuts it off from socket-2. At alarm time, the audio signal toggles logic output states of both gates N1 and N2. As a result, point D goes to logic 0 state. Diode D1 conducts, taking the voltage at junction of diodes D1 and D2 to near about 1 volt. Diode D3 ensures that its series combination with diode D2 puts them in blocking mode. Capacitor C3 meanwhile discharges via resistor R6 and the voltage at base of transistor T2 approaches towards ground level, cutting off transistor T2 and de-energising relay RL1. Now the power at output socket-1 would be cut off while it becomes available in socket-2.

If the above operation is repeated with switch S1 in 'Mode On,' the power would initially not be available in socket-1 (but available in socket-2). But after the alarm, the power would become available in socket-1 and not in socket-2.



# Infrared Cordless Headphone


PRADEEP G.

Using this low-cost project one can reproduce audio from TV without disturbing others. It does not use any wire connection between TV and headphones. In place of a pair of wires, it uses invisible infrared light to transmit audio signals from TV to headphones. Without using any lens, a range of up to 6 metres is possible. Range can be extended by using lenses and reflectors with IR sensors comprising transmitters and receivers.

IR transmitter uses two-stage transistor amplifier to drive two series-connected IR LEDs. An audio output transformer is used (in reverse) to couple audio output from TV to the IR transmitter. Transistors T1 and T2 amplify the audio signals received from TV through the audio transformer. Low-impedance output windings (lower gauge or thicker wires) are used for connection to TV side while high-impedance windings are connected to IR transmitter. This IR transmitter can be powered from a 9-volt mains adapter or battery. Red LED1 in transmitter circuit functions as a zener diode (0.65V) as well as supply-on indicator.

IR receiver uses 3-stage transistor amplifier. The first two transistors (T4 and T5) form audio signal amplifier while the third transistor T6 is used to drive a headphone. Adjust potmeter VR2 for max. clarity.

Direct photo-transistor towards IR LEDs of transmitter for max. range. A 9-volt battery can be used with receiver for portable operation.

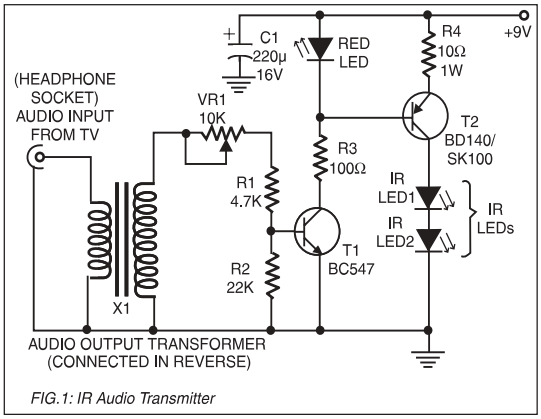



FIG.1: IR Audio Transmitter

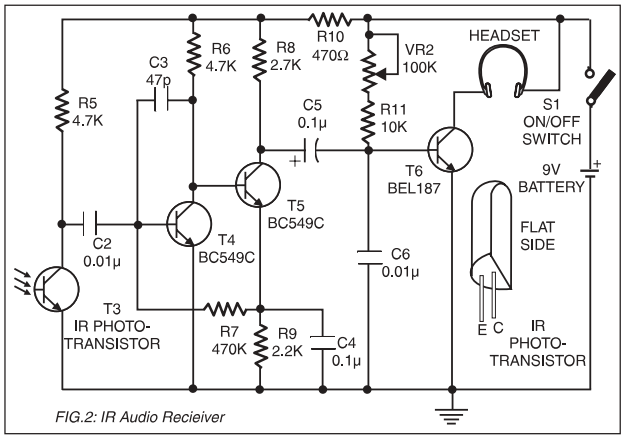



FIG.2: IR Audio Recieiver

# Dual-Channel Digital Volume Control

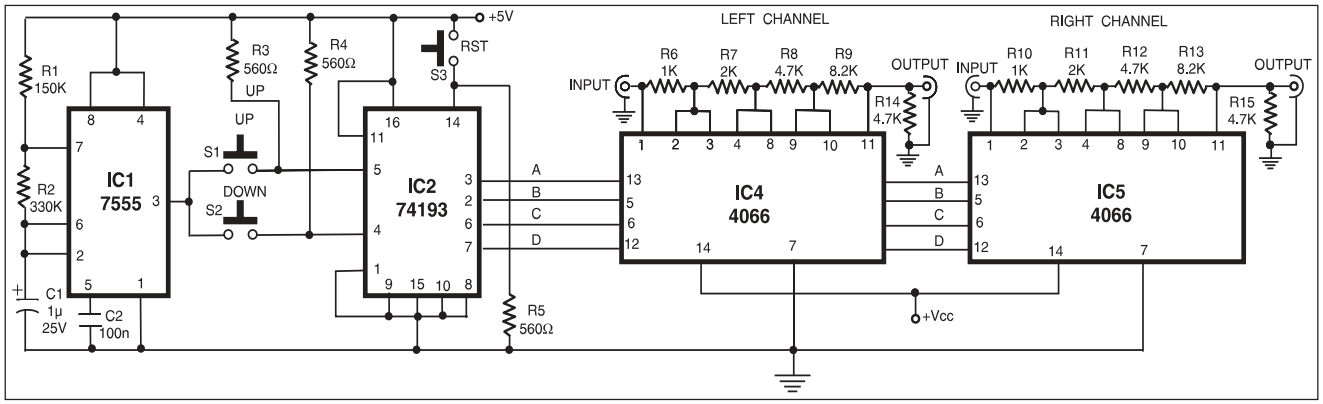
SHEENA K.

This circuit could be used for replacing your manual volume control in a stereo amplifier. In this circuit, push-to-on switch S1 controls the forward (volume increase) operation of both channels while a similar switch S2 controls reverse (volume decrease) operation of both channels.

Here IC1 timer 555 is configured as an astable flip-flop to provide low-frequency pulses to up/down clock input pins of pre-setable up/down counter 74LS193 (IC2) via push-to-on switches S1 and S2. To vary the pulse width of pulses from IC1, one may replace timing resistor R1 with a variable resistor.

Operation of switch S1 (up) causes the binary output to increment while operation of S2 (down) causes the binary output to decrement. The maximum count being 15 (all outputs logic 1) and minimum count being 0 (all outputs logic 0), it results in maximum and minimum volume respectively.

The active high outputs A, B, C and D of the counter are used for controlling two quad bi-polar analogue switches in each of the two CD4066 ICs (IC3 and IC4). Each of the output bits, when high, short a part of the resistor network comprising series resistors R6 through R9 for one channel and R10 through R13 for the other channel, and thereby control the output of the audio signals being fed to the inputs of stereo amplifier. Push-to-on switch S3 is used for resetting the output of counter to 0000, and thereby turning the volume of both channels to the minimum level.



# Simple Low-Cost Digital Code Lock

A. JEYABAL



Many digital code lock circuits have been published in this magazine. In those circuits a set of switches (conforming to code) are pressed one by one within the specified time to open the lock. In some other circuits, custom-built ICs are used and positive and negative logic pulses are keyed in sequence as per the code by two switches to open the lock.

A low-cost digital code lock circuit is presented in this article. Here the keying-in code is rather unique. Six switches are to be pressed to open the lock, but only two switches at a time. Thus a total of three sets of switches have to be pressed in a particular sequence. (Of these three sets, one set is repeated.) The salient features of this circuit are:

1. Use of 16 switches, which suggests that there is a microprocessor inside.

2. Elimination of power amplifier transistor to energise the relay.

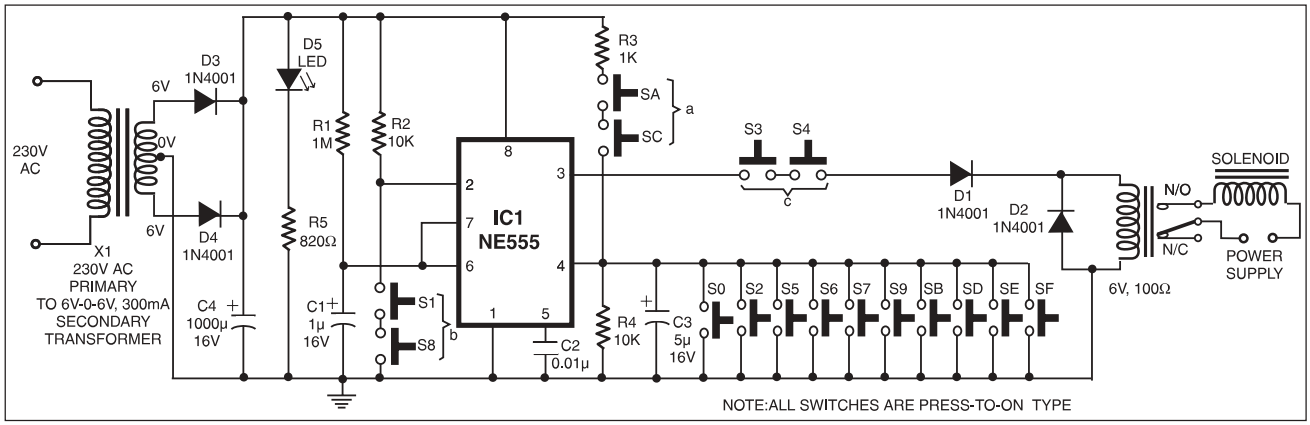
3. Low cost and small PCB size.

An essential property of this electronic code lock is that it works in monostable mode, i.e. once triggered, the output becomes high and remains so for a period of time, governed by the timing components, before returing to the quiescent low state. In this circuit, timer IC 555 with 8 pins is used. The IC is inexpensive and easily available. Its pin 2 is the triggering input pin which, when held below 1/3 of the supply voltage, drives the output to high state. The threshold pin 6, when held higher than 2/3 of the supply voltage, drives the output to low state. By applying a low-going pulse to the reset pin 4, the output at pin 3 can be brought to the quiescent low level. Thus the reset pin 4 should be held high for normal operation of the IC.

Three sets of switches SA-SC, S1-S8 and S3-S4 are pressed, in that order, to open the lock. On pressing the switches SA and SC simultaneously, capacitor C3 charges through the potential divider comprising resistors R3 and R4, and on releasing these two switches, capacitor C3 starts discharging through resistor R4. Capacitor C3 and resistor R4 are so selected that it takes about five seconds to fully discharge C3.

Depressing switches S1 and S8 in unison, within five seconds of releasing the switches SA and SC, pulls pin 2 to ground and IC 555 is triggered. The capacitor C1 starts charging through resistor R1. As a result, the output (pin 3) goes high for five seconds (i.e. the charging time T of the capacitor C1 to the threshold voltage, which is calculated by the relation T=1.1 R1 x C1 seconds).

Within these five seconds, switches SA and SC are to be pressed momentarily once again, followed by the depression of last code-switch pair S3-S4.




These switches connect the relay to output pin 3 and the relay is energised. The contacts of the relay close and the solenoid pulls in the latch (forming part of a lock) and the lock opens. The remaining switches are connected between reset pin 4 and ground. If any one of these switches is pressed, the IC is reset and the output goes to its quiescent low state. Possibilities of pressing these reset switches are more when a code breaker tries to open the lock.

LED D5 indicates the presence of power supply while resistor R5 is a current limiting resistor.

The given circuit can be recoded easily by rearranging connections to the switches as desired by the user.

# Electronic Jam

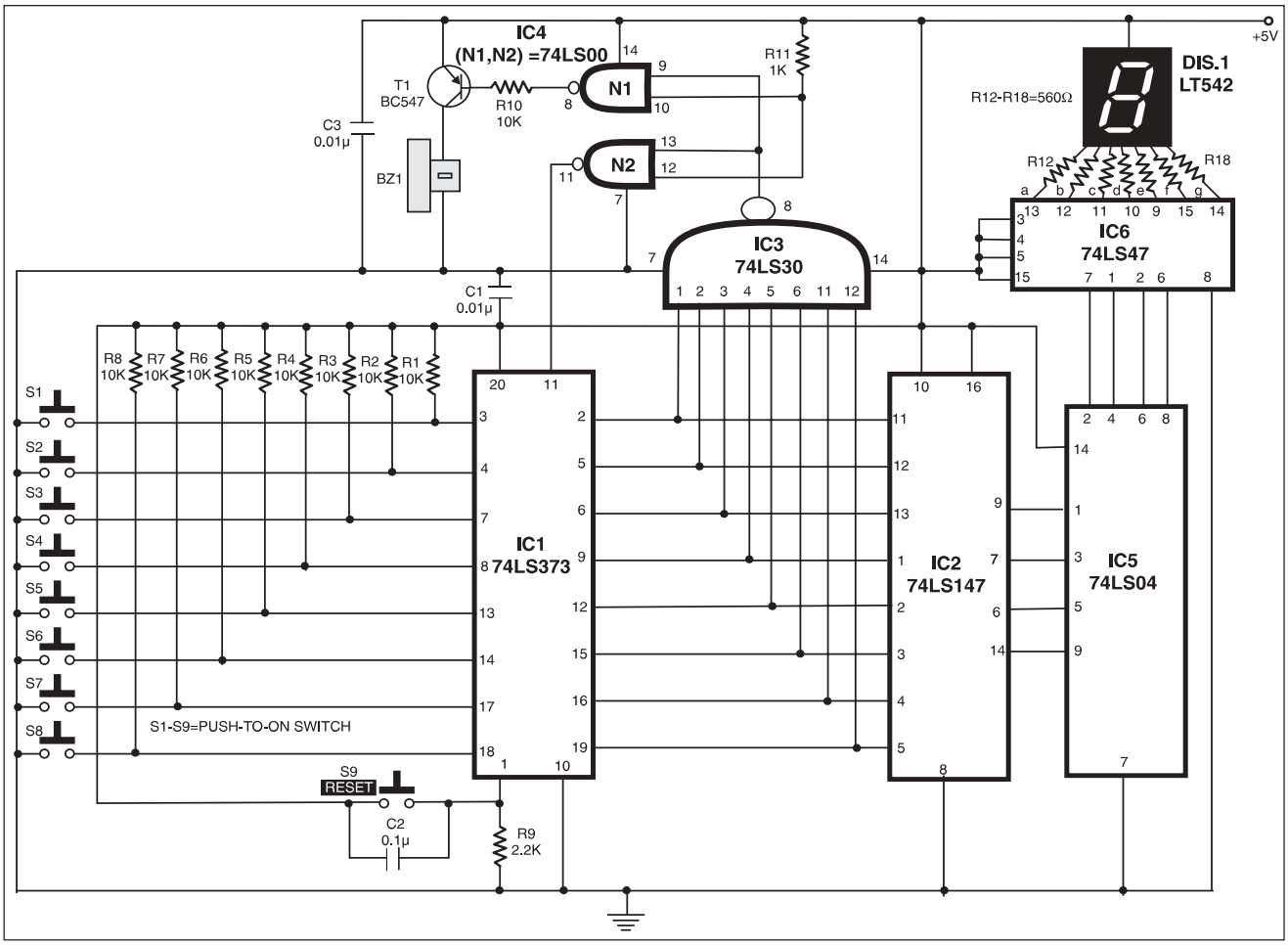

RAJESH K.P.

This jam circuit can be used in quiz contests wherein any participant who presses his button (switch) before the other contestants, gets the first chance to answer a question. The circuit given here permits up to eight contestants with each one allotted a distinct number (1 to 8). The display will show the number of the contestant pressing his button before the others. Simultaneously, a buzzer will also sound. Both, the display as well as the buzzer have to be reset manually using a common reset switch.

Initially, when reset switch S9 is momentarily pressed and released, all outputs of 74LS373 (IC1) transparent latch go 'high' since all the input data lines are returned to Vcc via resistors R1 through R8. All eight outputs of IC1 are connected to inputs of priority encoder 74LS147 (IC2) as well as 8-input NAND gate 74LS30 (IC3). The output of IC3 thus becomes logic 0 which, after inversion by NAND gate N2, is applied to latch-enable pin 11 of IC1. With all input pins of IC2 being logic 1, its BCD output is 0000, which is applied to 7-segment decoder/driver 74LS47 (IC6) after inversion by hex inverter gates inside 74LS04 (IC5). Thus, on reset the display shows 0.

When any one of the push-to-on switches—S1 through S8—is pressed, the corresponding output line of IC1 is latched at logic 0 level and the display indicates the number associated with the specific switch. At the same time, output pin 8 of IC3 becomes high, which causes outputs of both gates N1 and N2 to go to logic 0 state. Logic 0 output of gate N2 inhibits IC1, and thus pressing of any other switch S1 through S8 has no effect. Thus, the contestant who presses his switch first, jams the display to show only his number. In the unlikely event of simultaneous pressing (within few nano-seconds difference) of more than one switch, the higher priority number (switch no.) will be displayed. Simultaneously, the logic 0 output of gate N1 drives the buzzer via pnp transistor BC158 (T1). The buzzer as well the display can be reset (to show 0) by momentary pressing of reset switch S9 so that next round may start.

***Lab Note:*** The original circuit sent by the author has been modified as it did not jam the display, and a higher number switch (higher priority), even when pressed later, was able to change the displayed number.

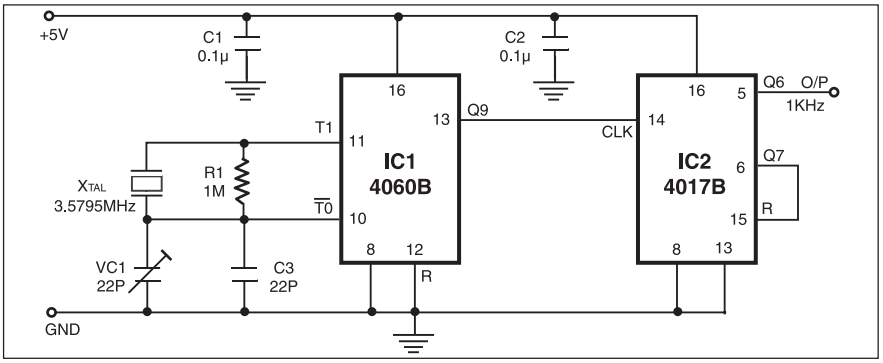


# CRYSTAL-CONTROLLED TIME-BASE GENERATOR


**PRATAP CHANDRA SAHU**

A digital frequency counter needs a time-base generator to count the frequency with high resolution. Normally, a crystal-based oscillator with divider IC chain or a similar circuit in the form of an ASIC (application-specific IC) is used for time-base generation. Here we've presented a simple circuit for accurate time-base generation using the readily available 3.5795MHz crystal commonly used in telecommunication equipment.

The 3.5795MHz crystal is used in conjunction with a CD4060-based crystal oscillator-cum-divider (IC1). The crystal frequency is divided by 512 by IC1, which is further divided by 7 by CD4017 (IC2). IC2 is reset as soon as its Q7 output goes high.




Thus the crystal frequency is divided by 3584, giving the final output frequency of around 998.8 Hz. This frequency can be trimmed to exactly 1 kHz with the help of trimmer capacitor VC1 as shown in the figure. The 1kHz signal can be further divided using decade counters to generate the required time period.

***EFY lab note**. To generate required gate for use in a frequency counter circuit, the final oscillator output needs to be followed by a toggle flip-flop. For example, a 1kHz clock, when applied to a toggle flip-flop, will generate gates with 1-sec 'on' period and 1-sec 'off' period.*

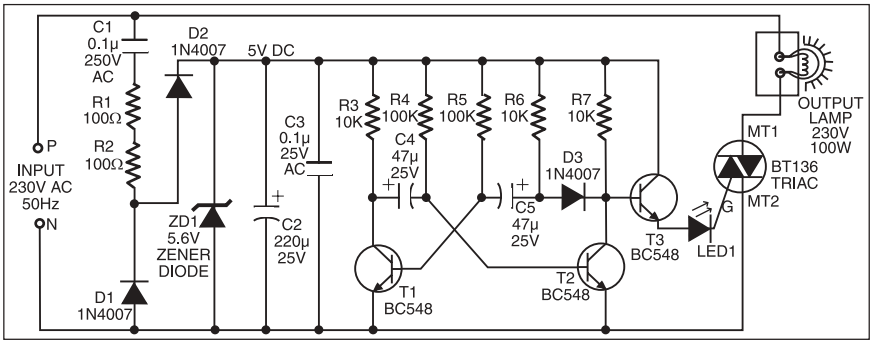
This circuit is estimated to cost below Rs 50.

# MAINS-OPERATED CHRISTMAS STAR

PRINCE PHILLIPS

Here is a low-cost circuit of Christmas star that can be easily constructed even by a novice. The main advantage of this circuit is that it doesn't require any step-down transformer or ICs.

Components like resistors R1 and R2, capacitors C1, C2, and C3, diodes D1 and D2, and zener ZD1 are used to develop a fairly steady 5V DC supply voltage that provides the required current to operate the multivibrator circuit and trigger triac BT136 via LED1.




The multivibrator circuit is constructed using two BC548 transistors (T1 and T2) and some passive components. The frequency of the multivibrator circuit is controlled by capacitors C4 and C5 and resistors R3 through R7. The output of the multivibrator circuit is connected to transistor T3, which, in turn, drives the triac via LED1. During positive half cycles of the multivibrator's output, transistor T3 energises triac BT136 and the lamp glows.

This circuit is estimated to cost Rs 75.

# MAINS MANAGER


SHIBASHISH PATEL

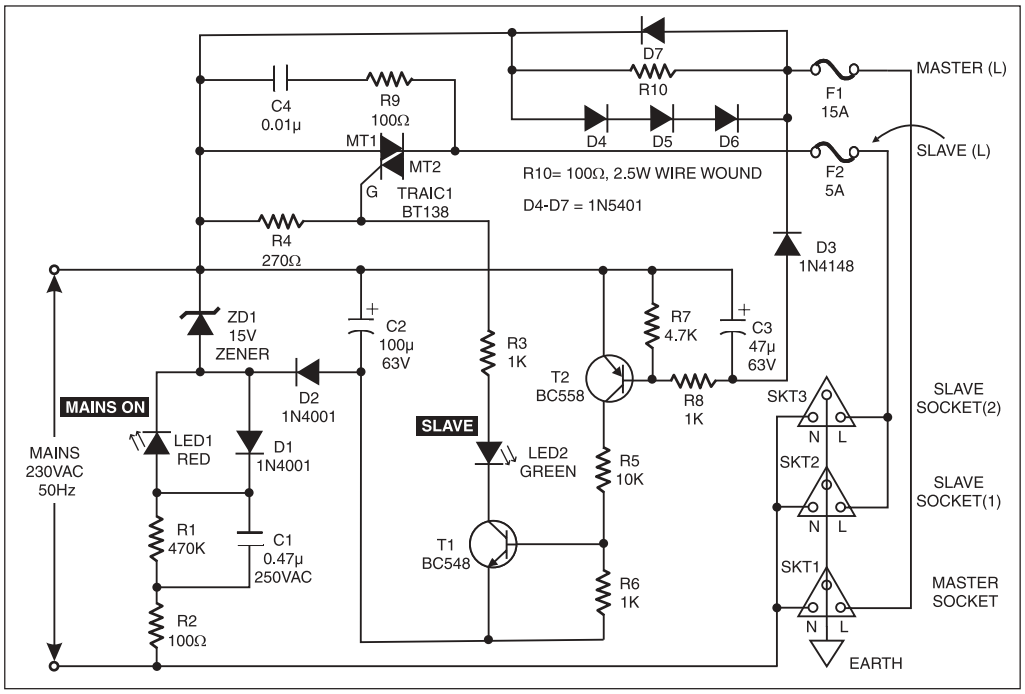
Very often we forget to switch off the peripherals like monitor, scanner, and printer while switching off our PC. The problem is that there are separate power switches to turn the peripherals off. Normally, the peripherals are connected to a single of those four-way trailing sockets that are plugged into a single wall socket. If that socket is accessible, all the devices could be switched off from there and none of the equipment used will require any modification.

Here is a mains manager circuit that allows you to turn all the equipment on or off by just operating the switch on any one of the devices; for example, when you switch off your PC, the monitor as well as other equipment will get powered down automatically. You may choose the main equipment to control other gadgets. The main equipment is to be directly plugged into the master socket, while all other equipment are to be connected via the slave socket. The mains supply from the wall socket is to be connected to the input of the mains manager circuit.

The unit operates by sensing the current drawn by the control equipment/load from the master socket. On sensing that the control equipment is on, it powers up the other (slave) sockets. The load on the master socket can be anywhere between 20 VA and 500 VA, while the load on the slave sockets can be 60 VA to 1200 VA.

During the positive half cycle of the mains AC supply, diodes D4, D5, and D6 have a voltage drop of about 1.8 volts when current is drawn from the master socket. Diode D7 carries the current during negative half cycles. Capacitor C3, in series with diode D3, is connected across the diode combination of D4 through D6, in addition to diode D7 as well as resistor R10. Thus current pulses during positive half-cycles, charge up the capacitor to 1.8 volts via diode D3. This voltage is sufficient to hold transistor T2 in forward biased condition for about 200 ms even after the controlling load on the master socket is switched off.

When transistor T2 is 'on', transistor T1 gets forward biased and is switched on. This, in turn, triggers Triac 1, which then powers the slave loads. Capacitor C4 and resistor R9 form a snubber network to ensure that the triac turns off cleanly with an inductive load.




LED1 indicates that the unit is operating. Capacitor C1 and zener ZD1 are effectively in series across the mains. The resulting 15V pulses across ZD1 are rectified by diode D2 and smoothened by capacitor C2 to provide the necessary DC supply for the circuit around transistors T1 and T2. Resistor R3 is used to limit the switching-on surge current, while resistor R1 serves as a bleeder for rapidly discharging capacitor C1 when the unit is unplugged. LED1 glows whenever the unit is plugged into the mains. Diode D1, in anti-parallel to LED1, carries the current during the opposite half cycles.

Don't plug anything into the master or slave sockets without testing the unit. If possible, plug the unit into the mains via an earth leakage circuit breaker. The mains LED1 should glow and the slave LED2 should remain off. Now connect a table lamp to the master socket and switch it 'on'. The lamp should operate as usual. The slave LED should turn 'on' whenever the lamp plugged into slave socket is switched on. Both lamps should be at full brightness without any flicker. If so, the unit is working correctly and can be put into use.

***Note.*** 1. The device connected to the master socket must have its power switch on the primary side of the internal transformer. Some electronic equipment have the power switch on the secondary side and hence these devices continue to draw a small current from the mains even when switched off. Thus such devices, if connected as the master, will not control the slave units correctly.

2. Though this unit removes the power from the equipment being controlled, it doesn't provide isolation from the mains. So, before working inside any equipment connected to this unit, it must be unplugged from the socket.

# AUTO SHUT-OFF FOR CASSETTE PLAYERS AND AMPLIFIERS


ARTHUR LOUIS

Here are two simple, low-cost circuits that can be used to shut off the mains supply to any audio or video equipment (such as tape recorder, CD player, and amplifier). These circuits are helpful to those in the habit of falling asleep with their music system on.

The circuits will also protect the equipment from getting damaged due to high-voltage spikes whenever there is a resumption of power after a break. This is possible because the equipment will get switched off automatically under such conditions but will not get switched on automatically on resumption of mains supply.

The circuit in Fig. 1 can be used to shut off any cassette player that has a reliable auto-stop mechanism. Whenever switch S1 is pressed momentarily, it extends the supply to the step-down transformer of the tape recorder and charges capacitor C1 through diode D1. This, in turn, makes transistor T1 conduct and energise relay RL1 to provide a parallel path to switch S1, so that supply to the step-down transformer continues even when switch S1 is released.

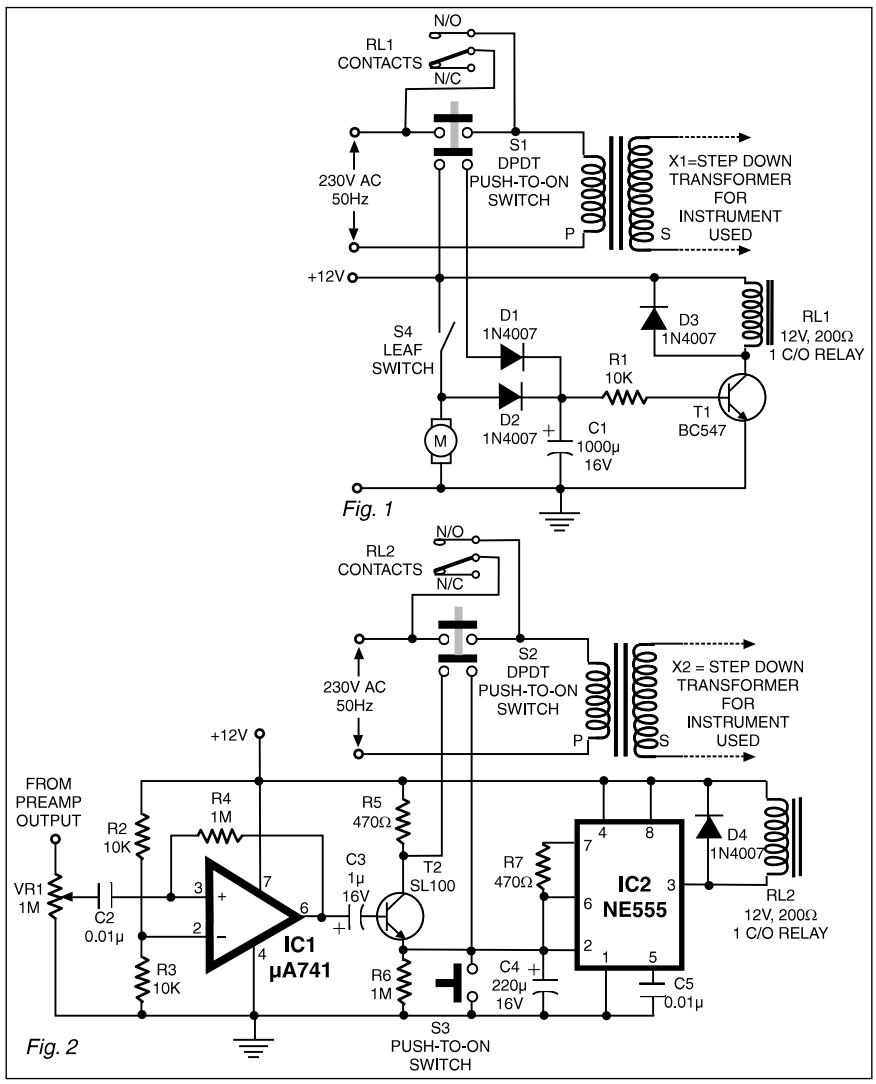
When any button on the cassette player is pressed, the capacitor charges through diode D2. This ensures conduction of transistor T1 and thus the continuity of operation of cassette player. However, whenever the auto-stop mechanism functions at the end of a tape, the leaf switch gets opened. This cuts the charging path for the capacitor and it starts discharging slowly. After about one minute, the relay opens and interrupts main power to the transformer. The time delay can be increased by increasing the value of capacitor C1.

If the appliance used is a two-in-one type (e.g. cassette player-cum-radio), just connect another diode in parallel with diodes D1 and D2 to provide an additional path for charging capacitor C1 via the tape-to-radio changeover switch, so that when radio is played the relay does not interrupt the power supply.

The other circuit, shown in Fig. 2, functions on the basis of the signal received from preamp of the appliance used. In this circuit, opamp µA741 is wired in inverting opamp configuration. It amplifies the signal received from the preamp. Timer NE555 is used to provide the necessary time delay of about one minute.

Preset VR1 is used to control the sensitivity of the circuit to differentiate between the noise and the signal. Resistor R4 offers feedback resistance to control the gain of the opamp. By increasing or decreasing the value of resistor R4, the gain can be increased or decreased, respectively. The preset time delay of timer NE555 (which is about one minute) can be increased by increasing the value of C4.

Initial energisation of relay RL2 and charging of capacitor C4 take place on depression of switch S3 in the same manner as charging of capacitor C1 (refer Fig. 1) on depression of switch S1. As a result, pins 2 and 6 of NE555 go high and the output of timer goes low to switch off mains supply from the relay to step-down transformer X2 of the appliance. Bleeder resistor R6 is used to discharge capacitor C4. Now if signals are received from the



preamplifier, these are amplified by 741 and fed to the base of transistor T2, which keeps capacitor C4 charged through resistor R5. When there is no signal, T2 will not conduct and the capacitor slowly discharges through R6. The output of 555 goes high to switch off the relay and thus the mains supply to transformer X2. Switch S2 can be depressed momentarily if the device needs to be manually switched off.

***Note.*** The 12V supply should be provided to the circuit from the equipment's power supply. Opamp 741 should be driven from the preamplifier of the gadget used, and not from its power amplifier output. Switches S1 and S2 are 2-pole push-to-on switches. These can also be fabricated from 2-pole on-off switches, which are widely used in cassette players, by removing the latch pin from them.❑

# CONDENSER MIC AUDIO AMPLIFIER


D. PRABAKARAN

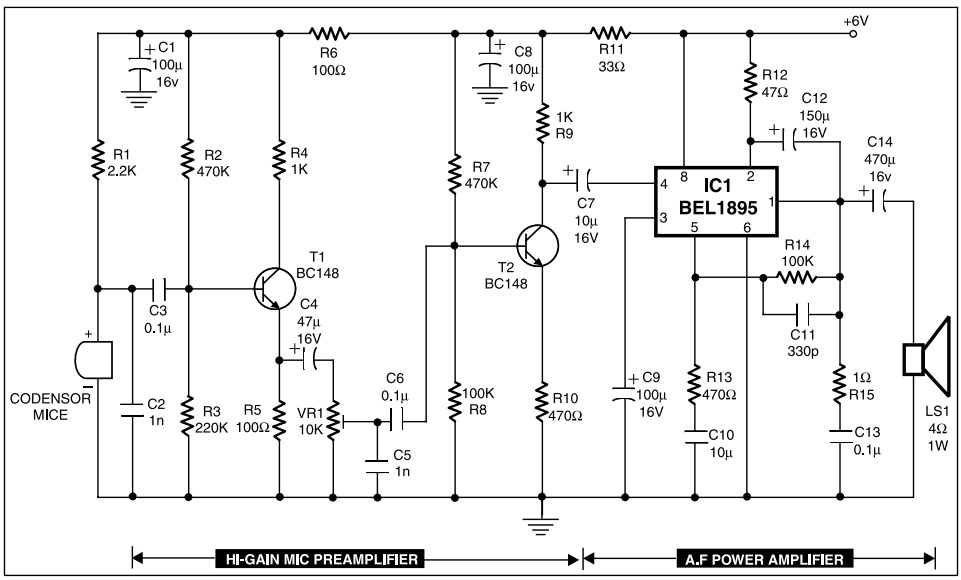
The compact, low-cost condenser mic audio amplifier described here provides good-quality audio of 0.5 watts at 4.5 volts. It can be used as part of intercoms, walkie-talkies, low-power transmitters, and packet radio receivers.

Transistors T1 and T2 form the mic preamplifier. Resistor R1 provides the necessary bias for the condenser mic while preset VR1 functions as gain control for varying its gain. In order to increase the audio power, the low-level audio output from the preamplifier stage is coupled via coupling capacitor C7 to the audio power amplifier built around BEL1895 IC.

BEL1895 is a monolithic audio power amplifier IC designed specifically for sensitive AM radio applications that delivers 1 watt into 4 ohms at 6V power supply voltage. It exhibits low distortion and noise and operates over 3V-9V supply voltage, which makes it ideal for battery operation. A turn-on pop reduction circuit prevents thud when the power supply is switched on.

Coupling capacitor C7 determines low-frequency response of the amplifier. Capacitor C9 acts as the ripple-rejection filter. Capacitor C13 couples the output available at pin 1 to the loudspeaker. R15-C13 combination acts as the damping circuit for output oscillations. Capacitor C12 provides the boot strapping function.

This circuit is suitable for low-power HAM radio transmitters to supply the necessary audio power for modulation. With simple modifications it can also be used in intercom circuits. ❑



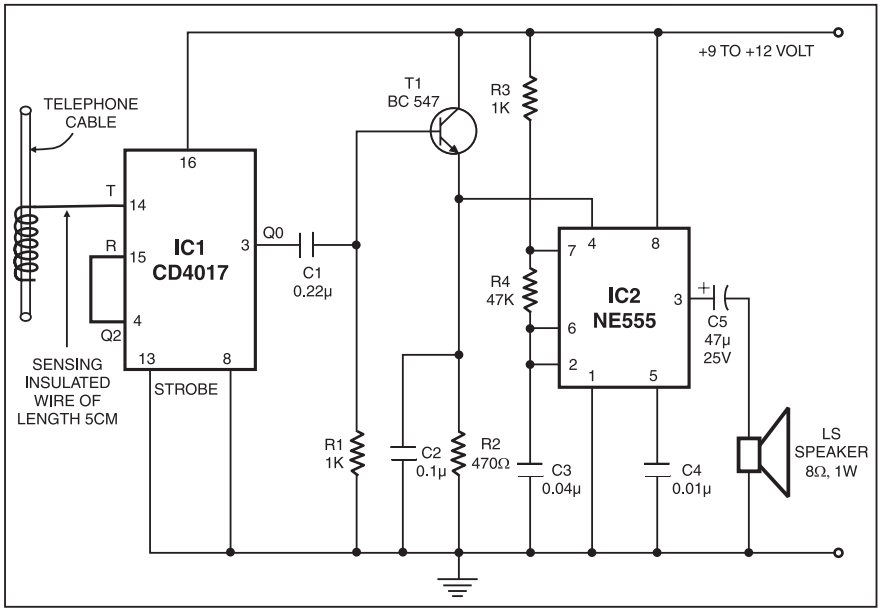
# CONTACTLESS RINGER FOR TELEPHONES


**PRADEEP G.**

Several circuits of a telephone extension ringer have earlier been published in EFY. The circuit presented here is distinct from these circuits in that it has no electrical contact with the telephone lines. It senses the induced fluctuating electric field of telephone lines when the phone rings.

The input is sensed by a 5-8cm long plastic insulated flexible wire that is wound 3-4 times on telephone cable. When the telephone rings, about 20Hz AC voltage is available on the telephone line, which causes field fluctuation up to a few centimetres outside the telephone cable also.

The 20Hz AC signal induced in the pick-up sensor is coupled to the clock pin of decade counter IC CD4017. The CD4017 is wired as a divide-by-two counter by connecting its pin 4 to reset pin 15.




As the input impedance of CMOS IC is extremely high, the induced electric field is sufficient to clock it. The output obtained at pin 3 of CD4017 is a 10Hz square wave (half of input 20Hz signals). This square wave signal is used to bias npn transistor BC547 in class-C mode of operation. The transistor conducts during the positive half cycle of square wave. The positive voltage (high) available at the emitter of the transistor pulls reset pin 4 of 555 timer IC high for the corresponding duration. As a result, the alarm is activated when the telephone rings.

When the handset is picked up or the telephone stops ringing, the transistor stops conduction. Then reset pin of IC2 goes low and the alarm is disabled.

# EXCLUSIVE-OR GATE APPLICATIONS


**ANAND TAMBOLI**

XOR gate is a derived logic gate that finds many applications in digital circuits. Here we have described use of XOR gate as controlled inverter, 9's BCD subtractor and up-/down-counter.

It can be seen from Fig. 1 and the accompanying truth table that XOR gate works as NOT (inverter) gate when its one input is held high, and as a buffer when the same input is pulled low. The common input (as shown in Fig. 2) can therefore be used as control input for XOR gates to behave as inverters or buffers. Here one of the inputs of each XOR gate are connected together to serve as the control signal, while the remaining inputs serve as input bits.

**TRUTH TABLE I**
**XOR Gate**

| A | B | Y |
|---|---|---|
| 0 | 0 | 0 |
| 0 | 1 | 1 |
| 1 | 0 | 1 |
| 1 | 1 | 0 |

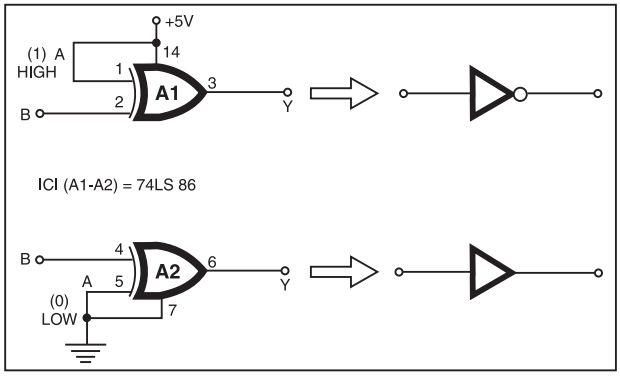



*Fig. 1: XOR gate as inverter and buffer*

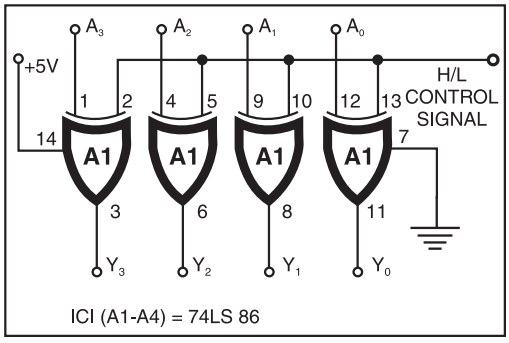



*Fig. 2: XOR gate as controlled inverter*

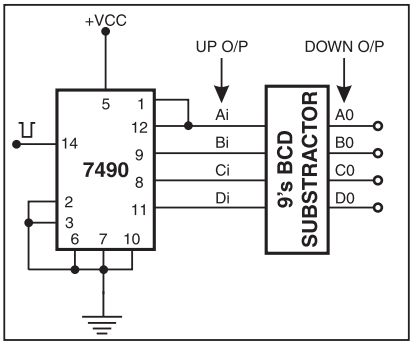



*Fig. 4: Up-/down-counter*

When control signal is zero, the output is same as the input (buffer mode). However, when control signal is held high, all bits are inverted. Thus the output is complement of the input. This complementing function is useful in subtraction circuits using 1's and 2's complements.

In another configuration of XOR gate shown in Fig. 3, the circuit works as a 9's BCD subtractor. Its output = 9 – input.

The circuit in Fig. 3 can be used as a down-counter when employed with 4-bit BCD counter IC 7490 as shown in Fig. 4. Truth table of this configuration is given alongside. As seen from the figure, this circuit can be built using a single XOR-IC (TTL-7486). The NOR gate used in the circuit is inevitable, but it can be replaced by a resistance-transistor logic (RTL) circuit or any other equivalent circuit.

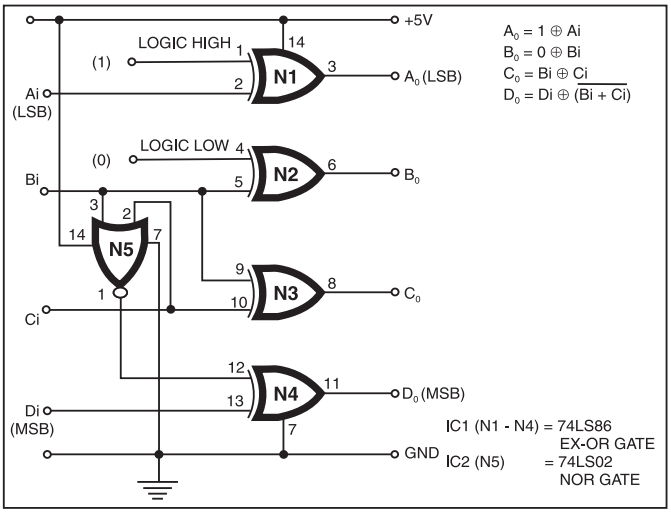



*Fig. 3: XOR gate as 9's BCD subtractor*

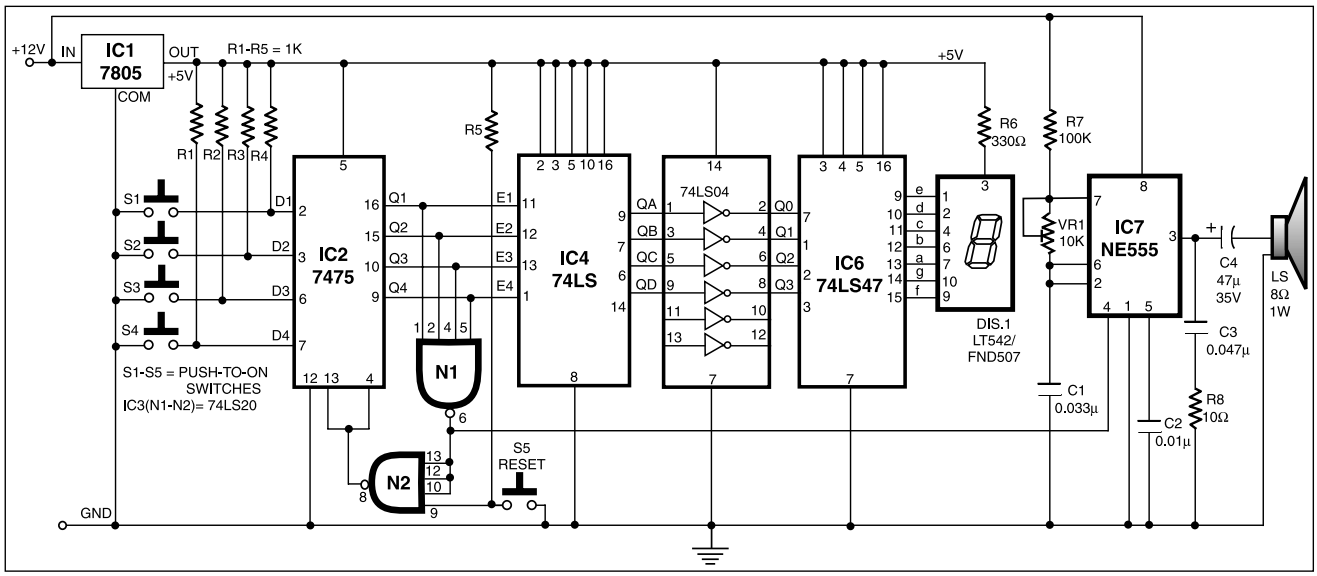
**TRUTH TABLE II**
**Down-Counter**

| DiCiBiAi | DEC. EQ. | D0C0B0A0 | DEC. EQ. |
|---|---|---|---|
| 0000 | 0 | 1001 | 9 |
| 0001 | 1 | 1000 | 8 |
| 0010 | 2 | 0111 | 7 |
| 0011 | 3 | 0110 | 6 |
| 0100 | 4 | 0101 | 5 |
| 0101 | 5 | 0100 | 4 |
| 0110 | 6 | 0011 | 3 |
| 0111 | 7 | 0010 | 2 |
| 1000 | 8 | 0001 | 1 |
| 1001 | 9 | 0000 | 0 |

# FASTEST FINGER FIRST INDICATOR


P. RAJESH BHAT

Quiz-type game shows are increasingly becoming popular on television these days. In such games, fastest finger first indicators (FFFIs) are used to test the player's reaction time. The player's designated number is displayed with an audio alarm when the player presses his entry button.




The circuit presented here determines as to which of the four contestants first pressed the button and locks out the remaining three entries. Simultaneously, an audio alarm and the correct decimal number display of the corresponding contestant are activated.

When a contestant presses his switch, the corresponding output of latch IC2 (7475) changes its logic state from 1 to 0. The combinational circuitry comprising dual 4-input NAND gates of IC3 (7420) locks out subsequent entries by producing the appropriate latch-disable signal.

Priority encoder IC4 (74147) encodes the active-low input condition into the corresponding binary coded decimal (BCD) number output. The outputs of IC4 after inversion by inverter gates inside hex inverter 74LS04 (IC5) are coupled to BCD-to-7-segment decoder/display driver IC6 (7447). The output of IC6 drives common-anode 7-segment LED display (DIS.1, FND507 or LT543).

The audio alarm generator comprises clock oscillator IC7 (555), whose output drives a loudspeaker. The oscillator frequency can be varied with the help of preset VR1. Logic 0 state at one of the outputs of IC2 produces logic 1 input condition at pin 4 of IC7, thereby enabling the audio oscillator.

IC7 needs +12V DC supply for sufficient alarm level. The remaining circuit operates on regulated +5V DC supply, which is obtained using IC1 (7805).

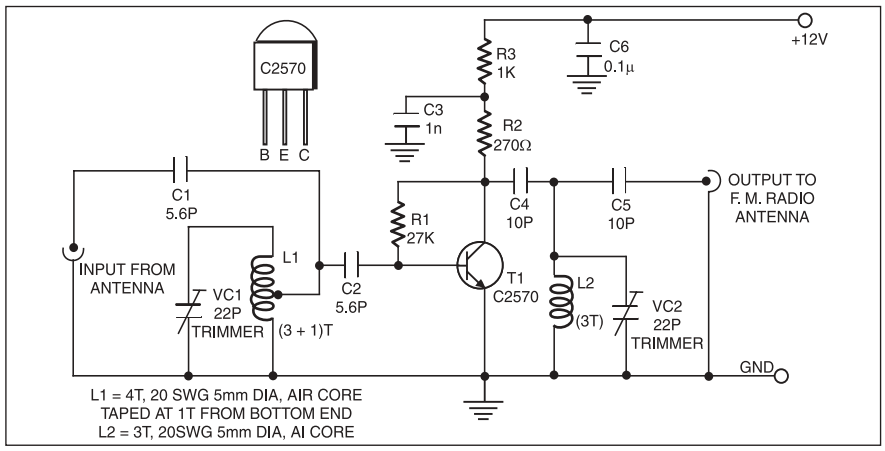
Once the organiser identifies the contestant who pressed the switch first, he disables the audio alarm and at the same time forces the digital display to '0' by pressing reset pushbutton S5.

With a slight modification, this circuit can accommodate more than four contestants. ❑

# FM BOOSTER

PRADEEP G.


Here is a low-cost circuit of an FM booster that can be used to listen to programmes from distant FM stations clearly. The circuit comprises a common-emitter tuned RF preamplifier wired around VHF/UHF transistor 2SC2570. (Only C2570 is annotated on the transistor body.)




Assemble the circuit on a good-quality PCB (preferably, glass-epoxy). Adjust input/output trimmers (VC1/VC2) for maximum gain.

Input coil L1 consists of four turns of 20SWG enamelled copper wire (slightly space wound) over 5mm diameter former. It is tapped at the first turn from ground lead side. Coil L2 is similar to L1, but has only three turns. Pin configuration of transistor 2SC2570 is shown in the figure.

# MUSIC-ON-HOLD FOR TELEPHONES


**SIBIN K. ZACHARIAH**

Here is a simple circuit for music-on-hold with automatic shut off facility. During telephone conversation if you are reminded of some urgent work, momentarily push switch S1 until red LED1 glows, keep the telephone handset on the cradle, and attend to the work on hand. A soft music is generated and passed into the telephone lines while the other-end subscriber holds. When you return, you can simply pick up the handset again and continue with the conversation.

The glowing of LED1, while the music is generated, indicates that the telephone is in hold position. As soon as the handset is picked up, LED1 is turned off and the music stops.

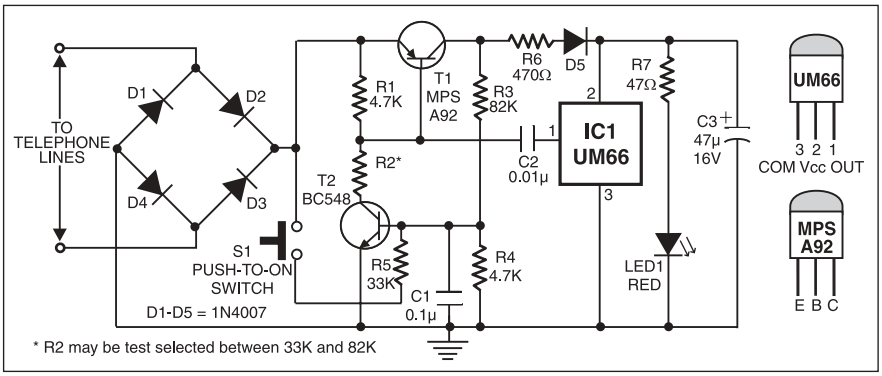
Normally, the voltage across telephone lines is about 50 volts. When we pick up the receiver (handset), it drops to about 9 volts. The minimum voltage required to activate this circuit is about 15 volts. If the voltage is less than 15 volts, the circuit automatically switches off. However, initially both transistors T1 and T2 are cut off. The transistor pair of T1 and T2 performs switching and latching action when switch S1 is momentarily pressed, provided the line voltage is more than 15 volts, i.e. when the handset is placed on the cradle.

Once the transistor pair of TI and T2 starts conducting, melody generator IC1 gets the supply and is activated. The music is coupled to the telephone lines via capacitor C2, resistor R1, and the bridge rectifier.

With the handset off-hook after a ring, momentary depression of switch S1 causes forward biasing of transistor T2. Meanwhile, if the handset is placed on the cradle, the current passing through R1 (connected across the emitter and base terminals of pnp transistor T1) develops enough voltage to forward bias transistor T1 and it starts conducting.

As a consequence, output voltage at the collector of transistor T1 sustains forward biasing of transistor T2, even if switch S1 is released. This latching action keeps both transistors T1 and T2 in conduction as long as the output of the bridge rectifier is greater than 15 volts.

If the handset is now lifted off-hook, the rectifier output drops to about 9 volts and hence latching action ceases and the circuit automatically switches off.



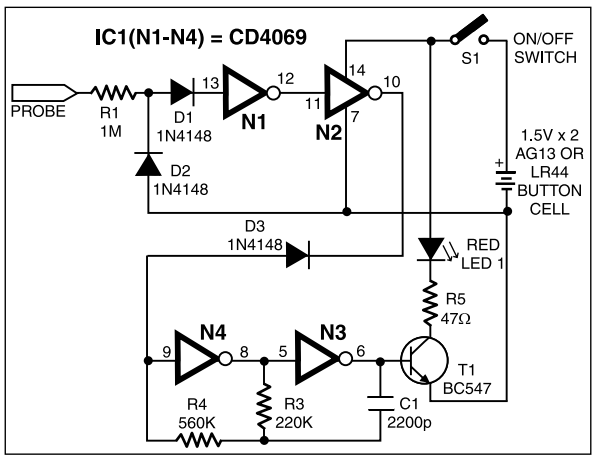

(***EFY lab note.*** *The value of resistor R2 determines the current through resistor R1 to develop adequate voltage (greater than 0.65 volts) for conduction of transistor T1. Hence it may be test selected between 33 kilo-ohms and 100 kilo-ohms to obtain instant latching.*)

The total cost of this circuit is around Rs 50.

# INVISIBLE BROKEN WIRE DETECTOR


K. UDHAYA KUMARAN, VU3GTH

Portable loads such as video cameras, halogen flood lights, electrical irons, hand drillers, grinders, and cutters are powered by connecting long 2- or 3-core cables to the mains plug. Due to prolonged usage, the power cord wires are subjected to mechanical strain and stress, which can lead to internal snapping of wires at any point. In such a case most people go for replacing the core/cable, as finding the exact location of a broken wire is difficult. In 3-core cables, it appears almost impossible to detect a broken wire and the point of break without physically disturbing all the three wires that are concealed in a PVC jacket.




The circuit presented here can easily and quickly detect a broken/faulty wire and its breakage point in 1-core, 2-core, and 3-core cables without physically disturbing wires. It is built using hex inverter CMOS CD4069. Gates N3 and N4 are used as a pulse generator that oscillates at around 1000 Hz in audio range. The frequency is determined by timing components comprising resistors R3 and R4, and capacitor C1. Gates N1 and N2 are used to sense the presence of 230V AC field around the live wire and buffer weak AC voltage picked from the test probe. The voltage at output pin 10 of gate N2 can enable or inhibit the oscillator circuit.

When the test probe is away from any high-voltage AC field, output pin 10 of gate N2 remains low. As a result, diode D3 conducts and inhibits the oscillator circuit from oscillating. Simultaneously, the output of gate N3 at pin 6 goes 'low' to cut off transistor T1. As a result, LED1 goes off. When the test probe is moved closer to 230V AC, 50Hz mains live wire, during every positive half-cycle, output pin 10 of gate N2 goes high.

Thus during every positive half-cycle of the mains frequency, the oscillator circuit is allowed to oscillate at around 1 kHz, making red LED (LED1) to blink. (Due to the persistence of vision, the LED appears to be glowing continuously.) This type of blinking reduces consumption of the current from button cells used for power supply.

A 3V DC supply is sufficient for powering the whole circuit. AG13 or LR44 type button cells, which are also used inside laser pointers or in LED-based continuity testers, can be used for the circuit. The circuit consumes 3 mA during the sensing of AC mains voltage.

For audio-visual indication, one may use a small buzzer (usually built inside quartz alarm time pieces) in parallel with one small (3mm) LCD in place of LED1 and resistor R5. In such a case, the current consumption of the circuit will be around 7 mA. Alternatively, one may use two 1.5V R6- or AA-type batteries. Using this gadget, one can also quickly detect fused small filament bulbs in serial loops powered by 230V AC mains.

The whole circuit can be accommodated in a small PVC pipe and used as a handy broken-wire detector. Before detecting broken faulty wires, take out any connected load and find out the faulty wire first by continuity method using any multimeter or continuity tester. Then connect 230V AC mains live wire at one end of the faulty wire, leaving the other end free. Connect neutral terminal of the mains AC to the remaining wires at one end. However, if any of the remaining wires is also found to be faulty, then both ends of these wires are connected to neutral. For single-wire testing, connecting neutral only to the live wire at one end is sufficient to detect the breakage point.

In this circuit, a 5cm (2-inch) long, thick, single-strand wire is used as the test probe. To detect the breakage point, turn on switch S1 and slowly move the test probe closer to the faulty wire, beginning with the input point of the live wire and proceeding towards its other end. LED1 starts glowing during the presence of AC voltage in faulty wire. When the breakage point is reached, LED1 immediately extinguishes due to the non-availability of mains AC voltage. The point where LED1 is turned off is the exact broken-wire point.

While testing a broken 3-core rounded cable wire, bend the probe's edge in the form of 'J' to increase its sensitivity and move the bent edge of the test probe closer over the cable. During testing avoid any strong electric field close to the circuit to avoid false detection. ❑

# IR REMOTE SWITCH

K.S. SANKAR

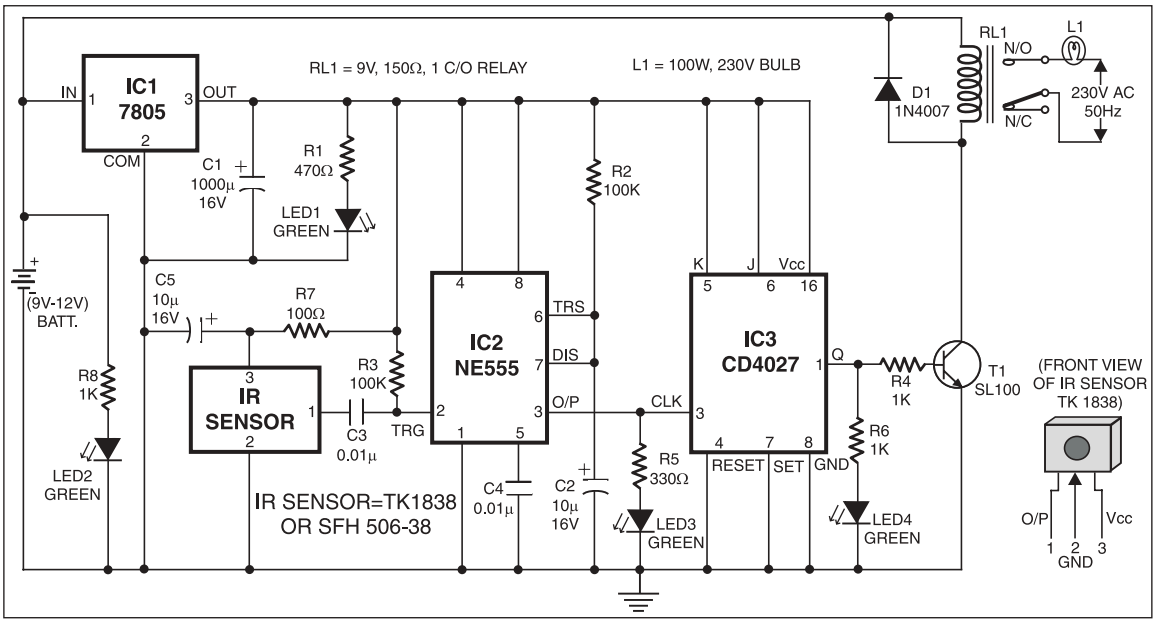
Imagine the convenience of selecting TV channels using your remote and then pointing the same remote to your switchboard to switch on/off the fan or the tubelight. Here is a simple circuit to remotely switch on/off any electrical device through a relay using the normal TV/VCR/VCP/VCD remote control unit. It works up to a distance of about 10 metres.

The circuit is built around a 3-pin IR IC receiver (Siemens SFH-506-38 or equivalent) that can detect 38kHz burst frequency generated by a TV remote. (This IR receiver module has been covered earlier in many projects published in EFY.)

The output pin of IR sensor goes low when it detects IR light, triggering the monostable (1-second) built around timer NE555. The output of the mono toggles the J-K flip flop, whose Q output drives the relay through SL100 npn transistor (T1).

LED2, LED3, and LED4 are used to display the status of each output stage during circuit operation. Back-EMF diode D5 is used for protection. Transistor T1 is configured as an open-collector output device to drive the relay rated at 12V DC.

The circuit draws the power from voltage regulator 7805. Capacitor C5 is soldered close to the IR sensor's pins to avoid noise and false triggering. Capacitor C3 and resistor R3 also avoid false triggering of monostable NE555. The monostable acts as a 1-second hysterisis unit to restrict the flip-flop from getting retriggered within one second. To activate any other 12V logic device, use the output across the relay coil terminals.



# LASER TORCH-BASED VOICE TRANSMITTER AND RECEIVER


**PRADEEP G.**

Using this circuit you can communicate with your neighbours wirelessly. Instead of RF signals, light from a laser torch is used as the carrier in the circuit. The laser torch can transmit light up to a distance of about 500 metres. The phototransistor of the receiver must be accurately oriented towards the laser beam from the torch. If there is any obstruction in the path of the laser beam, no sound will be heard from the receiver.

The transmitter circuit (Fig. 1) comprises condenser microphone transistor amplifier BC548 (T1) followed by an op-amp stage built around µA741 (IC1). The gain of the op-amp can be controlled with the help of 1-mega-ohm potmeter VR1. The AF output from IC1 is coupled to the base of transistor BD139 (T2), which, in turn, modulates the laser beam.

The transmitter uses 9V power supply. However, the 3-volt laser torch (after removal of its battery) can be directly connected to the circuit—with the body of the torch connected to the emitter of BD139 and the spring-loaded lead protruding from inside the torch to circuit ground.

The receiver circuit (Fig. 2) uses an npn phototransistor as the light sensor that is followed by a two-stage transistor preamplifier and LM386-based audio power amplifier. The receiver does not need any complicated alignment. Just keep the phototransistor oriented towards the remote transmitter's laser point and adjust the volume control for a clear sound.

To avoid 50Hz hum noise in the speaker, keep the phototransistor away from AC light sources such as bulbs. The reflected sunlight, however, does not cause any problem. But the sensor should not directly face the sun.

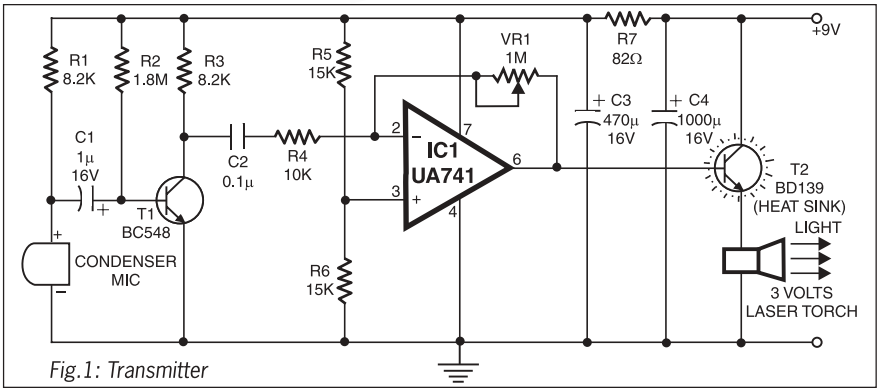



*Fig.1: Transmitter*

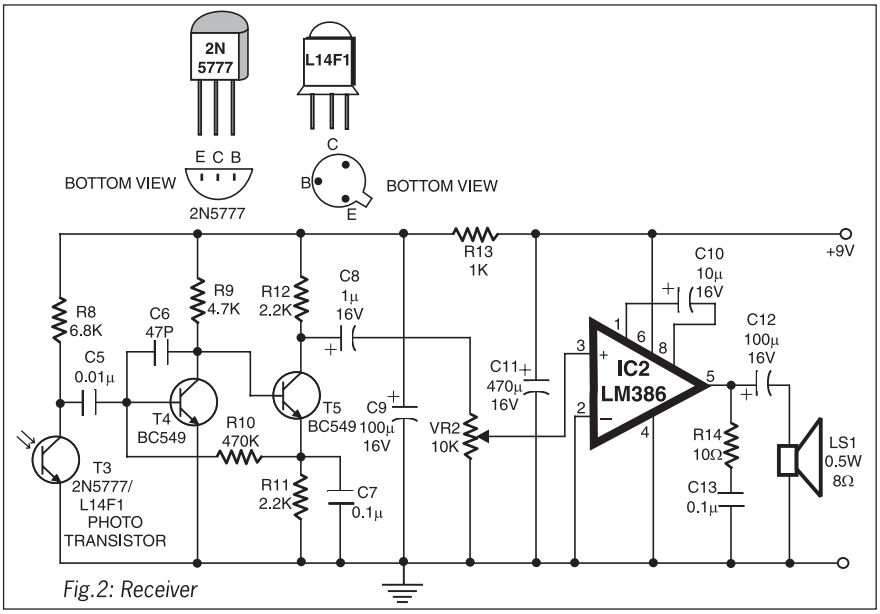


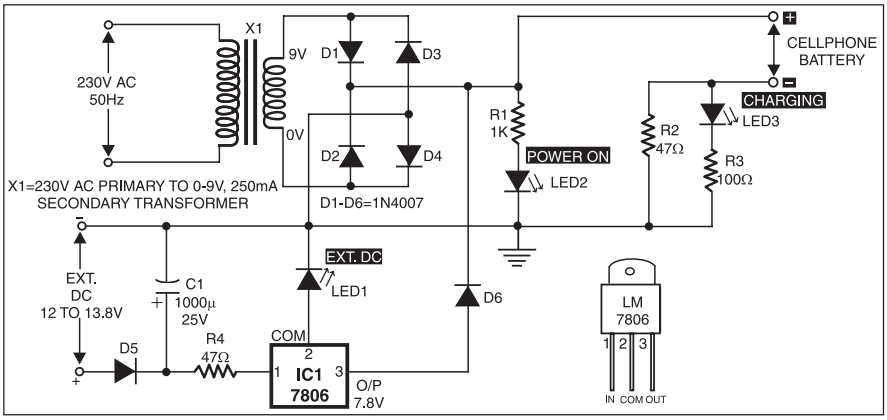

*Fig.2: Receiver*

# MOBILE PHONE BATTERY CHARGER


T.K. HAREENDRAN

Mobile phone chargers available in the market are quite expensive. The circuit presented here comes as a low-cost alternative to charge mobile telephones/battery packs with a rating of 7.2 volts, such as Nokia 6110/6150.

The 220-240V AC mains supply is downconverted to 9V AC by transformer X1. The transformer output is rectified by diodes D1 through D4 wired in bridge configuration and the positive DC supply is directly connected to the charger's output contact, while the negative terminal is connected through current limiting resistor R2.




LED2 works as a power indicator with resistor R1 serving as the current limiter and LED3 indicates the charging status. During the charging period, about 3 volts drop occurs across resistor R2, which turns on LED3 through resistor R3.

An external DC supply source (for instance, from a vehicle battery) can also be used to energise the charger, where resistor R4, after polarity protection diode D5, limits the input current to a safe value. The 3-terminal positive voltage regulator LM7806 (IC1) provides a constant voltage output of 7.8V DC since LED1 connected between the common terminal (pin 2) and ground rail of IC1 raises the output voltage to 7.8V DC. LED1 also serves as a power indicator for the external DC supply.

After constructing the circuit on a veroboard, enclose it in a suitable cabinet. A small heat sink is recommended for IC1.

# KNOCK ALARM


**PRADEEP G.**

This circuit (Fig. 1), used in conjunction with a thin piezoelectric plate, senses the vibration generated on knocking a surface (such as a door or a table) to activate the alarm. It uses readily-available, low-cost components and can also be used to safeguard motor vehicles. The piezoelectric plate is used as the sensor. It is the same as used in ordinary piezobuzzers and is easily available in the market.

The piezoelectric plate can convert any mechanical vibration into electrical variation. As it doesn't sense sound from a distance like a microphone, it avoids false triggering. The plate can be fixed on a door, cash box, cupboard, etc using adhesive. A 1-1.5m long, shielded wire is connected between the sensor plate and the input of the circuit. When someone knocks on the door, the piezoelectric sensor generates an electrical signal, which is amplified by transistors T1 through T3.

The amplified signal is rectified and filtered to produce a low-level DC voltage, which is further amplified by the remaining transistors. The final output from the collector of pnp transistor T6 is applied to reset pin 4 of 555 (IC1) that is wired as an astable multivibrator. Whenever the collector of transistor T6 goes high, the astable multivibrator activates to sound an alarm through the speaker. The value of resistor R12 is chosen between 220 and 680 ohms such that IC1 remains inactive in the absence of any perceptible knock.

When the circuit receives an input signal due to knocking, the alarm gets activated for about 10 seconds. This is the time that capacitor C5 connected between the emitter of transistor T4 and ground takes to discharge after a knock. The time delay can be changed by changing the value of capacitor C5. After about 10 seconds, the alarm is automatically reset.

The circuit operates off a 9V or a 12V battery eliminator. The proposed installation of the knock alarm is shown in Fig. 2.

This circuit costs around Rs 75.

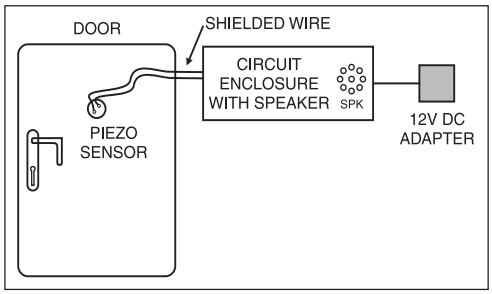



*Fig. 2: Proposed installation of knock alarm*

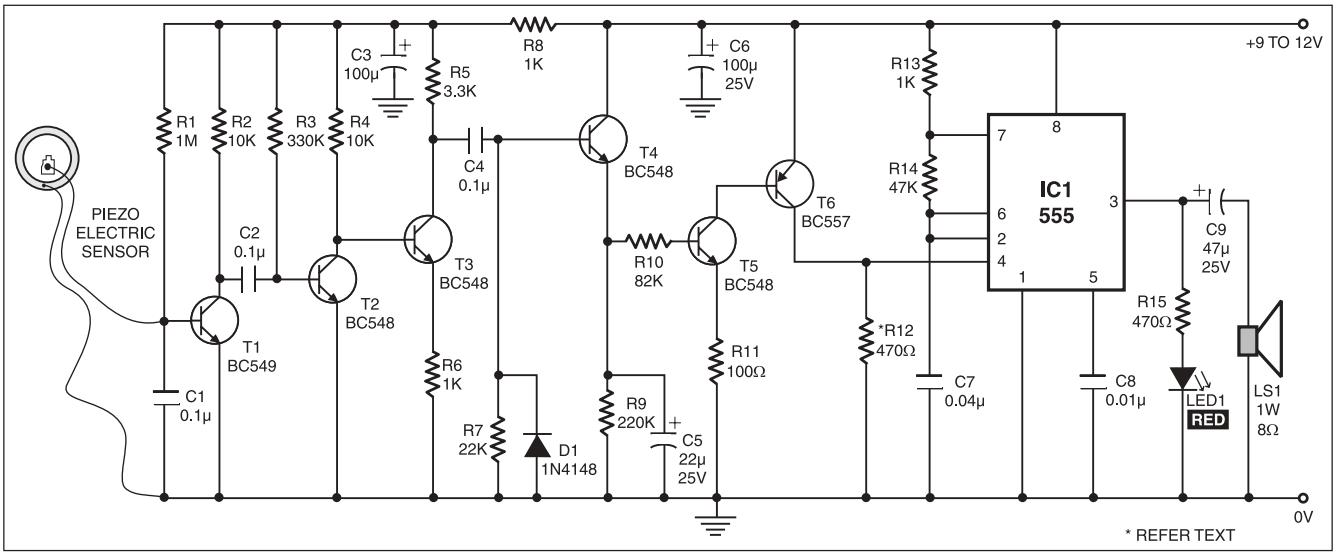
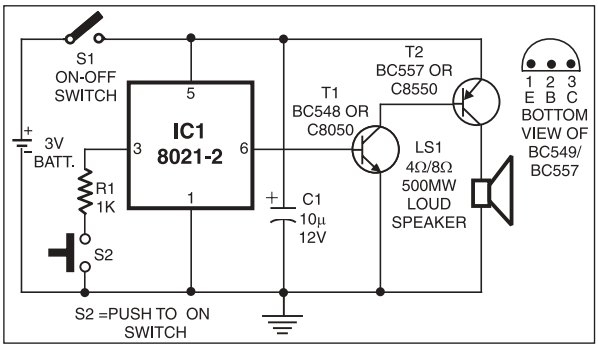



*Fig. 1: The circuit of knock alarm*

# DING-DONG BELL

PRAVEEN SHANKER

This simple and cost-effective door bell circuit is based on IC 8021-2 from Formox Semiconductors (Website address: fortech@mantramail.com). It is an 8-pin DIP IC whose only four pins, as shown in the circuit, have been used.




The IC has an in-built circuitry to produce ding-dong sound each time its pin 3 is pulled low. The sound is stored in the IC as bits, as in a ROM. The sound output from the IC can't however drive a speaker directly, as this puts strain on the device. Therefore a complementary-pair, two-transistor amplifier is used to amplify the sound to a fair level of audiblity. You may either use a piezo tweeter or an 8-ohm, 500mW speaker at the output.

During the standby period, the IC consumes nominal current of a few microamperes only. Therefore switch S1 may be kept closed. Each time switch S2 is pressed, ding dong sound is produced twice. If you try to press switch S2 a second time when the first ding dong sound is still being produed, it has no effect whatever and the two ding-dong bell sounds will be invariably produced.

The circuit costs no more than Rs 35 and the IC 8021-2 used in the circuit is readily available for around Rs 15 in the market.

# PC-BASED OSCILLOSCOPE

M.M. VIJAI ANAND

This circuit conditions different signals of frequency below 1 kHz and displays their waveforms on the PC's screen. The hardware is used to condition the input waveform and convert it to the digital format for interfacing to the PC. The software for acquiring the data into the PC and displaying the same on its screen is written in Turbo C.

The input waveform (limited to 5V peak-to-peak) is first applied to a full-wave rectifier comprising op-amps A1 and A2 of quad op-amp LM324 (IC4) and a zero-crossing detector built around LM3914 dot/bar display driver (IC8) simultaneously.

The full-wave rectifier rectifies the input signal such that the negative half cycle of the input signal is available in the positive side itself, so both the half cycles are read as positive when it is given as input to the ADC. During positive half cycle, diode D3 is on and diode D4 is off, and op-amps A1 and A2 act as inverters. Thus the output is a replica of the input. During the negative half cycle, diode D3 is off and diode D4 is on. With R2 = R3 = R4 = R5 = R6 = R = 330 ohms, the voltage (V) at inverting pin 2 of op-amp A1 is related to the input voltage (Vi) as follows:

$Vi/R + V/(2R) + V/R = 0$

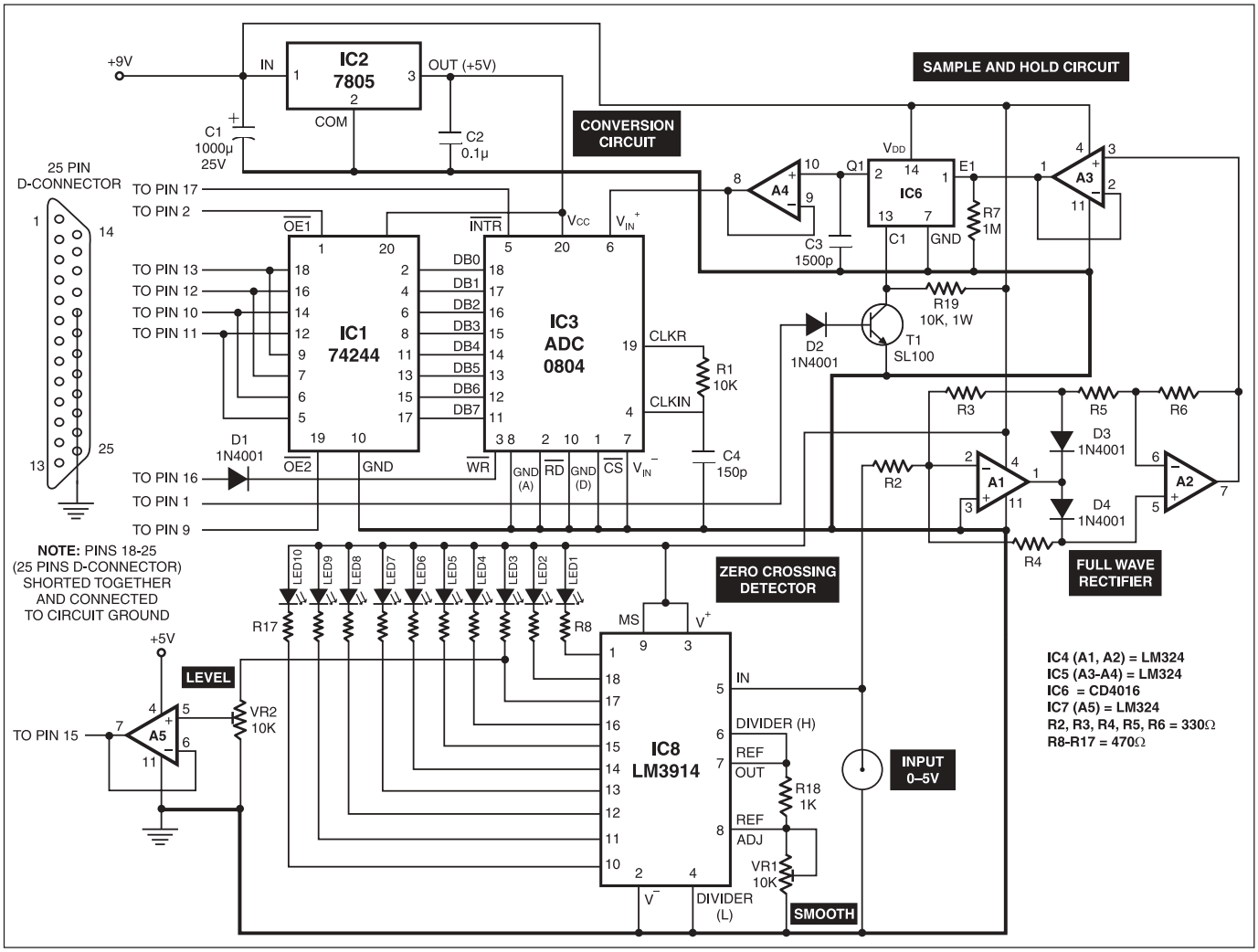
$V = -(2/3)Vi$

The final output voltage (Vo) at pin 7 of op-amp A2 is given by the following relationship:

$Vo = (1 + R/2R)(-2Vi/3) = -Vi$

As Vi is negative, the output voltage is positive.

The zero-crossing detector detects whether the cycle is positive or negative. It is the most critical part of the circuit and if it operates improperly, the symmetry of the analogue signal displayed in the PC monitor gets affected. At the zero-crossing instant when the input signal transits to negative side, the zero-crossing detector informs the PC by taking pin 15 of 25-pin 'D' connector of the parallel port high.



The input at pin 15 of 'D' connector goes low when the input signal transits to positive side. The zero-crossing detector communicates with the PC through bit D3 of the status port 379Hex.

The zero-crossing detector has been realised using LM3914 IC. You may adjust VR1 such that the last LED (LED10) goes off when the input signal transits negative side of the input waveform. The LM3914 itself rectifies the input signal and allows only positive half of the cycle.

The output from the full-wave rectifier is applied to the input of a sample-and-hold circuit comprising op-amps A3 and A4 of the LM324 (IC5), capacitor C3, transistor T1 (SL100), and analogue switch IC6 (CD4016). This circuit samples the input signal, i.e. it divides the waveform into a number of voltages or points and inputs each voltage level (with a delay) to the ADC for conversion into the digital format. Op-amps A3 and A4, along with a switch from IC CD4016 and a 1500pF capacitor with sampling time of 20 µs, are used as voltage followers/buffers.

When the base of transistor T1 is made low via strobe pin 1 (bit Do of I/O port 37A) of 25-pin D connector of the parallel port, the transistor stops conducting and the voltage at its collector goes high. The high voltage at the collector of transistor T1 closes the switch inside CD4016. As a consequence, the analogue input signal is applied to the capacitor, which charges towards the signal voltage.

When the switch is subsequently opened by applying a logic-high voltage from pin 1 of 'D' connector to the base of transistor T1, the capacitor retains the voltage with a loss of about 20 mV/sec and this voltage is given to input pin 6 of the ADC0804 (IC3) via buffer A4 for conversion to the digital format. When the number of sampling points in the input signal waveform is increased, the reconstructed waveform becomes more accurate.

The ADC0804 is compatible with microprocessors. It is a 20-pin IC that works with 5V supply. It converts the analogue input voltage to 8-bit digital output. The data bus is tristate buffered. With eight bits, the resolution is 5V/255 = 19.6 mV.

The inbuilt clock generator circuit produces a frequency of about 640 kHz with R1 = 10 kilo-ohms and C4 = 150 pF, which are the externally connected timing components. The conversion time obtained is approximately 100 µs. The functions of other pins are given below:

Pin 1 ($\overline{CS}$): This is active-low chip-select pin.

Pin 2 ($\overline{RD}$): This active-low pin enables the digital output buffers. When high, the 8-bit bus will be in Hi-Z state.

Pin 3 ($\overline{WR}$): This active-low pin is used to start the conversion.

Pin 9 (Vref/2): This is optional input pin. It is used only when the input signal range is small. When pin 9 is at 2V, the range is 0-4V, i.e. twice the voltage at pin 9.

Pin 6 (V+), Pin 7(V-): The actual input is the difference in voltages applied to these pins. The analogue input can range from 0 to 5V.

In this circuit, pins 1 and 2 are always made low, so the IC and the buses are always enabled. Pin 9 is made open, as we use analogue input with 0-5V range. Pin 7 is grounded.

Pin 5 ($\overline{INTR}$): This active-low pin indicates the end of conversion. It is connected to pin 17 (bit D3 of I/O port 37A) of 'D' connector. (Note that this bit is inverted.)

The start-of-conversion command via pin 16 of 'D' connector is applied to pin 3 of the ADC0804. Since we cannot read 8-bit digital data output from ADC through the 4-bit status port at a time, we divide it in two 4-bit parts and read. Hence the ADC data output is multiplexed through two 4-bit sections of octal buffers of IC1 (74244) with the help of output-enable signals from pins 2 and 9 of 'D' connector to pins 1 and 19 ($\overline{OE1}$ and $\overline{OE2}$, respectively) of IC1. The digital data output from IC1 is interfaced to the PC via pins 13 (D4), 12 (D5), 10 (D6), and 11 (D7) of status input port 379H of 'D' connector.

The circuit uses 9V and 5V regulated DC supply voltages as shown in the circuit diagram.

A PC printer port is an inexpensive platform for implementing low-frequency data acquisition projects. Each printer port consists of data, status, and control port addresses. These addresses are in sequential order; for example, if the data port address is 0x0378, the corresponding status port address is 0x0379 and the control port address is 0x037a. The port addresses for parallel ports are summarised below:

| Printer | Data port | Status port | Control port |
|---|---|---|---|
| LPT1 | 0x0378 | 0x0379 | 0x037a |
| LPT2 | 0x0278 | 0x0279 | 0x027a |
| LPT3 | 0x03bc | 0x03bd | 0x03be |

(***EFY Lab note***. For details of the parallel port pins, refer 'PC-based Dial Clock with Timer' project published in June 2002 issue of EFY.)

The software, written in C programming language, is user-friendly and easy-to-understand. It gets data from the developed hardware circuit and displays it in the graphical screen with some changes.

The C program includes two user-defined functions with the main function: graphics( ) and settings( ). The settings( ) function is used to adjust the voltage and time scale. The graphics( ) function is used to display the waveform on the screen. The sample control signal is used to close the switch in the sample-and-hold circuit, so the capacitor charges towards the analogue input voltage. After the sampling is over, the switch is opened using the same signal. Then the start-of-conversion control signal is given to start the conversion. The sampling time is approximately 20 µs and the conversion time is approximately 100 µs.

After the conversion is over, the 8-bit binary data for the specific voltage sample is available in the data bus of the ADC. Since the PC accepts only 4-bit data through the status port (379H), the 8-bit data must be split into two 4-bit data, which are accepted one after another. This is done by IC 74244, which is controlled by D0 and D7 bits of the data port. Then the two 4-bit data are packed to get the final 8-bit data.

The default BGI directory path is set as 'c:\tc\bgi'. The sampling time is decided by the 'for' loop that uses the samp value. The maximum delay produced should be greater than 20 µs, which is the maximum acquisition time of the capacitor. When the sample value is increased, the number of points on the input signal decreases and therefore the accuracy decreases. The time scale may be calibrated with 50Hz sine wave as reference.

This circuit costs around Rs 400.

## PROGRAM IN 'C' FOR PC OSCILLOSCOPE

```
/* PROGRAM FOR PC OSCILLOSCOPE */
/*by M.M.VIJAI ANAND B.E (E.E.E) C.I.T */
#include < dos.h >
#include < time.h >
#include < stdio.h >
#include  < graphics.h >
#include < string.h >
#include < stdlib.h >
#define data 0x0378
#define stat 0x0379
#define cont 0x037a
```

```
   void graphics(int[],int[]);       //FUNCTION TO DISPLAY GRAPH AND WAVEFORM

   void settings();                  //FUNCTION TO CHANGE THE SETTINGS(TIME AND VOLTAGE)

   long int samp = 7000;   //PLEASE CHECK THESE VALUES WHEN CONVERSION IS
                                     //        NOT PROPER( + -3000)

   float scale = 1;
   float times = 1;
   char again = 'a';
   int number = 800;

   void  main()
   {
   int i,j,k,a[1700],b[1700],c[1700],e[1700]; //This value 1700 is given when we want to compress the waveform

                  //done when we compress the time scale
   long int b1;
   clrscr();
   settings();
   while(again = = 'a')
   {
   for(i = 0;i < number;i + + )
   {
   outportb(cont,0x05^0x0b);
   outportb(cont,0x04^0x0b);
   e[i] = (inportb(stat)^0x80)&0x08;
   for(b1 = 0;b1 < = samp;b1 + + )                //sampling time is approximately 50 µsec
   {}

   outportb(cont,0x05^0x0b);
   outportb(cont,0x01^0x0b);
   outportb(cont,0x05^0x0b);
   while((inportb(cont)&0x08) = = 0x00) //converstion time is approximately 100 µsec
   {
   }

   outportb(data,0xf0);
   a[i] = (inportb(stat)^0x80)&0xf0;
   outportb(data,0x01);
   b[i] = (inportb(stat)^0x80)&0xf0;
   outportb(data,0xff);
   }
   for(i = 0;i < number;i + + )
   {
   a[i] = a[i] > > 4;
   c[i] = a[i] + b[i];
   c[i] = c[i]*0.0196*45/scale;
   }
   graphics(c,e);
   }

   }

   void graphics(int a1[],int e1[])
   {
   int gd = DETECT,gm,max,may,a,b,c,im,error,get = 5;

   char str[10],*st = "-",d;

   clrscr();
   initgraph(&gd,&gm,"c:\\tc\\bgi");                //use default bgi path
   error = graphresult();
   if(error ! =  grOk)
   {
   printf("Graphics error %s /n",grapherrormsg(error)); //reports error when

                  //graphics is not set
   printf("PRESS ANY KEY TO EXIT");
   getch();
   exit(1);
   }
   setbkcolor(LIGHTCYAN);
   setcolor(MAGENTA);

   settextstyle(0,0,2);
   max = getmaxx();
   may = getmaxy();
   may = may-20;
   outtextxy(0,may,"OSCILLOSCOPE");
   settextstyle(0,0,1);
   setcolor(BLUE);
   outtextxy(max-200,may + 2,"press 'a' for next sample");
   setcolor(BROWN);
   outtextxy(max-200,may + 10,"press any key to exit");
   setcolor(GREEN);
   settextstyle(0,0,0);
   for(a = 0;a < = may;a + = get)
   {line(0,a,800,a);
   }
   for(a = 0;a < = max;a + = get)
   {
   line(a,0,a,may);
   }
   setcolor(BROWN);
   setlinestyle(0,3,0);
   line(max/2,0,max/2,may);
   line(0,may/2,max,may/2);
   setcolor(RED);
   for(a = 0,c = 0;a < = max;a + = 50,c + + )
   {
   putpixel(a,may/2,BLUE);
   itoa((a-c*30)*times/2,str,10);
   outtextxy(a + 3,may/2 + 3,str);
   }
   for(b = (may/2)-45,c = 1;b > = 0;b- = 45,c + + )
   {
   itoa((c*scale),str,10);
   putpixel((max/2),b,BLUE);
   outtextxy((max/2) + 3,b + 3,str);
   }
   for(b = (may/2) + 45,c = 1;b < = 800;b + = 45,c + + )
   {
   itoa((c*scale),str,10);
   strcat(st,str);
   putpixel((max/2),b,BLUE);
   outtextxy((max/2) + 2,b + 2,st);
   strcpy(st,"-");
   }
   setcolor(MAGENTA);

   outtextxy(max-80,may/2 + 30,"time(msec)");
   settextstyle(0,1,0);
   outtextxy((max/2)-10,0,"volt(s)");

   setlinestyle(0,0,0);
   setcolor(RED);
   moveto(0,may/2);
   for(b = 0,c = 0;b < = number;c + = 1, b + + )
   {
   if(e1[b]! = 0x08)
   {
   lineto(c*times,((may/2)-a1[b]));
   }
   else
   {
   lineto(c*times,((may/2) + a1[b]));
   }
   }
   again  =  getch();
   closegraph();
   restorecrtmode();

   }

   void settings()
   {
   int gd = DETECT,gm,error,max,may,b;
   char c,d,e[2],m,*n;
   times = 1;
   initgraph(&gd,&gm,"c:\\tc\\bgi");        //default bgi directory path
   error = graphresult();
   if(error ! =  grOk)
   {
   printf("Graphics error %s /n",grapherrormsg(error));
   printf("PRESS ANY KEY TO EXIT");
   getch();
   exit(1);
   }
   max = getmaxx();
   setbkcolor(LIGHTBLUE);
   settextstyle(1,0,0);
   setcolor(BROWN);
   outtextxy(max/2-60,20,"SETTINGS");
   line(0,60,800,60);
   setcolor(MAGENTA);
   settextstyle(1,0,1);
   outtextxy((max/4)-70,80,"Voltage Scale");
   settextstyle(0,0,0);
   setcolor(BROWN);
   outtextxy(10,120,"DEFAULT      :");
   outtextxy(10,120,"              1 unit  =  1 volt");
   setcolor(RED);
   outtextxy(10,170,"TYPE 'C' TO CHANGE AND 'D' TO DEFAULT");
   c = getch();
   if(c = = 'c')
   {
   outtextxy(10,200,"TYPE 1  for   1 unit  =  2 volt");
   outtextxy(10,240,"TYPE 2  for   1 unit  =  4 volt");
   outtextxy(10,300,"TYPE 3  for   user defined");
   switch(getch())
   {
    case '1' :
              { scale = 2;
               break;
              }
    case '2' :
              {scale  =  4;
              break;
              }
    case '3' :
    {
     outtextxy(10,340,"TYPE VALUES FROM 1 TO 9 (minimize) or  m to (magnify)");
    d = getch();
    if(d = = 'm')
      {
       outtextxy(10,360,"TYPE a (1 unit  =  0.5 volt) or b (1 unit  =  0.25 volt)");
       switch(getch())
              {
               case 'a':
                  {
                  scale = 0.5;
                  break;
                  }
               case 'b':
                  {
                  scale = 0.25;
                  break;
                  }
              }
      }
    else
    {  e[0] = '0';
      e[1] =  '0';
      e[2] = d;
    scale = atoi(e);
    break;
    }
    }
    }
   }
   setcolor(BROWN);
   outtextxy(10,380,"TYPE C TO CHANGE TIME SETTINGS");
   m = getch();
   if( m =  = 'c')
   {
   cleardevice();
   outtextxy(10,20,"X AXIS 1 unit =  10msec CHANGE TO x(10msec)");
   outtextxy(10,40,"TYPE 'a' IF x IS (2 to 9) ,'b' IF x IS (10 to 99) AND 'c' IF x IS (.5 TO .9)");
   switch(getch())
      {
      case 'a':
        outtextxy(10,60,"x value is ....");
        n[0] = getch();
        times = atoi(n);
        itoa(times,n,10);
        outtextxy(10,70,n);
        break;
      case 'b':
        outtextxy(10,60,"x value is ....");
        n[0] = getch();
        n[1] = getch();
        times = atoi(n);
        itoa(times,n,10);
        outtextxy(10,70,n);
        break;

      case 'c':
              outtextxy(10,60,"x value is...");
              getch();
              n[0] = getch();
              times = atoi(n)*0.1;
              outtextxy(10,70,"scale decremented");
              break;

   }
   number = 800;
   if(times < 1)
   {number = number/times;
   }
   getch();
   }
   closegraph();
   restorecrtmode();
   }
```

# 9-LINE TELEPHONE SHARER


DHURJATI SINHA

This circuit is able to handle nine independent telephones (using a single telephone line pair) located at nine different locations, say, up to a distance of 100m from each other, for receiving and making outgoing calls, while maintaining conversation secrecy. This circuit is useful when a single telephone line is to be shared by more members residing in different rooms/apartments.

Normally, if one connects nine phones in parallel, ring signals are heard in all the nine telephones (it is also possible that the phones will not work due to higher load), and out of nine persons eight will find that the call is not for them. Further, one can overhear others' conversation, which is not desirable. To overcome these problems, the circuit given here proves beneficial, as the ring is heard only in the desired extension, say, extension number '1'.

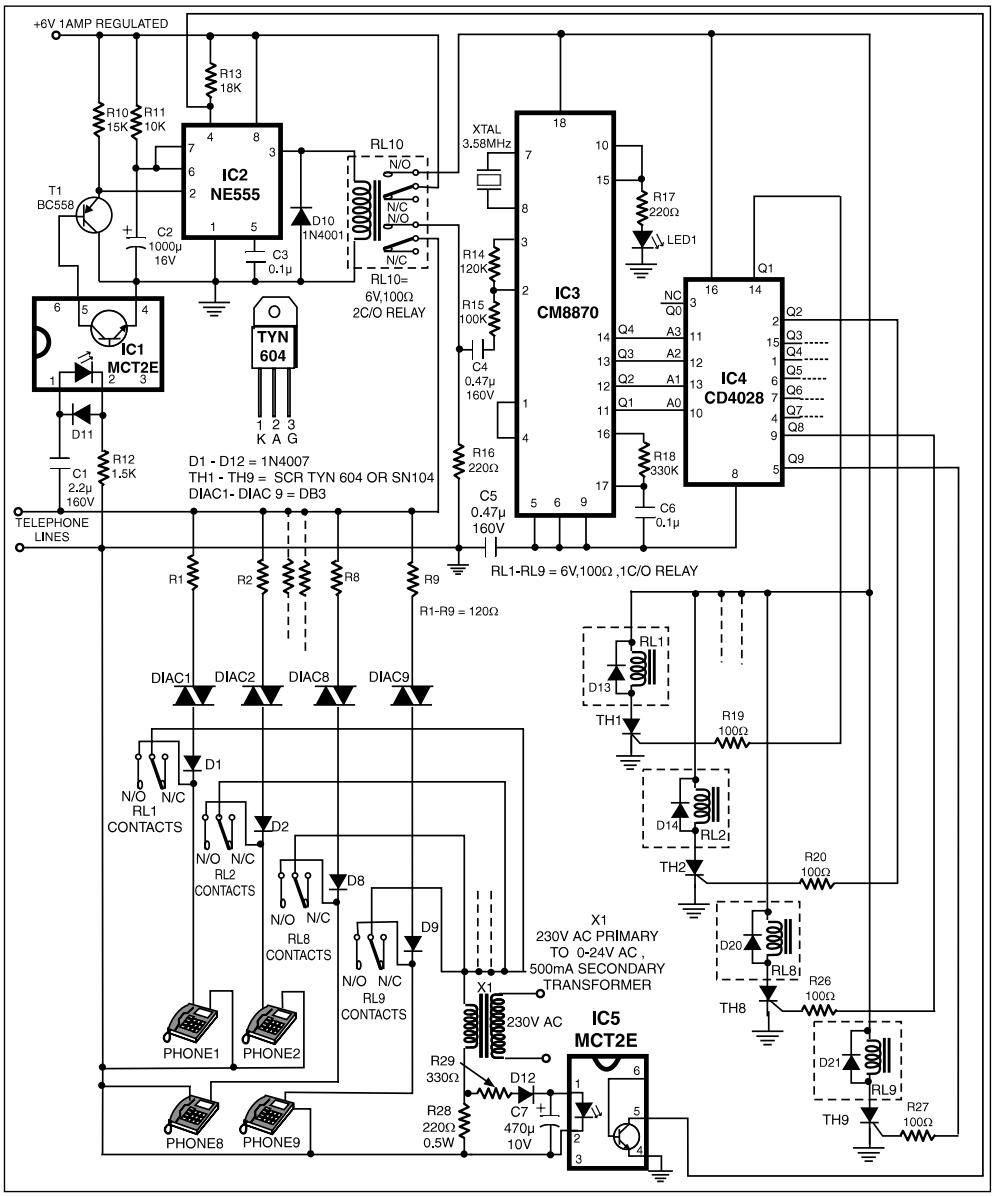
For making use of this facility, the calling subscriber is required to initially dial the normal phone number of the called subscriber. When the call is established, no ring-back tone is heard by the calling party. The calling subscriber has then to press the asterik (*) button on the telephone to activate the tone mode (if the phone normally works in dial mode) and dial extension number, say, '1', within 10 seconds. (In case the calling subscriber fails to dial the required extension number within 10 seconds, the line will be disconnected automatically.) Also, if the dialed extension phone is not lifted within 10 seconds, the ring-back tone will cease.

The ring signal on the main phone line is detected by opto-coupler MCT-2E (IC1), which in turn activates the 10-second 'on timer', formed by IC2 (555), and energises relay RL10 (6V, 100-ohm, 2 C/O). One of the 'N/O' contacts of the relay has been used to connect +6V rail to the processing circuitry and the other has been used to provide 220-ohm loop resistance to de-energise the ringer relay in telephone exchange, to cut off the ring.

When the caller dials the extension number (say, '1') in tone mode, tone receiver CM8870 (IC3) outputs code '0001', which is fed to the 4-bit BCD-to-10 line decimal decoder IC4 (CD4028). The output of IC4 at its output pin 14 (Q1) goes high and switches on the SCR (TH-1) and associated relay RL1. Relay RL1, in turn, connects, via its N/O contacts, the 50Hz extension ring signal, derived from the 230V AC mains, to the line of telephone '1'. This ring signal is available to telephone '1' only, because half of the signal is blocked by diode D1 and DIAC1 (which do not conduct below 35 volts).

As soon as phone '1' is lifted, the ring current increases and voltage drop across R28 (220-ohm, 1/2W resistor) increases and operates opto-coupler IC5 (MCT-2E). This in turn resets timer IC2 causing:

(a) interruption of the power supply for processing circuitry as well as the ring



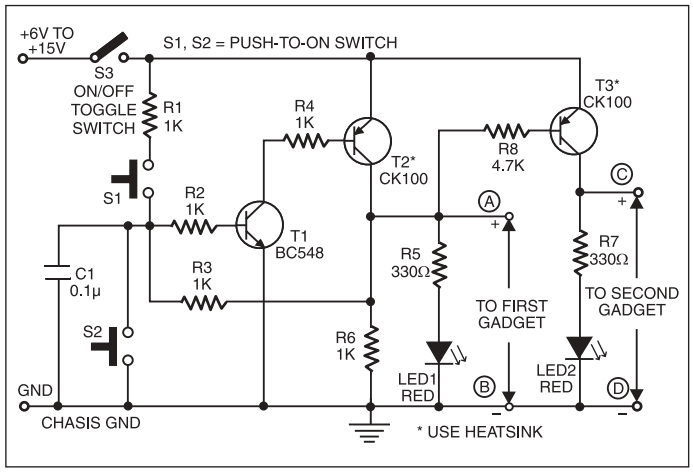
# SOLIDSTATE SWITCH FOR DC-OPERATED GADGETS


**PRAVEEN SHANKER**

This solidstate DC switch can be assembled using just three transistors and some passive components. It can be used to switch on one gadget while switching off the second gadget with momentary operation of switch. To reverse the operation, you just have to momentarily depress another switch.

The circuit operates over 6V-15V DC supply voltage. It uses positive feedback from transistor T2 to transistor T1 to keep this transistor pair in latched state (on/off), while the state of the third transistor stage is the complement of transistor T2's conduction state.

Initially when switch S3 is closed, both transistors T1 and T2 are off, as no forward bias is available to these, while the base of transistor T3 is effectively grounded via resistors R8 and R6 (shunted by the load of the first gadget). As a result, transistor T3 is forward biased and gadget 2 gets the supply. This is indicated by glowing of LED2.




When switch S1 is momentarily depressed, T1 gets the base drive and it grounds the base of transistor T2 via resistor R4. Hence transistor T2 (pnp) also conducts. The positive voltage available at the collector of transistor T2 is fed back to the base of transistor T1 via resistor R3. Hence a latch is formed and transistor T2 (as also transistor T1) continues to conduct, which activates gadget 1 and LED1 glows.

Conduction of transistor T2 causes its collector to be pulled towards positive rail. Since the collector of T2 is connected to the base of pnp transistor T3, it causes transistor T3 to cut off, switching off the supply to gadget 2) as well as extinguishing LED2. This status is maintained until switch S2 is momentarily pressed. Depression of switch S2 effectively grounds the base of transistor T1, which cuts off and thus virtually opens the base-emitter circuit of transistor T2 and thus cutting it off. This is the same condition as was obtained initially. This condition can be reversed by momentarily pressing switch S1 as explained earlier.

***EFY lab note**. During testing, it was noticed that for proper operation of the circuit, gadget 1 must draw a current of more than 100 mA (i.e. the resistance of gadget 1 must be less than 220 ohms) to sustain the latched 'on' state. But this stipulation is not applicable for gadget 2. A maximum current of 275 mA could be drawn by any gadget.*

The total cost of this circuit is around Rs 30.

# ELECTRONIC SECURITY SYSTEM


**K. BHARATHAN**

This reliable and easy-to-operate electronic security system can be used in banks, factories, commercial establishments, houses, etc.

The system comprises a monitoring system and several sensing zones. Each sensing zone is provided with a closed-loop switch known as sense switch. Sense switches are fixed on the doors of premises under security and connected to the monitoring system. As long as the doors are closed, sense switches are also closed. The monitoring system can be installed at a convenient central place for easy operation.

Fig. 1 shows the monitoring circuit only for zone 1 along with the common alarm circuit. For other zones, the monitoring circuit is identical, with only the prefixes of components changing as per zone number. Encircled points A, B, and C of each zone monitoring circuit need to be joined to the corresponding points of the alarm circuit (upper half of Fig. 1).

When zone 1 sensing switch S11, zone on/off slide switch S12, and system on/off switch S1 are all on, pnp transistor T12 reverse biases to go in cut-off condition, with its collector at around 0 volt. When the door fitted with sensor switch S11 is opened, transistor T12 gets forward biased and it conducts. Its collector voltage goes high, which forward biases transistor T10 via resistor R10 to turn it on. (Capacitor C10 serves as a filter capacitor.) As a result, the collector voltage of transistor T10 falls to forward bias transistor T11, which conducts and its collector voltage is sustained at a high level. Under this latched condition, sensor switch S11 and the state of transistor T12 have no effect. In this state, red LED11 of the zone remains lit.

Simultaneously, the high-level voltage from the collector of transistor T11 via diode D10 is applied to $V_{DD}$ pin 5 of siren sound generator IC1 (UM3561) whose pin 2 is grounded. Resistor R3 connected across pins 7 and 8 of IC1 determines the frequency of the in-built oscillator. As a result, IC1 starts generating the audio signal output at pin 3. The output voltage from IC1 is further amplified by Darlington pair of transistors T1 and T2. The amplified output of the Darlington pair drives the loudspeaker whose output volume can be controlled by potentiometer VR1. Capacitor C1 serves as a filter capacitor.

You can alter the alarm sound as desired by changing the connections of IC1 as shown in the table.

The circuit continues to sound the alarm until zone door is closed (to close switch S11) and the reset switch is pressed momentarily (which causes transistor T10 to cut off, returning the circuit to its initial state).

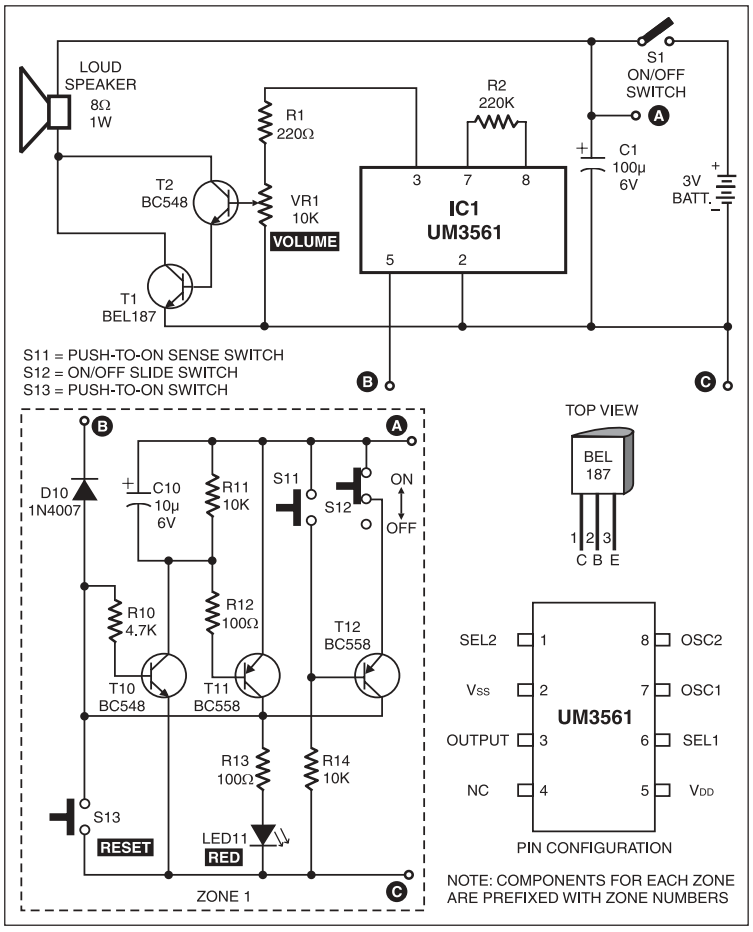



*Fig. 1: Monitoring circuit along with the alarm circuit*

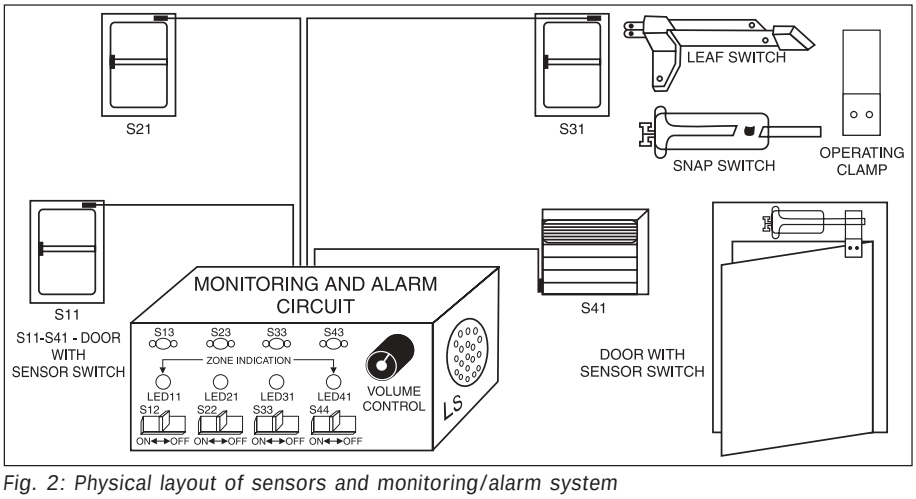



*Fig. 2: Physical layout of sensors and monitoring/alarm system*

The system operates off a 3V DC battery or recharging battery with charging circuit or battery eliminator. If desired, more operating zones can be added.

| Alarm sound | Circuit connections | |
|---|---|---|
| | IC pin 1 connected to | IC pin 6 connected to |
| Police siren | NC | NC |
| Ambulance siren | NC | $V_{DD}$ |
| Fire engine Sound | NC | $V_{SS}$ |
| Machinegun sound | $V_{SS}$ | NC |

***Note.*** *NC indicates no connection*

Initially keep the monitoring system switch S1 off. Keep all the zone doors fixed with sensing switches S11, S21, S31, S41, etc closed. This keeps the sensing switches for respective zones in closed position. Also keep zone slide switches S12, S22, S32, S42, etc in 'on' position. This puts the system in operation, guarding all the zone doors.

Now, if the door of a particular zone is opened, the monitoring system sounds an audible alarm and the LED corresponding to the zone glows to indicate that the door of the zone is open. The alarm and the LED indication will continue even after that particular door with the sensing switch is immediately closed, or even if that switch is removed/damaged or connecting wire is cut open.

Any particular zone in the monitoring system can be put to operation or out of operation by switching on or switching off the corresponding slide switch in the monitoring system.

The circuit for monitoring four zones costs around Rs 400.

# HOUSE SECURITY SYSTEM


MALAY BANERJEE

Here is a low-cost, invisible laser circuit to protect your house from thieves or trespassers. A laser pointer torch, which is easily available in the market, can be used to operate this device.

The block diagram of the unit shown in Fig. 1 depicts the overall arrangement for providing security to a house. A laser torch powered by 3V power-supply is used for generating a laser beam. A combination of plain mirrors M1 through M6 is used to direct the laser beam around the house to form a net. The laser beam is directed to finally fall on an LDR that forms part of the receiver unit as shown in Fig. 2. Any interruption of the beam by a thief/trespasser will result into energisation of the alarm. The 3V power-supply circuit is a conventional full-wave rectifier-filter circuit. Any alarm unit that operates on 230V AC can be connected at the output.

The receiver unit comprises two identical step-down transformers (X1 and X2), two 6V relays (RL1 and RL2), an LDR, a transistor, and a few other passive components. When switches S1 and S2 are activated, transformer X1, followed by a full-wave rectifier and smoothing capacitor C1, drives relay RL1 through the laser switch.

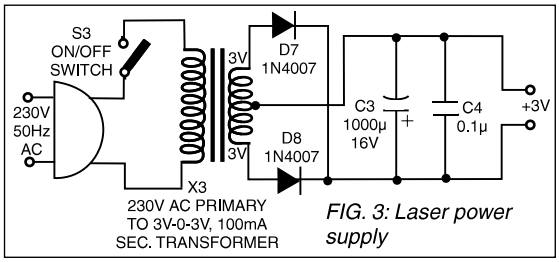



FIG. 3: Laser power supply

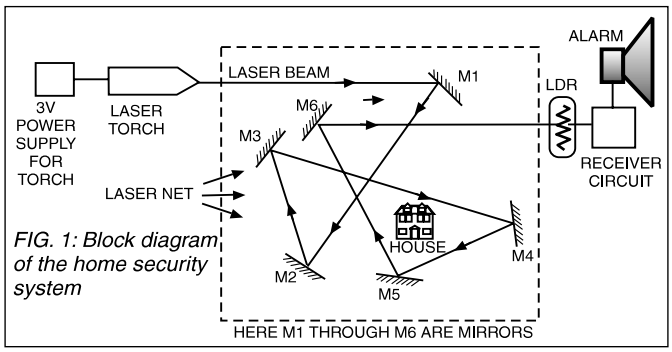



FIG. 1: Block diagram of the home security system

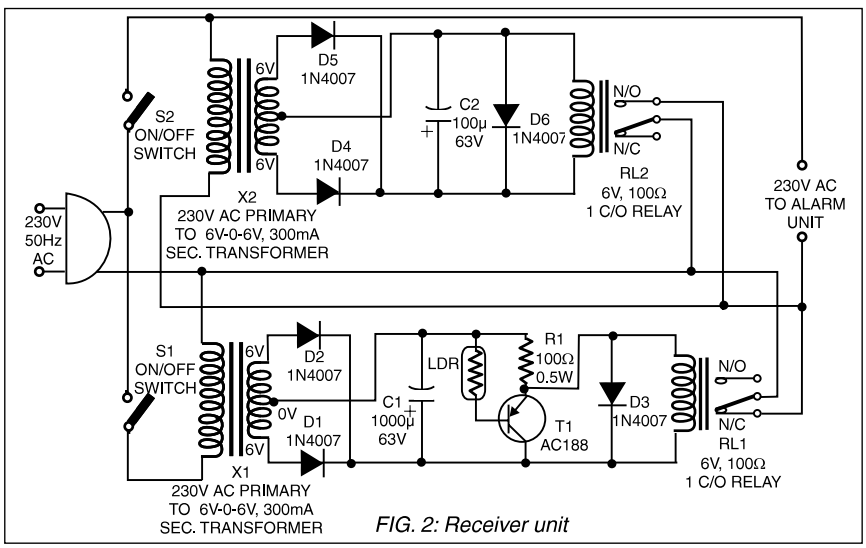



FIG. 2: Receiver unit

When the torch is switched on, the pointed laser beam is reflected from a definite point/place on the periphery of the house. Making use of a set of properly oriented mirrors one can form an invisible net of laser rays as shown in the block diagram. The final ray should fall on LDR of the circuit.

***Note.*** LDR should be kept in a long pipe to protect it from other sources of light, and its total distance from the source may be kept limited to 500 metres. The total cost of the circuit, including the laser torch, is Rs 400 or less. ❑

# SONG NUMBER DISPLAY


**PRABHASH K.P.**

Here's a circuit to display the song number in an audio system for quick reference to songs. It also serves the purpose of an extra visual indicator in modern audio systems.

When the power is switched on, the power-on-reset circuit comprising 3.3k resistor R20 and 1µF, 25V capacitor C6 resets the counters, showing '00' in the display. One can also reset the display to zero at any time by pressing reset switch S1.

When the first song starts playing, the output pins of IC1 (KA2281) go low and capacitor C5 starts charging. This forward biases transistor T1 and hence the input to IC3 at pin 1 goes to high state. As a result, the output of the counter goes to the next state, showing 01 on the display. The counter remains in this state until the song is completed.

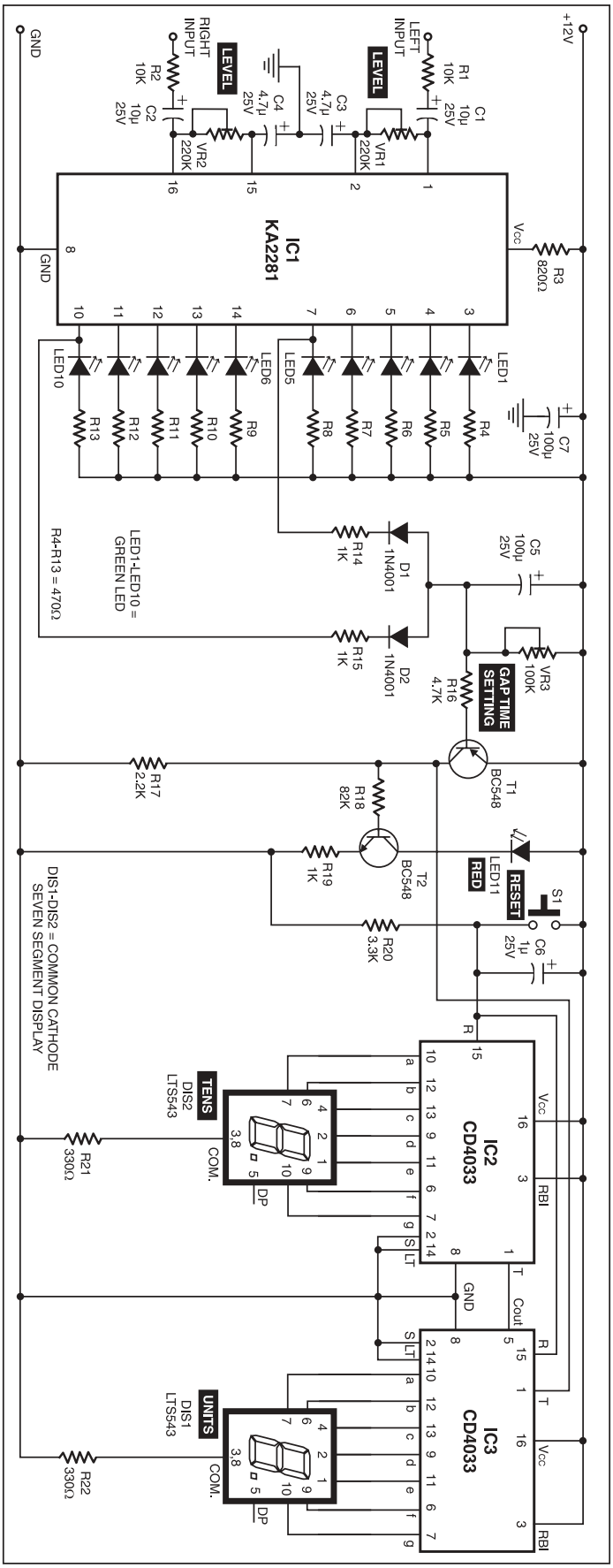
During the time gap before the next song starts playing, capacitor C5 discharges. After discharging of capacitor C5, the input to IC3 becomes low again. When the song starts, the process described above is repeated and the display shows 02. You can adjust VR3 to change the time gap setting. This must be set such that the circuit doesn't respond to short gaps, if any, within a song and responds only to long gaps between different songs.

Transistor T2 helps in gap-delay adjustment. The intensity of LED11 diminishes when a song is completed and the counter is ready to accept the next pulse.

Connect the input to the preamp output or equaliser output of the audio system. Adjust VR1 and VR2 to get the correct audio-level indication. If you are already using KA2281 for audio-level indication, just connect diodes D1 and D2 as shown in this circuit.

Note that the counter counts the songs by detecting the gaps. Therefore any long gap within a song may cause false triggering and the display will also be incremented. However, as this is very unlikely to happen, the circuit shows the correct song number almost all the time.

The circuit costs around Rs 100.



# SPELLER EFFECT SIGN DISPLAY


VIJAYA KUMAR P.

The circuit described here uses low-cost and easily available IC CD4017 to produce a speller type light display. In such displays, each letter of the sign sequentially lights up, one after the other, until all letters are glowing. After a few seconds, the letters switch off and the cycle repeats. This circuit provides a maximum of nine channels and therefore can be used to spell a word or sign having up to nine characters.

Timer IC1 (555) is configured in astable mode to produce clock signal for triggering IC2 (CD4017). Speed of switching on the display can be controlled by varying preset VR1.

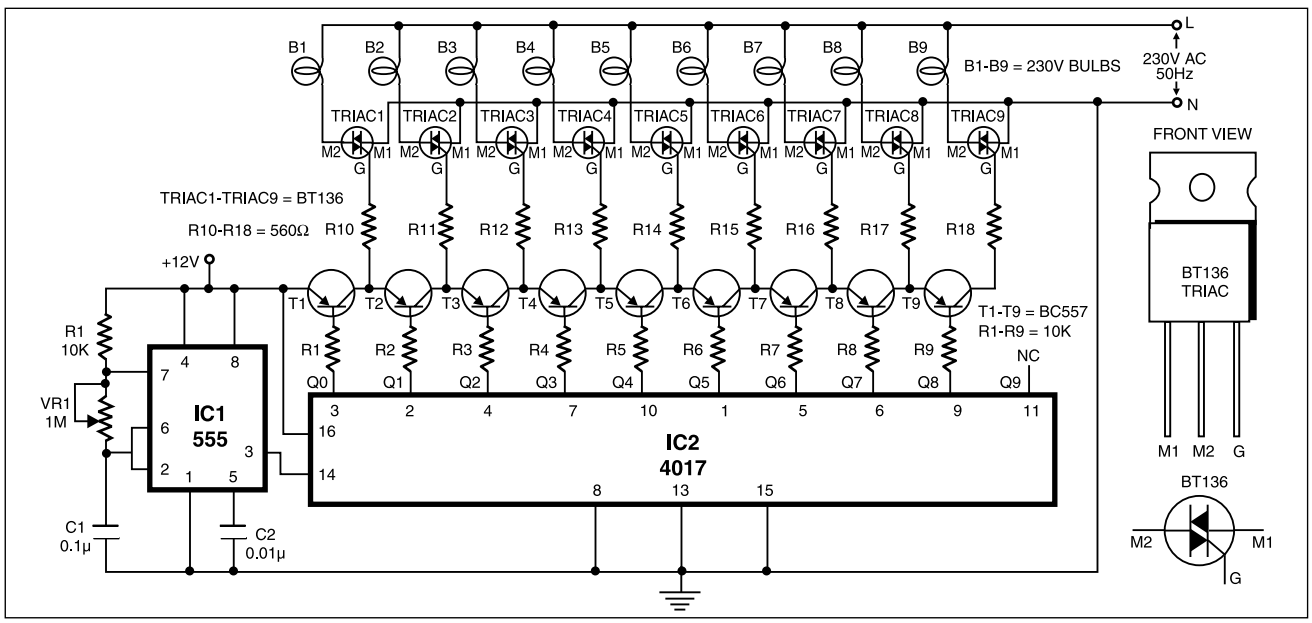
CD4017 is a decade counter having ten outputs, of which one output is high for each clock pulse. However, this produces running lights effect. To change this sequence to get the speller effect, pnp transistors T1 through T9 are wired as shown in the figure. Nine triacs (triac 1 through triac 9) are used to drive 230V bulbs. (In place of 230V bulbs, miniature lamps connected in series in the form of characters or letters can also be used, provided the voltage drop across the series combination is 230 volts.)

When any of the outputs of IC2 goes high, the corresponding transistor connected to the output goes off. When Q0 is high, transistor T1 goes off and its output at the collector goes low. Since the emitter of transistor T2 is connected to the collector of transistor T1, and collector and emitter terminals of transistors T1 through T9 are connected in series, all transistors next to transistor T1, i.e. transistors T2 through T9, do not get supply and hence all their outputs go low.

Next, when Q1 output goes high, transistor T2 goes off. Thus outputs of transistors T2 through T9 remain low. Since Q0 output at this instant is low, transistor T1 is forward biased and its output goes high to light up the first character.

Similarly, when Q2 output goes high, Q0 and Q1 outputs are low and therefore outputs of transistors T1 and T2 go high to light up the first and second characters.

This process continues until all transistors turn on, making all the characters to light up. The cycle repeats endlessly, producing the speller type light effect. ❑



# STEREO TAPE HEAD PREAMPLIFIER FOR PC SOUND CARD


T.K. HAREENDRAN

Here is a stereo tape head preamplifier circuit for your PC sound card that can playback your favourite audio cassette through the PC. Audio signals from this circuit can be directly connected to the stereo-input (line-input) socket of the PC sound card for further processing.

The circuit is built around a popular stereo head preamp IC LA3161. Weak electrical signals from the playback heads are fed to pins 1 and 8 of IC1 via DC decoupling capacitors C1 and C6, respectively. Components between pins 2 and 3 and pins 6 and 7 provide adequate equalisation to the signals for a normal tape playback.

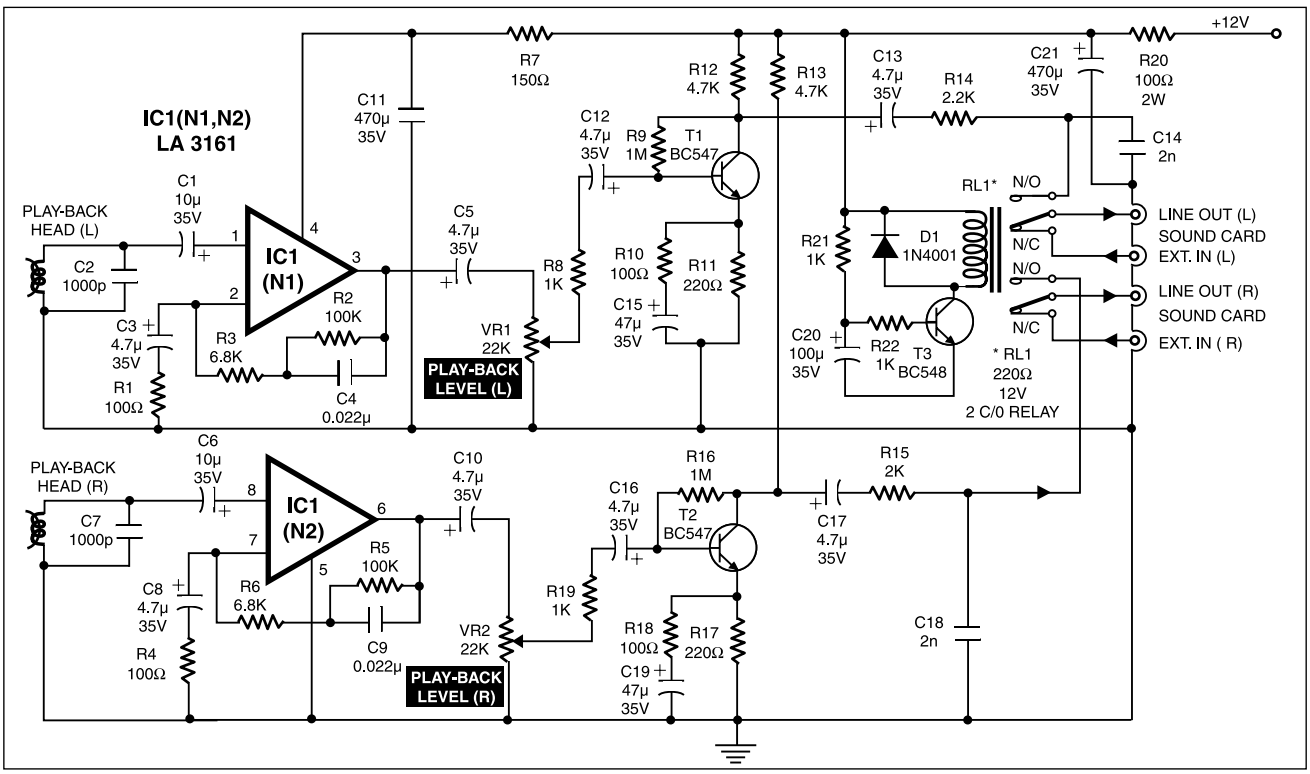
The amplified and equalised signals available at output pins 3 and 6 of IC1 are coupled to the inputs of line amplifier circuit built around transistors T1 (via capacitor C5, potmeter VR1, resistor R8, and capacitor C12) and T2 (via capacitor C10, potmeter VR2, resistor R19, and capacitor C16), respectively. Left and right playback levels can be adjusted by variable resistors VR1 and VR2. The audio signals are finally available at the negative ends of capacitors C13 and C17.

The circuit wired around relay driver transistor T3 serves as a simple source selector. This is added deliberately to help the user share the common PC sound card line-input terminal for operating some other audio device as well.

When the preamplifier is in 'off' state, switching relay RL1 is off and it allows connection of external signals to the sound card. When the preamplifier is turned 'on', the relay is energised by transistor T3 after a short delay determined by the values of resistor R21 and capacitor C23. On energisation, the relay contacts changeover the signals to internal source, i.e. the head preamplifier.

After constructing the whole circuit on a veroboard, enclose it in a mini metallic cabinet with level controls and sockets at suitable points. Use a regulated 1A, 12V DC power supply for powering the whole circuit including the tape deck mechanism. (A 1A, 18V AC secondary transformer with 4700µF, 40V electrolytic capacitor and 78M12 regulator is sufficient.)

You can use any kind of tape deck mechanism with this circuit. Use of good-quality playback head and well-screened wires are recommended. ❑



# LEAD-ACID BATTERY CHARGER WITH VOLTAGE ANALYSER


**D. MOHAN KUMAR**

Nowadays maintenance-free lead-acid batteries are common in vehicles, inverters, and UPS systems. If the battery is left in a poor state of charge, its useful life is shortened. It also reduces the capacity and rechargeability of the battery. For older types of batteries, a hygrometer can be used to check the specific gravity of the acid, which, in turn, indicates the charge condition of the battery. However, you cannot use a hygrometer for sealed-type maintenance-free batteries. The only way to know their charge level is by checking their terminal voltage.

The circuit presented here can replenish the charge in a battery within 6-8 hours. It also has a voltage analysing circuit for quick checking of voltage before start of charging, since overcharging may damage the battery. The voltage analyser gives an audio-visual indication of the battery voltage level and also warns about the critical voltage level at which the battery requires immediate charging.

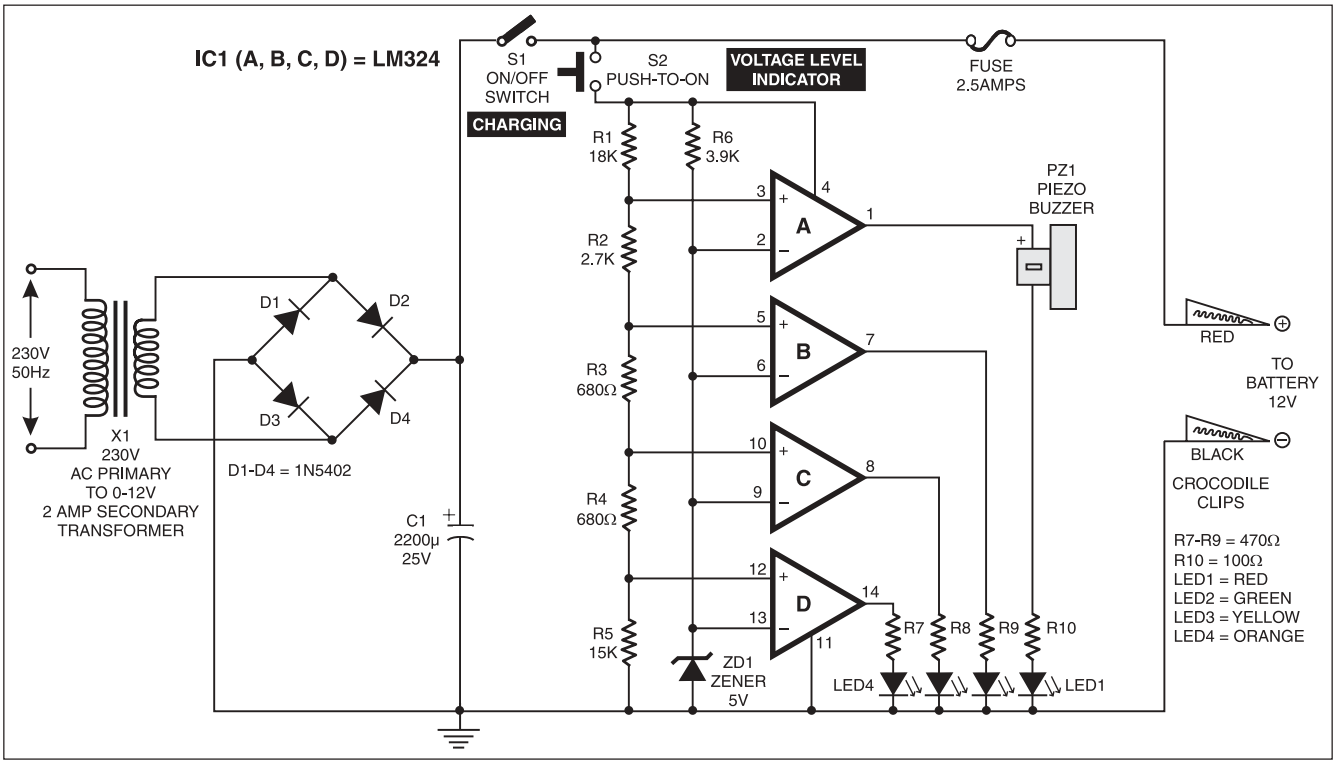
The charger circuit consists of a standard step-down 12V AC (2-amp) transformer and a bridge rectifier comprising diodes D1 through D4. Capacitor C1 smoothes the AC ripples to provide a clean DC for charging the battery.

The battery voltage analyser circuit is built around the popular quad op-amp LM324 that has four separate op-amps (A through D) with differential inputs. Op-amps have been used here as comparators. Switch S2 is a pushswitch, which is pressed momentarily to check the battery voltage level before charging the battery.

The non-inverting terminals of op-amps A through D are connected to the positive supply rail via a potential divider chain comprising resistors R1 through R5. Thus the voltage applied to any non-inverting input is the ratio of the resistance between that non-inverting terminal and ground to the total resistance (R1 + R2 + R3 + R4 + R5). The resistor chain provides a positive voltage of above 5V to the non-inverting inputs of all op-amps when battery voltage is 12.5V or more. A reference voltage of 5V is applied to the inverting inputs of op-amps via 5V zener diode ZD1.

When the circuit is connected to the battery and pushswitch S2 is pressed (with S1 open), the battery voltage is sampled by the analyser circuit. If the supply voltage sample applied to the non-inverting input of an op-amp exceeds the reference voltage applied to the inverting inputs, the output of the op-amp goes high and the LED connected at its output lights up.

| Battery voltage | Status of LEDs | | | | Comments |
|---|---|---|---|---|---|
| | Red | Green | Yellow | Orange | |
| < 9.8V | Off | Off | Off | Off | Buzzer off |
| > 9.8V | On | Off | Off | Off | Danger level |
| 11.5V | On | On | Off | Off | Low level |
| 12.0V | On | On | On | Off | Normal level |
| 12.5V | On | On | On | On | High level |



The different levels of battery voltages are indicated by LED1 through LED4. All the LEDs remain lit when the battery is fully charged (above 12.5V). The buzzer connected to the output of IC1 also sounds (when S2 is pressed with S1 kept open) as long as the voltage of battery is above 9.8V. If the voltage level goes below 9.8V, the buzzer goes off, which indicates that it's time to replace the battery. The status of LEDs for different battery voltages is shown in the table.

The circuit can be assembled on a general-purpose PCB or a veroboard. Use 4mm wire and crocodile clips to connect the charger to the battery. A 2.5-amp fuse connected to the output of the charger protects the analyser circuit against accidental polarity reversal.

The circuit costs around Rs 120 with all accessories.

# THREE-COLOUR DISPLAY USING BICOLOUR LEDs


PRIYANK MUDGAL

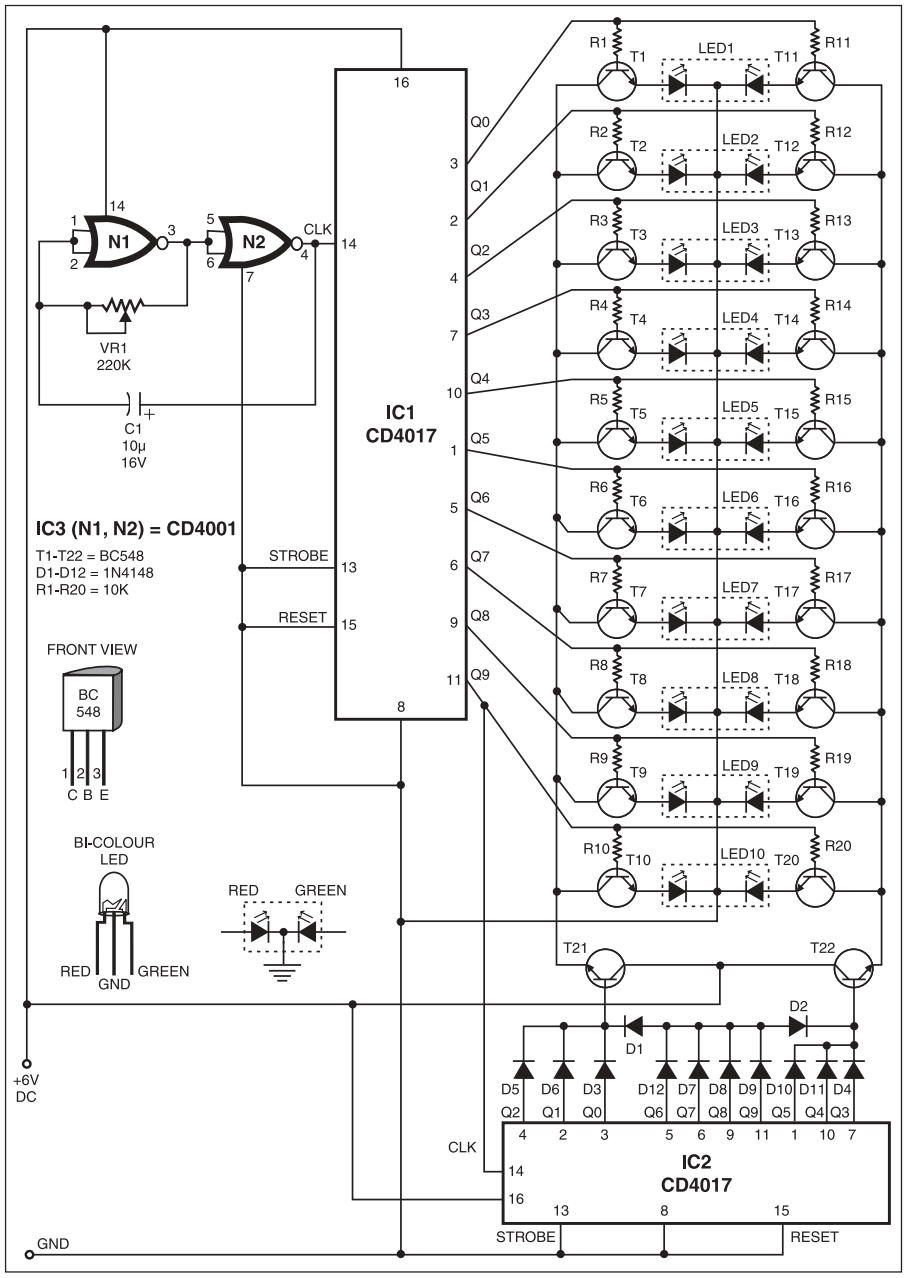
The circuit presented here uses bicolour LEDs to generate a display in three colours, namely, red, green, and yellowish green.

Transistors T1 through T20 form a grid to which common-cathode bicolour LEDs (LED1 through LED10) are connected. Transistors T1 through T10 have their collector terminals connected to the emitter of transistor T21. Similarly, transistors T11 through T20 have their collector terminals connected to the emitter of transistor T22. The bases of each pair of transistors (i.e. T1 and T11, T2 and T12,..., T10 and T20) are tied to outputs Q0, Q1,..., Q9, respectively, of IC1 (CD4017) through 10-kilo-ohm resistors as shown in the figure. Positive supply to collectors of transistors T1 through T10 is controlled by transistor T21. Similarly, positive supply to collectors of transistors T11 through T20 is controlled by transistor T22.

IC1 and IC2 are decade counters. Clock pulse to IC1 is provided by the oscillator circuit comprising NOR gates N1 and N2. The outputs of IC1 advance sequentially with each clock. (Any other source of squarewave pulses also serves the purpose.) IC2 is used to select the mode of display. Clock input pin 14 of IC2 is connected to Q9 output of IC1. Thus IC2 receives one pulse after every ten pulses received by IC1.

When the circuit is switched on, Q0 output of IC2 is active high. Thus transistor T21 gets forward biased via diode D3 and it conducts to extend positive supply to transistors T1 through T10. Transistors T1 through T10 are forward biased sequentially by Q0 through Q9 outputs of IC1, i.e. at a time only one of these ten transistors is forward biased (on). Thus only red LED parts of bicolour LEDs light up sequentially. (Transistor T22 is not conducting at this moment.)

When red LED part of LED10 glows, IC2 receives a clock pulse and its Q1 output goes high. Transistor T21 still conducts, as it is forward biased through diode D6, and next again via diode D5. Thus red LEDs complete two more glowing sequences.




After completion of the third glowing sequence of red LEDs, when Q3 output of IC2 goes high, transistor T21 stops conducting and T22 starts conducting with the next three sequences of green LEDs of bicolour LEDs (LED1 through LED10) glowing sequentially.

After completion of three sequences of green LEDs, output Q6 of IC2 goes high. Now both transistors T21 and T22 conduct due to diodes D1 and D2. Thus both red and green LEDs in bicolour LEDs (LED1 through LED10) glow sequentially. The effect of red and green LEDs glowing together is a distinct yellowish orange colour. This sequence repeats four times.

Thereafter, the whole sequence repeats, starting with red LEDs. Thus the bicolour-

LED display shows three colours—red, green, and yellowish green—one after the other.

The speed of display can be controlled by preset VR1. One can omit automatic selection of different colours by omitting IC2 and replacing connections to pins 3, 5, and 7 of IC2 with SPDT switches. (Thus diodes D3-D12 are also omitted.)

This circuit costs around Rs 250.

# ULTRA-BRIGHT LED LAMP


**N.S. HARISANKAR VU3NSH**

This ultra-bright white LED lamp works on 230V AC with minimal power consumption. It can be used to illuminate VU meters, SWR meters, etc.

Ultra-bright LEDs available in the market cost Rs 8 to 15. These LEDs emit a 1000-6000mCd bright white light like welding arc and work on 3 volts, 10 mA. Their maximum voltage is 3.6 volts and the current is 25 mA. Anti-static precautions should be taken when handling the LEDs. The LEDs in water-clear plastic package emit spotlight, while diffused type LEDs have a wide-angle radiation pattern.

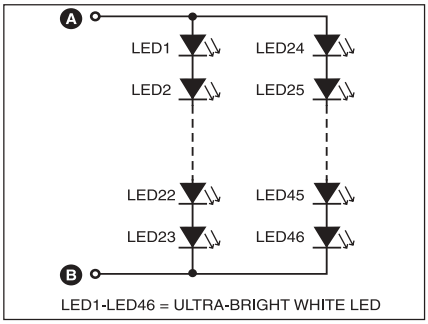



*Fig. 3: 46-LED combination*

This circuit (Fig. 1) employs capacitive reactance for limiting the current flow through the LEDs on application of mains voltage to the circuit. If we use only a series resistor for limiting the current with mains operation, the limiting resistor itself will dissipate around 2 to 3 watts of power, whereas no power is dissipated in a capacitor. The value of capacitor is calculated by using the following relationships:

$X_C = 1/(2\pi fC)$ ohms ——————(a)

$X_C = V_{RMS}/I$ ohms ——————(b)

where $X_C$ is capacitive reactance in ohms, C is capacitance in farads, I is the current through the LED in amperes, f is the mains frequency in Hz, and Vrms is the input mains voltage.

The 100-ohm, 2W series resistor avoids heavy 'inrush' current during transients. MOV at the input prevents surges or spikes, protecting the circuit. The 390-kilo-ohm, ½-watt resistor acts as a bleeder to provide discharge path for capacitor Cx when mains supply is disconnected. The zener diode at the output section prevents excess reverse voltage levels appearing across the LEDs during negative half cycles. During positive half cycle, the voltage across LEDs is limited to zener voltage.

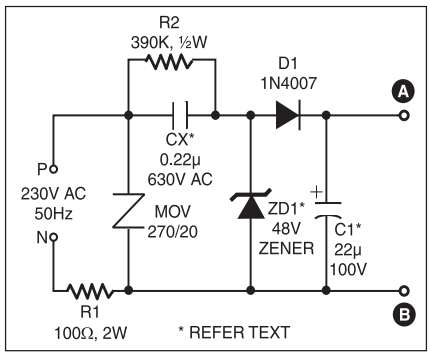



*Fig. 1: The circuit of ultra-bright white LED lamp*

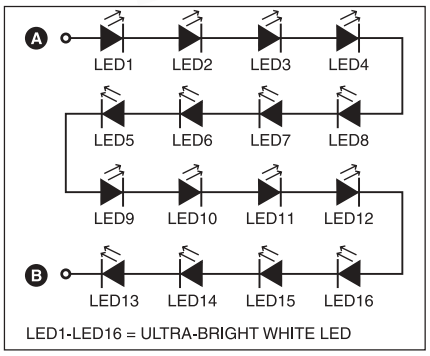



*Fig. 2: 16-LED combination*

Use AC capacitors for Cx. Filter capacitor C1 across the output provides flicker-free light. The circuit can be enclosed in a CFL round case, and thus it can be connected directly to AC bulb holder socket. A series combination of 16 LEDs (Fig. 2) gives a luminance (lux) equivalent of a 12W bulb. But if you have two series combinations of 23 LEDs in parallel (total 46 LEDs as shown in Fig. 3), it gives light equal to a 35W bulb. 15 LEDs are suitable for a table-lamp light.

Diode D1 (1N4007) and capacitor C1 act as rectifying and smoothing elements to provide DC voltage to the row of LEDs. For a 16-LED row, use Cx of 0.22 µF, 630V; C1 of 22 µF, 100V; and zener of 48V, 1W. Similarly, for 23 + 23 LED combination use Cx of 0.47 mF, 630V; C1 of 33 µF, 150V; and zener of 69V, 1W.

This circuit (inclusive of LEDs) costs Rs 200 to Rs 400.

# VERSATILE EMERGENCY LIGHT USING FLUORESCENT TUBES

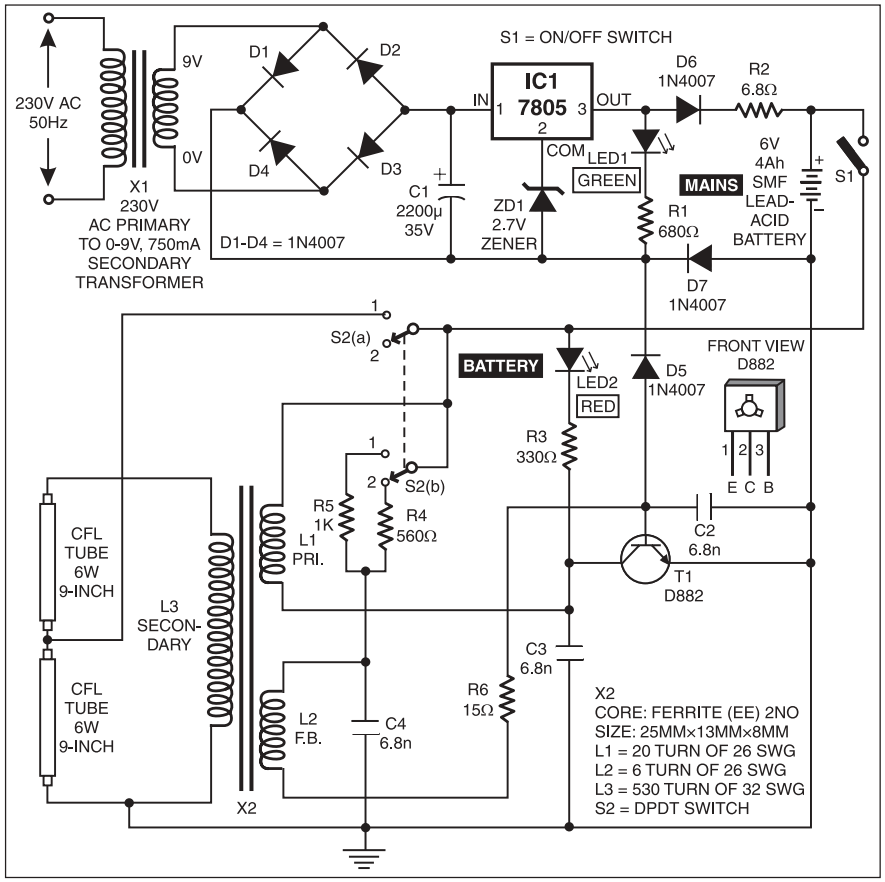

**VIVEK KALKUR M.**

Emergency lights using incandescent bulbs are inherently inefficient compared to those using fluorescent tubes. Here's a versatile emergency light using fluorescent tubes. You can operate it using readymade SUNCA or similar other inverter transformers, which are readily available in the market for around Rs 25. With this circuit you can drive two 6W, 22.8cm (9-inch) fluorescent tubes, with the option to use a single tube or a pair of tubes with the help of DPDT switch S2.

Step-down transformer X1, diodes D1-D4, capacitor C1, and 5V regulator IC1 (7805) form a regulated power supply. A 2.7V zener (ZD1) in common terminal of the regulator props up the output voltage to 7.7 volts. The regulated voltage is applied to the battery through diodes D6 and D7, which cause a drop of about 1.4V across them. Thus the effective charging voltage is about 6.3V, which prevents overcharging of the battery as the terminal voltage of the battery cannot exceed 6.3V.

When AC mains supply is present, the battery starts charging and green LED1 glows to indicate the same. Diode D5 reverse biases transistor T1 forming part of the inverter oscillator and thus the tubes don't glow.

When mains supply fails, transistor T1 starts oscillating and supplies power to inverter transformer X2 and the tubes glow.




An on/off switch (S1) is used to switch off the light when it is not required.

D882 (actually, 2SD882) is an npn power transistor in TO-126 package. It is mounted on a suitable heat-sink to prevent it from thermal runaway. For good illumination, use Toshiba's FL6D fluorescent tubes.

The circuit (excluding the cabinet) costs around Rs 350.

# CLAP-BASED SWITCHING FOR DEVICES


**MANOJ KUMAR SAHA**

It is quite difficult to find the switch board in a dark room to turn on the light. Here's a clap switch that allows you to switch on lights, fans, and motors sequentially by just clapping in the vicinity of the microphone used in the circuit.

The mains supply is stepped down to 15-0-15V AC by step-down transformer X1. The output of the transformer is rectified, filtered, and regulated by diodes D1 through D4, capacitors C1 through C4, and IC1 (regulator IC 7812) and IC2 (regulator IC 7912), respectively. Additional filtering is performed by capacitors C5 through C8 to get +12V, 0V (Gnd) and –12V DC required for the operation of the circuit.

The clap sound impulses are converted into electrical signals by a condenser microphone that forms a Wheatstone bridge together with resistors R4, R5, and R3. The microphone is suitably biased through resistor R3. The output of the microphone is coupled to op-amp IC 741 (IC3) having a voltage gain of 45. The output of IC3, after passing through capacitor C10, is free from any DC component of signal. Capacitors C15 and C17 are used for spike and surge suppression.

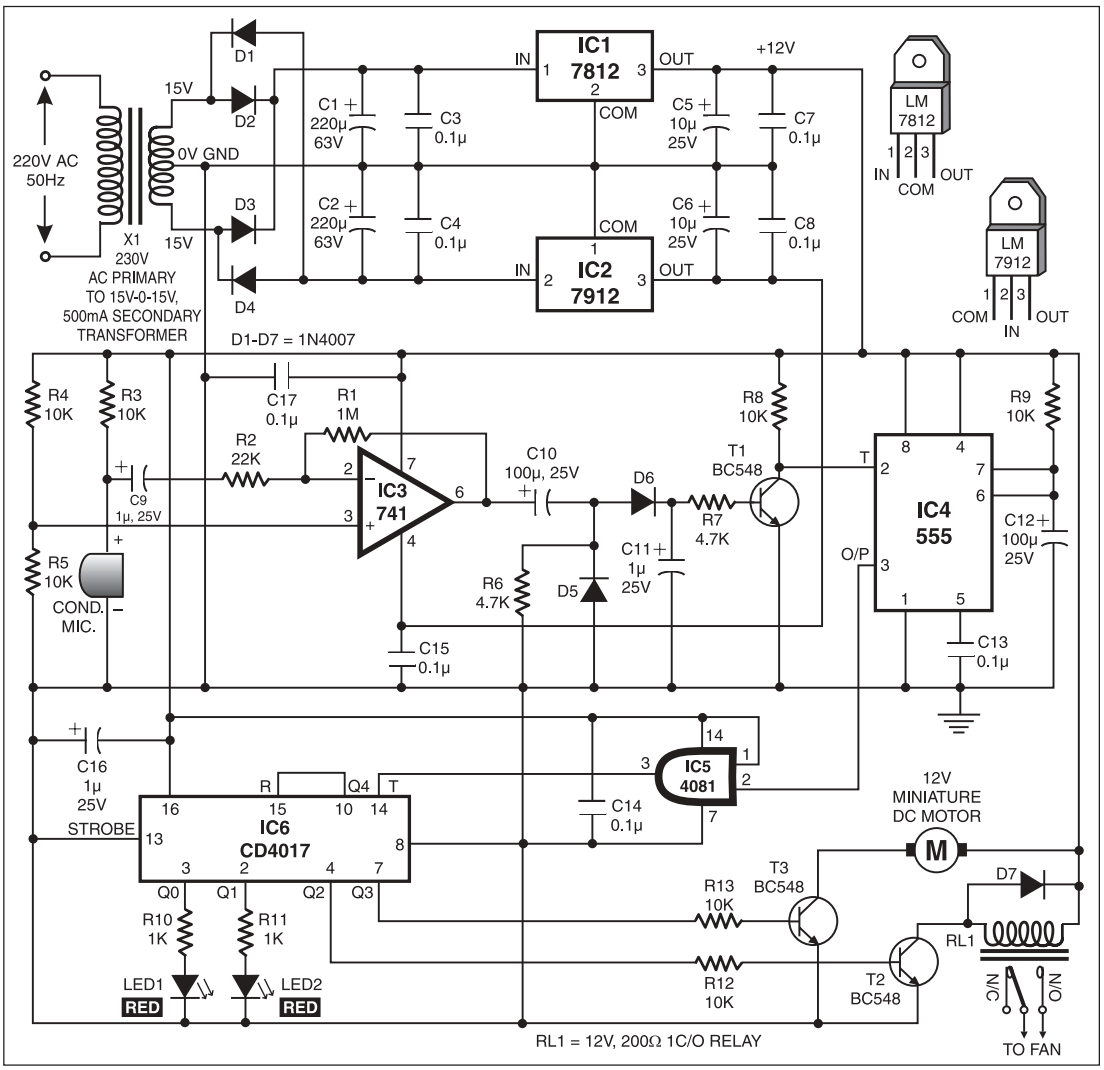
Diodes D5 and D6 and capacitor C11 form the detector circuit. Resistor R6 is used here for quick discharge of capacitor C10. The detected clap signal is used to switch on transistor T1. On conduction of transistor T1, its collector voltage falls to trigger timer IC4 connected as a monostable. The combination of resistor R9 and capacitor C12 determine the pulsewidth of the monostable (about one second, with the component values shown).

AND gate IC5 (4081) is used as a buffer between the output of IC4 and clock input to decade counter IC6 (CD4017). Thus each clap causes outputs of IC6 to advance in sequential manner and switch on the corresponding devices.

If you want a lamp to be switched on when output Q1 goes high (after first clap), then in place of R11 and LED2 use a relay driver circuit at Q1 output similar to that used for Q2 output (for fan).

As stated earlier, only one output of CD4017 can be high at any given time. Thus first clap causes LED1 to go off and LED2 to glow. The second clap causes only the fan to switch on via relay RL1. The third clap causes the miniature 12V motor to run. On fourth clap, Q4 output goes high momentarily to reset IC6 since Q4 output is connected to its reset pin 15. In reset state, LED1 connected to Q0 output lights up.

The circuit costs around Rs 150.



# DIY Kit 130. Door Minder

**INTRODUCTION**
This kit uses an infrared (IR) beam to monitor door & passage-ways or any other area. When the beam is broken a relay is tripped which can be used to sound a bell or alarm. Suitable for detecting customers entering a shop, cars coming up a driveway, etc. The IR beam is very strong. Distances over 25 yards can be monitored with this Kit. A 12VDC unregulated supply is required to power the kit. A 12V wall adaptor is fine. Provision has been made so that only one power supply needs to be used to power both units. The relays in this kit are rated to switch mains voltages.

The kit is constructed on single-sided printed circuit board. Protel Autotrax & Schematic were used in the design.

**CONSTRUCTION**
The kit is built on two separate PCBs – a transmitter PCB (K130T) and a receiver PCB (K130R). Refer to the parts list to see which components belong to which board. Use the component overlay on each PCB to place the components.

**Transmitter board**
Insert the lowest height components first. Leave the infrared LED and power jack until last. Make sure that the electrolytic capacitor and diode are inserted correctly. A socket is supplied for the IC.

The IR LED can be mounted vertically or at right angles to the PCB. This will depend on how the transmitter will be mounted when in use. In either case the LED must be inserted the right way round. The flat edge on the LED should line up with the flat edge on the PCB overlay.

Lastly, if the distance to be monitored is less than about 10 yards then you will need to fit the 5mm tubing over the IR LED. This narrows the radiating angle of the IR beam and makes it much more directional. The IR output is strong. It can easily bounce off walls etc to give false readings.

**Receiver board**
As with the transmitter board, start with the lowest height components first - resistors, diodes, capacitors and transistors. Note the polarity of the electrolytic capacitors and diodes. Next insert the screw terminal block, power jacks and relay. Last of all is the infrared receiver module. The orientation is clearly shown on the PCB overlay.

**CIRCUIT DESCRIPTION**
**Transmitter Board**
The transmitter board consists of two oscillators, one running at approx. 250Hz and the other running at 38KHz. The 38KHz frequency acts as a carrier wave and is required by the IR receiver module on the receiver board. This carrier wave is "ANDed" or modulated with the 250Hz frequency to produce an output signal that contains bursts of 38KHz at a rate of 250Hz. This signal is used to drive an infrared LED.

The oscillators are made using a 4093 quad 2-input NAND gate IC. One gate, IC1:D, is used for the 250Hz oscillator. Resistor R1 and capacitor C1 set the frequency. Another pair of gates, IC1:A and IC1:B, make up the 38KHz oscillator. Resistors R2 and R3 and capacitor C2 set the frequency. Gate IC1:C "ANDs" these signals together.

Transistor Q1 provides a constant current source for the IR LED. The zener, Z1, on the base of Q1 keeps the emitter at a constant voltage. Therefore the voltage across R6 is also constant. For the values shown the current through the LED, when on, is fixed at approx. 85mA. The peak current through the LED is set by R7 and detemines the range of the kit. Reduce it to say 22R to get greater range if needed.

**Receiver Board**
The receiver consists of an IR receiver module that detects the incoming IR beam. The IR signal is used to keep a capacitor charged which in turn holds a relay operated. When the beam is broken the capacitor discharges and the relay releases.

The IR detector module, RX1, is made up of an IR LED and an amplifier/filter circuit tuned to detect a 38KHz frequency. The output pin is low whenever a 38KHz signal is detected.

**When the IR beam is present the relay is operated**.

The output of RX1 is the 250Hz signal from the transmitter. This signal is passed via transistor Q1, capacitor C1and diode D2 to capacitor C2.

C2 is fully charged during the high portion of the signal. It starts to discharge during the low portion of the signal via LED L1, resistor R4 and transistor Q2. However the discharge time is much longer than the off time of the signal so the voltage across C2 is always enough to keep transistor Q2 on and therefore the relay operated.

When the beam is broken the output of RX1 is high. Transistor Q1 is off and capacitor C2 is no longer being recharged. It will eventually discharge to the point where transistor Q2 will turn off and the relay will release.

The "turn off" delay is determined by the time constant of resistor R5 and capacitor C3. With the values used it is approximately half a second.

Capacitor C1 prevents a steady DC voltage on the collector of Q1 from charging C2. This would occur if the beam was not present or the beam was a **continuous** 38KHz signal. In other words, the receiver module will only respond to a **pulsed** 38KHz signal.

LED L1 gives a visual indication when the IR beam is present and is used to help with installation and setup. It can, of course, be removed if it is not wanted.

# DIY Kit 130. Door Minder

Zener diode Z1, resistor R6 and capacitor C4 provides a stable 5.6V supply for the IR module. The relays provided with this kit are mains rated but use with care. RUDH 110V/10A; RWH 250V/12A, 120VAC/15A.

**INSTALLATION**

For ease of operation, the transmitter board can be powered from the receiver board when they are relatively close together. Two plugs have been supplied for this. Connecting wire for the length required must be supplied by you. Otherwise two separate power supplies are required.

The receiver board contains two DC jacks connected in parallel. Power from a 12VDC source is connected to one jack. A lead can then be connected to the other jack and run to the transmitter board. DC plugs to make up a lead are supplied with the kit.

Aligning the transmitter and receiver is simply a matter of pointing the transmitter IR LED at the receiver board and moving it around until the red LED on the receiver board lights. This indicates that the beam is being received. The relay will be operated.

**Note:**

With power applied the relay is normally operated and only releases when the beam is broken. The relay contacts labelled "NO" (Normally Open) and "NC" (Normally Closed) refer to when the relay is released.

With the relay operated, the "NO" contact will be connected to the "C" contact and the "NC" contact will be unconnected.

**IF IT DOES NOT WORK**

Poor soldering ("dry joints") is the most common reason that the circuit does not work. Check all soldered joints carefully under a good light. Re-solder any that look suspicious. Check that all components are in their correct position on the PCB. Are the electrolytic capacitors and diodes the right way round? Have you fitted the 5mm tubing over the IR LED on the transmitter board?

**Web Address & Email**

You can email us at **peter@kitsrus.com**

See our website at

**http://kitsrus.com**

This is an improved (natually!) version of the door minder published in *Silicon Chip*, april, 1999. It was developed originally by Oatley Electronics.

**PARTS LIST – K130**

**Transmitter board**

**Resistors (0.25W carbon)**

| | | |
|---|---|---|
| 47 yellow violet black | R6 | 1 |
| 1K brown black red | R5 | 1 |
| 1K8 brown grey red | R3 | 1 |
| 6K8 blue grey red | R2 | 1 |
| 47K yellow violet orange | R1,4 | 2 |

**Capacitors**

| | | |
|---|---|---|
| 1nF ceramic 102 | C2 | 1 |
| 100nF monobloc 104 | C1 | 1 |
| 100uF 25V electrolytic | C3 | 1 |

**Semiconductors**

| | | |
|---|---|---|
| 4.7V 400mW zener diode | Z1 | 1 |
| BC557 transistor, PNP | Q1 | 1 |
| 4093 CMOS IC | IC1 | 1 |
| Quad 2-input NAND with schmitt trigger inputs | | |
| IR LED, EL-1L7 | L1 | 1 |

**Miscellaneous**

| | | |
|---|---|---|
| 2.5mm DC jack | X1 | 1 |
| Tubing, 5mm x 25mm long to fit over L1 | | 1 |
| PCB, K130T | | 1 |

**Receiver Board**

**Resistors (0.25W carbon)**

| | | |
|---|---|---|
| 470 yellow violet brown | R3,6 | 2 |
| 6K8 blue grey red | R1,2,4 | 3 |
| 47K yellow violet orange | R5 | 1 |

**Capacitors**

| | | |
|---|---|---|
| 10uF 16V electrolytic | C1,2,3 | 3 |
| 100uF 25V electrolytic | C4,5 | 2 |

**Semiconductors**

| | | |
|---|---|---|
| 1N4148 signal diode | D1,2 | 2 |
| 1N4004 power diode | D3 | 1 |
| 5.6V 400mW zener diode | Z1 | 1 |
| BC547 transistor, NPN | Q2 | 1 |
| BC557 transistor, PNP | Q1 | 1 |
| LED, 5mm red | L1 | 1 |
| IR receiver module | RX1 | 1 |
| PIC-12043S | | |

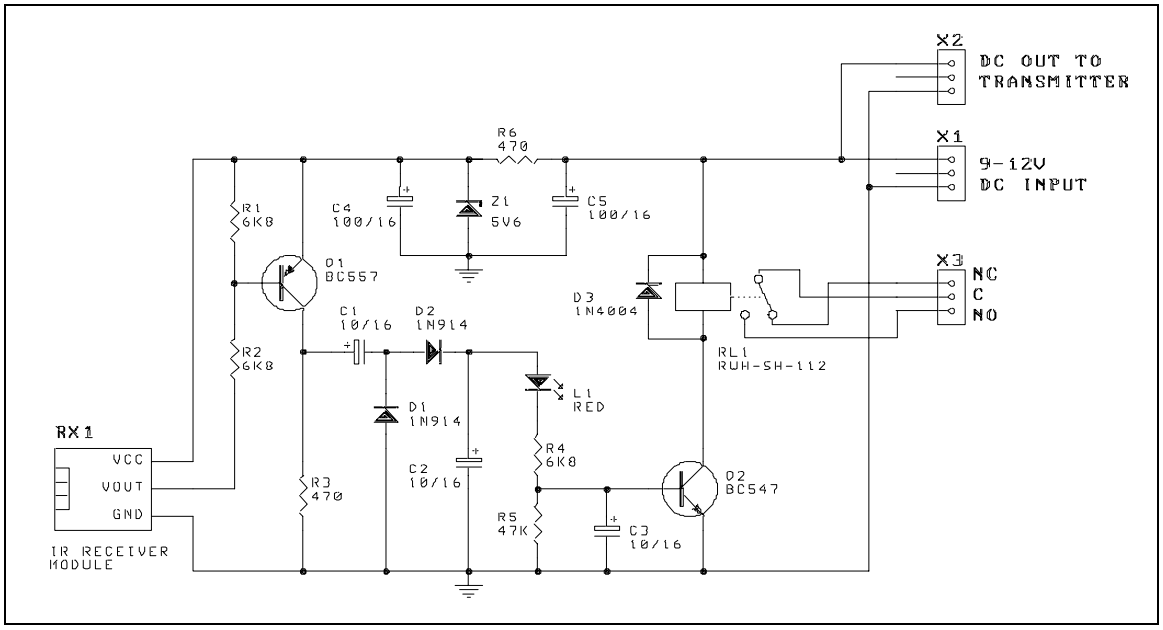
**Miscellaneous**

| | | |
|---|---|---|
| 2.5mm DC jack | X1,2 | 2 |
| Terminal block, 3-way | X3 | 1 |
| Relay, 12V SPDT | RL1 | 1 |
| "Goodsky" RWH-SH-112D, or RUDH-SS-112D | | |
| PCB, K130R | | 1 |

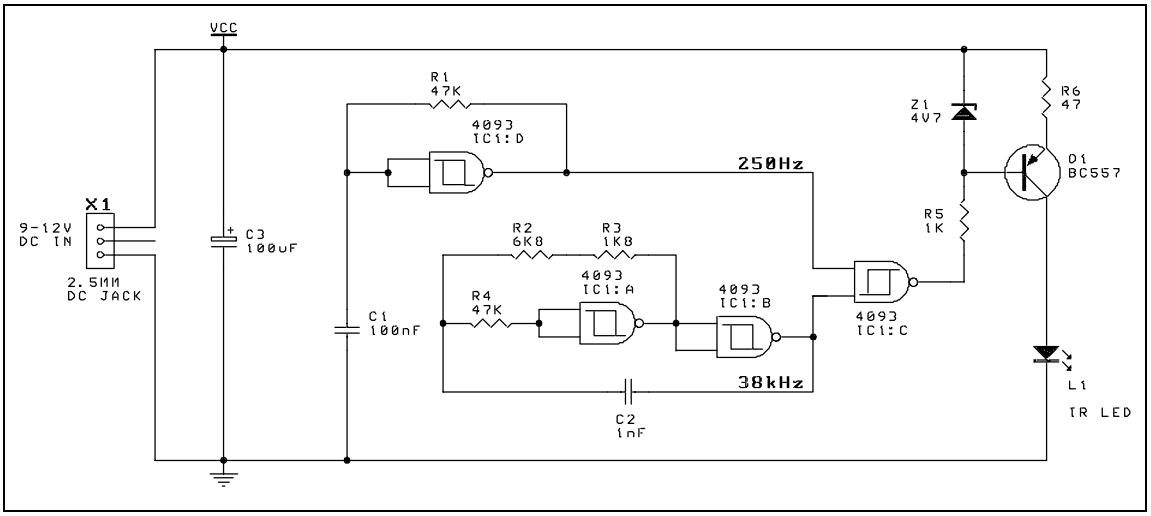
**Extras**

| | |
|---|---|
| 2.5mm DC plugs | 2 |
| (to make up optional power lead to the transmitter board) | |

# DIY Kit 130. Door Minder




## Door Minder - Receiver Module




## Door Minder - Transmitter Module

# Kit 49. ULTRASONIC MOVEMENT DETECTOR II

This is our second ultrasonic movement detector circuit. It also uses a crystal locked circuit to get maximum performance from the ultrasonic transmitter. However, the detection circuit is different. We think it is more sensitive.

ASSEMBLY INSTRUCTIONS

Check all the components against the parts list. It is generally easier to solder in the lowest height components first - the resistors and diodes. Make sure to get the diodes in the correct way. The black circle or band on the diodes must match the bar of the diode symbol on the overlay. There are a lot of resistors. Use the resistor color code printed on the back of the header card to work out each value.

The 40 kHz crystal can be inserted either way. Holes have been provided to secure its case to the PCB. Use one of the leads cut from the resistors. You can also quickly solder the wire to the case of the crystal.

The ultrasonic transmitter has a T suffix on the number stampedunderneath it. The receiver has an R suffix. Each may be soldered either way around on the pads under the PCB. Make sure to get them both pointing straight out at $90^{o}$ from the PCB.

CALIBRATION

Battery operation is not recommended for this detector. As the battery potential decreases the sensitivity will change. This is partcuarly so if the unit is triggered often (as in detecting a door opening.) It is better to use a power supply. Use 9V to 12V DC. You could go to a maximum of 15V but you may have to replace the 78L05 by a 7805.

Immediately after you connect the power let the unit stand for at least 20 seconds so whole circuit will settle down electrically. The schematic shows that the setting of the trimpot value is critical to the operation of the detector. We provide a 1M trimpot. Set it to about 400K by eye. We have found that below about 300K the detector is too sensitive and will self-trigger. Trial and error will show the best setting for your particular requirement.

Note that this circuit is very sensitive. Even air moving (hot air rising, wind blowing) will trigger it when the trimpot is set near the most sensitive position. That is why we say to set it for your particular need.

CIRCUIT DESCRIPTION

The transmitter sends out a steady ultrasonic tone at 40kHz. At this frequency the wavelength is about 6 mm.

Any reflected sound is detected by ultrasonic receiver. The signal is then amplified by IC1:A and IC1:B. IC1:A is self-biasing via C2 & R5. The time constant of the first amplifier is set to let the 40kHz signal through. Between the first & second amplifier there is a negative peak detector (diode D1 & R8) which follows the envelope of the 40kHz signal. If there is no movement the envelope is just a straight line. The time constant of IC1:B is much slower so that it will follow this envelope. All the amplifiers are AC coupled to prevent DC bias problems.

Then the signal is fed through a window detector IC1:C which detects both positive and negative pulses. When there is no movement the potential at pin 7 sits at half the supply potential and neither D2 or D3 can conduct. The potential at pin 8 is low. If the signal rises D3 conducts causing the output to go high. If the signal falls then D2 conducts which also causes the output to go high. Thus the name window detector circuit because it detects potentials which move both below and above a given range.

A low pass filter screens out unwanted spurious signals, then an amp IC1:D set up as a monostable flip flop converts any signal that gets through the filter into a substantial pulse to turn on the BC639. This turns on the LED and provides a Signal Out to drive a separate relay or any other device you may wish to signal to. The time constant of the monostable flip-flop is about half a second and is set by C8 & R10. D4 is used to separate the charge & discharge time constants. It lets the circuit switch on immediately movement is detected but allows about a 1/2 second delay for the reset.

WHAT TO DO IF IT DOES NOT WORK

Poor soldering is the most likely reason. Check all solder joints carefully under a good light. Next check that all components are in their correct position on the PCB. Thirdly, follow the track with a voltmeter to check the potential differences at various parts of the circuit.

Other items to check; are the IC's in the correct way. Check no IC pins are bent up. Are the diodes, transistors and the electrolytic capacitor in the correct way. Did you mix up the 78L05 with the BC639? Check the values of the resistors.

**COMPONENTS**

| | | |
|---|---|---|
| Resistors: carbon film, 5% | | |
| 47R | R17 | 1 |
| 1K brown black red | R6 R20 | 2 |
| 2K2 red red red | R4 | 1 |
| 10K brown black orange | R13 R16 R19 | 3 |
| 100K brown black yellow | R1 R2 R15 | 3 |
| 150K brown green yellow | R22 | 1 |
| 220K red red yellow | R5 | 1 |
| 1M Brown black green | R8 R9 R10 R14 | 4 |
| 1M2 brown red green | R12 R11 | 2 |
| 10M brown black blue | R3 R7 R18 R21 | 4 |
| Trimpot 1M | | 1 |
| Capacitors: | | |
| 100nF monoblock (104) | C1 C10 | 2 |
| 100pF monoblock (101) | C5 | 1 |
| 10nF (103) ceramic | C2 C6 C7 | 3 |
| 33pF ceramic | C11 C12 | 2 |
| 100 uF electrolytic | C9 | 1 |
| 470 nF (474) monoblock | C3 C4 C8 | 3 |
| Diode 1N4002/4 | D5 | 1 |

# Kit 49. ULTRASONIC MOVEMENT DETECTOR II

| | | |
|---|---|---|
| Diode 1N4148 | D1 2 3 4 | 4 |
| 9V battery snap | | 1 |
| 40kHz crystal | | 1 |
| 78L05 voltage regulator | | 1 |
| 14049 hex buffer IC | IC2 | 1 |
| BC639 | | 1 |
| LM324 IC | IC1 | 1 |
| 14 pin IC socket | | 1 |
| 16 pin IC socket | | 1 |
| 40kHz Ultrasonic Tx/Rx pair | 400ST/R160 | 1 |
| LED | | 1 |
| Kit 49 PCB | | 1 |
| Documentation | | |

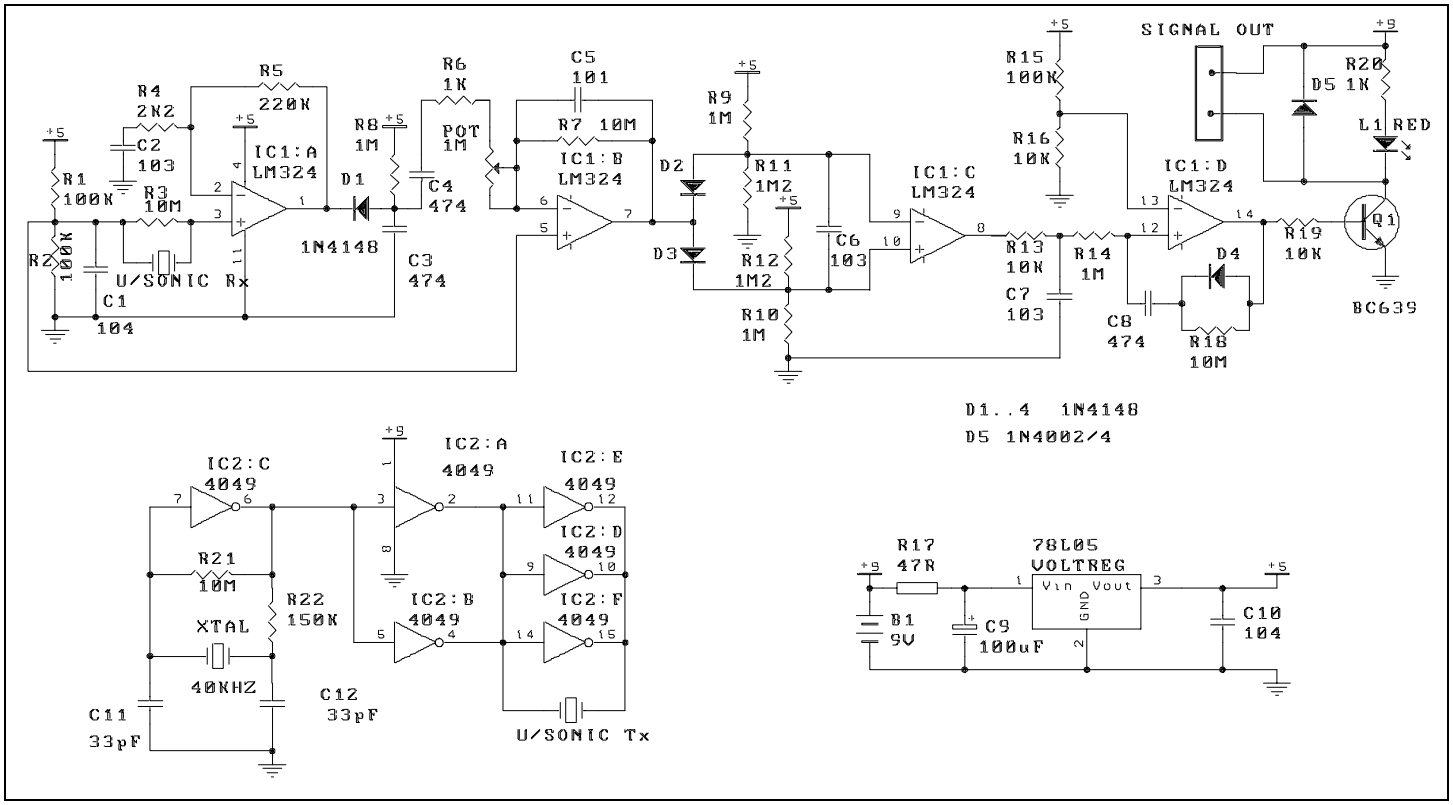
INFORMATION

The manufacturers specifications (pdf format) for the ultrasonic transducers see under Kit 20 or Kit 49 at

**http://kitsrus.com/kits.html**

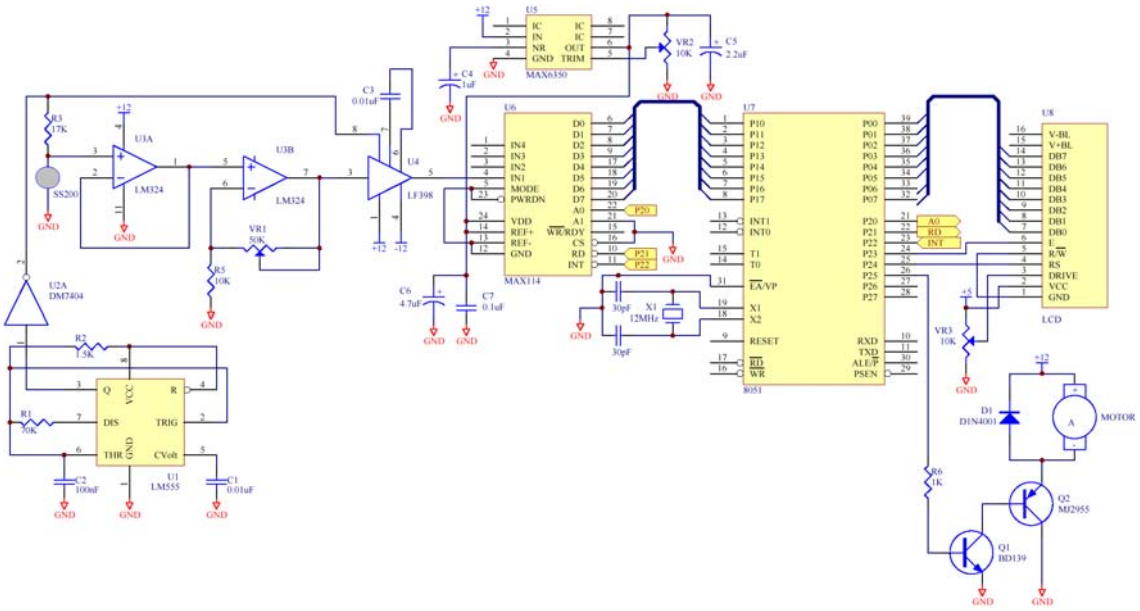
The data sheet for the LM324 & 4049 is available at

**www.nsc.com**



ELECTRONIC GARDENER Menu

by Andy Haryono Kartika




**Abstract**

This electronic gardener is used to maintain the soil moisture level, so the plants can grow healthy. The soil moisture value from the sensor is processed by microcontroller to activate the watering valve. If the soil moisture level is below from the expected value, then the watering valve will be active. But, if the soil moisture level is higher than the expected value, the watering valve will be off. We can control the expected soil moisture level by arranging the value of Pot VR1. The LCD (Liquid Crystal Display) will display the soil moisture level and the expected level. The main part of this electronic gardener is a microcontroller chip 8051. This chip is used as LCD driver, ADC driver and valve driver.


**Hardware Description**

I use the soil moisture sensor 200SS, which have an output range from 0 cb to 200 cb (centibar). This sensor must be operated with square wave signal, which generated by the timer circuit using LM555 and 7404. Basically, the sensor circuit is like an voltage divider circuit, so the resistance of the sensor is equal to the soil moisture value. Because the reading from the sensor is only valid when the bias signal from the timer circuit reach the peak amplitude, so I must use the sample and hold component LF398 to keep the value untill next reading cycle.

To convert the analog signal to digital signal, I use MAX114 which have four inputs channel and 8 bits output. The soil moisture level goes to the first channel, while the expected value goes to the second channel. The other channels can be connected to other sensors in different areas, so we can control the soil moisture in many areas. But, of course the sensors must be placed far enough from the others.

The micrcontroller chip drive the ADC chip by sending low signal to CS and RD pin of the MAX114 and get the digital data from DB0-DB7 pins after the INT pin goes low. These digital signals will be converted to decimal format, then send to the LCD. Because the function of the LCD is only to display data, so the RW pin can be

connected to ground. While the other control pins (i.e EN and RS pins) are controlled by the microcontroller.
The microcontroller will compare the expected soil moisture value with the sensor reading and determine whether the watering valve is active or not.

**Software Description**

I use Assembly program to design the controller for the microcontroller 8051. There are four main procedures included on the design.

1. ADC Driver

   This procedure is used to drive the ADC chip MAX114. Here is the listing program.

```
ADC   : JBC   P2.0,Two
        MOV   Moist,R3
        SETB  P2.0
        SJMP  Drive
Two   : MOV   Expect,R3
Drive : SETB  P2.1
Wait  : JB    P2.2,Wait
        MOV   R3,P0
        CLR   P2.1
        LJMP  One
        RET
```

2. Data Conversion

   This procedure is used to convert the soil moisture and expected level from ADC Driver procedure to the ASCII format, so they can be displayed on the LCD.

```
Convert: MOV   Repeat,#3
         MOV   A,R3
Loop :   MOV   B,#10
         DIV   AB
         MOV   DPL,B
         PUSH  DPL
         DJNZ  Repeat,Loop
         RET
```

3. LCD Driver

   This procedure is used to control the LCD. First the LCD must be initialized using InitLCD procedure. Then, to write an instruction register we must use IR procedure, while to write an data register we must use DR procedure. The main procedure of the LCD driver is Display procedure.

```
IR :     CLR   P2.3
         CLR   P2.4
         LCALL Delay
         SETB  P2.3
         MOV   P1,R4
         LCALL Delay
         CLR   P2.3
         LCALL Delay
         RET
DR :     CLR   P2.3
         SETB  P2.4
         LCALL Delay
         SETB  P2.3
         MOV   P1,R4
         LCALL Delay
         CLR   P2.3
         LCALL Delay
         RET
InitLCD: MOV   R4,#30h
         LCALL IR
```

```
          LCALL IR
          LCALL IR
          MOV   R4,#38h
          LCALL IR
          MOV   R4,#08h
          LCALL IR
          MOV   R4,#01H
          LCALL IR
          MOV   R4,#06h
          LCALL IR
          MOV   R4,#0Ch
          LCALL IR
          RET
Display:  MOV   R4,#8Ch         ; Column 12th Row 1st position
          LCALL IR
          MOV   R4,First_Moist  ; Display soil moisture level
          LCALL DR
          MOV   R4,Second_Moist
          LCALL DR
          MOV   R4,Third_Moist
          LCALL DR
          MOV   R4,#CCh         ; Column 12th Row 2nd position
          LCALL IR
          MOV   R4,First_Exp    ; Display expected level
          LCALL DR
          MOV   R4,Second_Exp
          LCALL DR
          MOV   R4,Third_Exp
          LCALL DR
```

4. Device Driver

   This procedure is used to control the watering valve. It will compare the expected level with soil moisture level, and decide whether the watering valve is active or not.

```
Valve:    MOV   A,Moist
          CJNE  A,Expect,Level
          JMP   Off
Level:    JNC   Off
          SETB  P2.5
          JMP   Exit
Off :     CLR   P2.5
Exit:     RET
```

5. Main Program

```
Program:  LCALL InitLCD
          LCALL ADC
          MOV   R3,Moist
          LCALL Convert
          POP   DPL
          MOV   First_Moist,DPL
          POP   DPL
          MOV   Second_Moist,DPL
          POP   DPL
          MOV   Third_Moist,DPL
          MOV   R3,Expect
          LCALL Convert
          POP   DPL
          MOV   First_Exp,DPL
          POP   DPL
          MOV   Second_Exp,DPL
          POP   DPL
```

```
MOV   Third_Exp,DPL
LCALL Display
```

**Parts List**

| Part Label | Part Description |
|---|---|
| U1 | LM555 Timer |
| U2 | 7404 Not Gate |
| U3 | LM324 Operational Amplifier |
| U4 | LF398 Sample And Hold |
| U5 | MAX6350 Voltage Reference |
| U6 | MAX114 Analog to Digital Converter |
| U7 | 8051 Microcontroller |
| U8 | M1632 Liquid Crystal Display |
| Q1 | BD139 |
| Q2 | MJ2955 |
| C1 | 0.01uF |
| C2 | 100nF |
| C3 | 0.01uF |
| C4 | 1uF |
| C5 | 2.2uF |
| C6 | 4.7uF |
| C7 | 0.1uF |
| D1 | D1N4001 |
| R1 | 70K |
| R2 | 1.5K |
| R3 | 17K |
| R4 | 10K |
| R5 | 1K |
| VR1 | Pot 50K |
| VR2 | Pot 10K |
| VR3 | Pot 10K |

Menu

OKI Semiconductor

This version: Jan. 1998
Previous version: Aug. 1996

# MSM80C85AHRS/GS/JS

**8-Bit CMOS MICROPROCESSOR**

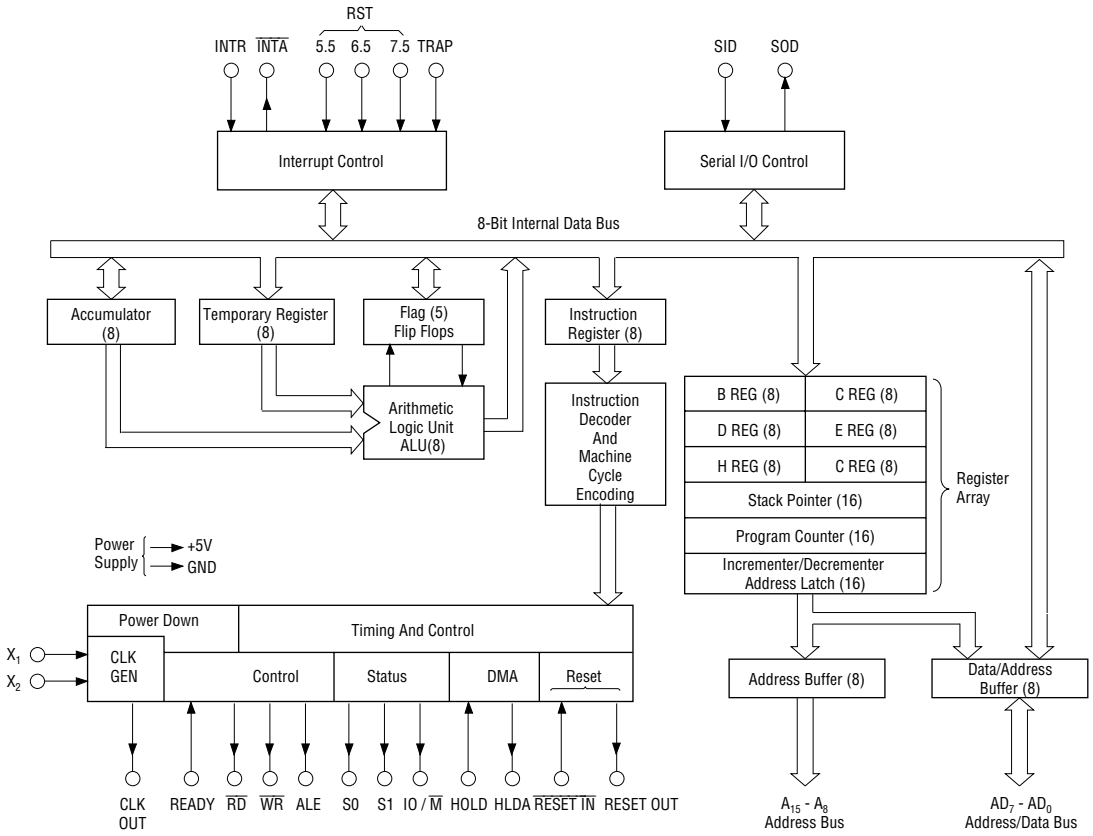
## GENRAL DESCRIPTION

The MSM80C85AH is a complete 8-bit parallel; central processor implemented in silicon gate C-MOS technology and compatible with MSM80C85A.
It is designed with higher processing speed (max.5 MHz) and lower power consumption compared with MSM80C85A and power down mode is provided, thereby offering a high level of system integration.
The MSM80C85AH uses a multiplexed address/data bus. The address is split between the 8-bit address bus and the 8-bit data bus. The on-chip address latch : of a MSM81C55-5 memory product allows a direct interface with the MSM80C85AH.

## FEATURES

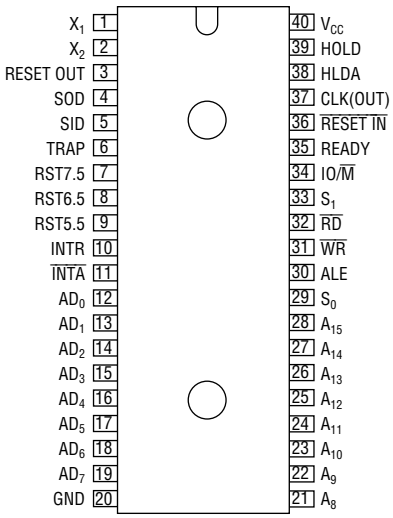
- Power down mode (HALT-HOLD)
- Low Power Dissipation: 50mW(Typ)
- Single + 3 to + 6 V Power Supply
- –40 to + 85°C, Operating Temperature
- Compatible with MSM80C85A
- 0.8 µs instruction Cycle ($V_{CC}$ = 5V)
- On-Chip Clock Generator (with External Crystal)
- On-Chip System Controller; Advanced Cycle Status Information Available for Large System Control
- Bug operation in MSM80C85AH is fixed
- Four Vectored interrupt (One is non-maskable) Plus the 8080A-compatible interrupt.
- Serial, In/Serial Out Port
- Decimal, Binary and Double Precision Arithmetic
- Addressing Capability to 64K Bytes of Memory
- TTL Compatible
- 40-pin Plastic DIP(DIP40-P-600-2.54): (Product name: MSM80C85AHRS)
- 44-pin Plastic QFJ(QFJ44-P-S650-1.27): (Product name: MSM80C85AHJS)
- 44-pin Plastic QFP(QFP44-P-910-0.80-2K): (Product name: MSM80C85AHGS-2K)

## FUNCTIONAL BLOCK DIAGRAM

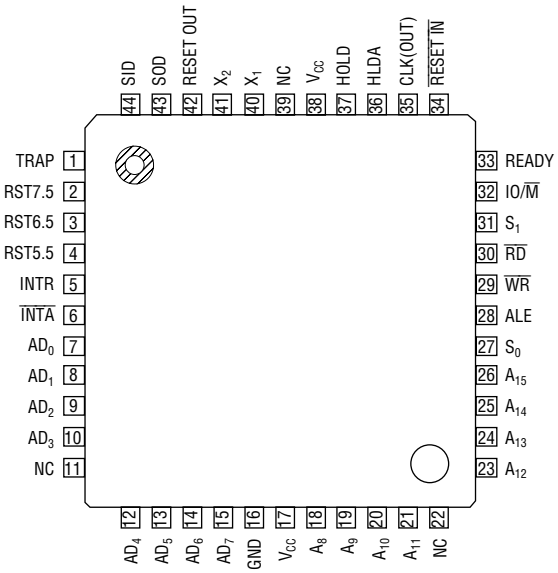


## PIN CONFIGURATION (TOP VIEW)

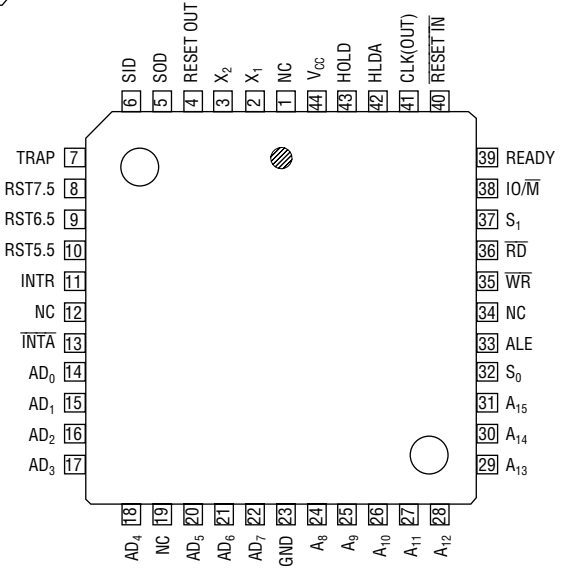
### 40 pin Plastic DIP




### 44 pin Plastic QFP




### 44 pin Plastic QFJ



## MSM80C85AH FUNCTIONAL PIN DEFINITION

The following describes the function of each pin:

| Symbol | Function |
|---|---|
| $A_8$ - $A_{15}$<br>(Output, 3-state) | Address Bus: The most significant 8-bits of the memory address or the 8-bits of the I/O address, 3-stated during Hold and Halt modes and during RESET. |
| $A_0$ - $A_7$<br>(Input/Output)<br>3-state | Multiplexed Address/Data Bus: Lower 8-bits of the memory address (or I/O address) appear on the bus during the first clock cycle (T state) of a machine cycle. It then becomes the data bus during the second and third clock cycles. |
| ALE<br>(Output) | Address Latch Enable: It occurs during the first clock state of a machine cycle and enables address to get latched into the on-chip latch peripherals. The falling edge of ALE is set to guarantee setup and hold times for the address information. The falling edge ALE can also be used to strobe the status information ALE is never 3-state. |
| $S_0$ , $S_1$ , IO/$\overline{M}$<br>(Output) | Machine cycle status:<br>IO/$\overline{M}$ $S_1$ $S_0$ States<br>0 0 1 Memory write<br>0 1 0 Memory read<br>1 0 1 I/O write<br>1 1 0 I/O read<br>0 1 1 Opcode fetch<br>IO/$\overline{M}$ $S_1$ $S_0$ States<br>1 1 1 Interrupt Acknowledge<br>. 0 0 Halt = 3-state<br>. × × Hold (high impedance)<br>. × × Reset × = unspecified<br>$S_1$ can be used as an advanced R/W status. IO/$\overline{M}$, $S_0$ and $S_1$ become valid at the beginning of a machine cycle and remain stable throughout the cycle. The falling edge of ALE may be used to latch the state of these lines. |
| $\overline{RD}$<br>(Output, 3-state) | READ control: A low level on $\overline{RD}$ indicates the selected memory or I/O device is to be read that the Data Bus is available for the data transfer, 3-stated during Hold and Halt modes and during RESET. |
| $\overline{WR}$<br>(Output, 3-state) | WRITE control: A low level on $\overline{WR}$ indicates the data on the Data Bus is to be written into the selected memory or I/O location. Data is set up at the trailing edge of $\overline{WR}$, 3-stated during Hold and Halt modes and during RESET. |
| READY<br>(Input) | If READY is high during a read or write cycle, it indicates that the memory or peripheral is ready to send or receive data. If READY is low, the cpu will wait an integral number of clock cycles for READY to go high before completing the read or write cycle READY must conform to specified setup and hold times. |
| HOLD<br>(Input) | HOLD indicates that another master is requesting the use of the address and data buses. The cpu, upon receiving the hold request, will relinquish the use of the bus as soon as the completion of the current bus transfer. Internal processing can continue. The processor can regain the bus only after the HOLD is removed. When the HOLD is acknowledged, the Address, Data, $\overline{RD}$, $\overline{WR}$, and IO/$\overline{M}$ lines are 3-stated. And status of power down is controlled by HOLD. |
| HLDA<br>(Output) | HOLD ACKNOWLEDGE: Indicates that the cpu has received the HOLD request and that it will relinquish the bus in the next clock cycle. HLDA goes low after the Hold request is removed. The cpu takes the bus one half clock cycle after HLDA goes low. |
| INTR<br>(Output) | INTERRUPT REQUEST: Is used as a general purpose interrupt. It is sampled on during the next to the last clock cycle of an instruction and during Hold and Halt states. If it is active, the Program Counter (PC) will be inhibited from incrementing and an $\overline{INTA}$ will be issued. During this cycle a RESTART or CALL instruction can be inserted to jump to the interrupt service routine. The INTR is enabled and disabled by software. It is disabled by Reset and immediately after an interrupt is accepted. Power down mode is reset by INTR. |
| $\overline{INTA}$<br>(Output) | INTERRUPT ACKNOWLEDGE: Is used instead of (and has the same timing as) $\overline{RD}$ during the instruction cycle after an INTR is accepted. |
| RST 5.5<br>RST 6.5<br>RST 7.5<br>(Input) | RESTART INTERRUPTS: These three inputs have the same timing as INTR except they cause an internal RESTART to be automatically inserted. The priority of these interrupts is ordered as shown in Table 1. These interrupts have a higher priority than INTR. In addition, they may be individually masked out using the SIM instruction. Power down mode is reset by these interrupts. |
| TRAP<br>(Input) | Trap interrupt is a nonmaskable RESTART interrupt. It is recognized at the same timing as INTR or RST 5.5 - 7.5. It is unaffected by any mask or Interrupt Disable. It has the highest priority of any interrupt. (See Table 1.) Power down mode is reset by input of TRAP. |

| Symbol | Function |
|---|---|
| $\overline{\text{RESET IN}}$ (Input) | Sets the Program Counter to zero and resets the Interrupt Enable and HLDA flip-flops and release power down mode. The data and address buses and the control lines are 3-stated during RESET and because of the asynchronous nature of $\overline{\text{RESET IN}}$, the processor's internal registers and flags may be altered by RESET with unpredictable results. $\overline{\text{RESET IN}}$ is a Schmitt-triggered input, allowing connection to an R-C network for power-on RESET delay. The cpu is held in the reset condition as long as $\overline{\text{RESET IN}}$ is applied. |
| RESET OUT (Output) | Indicated cpu is being reset. Can be used as a system reset. The signal is synchronized to the processor clock and lasts an integral number of clock periods. |
| $X_1$, $X_2$ (Input) | $X_1$ and $X_2$ are connected to a crystal to drive the internal clock generator. $X_1$ can also be an external clock input from a logic gate. The input frequency is divided by 2 to give the processor's internal operating frequency. |
| CLK (Output) | Clock Output for use as a system clock. The period of CLK is twice the $X_1$, $X_2$ input period. |
| SID (Input) | Serial input data line. The data on this line is loaded into accumulator bit 7 whenever a RIM instruction is executed. |
| SOD (Output) | Serial output data line. The output SOD is set or reset as specified by the SIM instruction. |
| $V_{CC}$ | + 5 Volt supply |
| GND | Ground Reference. |

**Table 1 Interrupt Priority, Restart Address, and Sensitivity**

| Name | Priority | Address Branched To (1) When Interrupt Occurs | Type Trigger |
|---|---|---|---|
| TRAP | 1 | 24H | Rising edge and high level unit sampled. |
| RST 7.5 | 2 | 3CH | Rising edge (latched). |
| RST 6.5 | 3 | 34H | High level unitl sampled. |
| RST 5.5 | 4 | 2CH | High level until sampled. |
| INTR | 5 | (2) | High level until sampled. |

Notes: (1) The processor pushes the PC on the stack before branching to the indicated address.
(2) The address branched to depends on the instruction provided to the cpu when the interrupt is acknowledged.

## FUNCTIONAL DESCRIPTION

The MSM80C85AH is a complete 8-bit parallel central processor. It is designed with silicon gate C-MOS technology and requires a single +5 volt supply. Its basic clock speed is 5 MHz, thus improving on the present MSM80C85A's performance with higher system speed and power down mode. Also it is designed to fit into a minimum system of two IC's: The CPU (MSM80C85AH), and a RAM/IO (MSM81C55-5)

The MSM80C85AH has twelve addressable 8-bit register pairs. Six others can be used interchangeably as 8-bit registers or 16-bit register pairs. The MSM80C85AH register set is as follows:

| Mnemonic | Register | Contents |
|---|---|---|
| ACC or A | Accumulator | 8-bits |
| PC | Program Counter | 16-bit address |
| BC, DE, HL | General-Purpose Registers; data pointer (HL) | 8-bit × 6 or 16-bits × 3 |
| SP | Stack Pointer | 16-bit address |
| Flags or F | Flag Register | 5 flags (8-bit space) |

The MSM80C85AH uses a multiplexed Data Bus. The address is spilt between the higher 8-bit Address Bus and the lower 8-bit Address/Data Bus. During the first T state (clock cycle) of a machine cycle the low order address is sent out on the Address/Data Bus. These lower 8-bits may be latched externally by the Address Latch Enable signal (ALE). During the rest of the machine cycle the data bus is used for mamory or I/O data.

The MSM80C85AH provides $\overline{RD}$, $\overline{WR}$, $S_0$, $S_1$, and IO/$\overline{M}$ signals for bus control. An Interrupt Acknowledge signal ($\overline{INTA}$) is also provided. Hold and all Interrupts are synchronized with the processor's internal clock. The MSM80C85AH also provides Serial Input Data (SID) and Serial Output Data (SOD) lines for a simple serial interface.
In addition to these features, the MSM80C85AH has three maskable, vector interrupt pins, one nonmaskable TRAP interrupt and power down mode with HALT and HOLD.

## INTERRUPT AND SERIAL I/O

The MSM80C85AH has 5 interrupt inputs: INTR, RST 5.5 RST 6.5, RST 7.5, and TRAP. INTR is identical in function to the 8080A INT. Each of the three RESTART inputs, 5.5, 6.5, and 7.5, has a programmable mask. TRAP is also a RESTART interrupt but it is nonmaskable.

The three maskable interrupts cause the internal execution of RESTART ( saving the program counter in the stack branching to the RESTART address) it the interrupts are enable and if the interrupt mask is not set. The nonmaskable TRAP causes the internal execution of a RESTART vector independent of the state of the interrupt enable or masks. (See Table 1.)

There are two different types of inputs in the restart interrupt. RST 5.5 and RST 6.5 are high level-sensitive like INTR (and INT on the 8080A) and are recognized with the same timing as INTR. RST 7.5 is rising edge-sensitive.

For RST 7.5, only a pulse is required to set an internal flip-flop which generates the internal interrupt request. The RST 7.5 request flip-flop remains set until the request is serviced. Then it is reset automatically, This flip-flop may also be reset by using the SIM instruction or by issuing a $\overline{\text{RESET IN}}$ to the MSM80C85AH. The RST 7.5 internal flip-flop will be set by a pulse on the RST 7.5 pin even when the RST 7.5 interrupt is masked out.

The interrupts are arranged in a flixed priority that determines which interrupt is to be recognized if more than one is pending, as follows: TRAP-highest priority, RST 7.5, RST 6.5, RST 5.5, INTR-lowest priority. This priority scheme does not take into account the priority of a routine that was started by a higher priority interrupt. RST 5.5 can interrupt an RST 7.5 routine if the interrupt are re-enabled before the end of the RST 7.5 routine.

The TRAP interrupt is useful for catastrophic evens such as power failure or bus error. The TRAP input is recognized just as any other interrupt but has the highest priority. It is not affected by any flag or mask. The TRAP input is both edge and level sensitive. The TRAP input must go high and remain high until it is acknowledged. It will not be recognized again until it goes low, then high again. This avoids any false triggering due to noise or logic glitches. Figure 3 illustrates the TRAP interrupt request circuitry within the MSM80C85AH. Note that the servicing of any interrupt (TRAP, RST 7.5, RST 6.5, RST 5.5,INTR) disables all future interrupts (except TRAPs) until an El instruction is executed.

The TRAP interrupt is special in that it disables interrupts, but preserves the previous interrupt enable status. Performing the first RIM instruction following a TRAP interrupt allows you to determine whether interrupts were enabled or disabled prior to the TRAP. All subsequent RIM instructions provide current interrupt enable status. Performing a RIM instruction following INTR or RST 5.5-7.5 will provide current interrupt Enable status, revealing that Interrupts are disabled.

The serial I/O system is also controlled by the RIM and SIM instructions. SID is read by RIM, and SIM sets the SOD data.

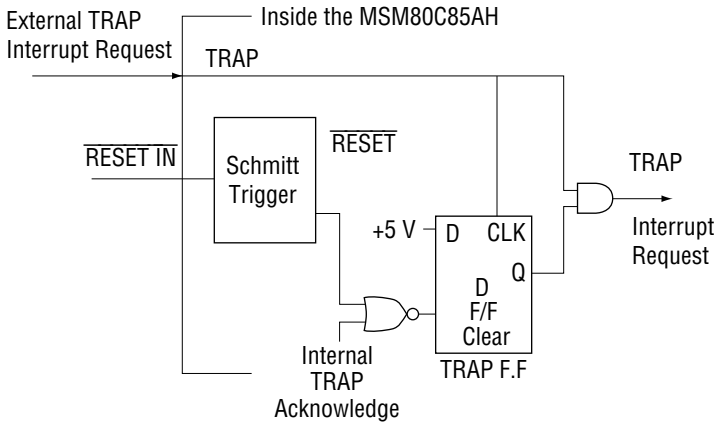



**Figure 3 Trap and $\overline{\text{RESET IN}}$ Circuit**

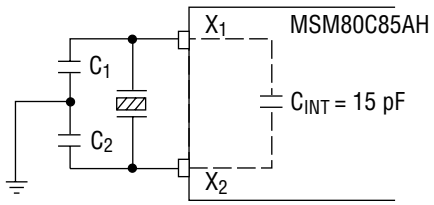
## DRIVING THE $X_1$ AND $X_2$ INPUTS

You may drive the clock inputs of the MSM80C85AH with a crystal, or an external clock source. The driving frequency must be at least 1 MHz, and must be twice the desired internal clock frequency; hence, the MSM80C85AH is operated with a 6 MHz crystal (for 3 MHz clock). If a crystal is used, it must have the following characteristics:

Parallel resonance at twice the clock frequency desired
$C_L$ (load capacitance) ≤ 30 pF
$C_S$ (shunt capacitance) ≤ 7 pF
$R_S$ (equivalent shunt resistance) ≤ 75 ohms
Drive level: 10 mW
Frequency tolerance: ±0.05% (suggested)

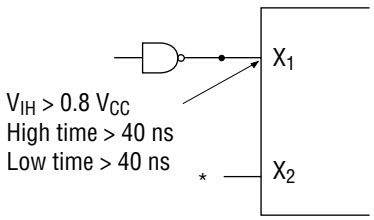
Note the use of the capacitors between $X_1$, $X_2$ and ground. These capacitors are required to assure oscillator startup at the correct frequency.
Figure 4 shows the recommended clock driver circuits. Note in B that a pull-up resistor is required to assure that the high level voltage of the input is at least 4 V.
For driving frequencies up to and including 6 MHz you may supply the driving signal to X, and leave $X_2$ open-circuited (Figure 4B). To prevent self-oscillation of the MSM80C85AH, be sure that $X_2$ is not coupled back to $X_1$ through the driving circuit.

**A. Quartz Crystal Clock Driver**




33 pF Capacitor required for crystal frequency 10 to 6.25 MHz
50 pF Capacitor required for crystal frequency 6.25 to 4 MHz
100 pF Capacitor required for crystal frequency <4 MHz

**B. 1 - 10 MHz Input Frequency External Clock Drive Circuit**




* $X_2$ Left Floating

Note: Since the constant values may vary depending on oscillator, consult the manufacturer of the oscillator used when designing a circuit.

**Figure 4 Clock Driver Circuits**

## BASIC SYSTEM TIMING

The MSM80C85AH has a multiplexed Data Bus. ALE is used as a strobe to sample the lower 8-bits of address on the Data Bus. Figure 5 shows an instruction fetch, memory read and I/O write cycle (as would occur during processing of the OUT instruction). Note that during the I/O write and read cycle that the I/O port address is copied on both the upper and lower half of the address.

There are seven possible types of machine cycles. Which of these seven takes place is defined by the status of the three status lines ($IO/\overline{M}$, $S_1$, $S_0$) and the three control signals ($\overline{RD}$, $\overline{WR}$,and $\overline{INTA}$). (See Table 2.) The status line can be used as advanced controls (for device selection, for example), since they become active at the $T_1$ state, at the outset of each machine cycle. Control lines $\overline{RD}$ and $\overline{WR}$ become active later, at the time when the transfer of data is to take place, so are used as command lines.

A machine cycle normally consists of three T states, with the exception of OPCODE FETCH, which normally has either four or six T states (unless WAIT or HOLD states are forced by the receipt of $\overline{READY}$ or HOLD inputs). Any T state must be one of ten possible states, shown in Table 3.

**Table 2 MSM80C85AH Machine Cycle Chart**

| Machine Cycle | | Status | | | Control | | |
|---|---|---|---|---|---|---|---|
| | | $IO/\overline{M}$ | $S_1$ | $S_0$ | $\overline{RD}$ | $\overline{WR}$ | $\overline{INTA}$ |
| Opcode Fetch | (OF) | 0 | 1 | 1 | 0 | 1 | 1 |
| Memory Read | (MR) | 0 | 1 | 0 | 0 | 1 | 1 |
| Memory Write | (MW) | 0 | 0 | 1 | 1 | 0 | 1 |
| I/O Read | (IOR) | 1 | 1 | 0 | 0 | 1 | 1 |
| I/O Write | (IOW) | 1 | 0 | 1 | 1 | 0 | 1 |
| Acknowledge of INTR | (INA) | 1 | 1 | 1 | 1 | 1 | 0 |
| Bus Idle | (BI): DAD | 0 | 1 | 0 | 1 | 1 | 1 |
| | ACK. OF RST, TRAP | 1 | 1 | 1 | 1 | 1 | 1 |
| | HALT | TS | 0 | 0 | TS | TS | 1 |

**Table 3 MSM80C85AH Machine State Chart**

| Machine State | Status & Buses | | | | Control | | |
|---|---|---|---|---|---|---|---|
| | $S_1$, $S_0$ | IO/$\overline{M}$ | $A_8 - A_{15}$ | $AD_0 - AD_7$ | $\overline{RD}$, $\overline{WR}$ | $\overline{INTA}$ | $\overline{ALE}$ |
| $T_1$ | × | × | × | × | 1 | 1 | 1 (1) |
| $T_2$ | × | × | × | × | × | × | 0 |
| $T_{WAIT}$ | × | × | × | × | × | × | 0 |
| $T_3$ | × | × | × | × | × | × | 0 |
| $T_4$ | 1 | 0 (2) | × | TS | 1 | 1 | 0 |
| $T_5$ | 1 | 0 (2) | × | TS | 1 | 1 | 0 |
| $T_6$ | 1 | 0 (2) | × | TS | 1 | 1 | 0 |
| $T_{RESET}$ | × | TS | TS | TS | TS | 1 | 0 |
| $T_{HALT}$ | 0 | TS | TS | TS | TS | 1 | 0 |
| $T_{HOLD}$ | × | TS | TS | TS | TS | 1 | 0 |

0 = Logic "0"
1 = Logic "1"
TS = High Impedance
× = Unspecified

**Notes:** (1) ALE not generated during 2nd and 3rd machine cycles of DAD instruction.
(2) IO/$\overline{M}$ = 1 during $T_4$ - $T_6$ of INA machine cycle.

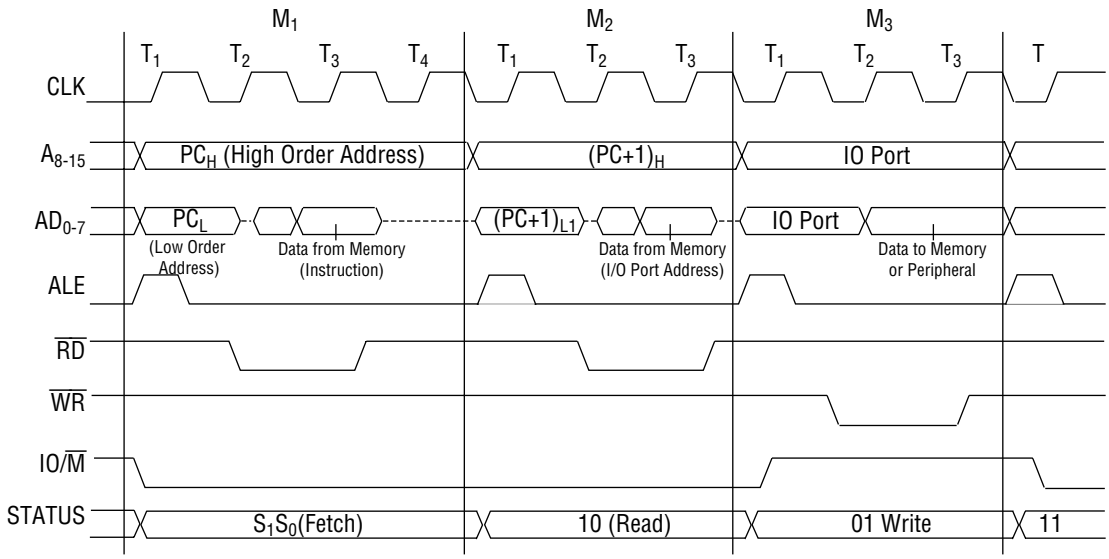



**Figure 5 MSM80C85AH Basic System Timing**

## POWER DOWN Mode

The MSM80C85AH is compatible with the MSM80C85A in function and POWER DOWN mode. This reduces power consumption further.
There are two methods available for starting this POWER DOWN mode. One is through software control by using the HALT command and the other is under hardware control by using the pin HOLD. This mode is released by the HOLD, RESET, and interrupt pins (TRAP, RST7.5, RST6.5 RST5.5, or INTR). (See Table 4.)

Since the sequence of HALT, HOLD, RESET, and INTERRUPT is compatible with MSM80C85A, every the POWER DOWN mode can be used with no special attention.

**Table 4 POWER DOWN Mode Releasing Method**

| | |
|---|---|
| Start by means of Halt command | Released by using pins RESET and INTERRUPT (not by pin HOLD) |
| Start by means of HOLD pin | Released by using RESET and HOLD pins (not by interrupt pins) |

### (1) Start by means of HALT command (See Figures 6 and 7.)

The POWER DOWN mode can be started by executing the HALT command.
At this time, the system is put into the HOLD status and therefore the POWER DOWN mode cannot be released even when the HOLD is released later.
In this case, the POWER DOWN mode can be released by means of the RESET or interrupt.

### (2) Start by means of HOLD pin (See Figure 8.)

During the execution of commands other than the HALT, the POWER DOWN mode is started when the system is put into HOLD status by means of the HOLD pin.
Since no interrupt works during the execution of the HOLD, the POWER DOWN mode cannot be released by means of interrupt pins. In this case, the POWER DOWN mode can be released either by means of the RESET pin or by releasing the HOLD status by means of HOLD pin.

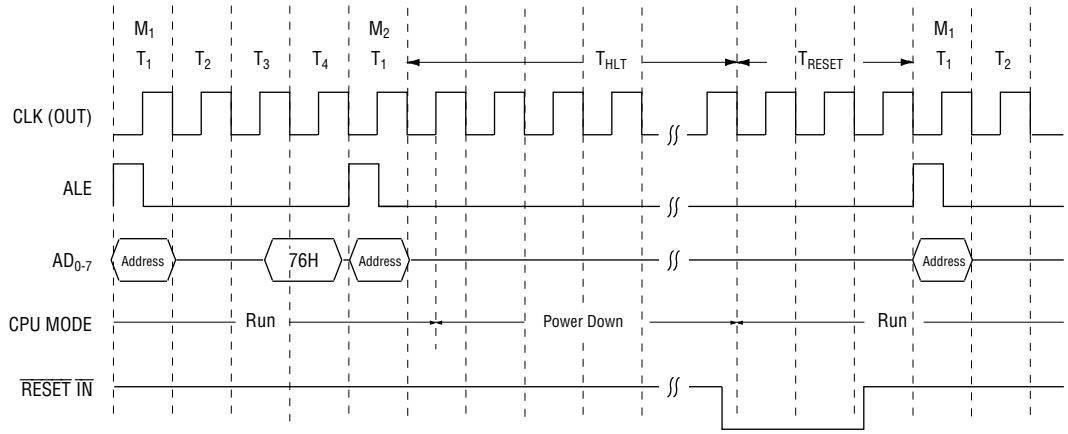



**Figure 6 Started by HALT and Released by $\overline{\text{RESET IN}}$**

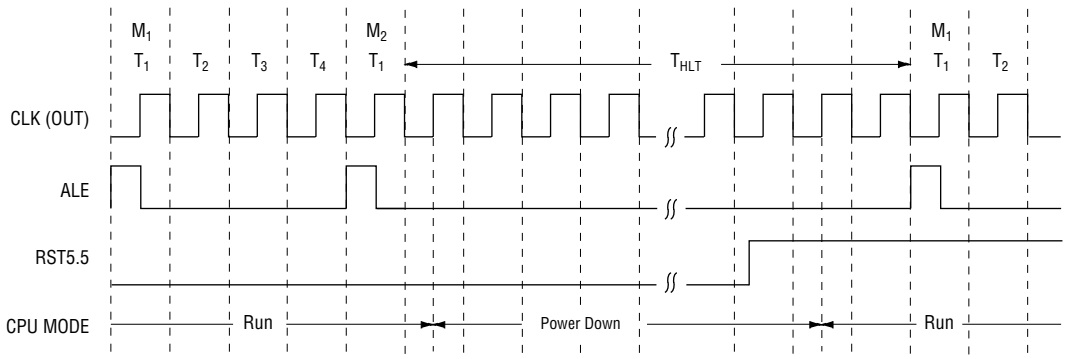



**Figure 7 Started by HALT and Released by RST5.5**

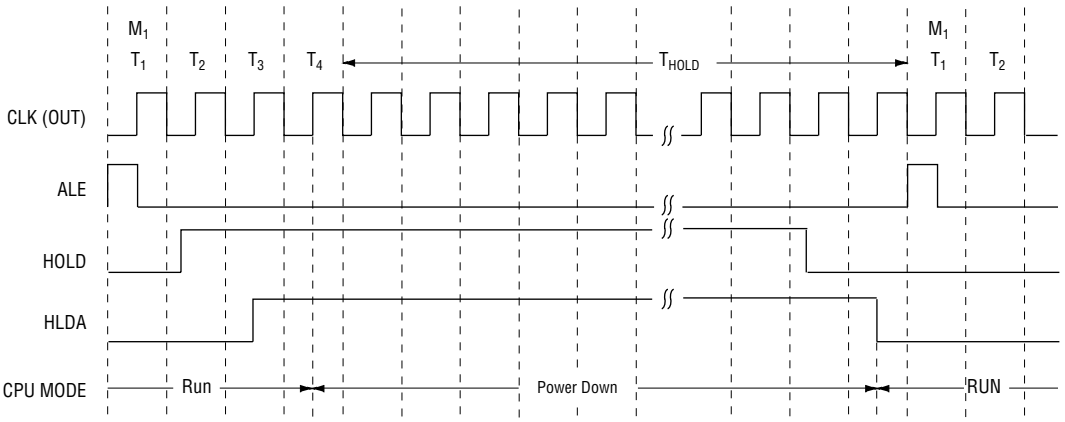



**Figure 8 Started and Released by HOLD**

## ABSOLUTE MAXIMUM RATINGS

| Parameter | Symbol | Condition | Limits | | | Units |
|---|---|---|---|---|---|---|
| | | | MSM80C85AHRS | MSM80C85AHGS | MSM80C85AHJS | |
| Power Supply Voltage | $V_{CC}$ | With respect to GND | –0.5 - 7 | | | V |
| Input Voltage | $V_{IN}$ | | –0.5 - $V_{CC}$ +0.5 | | | V |
| Output Voltage | $V_{OUT}$ | | –0.5 - $V_{CC}$ +0.5 | | | V |
| Storage Temperature | $T_{STG}$ | — | –55 - +150 | | | °C |
| Power Dissipation | $P_D$ | Ta = 25°C | 1.0 | 0.7 | 1.0 | W |

## OPERATING RANGE

| Parameter | Symbol | Limits | Unit |
|---|---|---|---|
| Power Supply Voltage | $V_{CC}$ | 3 - 6 | V |
| Operating Temperature | $T_{OP}$ | –40 - +85 | °C |

## RECOMMENDED OPERATING CONDITIONS

| Parameter | Symbol | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|
| Power Supply Voltage | $V_{CC}$ | 4.5 | 5 | 5.5 | V |
| Operating Temperature | $T_{OP}$ | –40 | +25 | +85 | °C |
| "L" Input Voltage | $V_{IL}$ | –0.3 | — | +0.8 | V |
| "H" Output Voltage | $V_{IH}$ | 2.2 | — | $V_{CC}$ +0.3 | V |
| "L" $\overline{\text{RESET IN}}$ Input Voltage | $V_{ILR}$ | –0.3 | — | +0.8 | V |
| "H" $\overline{\text{RESET IN}}$ Input Voltage | $V_{IHR}$ | 3.0 | — | $V_{CC}$ +0.3 | V |

## DC CHARACTERISTICS

| Parameter | Symbol | Conditions | | Min. | Typ. | Max. | Unit |
|---|---|---|---|---|---|---|---|
| "L" Output Voltage | $V_{OL}$ | $I_{OL}$ = 2.5 mA | $V_{CC}$ = 4.5 V - 5.5 V, Ta = –40°C - +85°C | — | — | 0.4 | V |
| "H" Output Voltage | $V_{OH}$ | $I_{OH}$ = –2.5 mA | | 3.0 | — | — | V |
| | | $I_{OH}$ = –100 µA | | $V_{CC}$ - 0.4 | — | — | V |
| Input Leak Current | $I_{LI}$ | $0 \le V_{IN} \le V_{CC}$ | | –10 | — | 10 | µA |
| Output Leak Current | $I_{LO}$ | $0 \le V_{OUT} \le V_{CC}$ | | –10 | — | 10 | µA |
| Operating Supply Current | $I_{CC}$ | $T_{cyc}$ = 200 ns, $C_L$ = 0 pF at reset | | — | 10 | 20 | mA |
| | | $T_{cyc}$ = 200 ns, $C_L$ = 0 pF at power down mode | | — | 5 | 10 | mA |

## AC CHARACTERISTICS

(Ta = –40°C ~ 85°C, $V_{CC}$ = 4.5 V ~ 5.5 V)

| Parameter | Symbol | Condition | Min. | Max. | Unit |
|---|---|---|---|---|---|
| CLY Cycle Period | $t_{CYC}$ | | 200 | 2000 | ns |
| CLY Low Time | $t_1$ | | 40 | — | ns |
| CLY High Time | $t_2$ | | 70 | — | ns |
| CLY Rise and Fall Time | $t_r$, $t_f$ | | — | 30 | ns |
| $X_1$ Rising to CLK Rising | $t_{XKR}$ | | 25 | 120 | ns |
| X1 Rising to CKK Falling | $t_{XKF}$ | | 30 | 150 | ns |
| $A_{8\sim15}$ Valid to Leading Edge of Control (1) | $t_{AC}$ | | 115 | — | ns |
| $AD_{0\sim7}$ Valid to Leading Edge of Control | $t_{ACL}$ | | 115 | — | ns |
| $AD_{0\sim15}$ Valid Data in | $t_{AD}$ | | — | 350 | ns |
| Address Float After Leading Edge of $\overline{RD}$ $\overline{INTA}$ | $t_{AFR}$ | | — | 0 | ns |
| $A_{8\sim15}$ Valid Before Trailing Edge of ALE (1) | $t_{AL}$ | | 50 | — | ns |
| $AD_{0\sim7}$ Valid Before Trailing Edge of ALE | $t_{ALL}$ | | 50 | — | ns |
| READY Valid from Address Valid | $t_{ARY}$ | | — | 100 | ns |
| Address ($A_{8\sim15}$) Valid After Control | $t_{CA}$ | | 60 | — | ns |
| Width of Control Law ($\overline{RD}$, $\overline{WR}$, $\overline{INTA}$) | $t_{CC}$ | | 230 | — | ns |
| Trailing Edge of Control to Leading Edges of ALE | $t_{CL}$ | | 25 | — | ns |
| Data Valid to Trailing Edge of $\overline{WR}$ | $t_{DW}$ | | 230 | — | ns |
| HLDA to Bus Enable | $t_{HABE}$ | | — | 150 | ns |
| Bus Float After HLDA | $t_{HABF}$ | $t_{CYC}$=200 ns | — | 150 | ns |
| HLDA Valid to Trailing Edge of CLK | $t_{HACK}$ | CL=150 pF | 40 | — | ns |
| HOLD Hold Time | $t_{HDH}$ | | 0 | — | ns |
| HOLD Step Up Time to Trailing Edge of CLK | $t_{HDS}$ | | 120 | — | ns |
| INTR Hold Time | $t_{INH}$ | | 0 | — | ns |
| INTR, RST and TRAP Setup Time to Falling Edge of CLK | $t_{INS}$ | | 150 | — | ns |
| Address Hold Time After ALE | $t_{LA}$ | | 50 | — | ns |
| Trailing Edge of ALE to Leading Edge of Control | $t_{LC}$ | | 60 | — | ns |
| ALE Low During CLK High | $t_{LCK}$ | | 50 | — | ns |
| ALE to Valid Data During Read | $t_{LDR}$ | | — | 270 | ns |
| ALE to Valid Data During Write | $t_{LDW}$ | | — | 140 | ns |
| ALE Width | $t_{LL}$ | | 80 | — | ns |
| ALE to READY Stable | $t_{LRY}$ | | — | 30 | ns |
| Trailing Edge of $\overline{RD}$ to Re-enabling of Address | $t_{RAE}$ | | 90 | — | ns |
| RD (or INTA) to Valid Data | $t_{RD}$ | | — | 150 | ns |
| Control Trailing Edge to Leading Edge of Next Control | $t_{RV}$ | | 220 | — | ns |
| Data Hold Time After $\overline{RD}$ $\overline{INTA}$ (7) | $t_{RDH}$ | | 0 | — | ns |
| READY Hold Time | $t_{RYH}$ | | 0 | — | ns |
| READY Setup Time to Leading Edge of CLK | $t_{RYS}$ | | 100 | — | ns |
| Data Valid After Trailing Edge of $\overline{WR}$ | $t_{WD}$ | | 60 | — | ns |
| LEADING Edge of $\overline{WR}$ to Data Vaild | $t_{WDL}$ | | — | 20 | ns |

Notes: (1) $A_8$ - $A_{15}$ address Specs apply to IO/$\overline{M}$, $S_0$ and $S_1$.
(2) Test condition: $t_{CYC}$=200 ns $C_L$=150 pF
(3) For all output timing where $C_L$=150 pF use the following correction factors:
25 pF $\leq C_L <$ 150 pF : –0.10ns/pF
150 pF $< C_L \leq$ 200 pF : +0.30ns/pF
(4) Output timings are measured with purely capacitive load.
(5) All timings are measured to output voltage $V_L$=0.8 V, $V_H$=2.2 V, and 1.5 V with 10 ns rise and fall time on inputs.
(6) To calculate timing specifications at other values of $t_{CYC}$ use Table 7.
(7) Data hold time is guaranteed under all loading conditions.

**Input Waveform for A.C. Tests:**

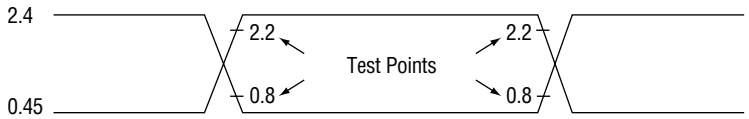



**Table 7 Bus Timing Specification as a $T_{CYC}$ Dependent**

(Ta = -40°C - +85°C, $V_{CC}$ = 4.5 V - 5.5 V, $C_L$ = 150 pF)

| | MSM80C85AH | | |
|---|---|---|---|
| $t_{AL}$ | — | (1/2)T - 50 | Min |
| $t_{LA}$ | — | (1/2)T - 50 | Min |
| $t_{LL}$ | — | (1/2)T - 20 | Min |
| $t_{LCK}$ | — | (1/2)T - 50 | Min |
| $t_{LC}$ | — | (1/2)T - 40 | Min |
| $t_{AD}$ | — | (5/2+N)T - 150 | Max |
| $t_{RD}$ | — | (3/2+N)T - 150 | Max |
| $t_{RAE}$ | — | (1/2)T - 10 | Min |
| $t_{CA}$ | — | (1/2)T - 40 | Min |
| $t_{DW}$ | — | (3/2+N)T -70 | Min |
| $t_{WD}$ | — | (1/2)T - 40 | Min |
| $t_{CC}$ | — | (3/2+N)T - 70 | Min |
| $t_{CL}$ | — | (1/2)T - 75 | Min |
| $t_{ARY}$ | — | (3/2)T - 200 | Max |
| $t_{HACK}$ | — | (1/2)T - 60 | Min |
| $t_{HABF}$ | — | (1/2)T + 50 | Max |
| $t_{HABE}$ | — | (1/2)T + 50 | Max |
| $t_{AC}$ | — | (2/2)T - 85 | Min |
| $t_1$ | — | (1/2)T - 60 | Min |
| $t_2$ | — | (1/2)T - 30 | Min |
| $t_{RV}$ | — | (3/2)T - 80 | Min |
| $t_{LDR}$ | — | (2+N)T -130 | Max |

**Note:** N is equal to the total WAIT states.
T = $t_{CYC}$

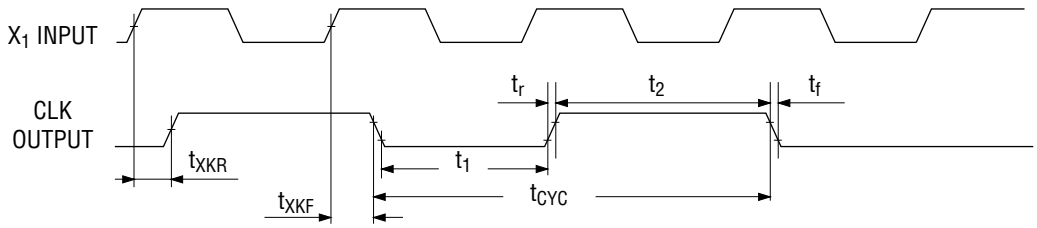
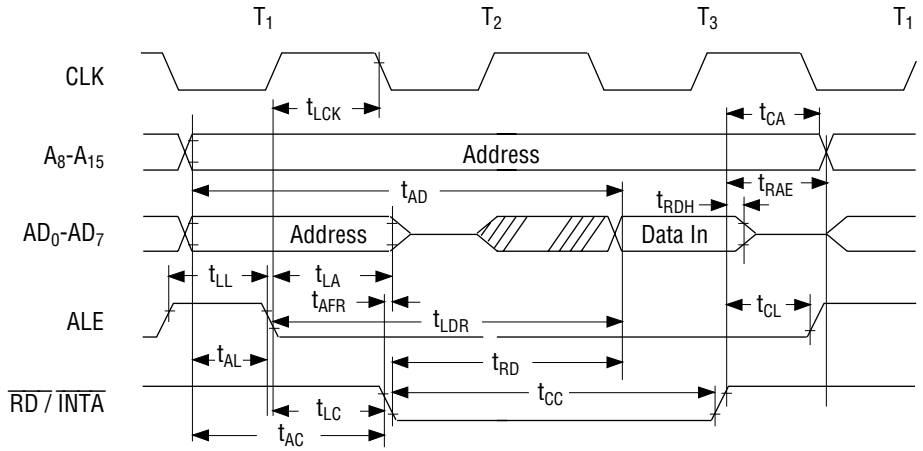
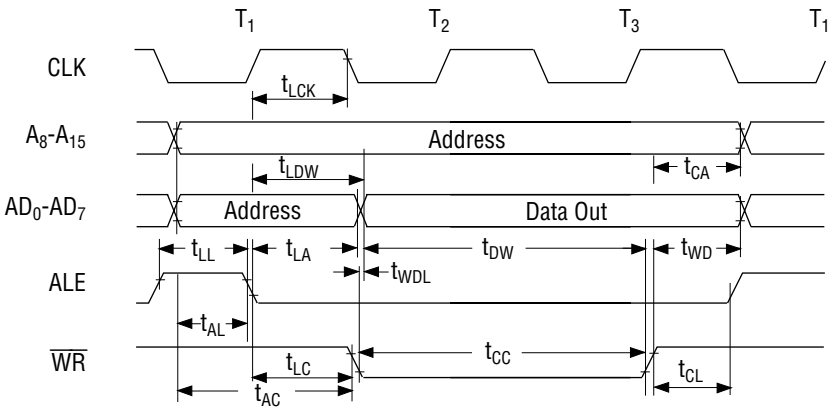



**Figure 6 Clock Timing Waveform**

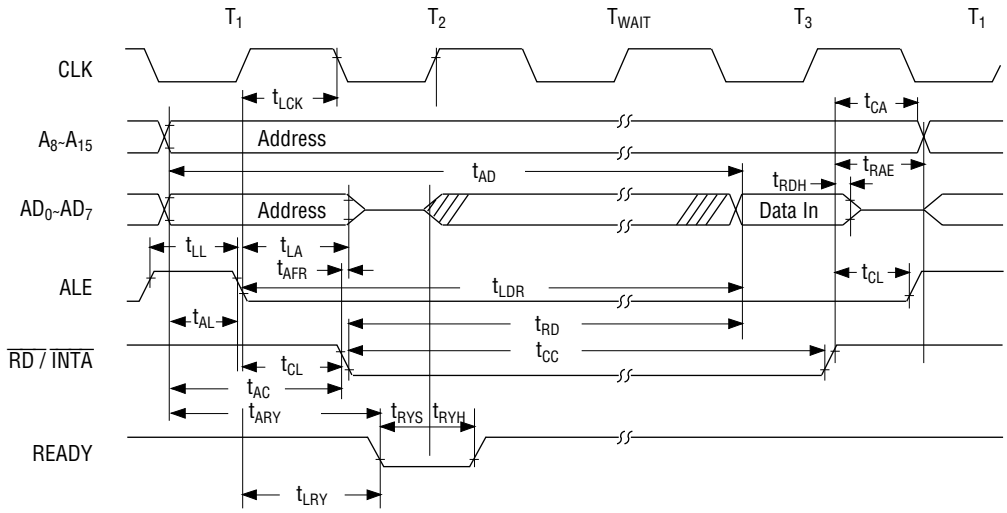
## READ OPERATION




## WRITE OPERATION



Read operation with Wait Cycle (Typical)–
same READY timing applies to WRITE operation




**Note:** READY must remain stable during setup and hold times.

**Figure 7 MSM80C85AH Bus Timing, With and Without Wait**

## HOLD OPERATION

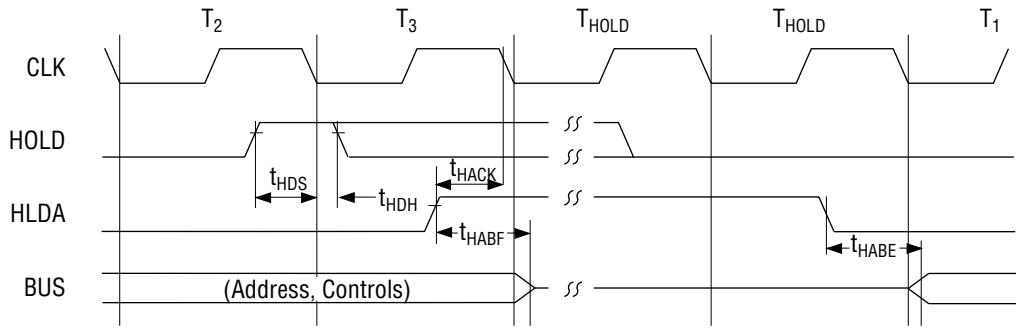
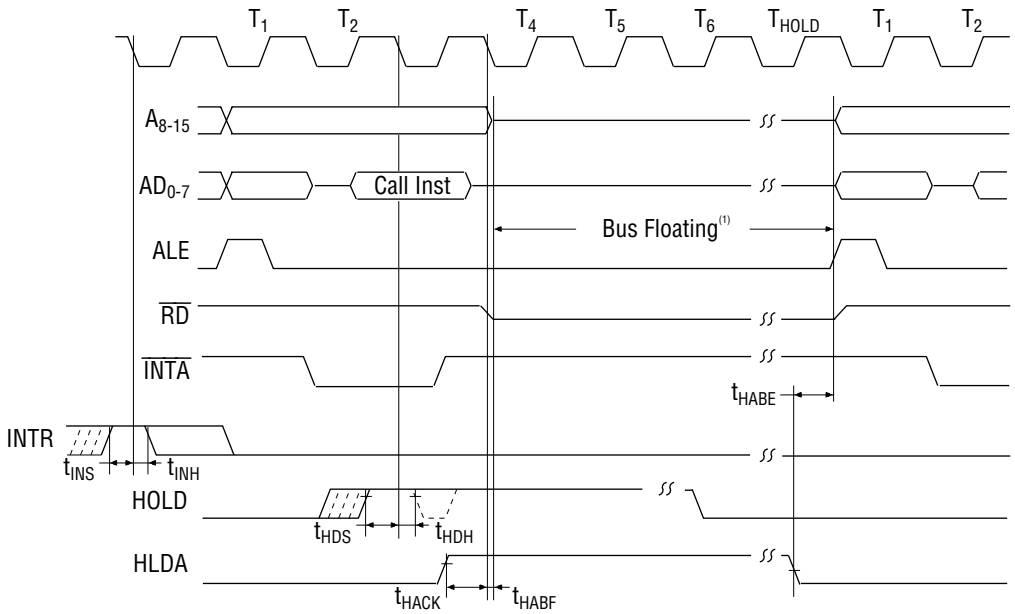



**Figure 8 MSM80C85AH Hold Timing**




**NOTE:** (1) IO/$\overline{M}$ is also floating during this time.

**Figure 9 MSM80C85AH Interrupt and Hold Timing**

**Table 8 Instruction Set Summary**

| Mnemonic | Description | Instruction Code (1) $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ | Clock (2) Cycles |
|---|---|---|---|---|---|---|---|---|---|---|
| MOVE, LOAD, AND STORE | | | | | | | | | | |
| MOVr1 r2 | Move register to register | 0 | 1 | D | D | D | S | S | S | 4 |
| MOV M r | Move register to memory | 0 | 1 | 1 | 1 | 0 | S | S | S | 7 |
| MOV r M | Move memory to register | 0 | 1 | D | D | D | 1 | 1 | 0 | 7 |
| MVI r | Move immediate register | 0 | 0 | D | D | D | 1 | 1 | 0 | 7 |
| MVI M | Move immediate memory | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 10 |
| LXI B | Load immediate register Pair B & C | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 10 |
| LXI D | Load immediate register Pair D & E | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 10 |
| LXI H | Load immediate register Pair H & L | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 10 |
| LXI SP | Load immediate stack pointer | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 | 10 |
| STAX B | Store A indirect | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 7 |
| STAX D | Store A indirect | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 7 |
| LDAX B | Load A indirect | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 7 |
| LDAX D | Load A indirect | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 7 |
| STA | Store A direct | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 13 |
| LDA | Load A direct | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 13 |
| SHLD | Store H & L direct | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 16 |
| LHLD | Load H & L direct | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 16 |
| XCHG | Exchange D & E H & L registers | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 4 |
| STACK OPS | | | | | | | | | | |
| PUSH B | Push register Pair B & C on stack | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 1 | 12 |
| PUSH D | Push register Pair D & E on stack | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 12 |
| PUSH H | Push register Pair H & L on stack | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 12 |
| PUSH PSW | Push A and Flags on stack | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 12 |
| POP B | Pop register Pair B & C off stack | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 10 |
| POP D | Pop register Pair D & E off stack | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 10 |
| POP H | Pop register Pair H & L off stack | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 10 |
| POP PSW | Pop A and Flags off stack | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 10 |
| XTHL | Exchange top of stack H & L | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 16 |
| SPHL | H & L to stack pointer | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 6 |
| JUMP | | | | | | | | | | |
| JMP | Jump unconditional | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 10 |
| JC | Jump on carry | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 0 | 7/10 |
| JNC | Jump on no carry | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 7/10 |
| JZ | Jump on zero | 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 7/10 |
| JNZ | Jump on no zero | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 7/10 |
| JP | Jump on positive | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 7/10 |
| JM | Jump on minus | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 7/10 |
| JPE | Jump on parity even | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 7/10 |
| JPO | Jump on parity odd | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 7/10 |
| PCHL | H & L to program counter | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 6 |
| CALL | | | | | | | | | | |
| CALL | Call unconditional | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 1 | 18 |
| CC | Call on carry | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 9/18 |
| CNC | Call on no carry | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 9/18 |
| CZ | Call on zero | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 9/18 |
| CNZ | Call on no zero | 1 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 9/18 |
| CP | Call on positive | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 9/18 |
| CM | Call on minus | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 9/18 |
| CPE | Call on parity even | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 9/18 |
| CPO | Call on parity odd | 1 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 9/18 |

**Table 8 Instruction Set Summary cont'd**

| Mnemonic | Description | Instruction Code (1) $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ | Clock (2) Cycles |
|---|---|---|---|---|---|---|---|---|---|---|
| RETURN | | | | | | | | | | |
| RET | Return | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 10 |
| RC | Return on carry | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 0 | 6/12 |
| RNC | Return on no carry | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 6/12 |
| RZ | Return on zero | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 6/12 |
| RNZ | Return on no zero | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 6/12 |
| RP | Return on positive | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 6/12 |
| RM | Return on minus | 1 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 6/12 |
| RPE | Return on parity even | 1 | 1 | 1 | 0 | 1 | 0 | 0 | 0 | 6/12 |
| RPO | Return on parity odd | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 6/12 |
| RESTART | | | | | | | | | | |
| RST | Restart | 1 | 1 | A | A | A | 1 | 1 | 1 | 12 |
| INPUT/OUTPUT | | | | | | | | | | |
| IN | Input | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 10 |
| OUT | Output | 1 | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 10 |
| INCREMENT AND DECREMENT | | | | | | | | | | |
| INR r | Increment register | 0 | 0 | D | D | D | 1 | 0 | 0 | 4 |
| DCR r | Decrement register | 0 | 0 | D | D | D | 1 | 0 | 1 | 4 |
| INR M | Increment memory | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 0 | 10 |
| DCR M | Decrement memory | 0 | 0 | 1 | 1 | 0 | 1 | 0 | 1 | 10 |
| INX B | Increment B & C registers | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 6 |
| INX D | Increment D & E registers | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 6 |
| INX H | Increment H & L registers | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 6 |
| INX SP | Increment stack pointer | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 6 |
| DCX B | Decrement B & C | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 6 |
| DCX D | Decrement D & E | 0 | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 6 |
| DCX H | Decrement H & L | 0 | 0 | 1 | 0 | 1 | 0 | 1 | 1 | 6 |
| DCX SP | Decrement stack pointer | 0 | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 6 |
| ADD | | | | | | | | | | |
| ADD r | Add register to A | 1 | 0 | 0 | 0 | 0 | S | S | S | 4 |
| ADC r | Add register to A with carry | 1 | 0 | 0 | 0 | 1 | S | S | S | 4 |
| ADD M | Add memory to A | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 7 |
| ADC M | Add memory to A with carry | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 7 |
| ADI | Add immediate to A | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 7 |
| ACI | Add immediate to A with carry | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 7 |
| DAD B | Add B & C to H & L | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 10 |
| DAD D | Add D & E to H & L | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 10 |
| DAD H | Add H & L to H & L | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 10 |
| DAD SP | Add stack pointer to H & L | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | 10 |
| SUBTRACT | | | | | | | | | | |
| SUB r | Subtract register from A | 1 | 0 | 0 | 1 | 0 | S | S | S | 4 |
| SBB r | Subtract register from A with borrow | 1 | 0 | 0 | 1 | 1 | S | S | S | 4 |
| SUB M | Subtract memory from A | 1 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 7 |
| SBB M | Subtract memory from A with borrow | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 7 |
| SUI | Subtract immediate from A | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 0 | 7 |
| SBI | Subtract immediate from A with borrow | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 7 |

**Table 8 Instruction Set Summary cont'd**

| Mnemonic | Description | Instruction Code (1) $D_7$ | $D_6$ | $D_5$ | $D_4$ | $D_3$ | $D_2$ | $D_1$ | $D_0$ | Clock (2) Cycles |
|---|---|---|---|---|---|---|---|---|---|---|
| LOGICAL | | | | | | | | | | |
| ANA r | Add register with A | 1 | 0 | 1 | 0 | 0 | S | S | S | 4 |
| XRA r | Exclusive Or register with A | 1 | 0 | 1 | 0 | 1 | S | S | S | 4 |
| ORA r | Or register with A | 1 | 0 | 1 | 1 | 0 | S | S | S | 4 |
| CMP r | Compare register with A | 1 | 0 | 1 | 1 | 1 | S | S | S | 4 |
| ANA M | And memory with A | 1 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 7 |
| XRA M | Exclusive Or Memory with A | 1 | 0 | 1 | 0 | 1 | 1 | 1 | 0 | 7 |
| ORA M | Or memory with A | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 7 |
| CMP M | Compare memory with A | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | 7 |
| ANI | And immediate with A | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 7 |
| XRI | Exclusive Or immediate with A | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 7 |
| ORI | Or immediate with A | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 7 |
| CPI | Compare immediate with A | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 0 | 7 |
| ROTATE | | | | | | | | | | |
| RLC | Rotate A left | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 4 |
| RRC | Rotate A right | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 4 |
| RAL | Rotate A left through carry | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 4 |
| RAR | Rotate A right through carry | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 4 |
| SPECIALS | | | | | | | | | | |
| CMA | Complement A | 0 | 0 | 1 | 0 | 1 | 1 | 1 | 1 | 4 |
| STC | Set carry | 0 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 4 |
| CMC | Complement carry | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 4 |
| DAA | Decimal adjust A | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 4 |
| CONTROL | | | | | | | | | | |
| EI | Enable Interrupts | 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 4 |
| DI | Disable Interrupts | 1 | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 4 |
| NOP | No-operation | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 4 |
| HLT | Halt (Power down) | 0 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 5 |
| RIM | Read Interrupt Mask | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 4 |
| SIM | Set Interrupt Mask | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 4 |

Notes: (1) DDD or SSS. B 000. C 001. D 010. E 011. H 100. L 101. Memory 110. A 111.
(2) Two possible cycle times, (6/12) indicate instruction cycles dependent on condition flags.

## Precautions for operation

(1) When the oscillation circuit is to be used, keep the RES input low until the oscillation is sufficiently stabilized after power is turned on.
(2) When power is turned on, the output level (SOD etc.) is unknown before the equipment is reset.
(3) Bug of MSM80C85A–2 at power down has fixed.
(4) Because Spike Noise would be output on HLDA, RESET OUT and CLK pins, depending on the customers condition of usage; please take into account this issue at System Board design.

## SUPPLEMENTARY EXPLANATION

(1) SIM instruction: The execution of the SIM instruction uses the contents of the accumulator to mask MSM80C85AH'S interrupts.

**Accumulator Setting Value**

| Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | — | — | R7.5 | MSE | M7.5 | M6.5 | M5.5 |

R7.5 (Reset interrupt 7.5 Flip-flop): When this bit is set to 1, the edge detecting flip-flop of RST 7.5 interrupt is reset.
MSE (Mask Set Enable): When this bit is set to 1, the interrupt mask bits are valid.
M7.5 (Mask RST7.5): When this bit is set to 1 and MSE bit is set to 1, RST7.5 interrupt is masked.
M6.5 (Mask RST6.5): When this bit is set to 1 and MSE bit is set to 1, RST6.5 interrupt is masked.
M5.5 (Mask RST5.5): When this bit is set to 1 and MSE bit is set to 1, RST 5.5 interrupt is masked.

(2) RIM instruction: When the contents of the accumulator are read out after RIM instruction has been executed, MSM80C85AH interrupt status can be known.

**Accumulator Reading Value**

| Bit 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|
| — | 17.5 | 16.5 | 15.5 | IE | M7.5 | M6.5 | M5.5 |

17.5 (Pending RST7.5): When RST7.5 interrupt is pending, "1" is read out.
16.5 (Pending RST6.5): When RST6.5 interrupt is pending, "1" is read out.
15.5 (Pending RST5.5): When RST5.5 interrupt is pending, "1" is read out.
IE (Interrupt Enable Flag): When interrupt is Enable, "1" is read out.
M7.5 (Mask RST7.5): When RST7.5 interrupt is masked, "1" is read out.
M6.5 (Mask RST6.5): When RST6.5 interrupt is masked, "1" is read out.
M5.5 (Mask RST5.5): When RST5.5 interrupt is masked ,"1" is read out.

## NOTICE ON REPLACING LOW-SPEED DEVICES WITH HIGH-SPEED DEVICES

The conventional low speed devices are replaced by high-speed devices as shown below.
When you want to replace your low speed devices with high-speed devices, read the replacement notice given on the next pages.

| High-speed device (New) | Low-speed device (Old) | Remarks |
|---|---|---|
| M80C85AH | M80C85A/M80C85A-2 | 8bit MPU |
| M80C86A-10 | M80C86A/M80C86A-2 | 16bit MPU |
| M80C88A-10 | M80C88A/M80C88A-2 | 8bit MPU |
| M82C84A-2 | M82C84A/M82C84A-5 | Clock generator |
| M81C55-5 | M81C55 | RAM.I/O, timer |
| M82C37B-5 | M82C37A/M82C37A-5 | DMA controller |
| M82C51A-2 | M82C51A | USART |
| M82C53-2 | M82C53-5 | Timer |
| M82C55A-2 | M82C55A-5 | PPI |

### Differences between MSM80C85AH and MSM80C85A/MSM80C85A-2

#### 1) Manufacturing Process

| Item | MSM80C85A | MSM80C85A-2 | MSM80C85AH |
|---|---|---|---|
| Manufacturing Process | 3μSi-CMOS | 2.5μSi-CMOS | 2μSi-CMOS |

#### 2) Functions

| Item | MSM80C85A | MSM80C85A-2 | MSM80C85AH |
|---|---|---|---|
| Power-down Function | Not provided | Provided (but may malfunction when HOLD is used) | Provided (The malfunction has been removed.) |
| Address output during T4 to T6 cycles | Undefined (compatible with Intel devices) | Not fixed | The contents of data in T3 cycle are retained (for low power consumption). |

#### 3) Electrical Characteristics
#### 3-1) Operating Conditions

| Parameter | Symbol | MSM80C85A | MSM80C85A-2 | MSM80C85AH |
|---|---|---|---|---|
| Power Supply Voltage | $V_{CC}$ | 4 to 6 V | 3 to 6 V | 3 to 6 V |

#### 3-2) DC Characteristics

| Parameter | Symbol | MSM80C85A | MSM80C85A-2 | MSM80C85AH |
|---|---|---|---|---|
| ''L''Level Output Voltage | $V_{OL}$ | 0.45 V maximum (+2 mA) | 0.45 V maximum (+2 mA) | 0.40 V maximum (+2.5 mA) |
| ''H''Level Output Voltage | $V_{OH}$ | 2.4 V minimum (-400 μA) | 2.4 V minimum (-400 μA) | 3.0 V maximum (-2.5 mA) |
| ''H''Level Output Voltage | $V_{OH}$ | 4.2 V minimum (-40 μA) | 4.2 V minimum (-40 μA) | $V_{CC}$-0.2 V minimum (-100 μA) |
| Supply Current (at RES) | $I_{CC}$ | 22 mA maximum (@3 MHz) | 20 mA maximum (@5 MHz) | 20 mA maximum (@5 MHz) |
| Supply Current (in PD) | $I_{CC}$ | None | 7 mA maximum (@5 MHz) | 10 mA maximum (@5 MHz) |

Notes: "at RES" means "at reset time" and "in PD" means "in power down mode".

As shown above, the $V_{OL}$ and $V_{OH}$ ranges the MSM80C85AH contain those of the MSM80C85A/MSM80C85A-2. Although the supply current range (at a power failure) of the MSM80C85AH does not contain that of the MSM80C85A-2, this does not affect the actual use of the MSM80C85AH.

#### 3-3) AC Characteristics

The AC characteristics (5 MHz) of the MSM80C85AH satisfy that (3 MHz) of the MSM80C85A. The MSM80C85AH also satisfies that (5MHz) of the MSM80C85A.

**AC Charasteristics**

| Symbol | | MSM80C85A | MSM80C85A-2 | MSM80C85AH |
|---|---|---|---|---|
| $t_{CYC}$ | Min | *320 ns* | 200 ns | 200 ns |
| $t_1$ | Min | *80 ns* | 40 ns | 40 ns |
| $t_2$ | Min | *120 ns* | 70 ns | 70 ns |
| $t_{XKR}$ | Min | *30 ns* | 25 ns | 25 ns |
| $t_{AC}$ | Max | *270 ns* | 115 ns | 115 ns |
| $t_{ACL}$ | Min | *240 ns* | 115 ns | 115 ns |
| $t_{AD}$ | Max | *575 ns* | 330 ns | 350 ns |
| $t_{AL}$ | Min | *115 ns* | 50 ns | 50 ns |
| $t_{ALL}$ | Min | *90 ns* | 50 ns | 50 ns |
| $t_{ARY}$ | Max | *220 ns* | 100 ns | 100 ns |
| $t_{CA}$ | Min | *120 ns* | 60 ns | 60 ns |
| $t_{CC}$ | Min | *400 ns* | 230 ns | 230 ns |
| $t_{CL}$ | Min | *50 ns* | 25 ns | 25 ns |
| $t_{DW}$ | Min | *420 ns* | 230 ns | 230 ns |
| $t_{HABE}$ | Min | *210 ns* | 150 ns | 150 ns |
| $t_{HABF}$ | Max | *210 ns* | 150 ns | 150 ns |
| $t_{HACK}$ | Min | *110 ns* | 40 ns | 40 ns |
| $t_{HDS}$ | Min | *170 ns* | 120 ns | 120 ns |
| $t_{INS}$ | Min | *160 ns* | 150 ns | 150 ns |
| $t_{LA}$ | Min | *100 ns* | 50 ns | 50 ns |
| $t_{LC}$ | Min | *130 ns* | 60 ns | 60 ns |
| $t_{LCK}$ | Min | *100 ns* | 50 ns | 50 ns |
| $t_{LDR}$ | Max | *460 ns* | 250 ns | 270 ns |
| $t_{LDW}$ | Max | *200 ns* | 140 ns | 140 ns |
| $t_{LL}$ | Min | *140 ns* | 80 ns | 80 ns |
| $t_{LRY}$ | Max | *110 ns* | 30 ns | 30 ns |
| $t_{RAE}$ | Min | *150 ns* | 90 ns | 90 ns |
| $t_{RD}$ | Max | *300 ns* | 150 ns | 150 ns |
| $t_{RV}$ | Min | *400 ns* | 220 ns | 220 ns |
| $t_{WD}$ | Min | *100 ns* | 60 ns | 60 ns |
| $t_{WDL}$ | Max | *40 ns* | 20 ns | 20 ns |

Notes: The italicized or underlined values indicate that they are different from those of the MSM80C85AH.

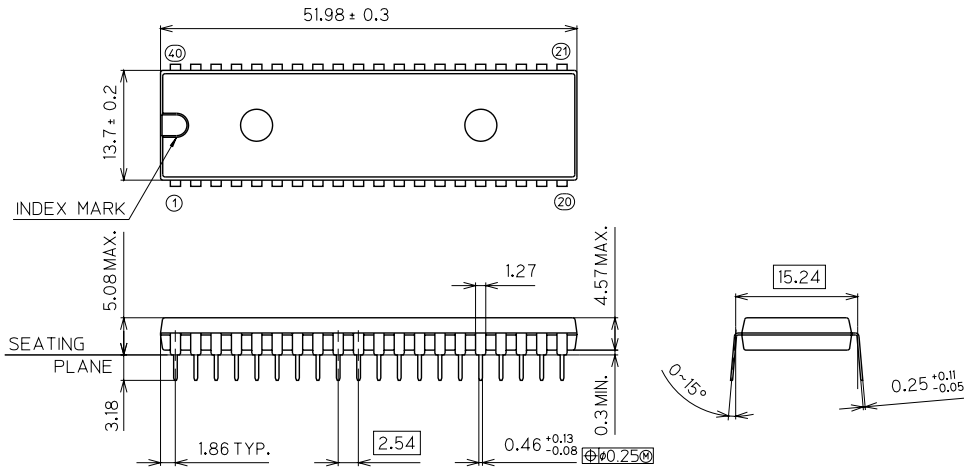
#### 4) Other notes

1) As the MSM80C85AH employs the 2 µ process, its noise characteristics may be a little different from those of the MSM80C85A. When devices are replaced for upgrading, it is recommended to perform noise evaluation. Especially, HLDA, RESOUT, and CLKOUT pins must be evaluated.
2) The MSM80C85AH basically satisfies the characteristics of the MSM80C85A-2 and the MSM80C85A, but their timings are a little different, Therefore, when critical timing is required in designing, it is recommended to evaluate operating margins at various temperatures and voltages.

## PACKAGE DIMENSIONS

(Unit : mm)

DIP40-P-600-2.54



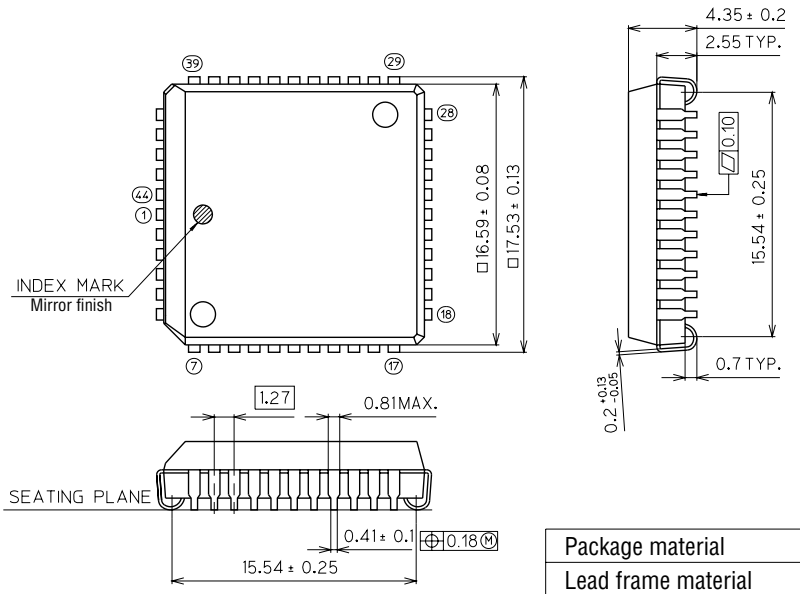

| | |
|---|---|
| Package material | Epoxy resin |
| Lead frame material | 42 alloy |
| Pin treatment | Solder plating |
| Solder plate thickness | 5 μm or more |
| Package weight (g) | 6.10 TYP. |

Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, SOJ, QFJ (PLCC), SHP and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage.
Therefore, before you perform reflow mounting, contact Oki's responsible sales person for the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

(Unit : mm)

QFJ44-P-S650-1.27



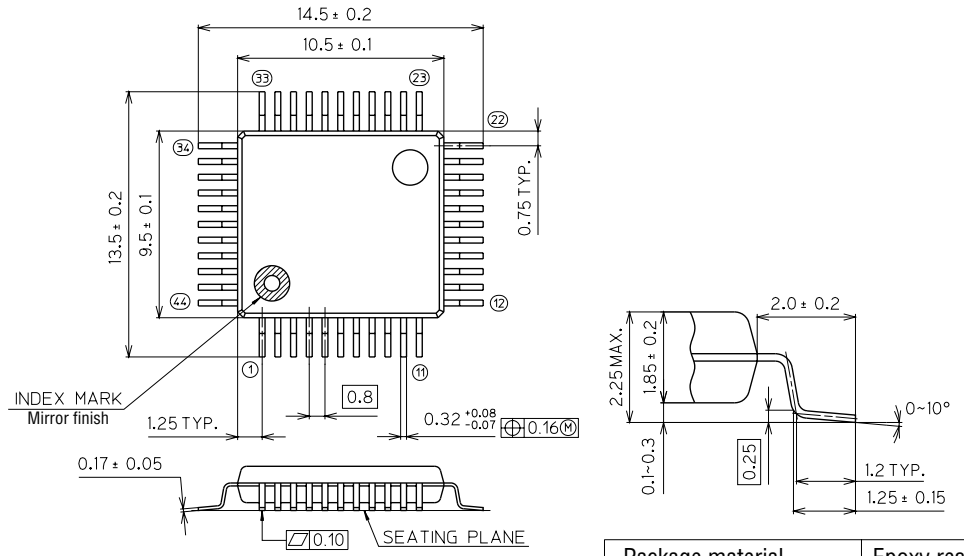

| | |
|---|---|
| Package material | Epoxy resin |
| Lead frame material | Cu alloy |
| Pin treatment | Solder plating |
| Solder plate thickness | 5 μm or more |
| Package weight (g) | 2.00 TYP. |

Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, SOJ, QFJ (PLCC), SHP and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage.
Therefore, before you perform reflow mounting, contact Oki's responsible sales person for the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

(Unit : mm)

QFP44-P-910-0.80-2K




| | |
|---|---|
| Package material | Epoxy resin |
| Lead frame material | 42 alloy |
| Pin treatment | Solder plating |
| Solder plate thickness | 5 μm or more |
| Package weight (g) | 0.41 TYP. |

Notes for Mounting the Surface Mount Type Package

The SOP, QFP, TSOP, SOJ, QFJ (PLCC), SHP and BGA are surface mount type packages, which are very susceptible to heat in reflow mounting and humidity absorbed in storage.
Therefore, before you perform reflow mounting, contact Oki's responsible sales person for the product name, package name, pin number, package code and desired mounting conditions (reflow method, temperature and times).

HARRIS SEMICONDUCTOR

# 80C86

March 1997

## CMOS 16-Bit Microprocessor

### Features

- **Compatible with NMOS 8086**
- **Completely Static CMOS Design**
  - **DC . . . . . 5MHz (80C86)**
  - **DC . . . . . 8MHz (80C86-2)**
- **Low Power Operation**
  - **ICCSB . . . . . 500μA Max**
  - **ICCOP . . . . . 10mA/MHz Typ**
- **1MByte of Direct Memory Addressing Capability**
- **24 Operand Addressing Modes**
- **Bit, Byte, Word and Block Move Operations**
- **8-Bit and 16-Bit Signed/Unsigned Arithmetic**
  - **Binary, or Decimal**
  - **Multiply and Divide**
- **Wide Operating Temperature Range**
  - **C80C86 . . . . . $0^oC$ to $+70^oC$**
  - **I80C86 . . . . . $-40^oC$ to $+85^oC$**
  - **M80C86 . . . . . $-55^oC$ to $+125^oC$**

### Description

The Harris 80C86 high performance 16-bit CMOS CPU is manufactured using a self-aligned silicon gate CMOS process (Scaled SAJI IV). Two modes of operation, minimum for small systems and maximum for larger applications such as multiprocessing, allow user configuration to achieve the highest performance level. Full TTL compatibility (with the exception of CLOCK) and industry standard operation allow use of existing NMOS 8086 hardware and software designs.

### Ordering Information

| PACKAGE | TEMP. RANGE | 5MHz | 8MHz | PKG. NO. |
|---|---|---|---|---|
| PDIP | $0^oC$ to $+70^oC$ | CP80C86 | CP80C86-2 | E40.6 |
| | $-40^oC$ to $+85^oC$ | IP80C86 | IP80C86-2 | E40.6 |
| PLCC | $0^oC$ to $+70^oC$ | CS80C86 | CS80C86-2 | N44.65 |
| | $-40^oC$ to $+85^oC$ | IS80C86 | IS80C86-2 | N44.65 |
| CERDIP | $0^oC$ to $+70^oC$ | CD80C86 | CD80C86-2 | F40.6 |
| | $-40^oC$ to $+85^oC$ | ID80C86 | ID80C86-2 | F40.6 |
| | $-55^oC$ to $+125^oC$ | MD80C86/B | MD80C86-2/B | F40.6 |
| SMD# | $-55^oC$ to $+125^oC$ | 8405201QA | 8405202QA | F40.6 |
| CLCC | $-55^oC$ to $+125^oC$ | MR80C86/B | MR80C86-2/B | J44.A |
| SMD# | $-55^oC$ to $+125^oC$ | 8405201XA | 8405202XA | J44.A |

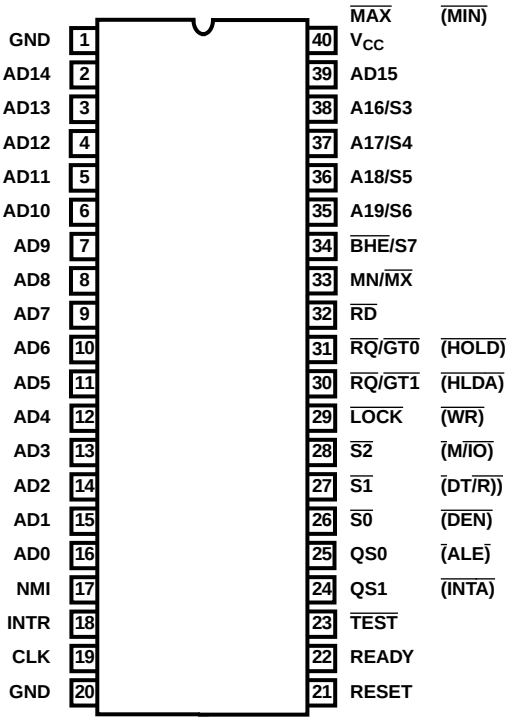
CAUTION: These devices are sensitive to electrostatic discharge. Users should follow proper IC Handling Procedures.



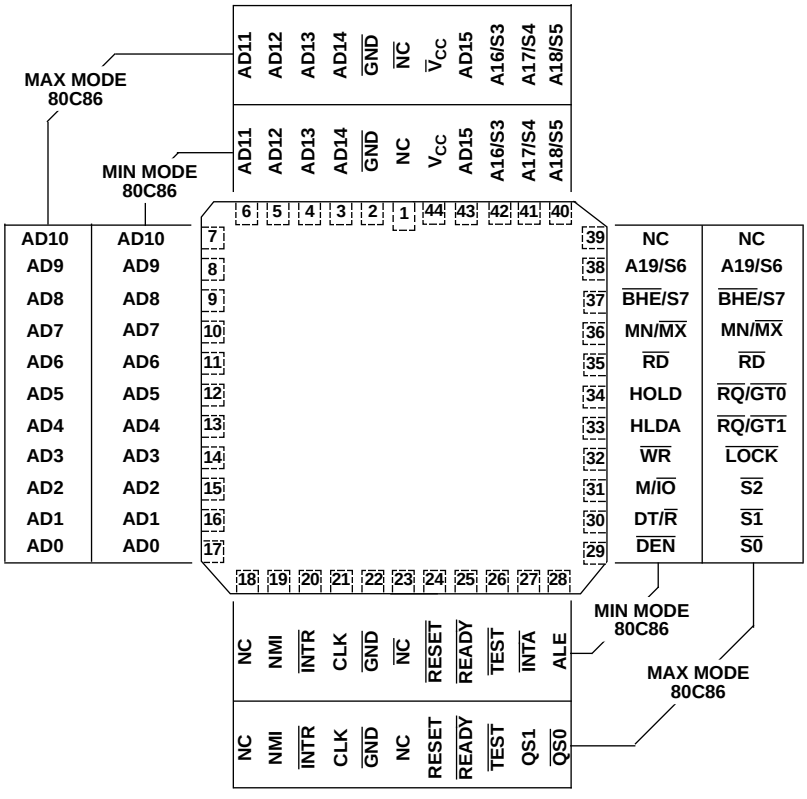
File Number **2957.1**

## Pinouts

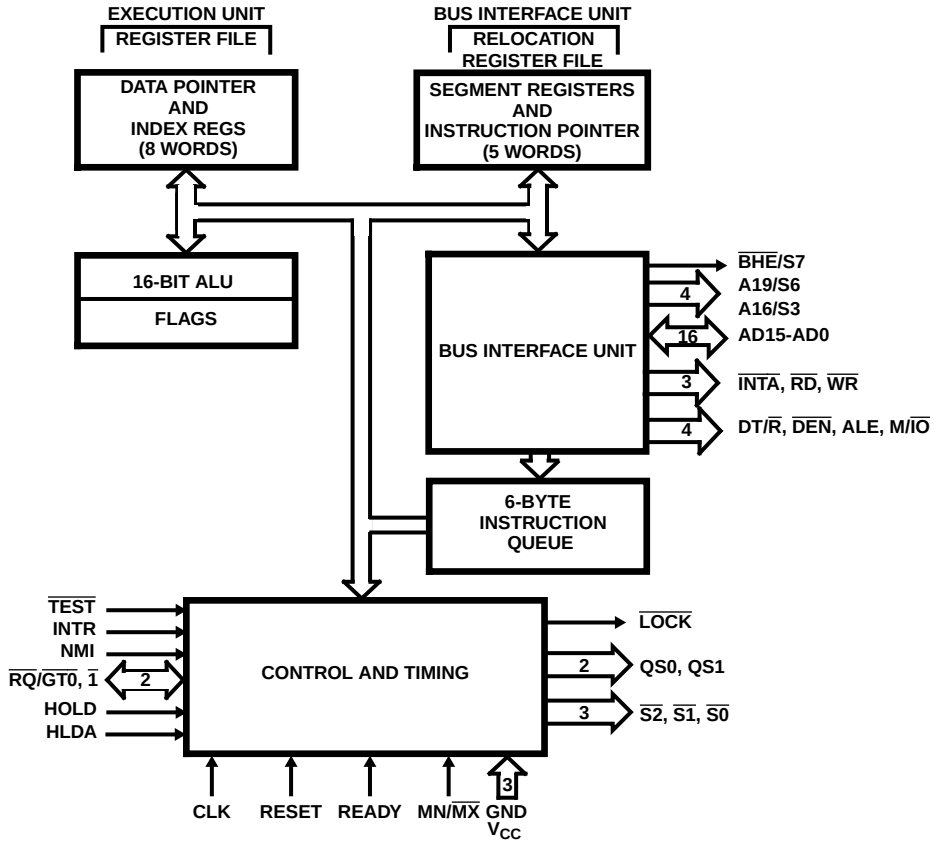
**80C86 (DIP)**
TOP VIEW

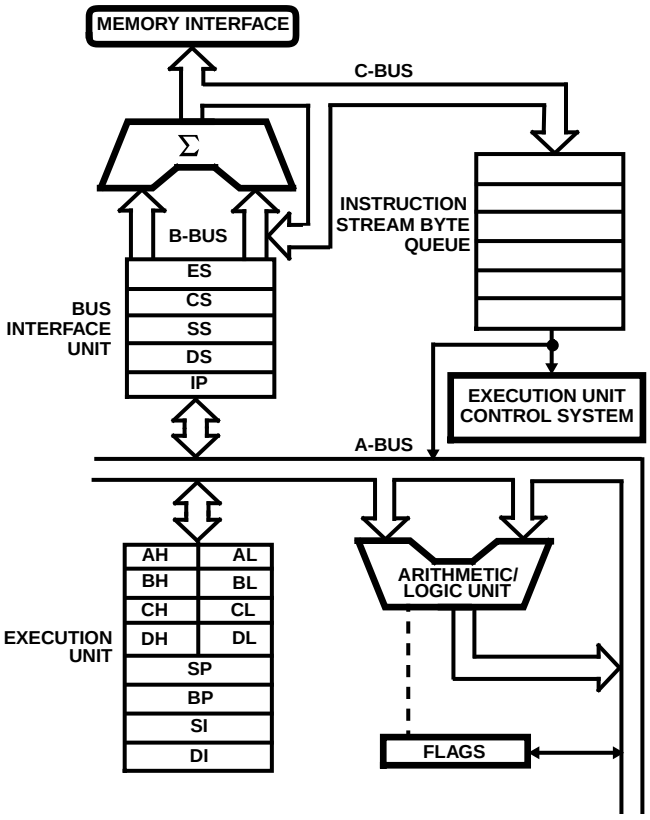



**80C86 (PLCC, CLCC)**
TOP VIEW



## Functional Diagram



## Pin Description

The following pin function descriptions are for 80C86 systems in either minimum or maximum mode. The “Local Bus” in these description is the direct multiplexed bus interface connection to the 80C86 (without regard to additional bus buffers).

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| AD15-AD0 | 2-16, 39 | I/O | ADDRESS DATA BUS: These lines constitute the time multiplexed memory/IO address (T1) and data (T2, T3, TW, T4) bus. A0 is analogous to $\overline{BHE}$ for the lower byte of the data bus, pins D7-D0. It is LOW during Ti when a byte is to be transferred on the lower portion of the bus in memory or I/O operations. Eight-bit oriented devices tied to the lower half would normally use A0 to condition chip select functions (See $\overline{BHE}$). These lines are active HIGH and are held at high impedance to the last valid logic level during interrupt acknowledge and local bus “hold acknowledge” or “grant sequence”. |
| A19/S6<br>A18/S5<br>A17/S4<br>A16/S3 | 35-38 | O | ADDRESS/STATUS: During T1, these are the four most significant address lines for memory operations. During I/O operations these lines are LOW. During memory and I/O operations, status information is available on these lines during T2, T3, TW, T4. S6 is always LOW. The status of the interrupt enable FLAG bit (S5) is updated at the beginning of each clock cycle. S4 and S3 are encoded as shown.<br>This information indicates which segment register is presently being used for data accessing.<br>These lines are held at high impedance to the last valid logic level during local bus “hold acknowledge” or “grant sequence”. |
| $\overline{BHE}$/S7 | 34 | O | BUS HIGH ENABLE/STATUS: During T1 the bus high enable signal ($\overline{BHE}$) should be used to enable data onto the most significant half of the data bus, pins D15-D8. Eight bit oriented devices tied to the upper half of the bus would normally use $\overline{BHE}$ to condition chip select functions. $\overline{BHE}$ is LOW during T1 for read, write, and interrupt acknowledge cycles when a byte is to be transferred on the high portion of the bus. The S7 status information is available during T2, T3 and T4. The signal is active LOW, and is held at high impedance to the last valid logic level during interrupt acknowledge and local bus “hold acknowledge” or “grant sequence”, it is LOW during T1 for the first interrupt acknowledge cycle. |
| $\overline{RD}$ | 32 | O | READ: Read strobe indicates that the processor is performing a memory or I/O read cycle, depending on the state of the M/$\overline{IO}$ or $\overline{S2}$ pin. This signal is used to read devices which reside on the 80C86 local bus. $\overline{RD}$ is active LOW during T2, T3 and TW of any read cycle, and is guaranteed to remain HIGH in T2 until the 80C86 local bus has floated.<br>This line is held at a high impedance logic one state during “hold acknowledge” or “grand sequence”. |
| READY | 22 | I | READY: is the acknowledgment from the addressed memory or I/O device that will complete the data transfer. The RDY signal from memory or I/O is synchronized by the 82C84A Clock Generator to form READY. This signal is active HIGH. The 80C86 READY input is not synchronized. Correct operation is not guaranteed if the Setup and Hold Times are not met. |

| S4 | S3 | CHARACTERISTICS |
|---|---|---|
| 0 | 0 | Alternate Data |
| 0 | 1 | Stack |
| 1 | 0 | Code or None |
| 1 | 1 | Data |

| BHE | A0 | CHARACTERISTICS |
|---|---|---|
| 0 | 0 | Whole Word |
| 0 | 1 | Upper Byte From/to Odd Address |
| 1 | 0 | Lower Byte From/to Even address |
| 1 | 1 | None |

## Pin Description (Continued)

The following pin function descriptions are for 80C86 systems in either minimum or maximum mode. The “Local Bus” in these description is the direct multiplexed bus interface connection to the 80C86 (without regard to additional bus buffers).

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| INTR | 18 | I | INTERRUPT REQUEST: is a level triggered input which is sampled during the last clock cycle of each instruction to determine if the processor should enter into an interrupt acknowledge operation. A subroutine is vectored to via an interrupt vector lookup table located in system memory. It can be internally masked by software resetting the interrupt enable bit. INTR is internally synchronized. This signal is active HIGH. |
| $\overline{TEST}$ | 23 | I | TEST: input is examined by the “Wait” instruction. If the $\overline{TEST}$ input is LOW execution continues, otherwise the processor waits in an “Idle” state. This input is synchronized internally during each clock cycle on the leading edge of CLK. |
| NMI | 17 | I | NON-MASKABLE INTERRUPT: is an edge triggered input which causes a type 2 interrupt. A subroutine is vectored to via an interrupt vector lookup table located in system memory. NMI is not maskable internally by software. A transition from LOW to HIGH initiates the interrupt at the end of the current instruction. This input is internally synchronized. |
| RESET | 21 | I | RESET: causes the processor to immediately terminate its present activity. The signal must transition LOW to HIGH and remain active HIGH for at least four clock cycles. It restarts execution, as described in the Instruction Set description, when RESET returns LOW. RESET is internally synchronized. |
| CLK | 19 | I | CLOCK: provides the basic timing for the processor and bus controller. It is asymmetric with a 33% duty cycle to provide optimized internal timing. |
| VCC | 40 | | VCC: +5V power supply pin. A 0.1µF capacitor between pins 20 and 40 is recommended for decoupling. |
| GND | 1, 20 | | GND: Ground. Note: both must be connected. A 0.1µF capacitor between pins 1 and 20 is recommended for decoupling. |
| MN/$\overline{MX}$ | 33 | I | MINIMUM/MAXIMUM: Indicates what mode the processor is to operate in. The two modes are discussed in the following sections. |

### Minimum Mode System

The following pin function descriptions are for the 80C86 in minimum mode (i.e., MN/$\overline{MX}$ = $V_{CC}$). Only the pin functions which are unique to minimum mode are described; all other pin functions are as described below.

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| M/$\overline{IO}$ | 28 | O | STATUS LINE: logically equivalent to $\overline{S2}$ in the maximum mode. It is used to distinguish a memory access from an I/O access. M/$\overline{IO}$ becomes valid in the T4 preceding a bus cycle and remains valid until the final T4 of the cycle (M = HIGH, I/O = LOW). M/$\overline{IO}$ is held to a high impedance logic one during local bus “hold acknowledge”. |
| $\overline{WR}$ | 29 | O | WRITE: indicates that the processor is performing a write memory or write I/O cycle, depending on the state of the M/$\overline{IO}$ signal. $\overline{WR}$ is active for T2, T3 and TW of any write cycle. It is active LOW, and is held to high impedance logic one during local bus “hold acknowledge”. |
| $\overline{INTA}$ | 24 | O | INTERRUPT ACKNOWLEDGE: is used as a read strobe for interrupt acknowledge cycles. It is active LOW during T2, T3 and TW of each interrupt acknowledge cycle. Note that $\overline{INTA}$ is never floated. |
| ALE | 25 | O | ADDRESS LATCH ENABLE: is provided by the processor to latch the address into the 82C82/82C83 address latch. It is a HIGH pulse active during clock LOW of T1 of any bus cycle. Note that ALE is never floated. |

## Minimum Mode System (Continued)

The following pin function descriptions are for the 80C86 in minimum mode (i.e., MN/$\overline{MX}$ = $V_{CC}$). Only the pin functions which are unique to minimum mode are described; all other pin functions are as described below.

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| DT/$\overline{R}$ | 27 | O | DATA TRANSMIT/RECEIVE: is needed in a minimum system that desires to use a data bus transceiver. It is used to control the direction of data flow through the transceiver. Logically, DT/$\overline{R}$ is equivalent to $\overline{S1}$ in maximum mode, and its timing is the same as for M/$\overline{IO}$ (T = HIGH, R = LOW). DT/$\overline{R}$ is held to a high impedance logic one during local bus "hold acknowledge". |
| $\overline{DEN}$ | 26 | O | DATA ENABLE: provided as an output enable for a bus transceiver in a minimum system which uses the transceiver. $\overline{DEN}$ is active LOW during each memory and I/O access and for $\overline{INTA}$ cycles. For a read or $\overline{INTA}$ cycle it is active from the middle of T2 until the middle of T4, while for a write cycle it is active from the beginning of T2 until the middle of T4. $\overline{DEN}$ is held to a high impedance logic one during local bus "hold acknowledge". |
| HOLD<br>HLDA | 31, 30 | I<br>O | HOLD: indicates that another master is requesting a local bus "hold". To be an acknowledged, HOLD must be active HIGH. The processor receiving the "hold" will issue a "hold acknowledge" (HLDA) in the middle of a T4 or TI clock cycle. Simultaneously with the issuance of HLDA, the processor will float the local bus and control lines. After HOLD is detected as being LOW, the processor will lower HLDA, and when the processor needs to run another cycle, it will again drive the local bus and control lines.<br>HOLD is not an asynchronous input. External synchronization should be provided if the system cannot otherwise guarantee the setup time. |

## Maximum Mode System

The following pin function descriptions are for the 80C86 system in maximum mode (i.e., MN/$\overline{MX}$ - GND). Only the pin functions which are unique to maximum mode are described below.

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| $\overline{S0}$<br>$\overline{S1}$<br>$\overline{S2}$ | 26<br>27<br>28 | O<br>O<br>O | STATUS: is active during T4, T1 and T2 and is returned to the passive state (1, 1, 1) during T3 or during TW when READY is HIGH. This status is used by the 82C88 Bus Controller to generate all memory and I/O access control signals. Any change by $\overline{S2}$, $\overline{S1}$ or $\overline{S0}$ during T4 is used to indicate the beginning of a bus cycle, and the return to the passive state in T3 or TW is used to indicate the end of a bus cycle.<br>These signals are held at a high impedance logic one state during "grant sequence". |

| $\overline{S2}$ | $\overline{S1}$ | $\overline{S0}$ | CHARACTERISTICS |
|---|---|---|---|
| 0 | 0 | 0 | Interrupt Acknowledge |
| 0 | 0 | 1 | Read I/O Port |
| 0 | 1 | 0 | Write I/O Port |
| 0 | 1 | 1 | Halt |
| 1 | 0 | 0 | Code Access |
| 1 | 0 | 1 | Read Memory |
| 1 | 1 | 0 | Write Memory |
| 1 | 1 | 1 | Passive |

## Maximum Mode System (Continued)

The following pin function descriptions are for the 80C86 system in maximum mode (i.e., MN/$\overline{MX}$ - GND). Only the pin functions which are unique to maximum mode are described below.

| SYMBOL | PIN NUMBER | TYPE | DESCRIPTION |
|---|---|---|---|
| $\overline{RQ}$/$\overline{GT0}$<br>$\overline{RQ}$/$\overline{GT1}$ | 31, 30 | I/O | REQUEST/GRANT: pins are used by other local bus masters to force the processor to release the local bus at the end of the processor's current bus cycle. Each pin is bidirectional with $\overline{RQ}$/$\overline{GTO}$ having higher priority than $\overline{RQ}$/$\overline{GT}$1. $\overline{RQ}$/$\overline{GT}$ has an internal pull-up bus hold device so it may be left unconnected. The request/grant sequence is as follows (see $\overline{RQ}$/$\overline{GT}$ Sequence Timing)<br>1. A pulse of 1 CLK wide from another local bus master indicates a local bus request ("hold") to the 80C86 (pulse 1).<br>2. During a T4 or TI clock cycle, a pulse 1 CLK wide from the 80C86 to the requesting master (pulse 2) indicates that the 80C86 has allowed the local bus to float and that it will enter the "grant sequence" state at the next CLK. The CPU's bus interface unit is disconnected logically from the local bus during "grant sequence".<br>3. A pulse 1 CLK wide from the requesting master indicates to the 80C86 (pulse 3) that the "hold" request is about to end and that the 80C86 can reclaim the local bus at the next CLK. The CPU then enters T4 (or TI if no bus cycles pending).<br>Each Master-Master exchange of the local bus is a sequence of 3 pulses. There must be one idle CLK cycle after each bus exchange. Pulses are active low.<br>If the request is made while the CPU is performing a memory cycle, it will release the local bus during T4 of the cycle when all the following conditions are met:<br><br>1. Request occurs on or before T2.<br>2. Current cycle is not the low byte of a word (on an odd address).<br>3. Current cycle is not the first acknowledge of an interrupt acknowledge sequence.<br>4. A locked instruction is not currently executing.<br>If the local bus is idle when the request is made the two possible events will follow:<br>1. Local bus will be released during the next cycle.<br>2. A memory cycle will start within three clocks. Now the four rules for a currently active memory cycle apply with condition number 1 already satisfied. |
| $\overline{LOCK}$ | 29 | O | LOCK: output indicates that other system bus masters are not to gain control of the system bus while $\overline{LOCK}$ is active LOW. The $\overline{LOCK}$ signal is activated by the "LOCK" prefix instruction and remains active until the completion of the next instruction. This signal is active LOW, and is held at a high impedance logic one state during "grant sequence". In MAX mode, $\overline{LOCK}$ is automatically generated during T2 of the first $\overline{INTA}$ cycle and removed during T2 of the second $\overline{INTA}$ cycle. |
| QS1, QSO | 24, 25 | O | QUEUE STATUS: The queue status is valid during the CLK cycle after which the queue operation is performed.<br>QS1 and QS0 provide status to allow external tracking of the internal 80C86 instruction queue. Note that QS1, QS0 never become high impedance. |

| QSI | QSO | |
|---|---|---|
| 0 | 0 | No Operation |
| 0 | 1 | First byte of op code from queue |
| 1 | 0 | Empty the queue |
| 1 | 1 | Subsequent byte from queue |

## Functional Description

### Static Operation

All 80C86 circuitry is of static design. Internal registers, counters and latches are static and require no refresh as with dynamic circuit design. This eliminates the minimum operating frequency restriction placed on other microprocessors. The CMOS 80C86 can operate from DC to the specified upper frequency limit. The processor clock may be stopped in either state (HIGH/LOW) and held there indefinitely. This type of operation is especially useful for system debug or power critical applications.

The 80C86 can be single stepped using only the CPU clock. This state can be maintained as long as is necessary. Single step clock operation allows simple interface circuitry to provide critical information for bringing up your system.

Static design also allows very low frequency operation (down to DC). In a power critical situation, this can provide extremely low power operation since 80C86 power dissipation is directly related to operating frequency. As the system frequency is reduced, so is the operating power until, ultimately, at a DC input frequency, the 80C86 power requirement is the standby current, (500µA maximum).

### Internal Architecture

The internal functions of the 80C86 processor are partitioned logically into two processing units. The first is the Bus Interface Unit (BIU) and the second is the Execution Unit (EU) as shown in the CPU functional diagram.

These units can interact directly, but for the most part perform as separate asynchronous operational processors. The bus interface unit provides the functions related to instruction fetching and queuing, operand fetch and store, and address relocation. This unit also provides the basic bus control. The overlap of instruction pre-fetching provided by this unit serves to increase processor performance through improved bus bandwidth utilization. Up to 6 bytes of the instruction stream can be queued while waiting for decoding and execution.

The instruction stream queuing mechanism allows the BIU to keep the memory utilized very efficiently. Whenever there is space for at least 2 bytes in the queue, the BIU will attempt a word fetch memory cycle. This greatly reduces “dead-time” on the memory bus. The queue acts as a First-In-First-Out (FIFO) buffer, from which the EU extracts instruction bytes as required. If the queue is empty (following a branch instruction, for example), the first byte into the queue immediately becomes available to the EU.

The execution unit receives pre-fetched instructions from the BIU queue and provides un-relocated operand addresses to the BIU. Memory operands are passed through the BIU for processing by the EU, which passes results to the BIU for storage.

### Memory Organization

The processor provides a 20-bit address to memory, which locates the byte being referenced. The memory is organized as a linear array of up to 1 million bytes, addressed as 00000(H) to FFFFF(H). The memory is logically divided into code, data, extra and stack segments of up to 64K bytes each, with each segment falling on 16-byte boundaries. (See Figure 1).

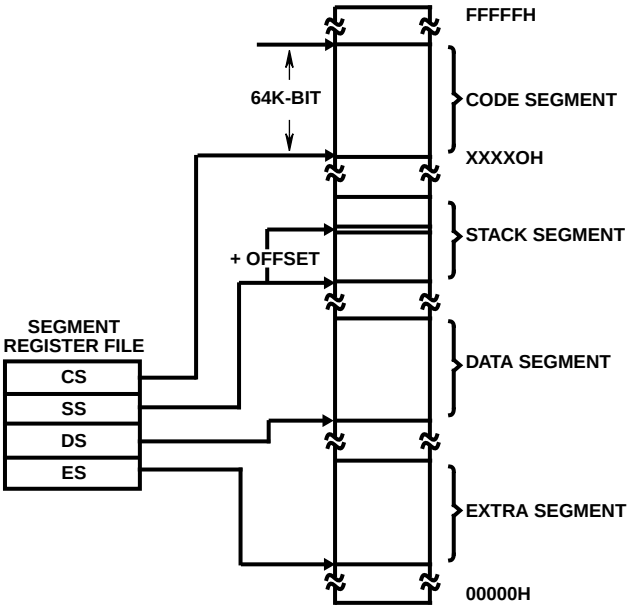



**FIGURE 1. 80C86 MEMORY ORGANIZATION**

**TABLE 1.**

| TYPE OF MEMORY REFERENCE | DEFAULT SEGMENT BASE | ALTERNATE SEGMENT BASE | OFFSET |
|---|---|---|---|
| Instruction Fetch | CS | None | IP |
| Stack Operation | SS | None | SP |
| Variable (except following) | DS | CS, ES, SS | Effective Address |
| String Source | DS | CS, ES, SS | SI |
| String Destination | ES | None | DI |
| BP Used As Base Register | SS | CS, DS, ES | Effective Address |

All memory references are made relative to base addresses contained in high speed segment registers. The segment types were chosen based on the addressing needs of programs. The segment register to be selected is automatically chosen according to the specific rules of Table 1. All information in one segment type share the same logical attributes (e.g. code or data). By structuring memory into re-locatable areas of similar characteristics and by automatically selecting segment registers, programs are shorter, faster and more structured. (See Table 1).

Word (16-bit) operands can be located on even or odd address boundaries and are thus, not constrained to even boundaries as is the case in many 16-bit computers. For address and data operands, the least significant byte of the word is stored in the lower valued address location and the most significant byte in the next higher address location. The BIU automatically performs the proper number of memory

accesses; one, if the word operand is on an even byte boundary and two, if it is on an odd byte boundary. Except for the performance penalty**,** this double access is transparent to the software. The performance penalty does not occur for instruction fetches; only word operands.

Physically, the memory is organized as a high bank (D15-D8) and a low bank (D7-D0) of 512K bytes addressed in parallel by the processor's address lines.

Byte data with even addresses is transferred on the D7-D0 bus lines, while odd addressed byte data (A0 HIGH) is transferred on the D15-D8 bus lines. The processor provides two enable signals, BHE and $A_0$, to selectively allow reading from or writing into either an odd byte location, even byte location, or both. The instruction stream is fetched from memory as words and is addressed internally by the processor at the byte level as necessary.

In referencing word data, the BIU requires one or two memory cycles depending on whether the starting byte of the word is on an even or odd address, respectively. Consequently, in referencing word operands performance can be optimized by locating data on even address boundaries. This is an especially useful technique for using the stack, since odd address references to the stack may adversely affect the context switching time for interrupt processing or task multiplexing.

Certain locations in memory are reserved for specific CPU operations (See Figure 2). Locations from address FFFF0H through FFFFFH are reserved for operations including a jump to the initial program loading routine. Following RESET, the CPU will always begin execution at location FFFF0H where the jump must be located. Locations 00000H through 003FFH are reserved for interrupt operations. Each of the 256 possible interrupt service routines is accessed thru its own pair of 16-bit pointers (segment address pointer and offset address pointer). The first pointer, used as the offset address, is loaded into the IP and the second pointer, which designates the base address is loaded into the CS. At this point program control is transferred to the interrupt routine. The pointer elements are assumed to have been stored at the respective places in reserved memory prior to occurrence of interrupts.

### Minimum and Maximum Operation Modes

The requirements for supporting minimum and maximum 80C86 systems are sufficiently different that they cannot be met efficiently using 40 uniquely defined pins. Consequently, the 80C86 is equipped with a strap pin (MN/$\overline{MX}$) which defines the system configuration. The definition of a certain subset of the pins changes, dependent on the condition of the strap pin. When the MN/$\overline{MX}$ pin is strapped to GND, the 80C86 defines pins 24 through 31 and 34 in maximum mode. When the MN/$\overline{MX}$ pin is strapped to $V_{CC}$, the 80C86 generates bus control signals itself on pins 24 through 31 and 34.

The minimum mode 80C86 can be used with either a multiplexed or demultiplexed bus. This architecture provides the 80C86 processing power in a highly integrated form.

The demultiplexed mode requires two 82C82 latches (for 64K addressability) or three 82C82 latches (for a full megabyte of addressing). An 82C86 or 82C87 transceiver can also be used if data bus buffering is required. (See Figure 6A.) The 80C86 provides $\overline{DEN}$ and DT/$\overline{R}$ to control the transceiver, and ALE to latch the addresses. This configuration of the minimum mode provides the standard demultiplexed bus structure with heavy bus buffering and relaxed bus timing requirements.

The maximum mode employs the 82C88 bus controller (See Figure 6B). The 82C88 decodes status lines $\overline{S0}$, $\overline{S1}$ and $\overline{S2}$, and provides the system with all bus control signals.

Moving the bus control to the 82C88 provides better source and sink current capability to the control lines, and frees the 80C86 pins for extended large system features. Hardware lock, queue status, and two request/grant interfaces are provided by the 80C86 in maximum mode. These features allow coprocessors in local bus and remote bus configurations.

### Bus Operation

The 80C86 has a combined address and data bus commonly referred to as a time multiplexed bus. This technique provides the most efficient use of pins on the processor while permitting the use of a standard 40 lead package. This "local bus" can be buffered directly and used throughout the system with address latching provided on memory and I/O modules. In addition, the bus can also be demultiplexed at the processor with a single set of 82C82 address latches if a standard non-multiplexed bus is desired for the system.

Each processor bus cycle consists of at least four CLK cycles. These are referred to as T1, T2, T3 and T4 (see Figure 3). The address is emitted from the processor during T1 and data transfer occurs on the bus during T3 and T4. T2 is used primarily for changing the direction of the bus during read operations. In the event that a "NOT READY" indication is given by the addressed device, "Wait" states (TW) are inserted between T3 and T4. Each inserted wait state is the same duration as a CLK cycle. Periods can occur between 80C86 driven bus cycles. These are referred to as idle" states ($T_I$) or inactive CLK cycles. The processor uses these cycles for internal housekeeping and processing.

During T1 of any bus cycle, the ALE (Address Latch Enable) signal is emitted (by either the processor or the 82C88 bus controller, depending on the MN/$\overline{MX}$ strap). At the trailing edge of this pulse, a valid address and certain status information for the cycle may be latched.

Status bits $\overline{S0}$, $\overline{S1}$ and $\overline{S2}$ are used by the bus controller, in maximum mode, to identify the type of bus transaction according to Table 2.

**TABLE 2.**

| $\overline{S2}$ | $\overline{S1}$ | $\overline{S0}$ | CHARACTERISTICS |
|---|---|---|---|
| 0 | 0 | 0 | Interrupt |
| 0 | 0 | 1 | Read I/O |
| 0 | 1 | 0 | Write I/O |
| 0 | 1 | 1 | Halt |
| 1 | 0 | 0 | Instruction Fetch |
| 1 | 0 | 1 | Read Data from Memory |
| 1 | 1 | 0 | Write Data to Memory |
| 1 | 1 | 1 | Passive (No Bus Cycle) |

Status bits S3 through S7 are time multiplexed with high order address bits and the $\overline{BHE}$ signal, and are therefore valid during T2 through T4. S3 and S4 indicate which segment register (see Instruction Set Description) was used for this bus cycle in forming the address, according to Table 3.

S5 is a reflection of the PSW interrupt enable bit. S3 is always zero and S7 is a spare status bit.

**TABLE 3.**

| S4 | S3 | CHARACTERISTICS |
|---|---|---|
| 0 | 0 | Alternate Data (Extra Segment) |
| 0 | 1 | Stack |
| 1 | 0 | Code or None |
| 1 | 1 | Data |

### I/O Addressing

In the 80C86, I/O operations can address up to a maximum of 64K I/O byte registers or 32K I/O word registers. The I/O address appears in the same format as the memory address on bus lines A15-A0. The address lines A19-A16 are zero in I/O operations. The variable I/O instructions which use register DX as a pointer have full address capability while the direct I/O instructions directly address one or two of the 256 I/O byte locations in page 0 of the I/O address space.

I/O ports are addressed in the same manner as memory locations. Even addressed bytes are transferred on the D7-D0 bus lines and odd addressed bytes on D15-D8. Care must be taken to ensure that each register within an 8-bit peripheral located on the lower portion of the bus be addressed as even.

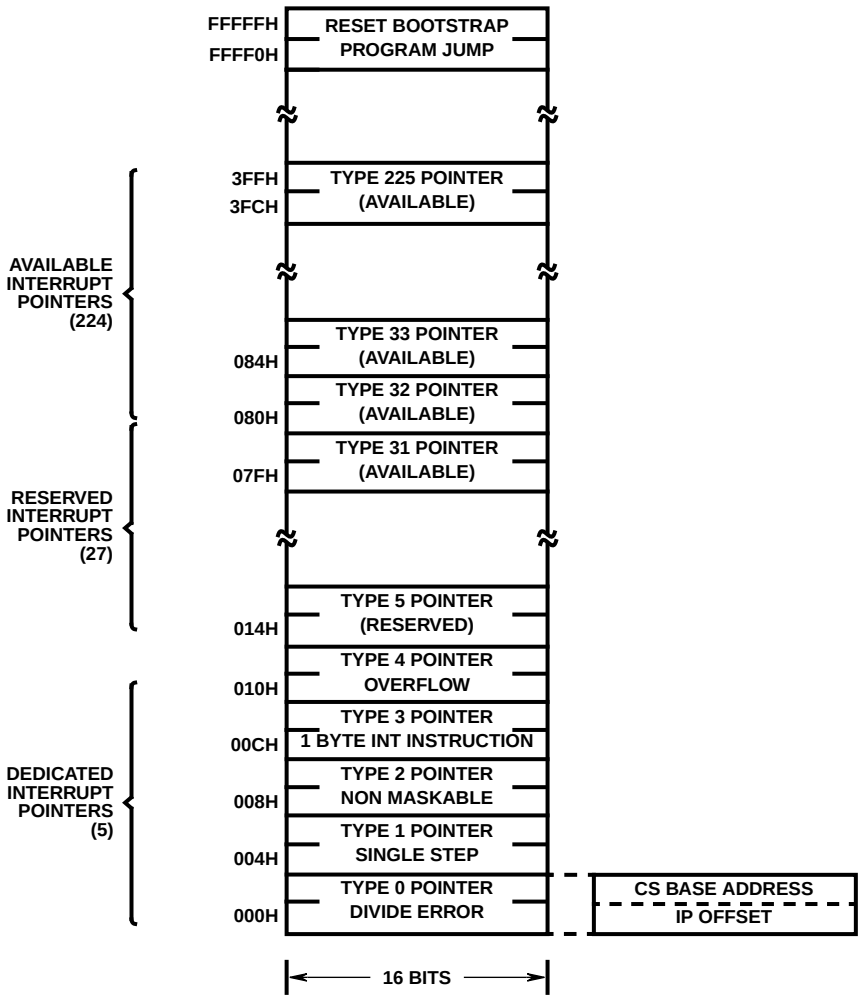



**FIGURE 2. RESERVED MEMORY LOCATIONS**

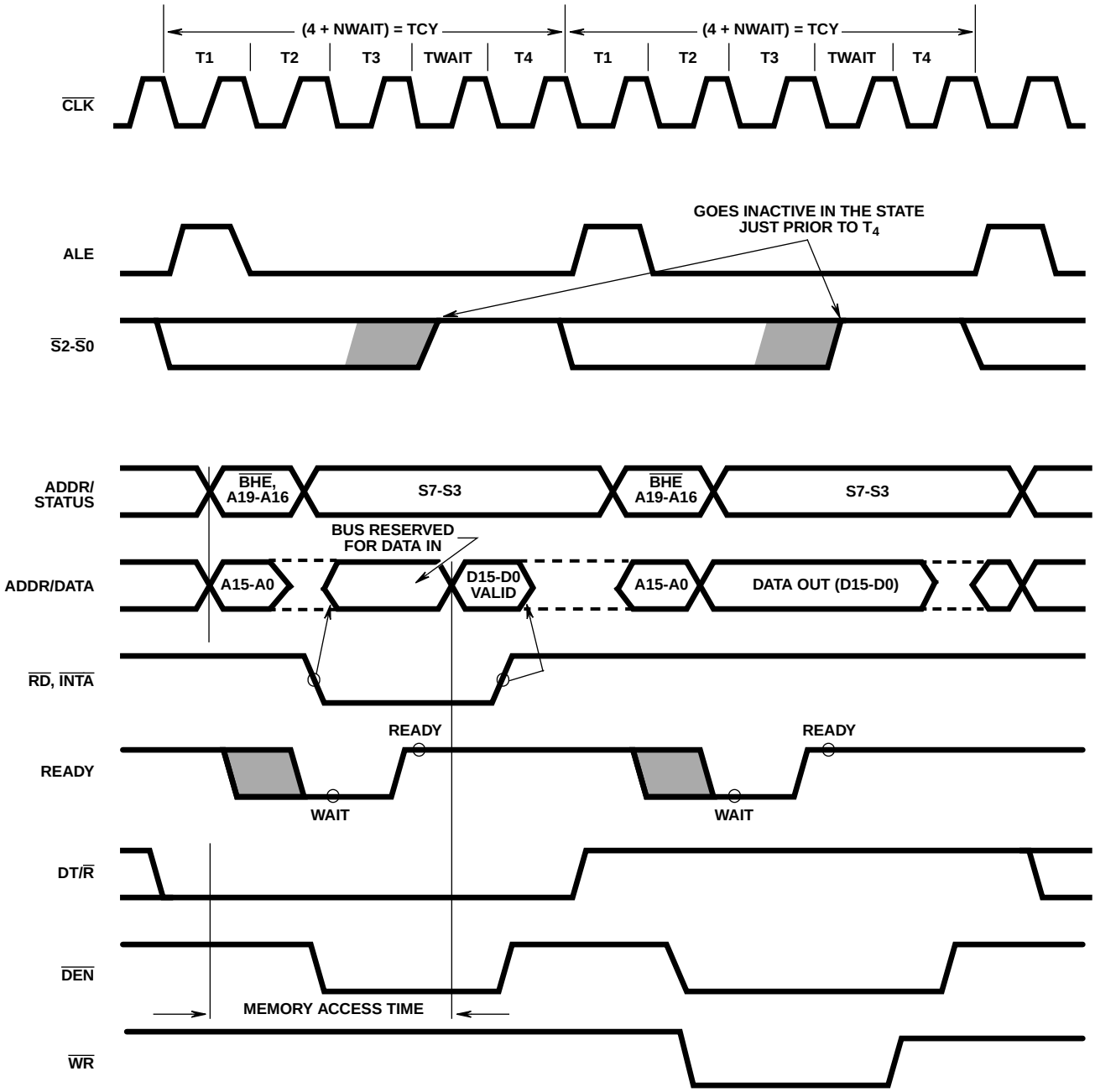



FIGURE 3. BASIC SYSTEM TIMING

## External Interface

### Processor RESET and Initialization

Processor initialization or start up is accomplished with activation (HIGH) of the RESET pin. The 80C86 RESET is required to be HIGH for greater than 4 CLK cycles. The 80C86 will terminate operations on the high-going edge of RESET and will remain dormant as long as RESET is HIGH. The low-going transition of RESET triggers an internal reset sequence for approximately 7 clock cycles. After this interval, the 80C86 operates normally beginning with the instruction in absolute location FFFF0H. (See Figure 2). The RESET input is internally synchronized to the processor clock. At initialization, the HIGH-to-LOW transition of RESET must occur no sooner than 50μs (or 4 CLK cycles, whichever is greater) after power-up, to allow complete initialization of the 80C86.

NMI will not be recognized prior to the second CLK cycle following the end of RESET. If NMI is asserted sooner than nine clock cycles after the end of RESET, the processor may execute one instruction before responding to the interrupt.

### Bus Hold Circuitry

To avoid high current conditions caused by floating inputs to CMOS devices and to eliminate need for pull-up/down resistors, "bus-hold" circuitry has been used on the 80C86 pins 2-16, 26-32 and 34-39. (See Figure 4A and Figure 4B). These circuits will maintain the last valid logic state if no driving source is present (i.e., an unconnected pin or a driving source which goes to a high impedance state). To overdrive the "bus hold" circuits, an external driver must be capable of supplying approximately 400μA minimum sink or source current at valid input voltage levels. Since this "bus hold" circuitry is active and not a "resistive" type element, the associated power supply current is negligible and power dissipation is significantly reduced when compared to the use of passive pull-up resistors.

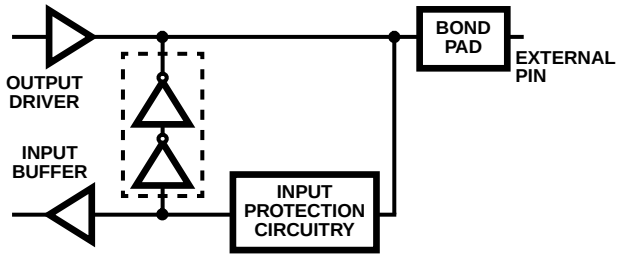



**FIGURE 4A. BUS HOLD CIRCUITRY PIN 2-16, 34-39**

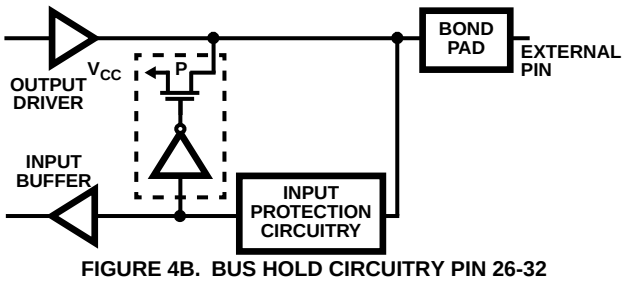



**FIGURE 4B. BUS HOLD CIRCUITRY PIN 26-32**

### Interrupt Operations

Interrupt operations fall into two classes: software or hardware initiated. The software initiated interrupts and software aspects of hardware interrupts are specified in the Instruction Set Description. Hardware interrupts can be classified as non-maskable or maskable.

Interrupts result in a transfer of control to a new program location. A 256-element table containing address pointers to the interrupt service program locations resides in absolute locations 0 through 3FFH, which are reserved for this purpose. Each element in the table is 4 bytes in size and corresponds to an interrupt "type". An interrupting device supplies an 8-bit type number during the interrupt acknowledge sequence, which is used to "vector" through the appropriate element to the new interrupt service program location. All flags and both the Code Segment and Instruction Pointer register are saved as part of the $\overline{\text{INTA}}$ sequence. These are restored upon execution of an Interrupt Return (IRET) instruction.

### Non-Maskable Interrupt (NMI)

The processor provides a single non-maskable interrupt pin (NMI) which has higher priority than the maskable interrupt request pin (INTR). A typical use would be to activate a power failure routine. The NMI is edge-triggered on a LOW-to-HIGH transition. The activation of this pin causes a type 2 interrupt.

NMI is required to have a duration in the HIGH state of greater than two CLK cycles, but is not required to be synchronized to the clock. Any positive transition of NMI is latched on-chip and will be serviced at the end of the current instruction or between whole moves of a block-type instruction. Worst case response to NMI would be for multiply, divide, and variable shift instructions. There is no specification on the occurrence of the low-going edge; it may occur before, during or after the servicing of NMI. Another positive edge triggers another response if it occurs after the start of the NMI procedure. The signal must be free of logical spikes in general and be free of bounces on the low-going edge to avoid triggering extraneous responses.

### Maskable Interrupt (INTR)

The 80C86 provides a single interrupt request input (INTR) which can be masked internally by software with the resetting of the interrupt enable flag (IF) status bit. The interrupt request signal is level triggered. It is internally synchronized during each clock cycle on the high-going edge of CLK. To be responded to, INTR must be present (HIGH) during the clock period preceding the end of the current instruction or the end of a whole move for a block type instruction. INTR may be removed anytime after the falling edge of the first $\overline{\text{INTA}}$ signal. During the interrupt response sequence further interrupts are disabled. The enable bit is reset as part of the response to any interrupt (INTR, NMI, software interrupt or single-step), although the FLAGS register which is automatically pushed onto the stack reflects the state of the processor prior to the interrupt. Until the old FLAGS register is restored, the enable bit will be zero unless specifically set by an instruction.

During the response sequence (Figure 5) the processor executes two successive (back-to-back) interrupt acknowledge cycles. The 80C86 emits the $\overline{LOCK}$ signal (Max mode only) from T2 of the first bus cycle until T2 of the second. A local bus "hold" request will not be honored until the end of the second bus cycle. In the second bus cycle, a byte is supplied to the 80C86 by the 82C59A Interrupt Controller, which identifies the source (type) of the interrupt. This byte is multiplied by four and used as a pointer into the interrupt vector lookup table. An INTR signal left HIGH will be continually responded to within the limitations of the enable bit and sample period. The INTERRUPT RETURN instruction includes a FLAGS pop which returns the status of the original interrupt enable bit when it restores the FLAGS.

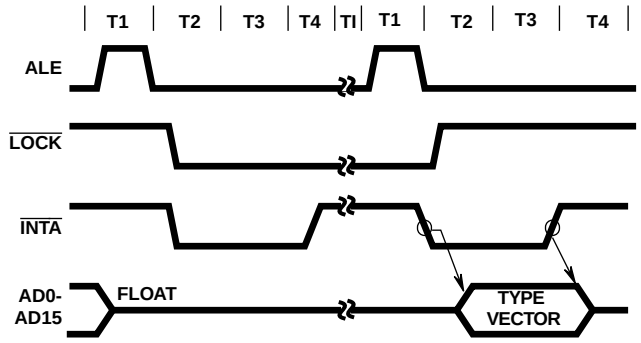



**FIGURE 5. INTERRUPT ACKNOWLEDGE SEQUENCE**

### Halt

When a software "HALT" instruction is executed the processor indicates that it is entering the "HALT" state in one of two ways depending upon which mode is strapped. In minimum mode, the processor issues one ALE with no qualifying bus control signals. In maximum mode the processor issues appropriate HALT status on $\overline{S2}$, $\overline{S1}$, $\overline{S0}$ and the 82C88 bus controller issues one ALE. The 80C86 will not leave the "HALT" state when a local bus "hold" is entered while in "HALT". In this case, the processor reissues the HALT indicator at the end of the local bus hold. An NMI or interrupt request (when interrupts enabled) or RESET will force the 80C86 out of the "HALT" state.

## *Read/Modify/Write (Semaphore)*

### Operations Via $\overline{Lock}$

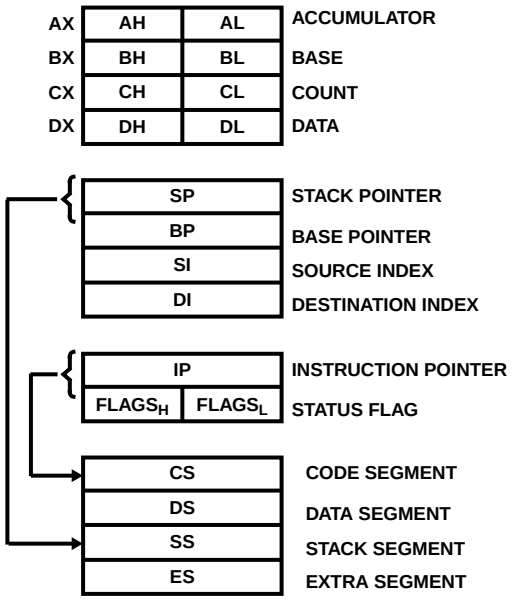
The LOCK status information is provided by the processor when consecutive bus cycles are required during the execution of an instruction. This gives the processor the capability of performing read/modify/write operations on memory (via the Exchange Register With Memory instruction, for example) without another system bus master receiving intervening memory cycles. This is useful in multiprocessor system configurations to accomplish "test and set lock" operations. The LOCK signal is activated (forced LOW) in the clock cycle following decoding of the software "LOCK" prefix instruction. It is deactivated at the end of the last bus cycle of the instruction following the "LOCK" prefix instruction. While LOCK is active a request on a RQ/GT pin will be recorded and then honored at the end of the LOCK.

### External Synchronization Via TEST

As an alternative to interrupts, the 80C86 provides a single software-testable input pin ($\overline{TEST}$). This input is utilized by executing a WAIT instruction. The single WAIT instruction is repeatedly executed until the $\overline{TEST}$ input goes active (LOW). The execution of WAIT does not consume bus cycles once the queue is full.

If a local bus request occurs during WAIT execution, the 80C86 three-states all output drivers while inputs and I/O pins are held at valid logic levels by internal bus-hold circuits. If interrupts are enabled, the 80C86 will recognize interrupts and process them when it regains control of the bus. The WAIT instruction is then refetched, and re-executed.

**TABLE 4. 80C86 REGISTER**




### Basic System Timing

Typical system configurations for the processor operating in minimum mode and in maximum mode are shown in Figures 6A and 6B, respectively. In minimum mode, the MN/$\overline{MX}$ pin is strapped to VCC and the processor emits bus control signals (e.g. $\overline{RD}$, $\overline{WR}$, etc.) directly. In maximum mode, the MN/$\overline{MX}$ pin is strapped to GND and the processor emits coded status information which the 82C88 bus controller uses to generate MULTIBUS compatible bus control signals. Figure 3 shows the signal timing relationships.

### System Timing - Minimum System

The read cycle begins in T1 with the assertion of the Address Latch Enable (ALE) signal. The trailing (low-going) edge of this signal is used to latch the address information, which is valid on the address/data bus (AD0-AD15) at this time, into the 82C82/82C83 latch. The $\overline{BHE}$ and A0 signals address the low, high or both bytes. From T1 to T4 the M/$\overline{IO}$ signal indicates a memory or I/O operation. At T2, the address is removed from the address/data bus and the bus is held at the last valid logic state by internal bus hold

devices. The read control signal is also asserted at T2. The read ($\overline{RD}$) signal causes the addressed device to enable its data bus drivers to the local bus. Some time later, valid data will be available on the bus and the addressed device will drive the READY line HIGH**.** When the processor returns the read signal to a HIGH level, the addressed device will again three-state its bus drivers. If a transceiver (82C86/82C87) is required to buffer the 80C86 local bus, signals DT/R and $\overline{DEN}$ are provided by the 80C86.

A write cycle also begins with the assertion of ALE and the emission of the address. The M/$\overline{IO}$ signal is again asserted to indicate a memory or I/O write operation. In T2, immediately following the address emission, the processor emits the data to be written into the addressed location. This data remains valid until at least the middle of T4. During T2, T3 and TW, the processor asserts the write control signal. The write ($\overline{WR}$) signal becomes active at the beginning of T2 as opposed to the read which is delayed somewhat into T2 to provide time for output drivers to become inactive.

The $\overline{BHE}$ and A0 signals are used to select the proper byte(s) of the memory/IO word to be read or written according to Table 5.

**TABLE 5.**

| $\overline{BHE}$ | A0 | CHARACTERISTICS |
|---|---|---|
| 0 | 0 | Whole word |
| 0 | 1 | Upper Byte From/To Odd Address |
| 1 | 0 | Lower Byte From/To Even Address |
| 1 | 1 | None |

I/O ports are addressed in the same manner as memory location. Even addressed bytes are transferred on the D7-D0 bus lines and odd address bytes on D15-D8.

The basic difference between the interrupt acknowledge cycle and a read cycle is that the interrupt acknowledge signal ($\overline{INTA}$) is asserted in place of the read ($\overline{RD}$) signal and the address bus is held at the last valid logic state by internal bus hold devices. (See Figure 4). In the second of two successive $\overline{INTA}$ cycles a byte of information is read from the data bus (D7-D0) as supplied by the interrupt system logic (i.e., 82C59A Priority Interrupt Controller). This byte identifies the source (type) of the interrupt. It is multiplied by four and used as a pointer into an interrupt vector lookup table, as described earlier.

**Bus Timing - Medium Size Systems**

For medium complexity systems the MN/$\overline{MX}$ pin is connected to GND and the 82C88 Bus Controller is added to the system as well as an 82C82/82C83 latch for latching the system address, and an 82C86/82C87 transceiver to allow for bus loading greater than the 80C86 is capable of handling. Signals ALE, $\overline{DEN}$, and DT/$\overline{R}$ are generated by the 82C88 instead of the processor in this configuration, although their timing remains relatively the same. The 80C86 status outputs ($\overline{S2}$, $\overline{S1}$ and $\overline{S0}$) provide type-of-cycle information and become 82C88 inputs. This bus cycle information specifies read (code, data or I/O), write (data or I/O), interrupt acknowledge, or software halt. The 82C88 issues control signals specifying memory read or write, I/O read or write, or interrupt acknowledge. The 82C88 provides two types of write strobes, normal and advanced, to be applied as required. The normal write strobes have data valid at the leading edge of write. The advanced write strobes have the same timing as read strobes, and hence, data is not valid at the leading edge of write. The 82C86/82C87 transceiver receives the usual T and $\overline{OE}$ inputs from the 82C88 DT/$\overline{R}$ and DEN signals.

The pointer into the interrupt vector table, which is passed during the second $\overline{INTA}$ cycle, can be derived from an 82C59A located on either the local bus or the system bus. If the master 82C59A Priority Interrupt Controller is positioned on the local bus, the 82C86/82C87 transceiver must be disabled when reading from the master 82C59A during the interrupt acknowledge sequence and software “poll”.

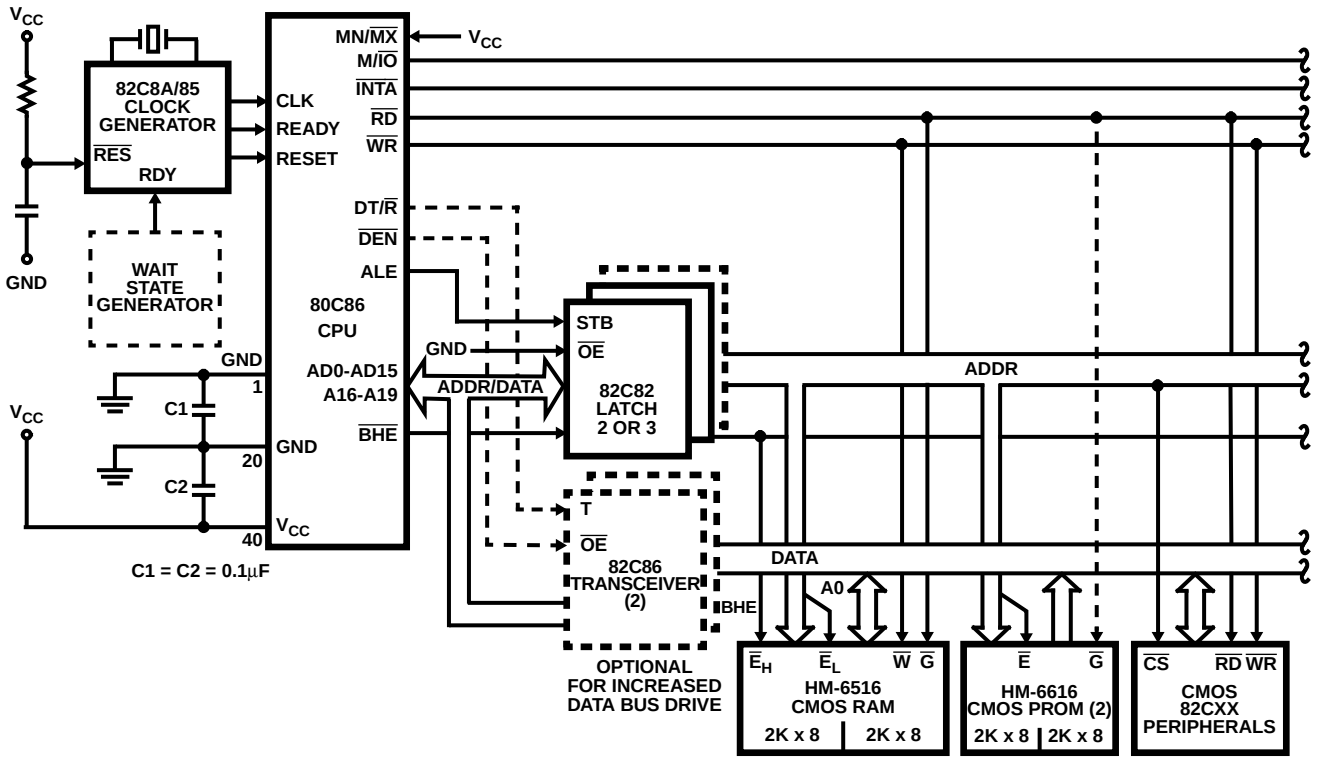



FIGURE 6A. MINIMUM MODE 80C86 TYPICAL CONFIGURATION

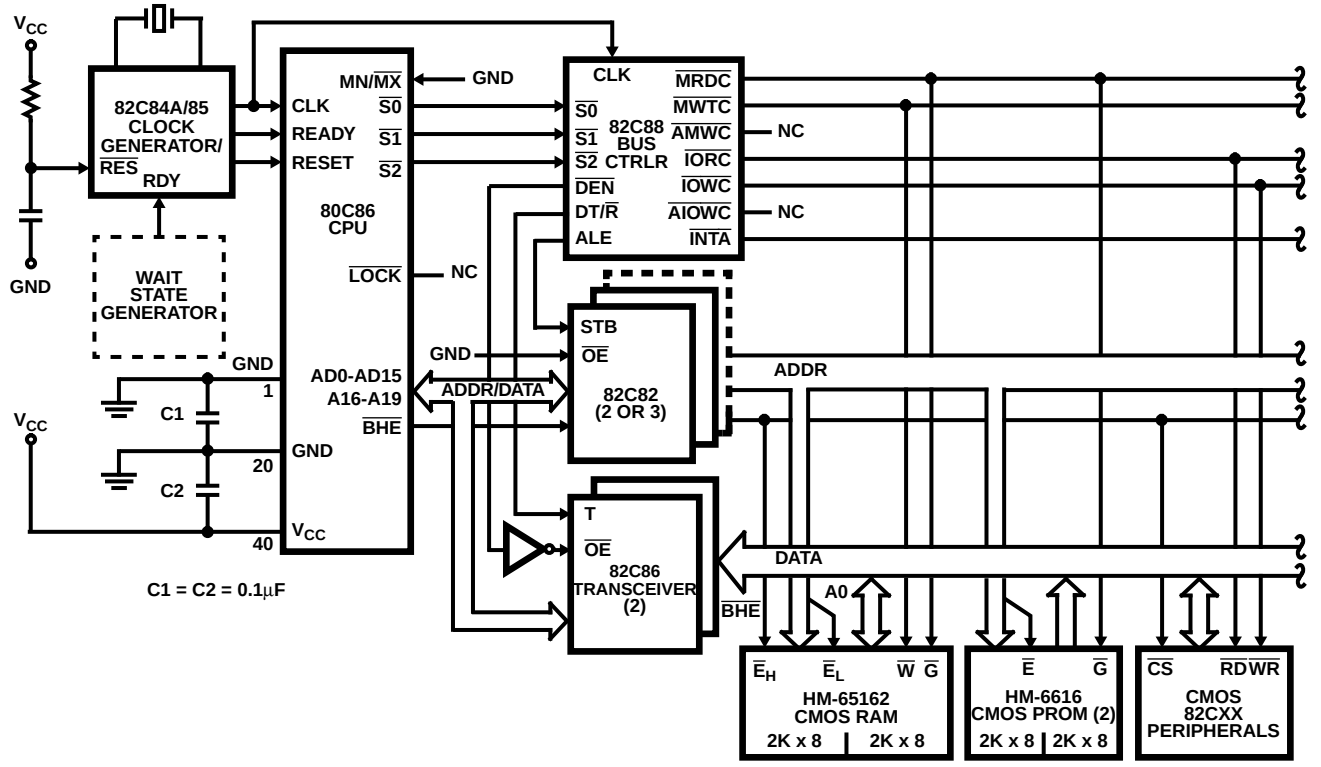



FIGURE 6B. MAXIMUM MODE 80C86 TYPICAL CONFIGURATION

## Absolute Maximum Ratings

Supply Voltage . . . . . +8.0V
Input, Output or I/O Voltage . . . . . GND -0.5V to $V_{CC}$ +0.5V
Storage Temperature Range . . . . . $-65^oC$ to $+150^oC$
Junction Temperature
Ceramic Packages . . . . . $+175^oC$
Plastic Packages . . . . . $+150^oC$
Lead Temperature (Soldering 10s) . . . . . $+300^oC$
(Lead tips only for surface mount packages)
ESD Classification . . . . . Class 1

## Thermal Information

| Thermal Resistance (Typical, Note 1) | $\theta_{JA}$ ($^oC/W$) | $\theta_{JC}$ ($^oC/W$) |
|---|---|---|
| PDIP Package | 50 | N/A |
| PLCC Package | 46 | N/A |
| SBDIP Package | 30 | 6 |
| CLCC Package | 40 | 6 |

Gate Count . . . . . 9750 Gates

*CAUTION: Stresses above those listed in "Absolute Maximum Ratings" may cause permanent damage to the device. This is a stress only rating and operation of the device at these or any other conditions above those indicated in the operational sections of this specification is not implied.*

NOTE:

1. $\theta_{JA}$ is measured with the component mounted on an evaluation PC board in free air.

## Operating Conditions

Operating Supply Voltage . . . . . +4.5V to +5.5V
M80C86-2 ONLY . . . . . +4.75V to +5.25V

Operating Temperature Range: C80C86/-2 . . . . . $0^oC$ to $+70^oC$
I80C86/-2 . . . . . $-40^oC$ to $+85^oC$
M80C86/-2 . . . . . $-55^oC$ to $+125^oC$

## DC Electrical Specifications

$V_{CC}$ = 5.0V, ±10%; $T_A$ = $0^oC$ to $+70^oC$ (C80C86, C80C86-2)
$V_{CC}$ = 5.0V, ±10%; $T_A$ = $-40^oC$ to $+85^oC$ (I80C86, I80C86-2)
$V_{CC}$ = 5.0V, ±10%; $T_A$ = $-55^oC$ to $+125^oC$ (M80C86)
$V_{CC}$ = 5.0V, ±5%; $T_A$ = $-55^oC$ to $+125^oC$ (M80C86-2)

| SYMBOL | PARAMETER | MIN | MAX | UNITS | TEST CONDITION |
|---|---|---|---|---|---|
| $V_{IH}$ | Logical One Input Voltage<br>Logical One Input Voltage | 2.0<br>2.2 | | V<br>V | C80C86, I80C86 (Note 5)<br>M80C86 (Note 5) |
| $V_{IL}$ | Logical Zero Input Voltage | | 0.8 | V | |
| $V_{IHC}$ | CLK Logical One Input Voltage | $V_{CC}$ -0.8 | | V | |
| $V_{ILC}$ | CLK Logical Zero Input Voltage | | 0.8 | V | |
| $V_{OH}$ | Output High Voltage | 3.0<br>$V_{CC}$ -0.4 | | V<br>V | $I_{OH}$ = -2.5mA<br>$I_{OH}$ = -100µA |
| $V_{OL}$ | Output Low Voltage | | 0.4 | V | $I_{OL}$ = +2.5mA |
| $I_I$ | Input Leakage Current | -1.0 | 1.0 | µA | $V_{IN}$ = GND or $V_{CC}$ DIP Pins 17-19, 21-23, 33 |
| $I_{BHH}$ | Input Current-Bus Hold High | -40 | -400 | µA | $V_{IN}$ = - 3.0V (Note 1) |
| $I_{BHL}$ | Input Current-Bus Hold Low | 40 | 400 | µA | $V_{IN}$ = - 0.8V (Note 2) |
| $I_O$ | Output Leakage Current | - | -10.0 | µA | $V_{OUT}$ = GND (Note 4) |
| $I_{CCSB}$ | Standby Power Supply Current | - | 500 | µA | $V_{CC}$ = - 5.5V (Note 3) |
| $I_{CCOP}$ | Operating Power Supply Current | - | 10 | mA/MHz | FREQ = Max, $V_{IN}$ = $V_{CC}$ or GND, Outputs Open |

## Capacitance $T_A$ = $25^oC$

| SYMBOL | PARAMETER | TYPICAL | UNITS | TEST CONDITIONS |
|---|---|---|---|---|
| $C_{IN}$ | Input Capacitance | 25 | pF | FREQ = 1MHz. All measurements are referenced to device GND |
| $C_{OUT}$ | Output Capacitance | 25 | pF | FREQ = 1MHz. All measurements are referenced to device GND |
| $C_{I/O}$ | I/O Capacitance | 25 | pF | FREQ = 1MHz. All measurements are referenced to device GND |

NOTES:

2. IBHH should be measured after raising $V_{IN}$ to $V_{CC}$ and then lowering to 3.0V on the following pins 2-16, 26-32, 34-39.
3. IBHL should be measured after lowering $V_{IN}$ to GND and then raising to 0.8V on the following pins: 2-16, 34-39.
4. ICCSB tested during clock high time after halt instruction executed. $V_{IN}$ = $V_{CC}$ or GND, $V_{CC}$ = 5.5V, Outputs unloaded.
5. IO should be measured by putting the pin in a high impedance state and then driving $V_{OUT}$ to GND on the following pins: 26-29 and 32.
6. MN/$\overline{MX}$ is a strap option and should be held to $V_{CC}$ or GND.

## AC Electrical Specifications

$V_{CC}$ = 5.0V ±10%; $T_A$ = 0°C to +70°C (C80C86, C80C86-2)
$V_{CC}$ = 5.0V ±100%; $T_A$ = -40°C to +85°C (I80C86, I80C86-2)
$V_{CC}$ = 5.0V ±100%; $T_A$ = -55°C to +125°C (M80C86)
$V_{CC}$ = 5.0V ±5%; $T_A$ = -55°C to +125°C (M80C86-2)

**MINIMUM COMPLEXITY SYSTEM**

| SYMBOL | | PARAMETER | 80C86 | | 80C86-2 | | UNITS | TEST CONDITIONS |
|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | | |
| **TIMING REQUIREMENTS** | | | | | | | | |
| (1) | TCLCL | Cycle Period | 200 | | 125 | | ns | |
| (2) | TCLCH | CLK Low Time | 118 | | 68 | | ns | |
| (3) | TCHCL | CLK High Time | 69 | | 44 | | ns | |
| (4) | TCH1CH2 | CLK Rise Time | | 10 | | 10 | ns | From 1.0V to 3.5V |
| (5) | TCL2C1 | CLK Fall Time | | 10 | | 10 | ns | From 3.5V to 1.0V |
| (6) | TDVCL | Data In Setup Time | 30 | | 20 | | ns | |
| (7) | TCLDX1 | Data In Hold Time | 10 | | 10 | | ns | |
| (8) | TR1VCL | RDY Setup Time into 82C84A (Notes 7, 8) | 35 | | 35 | | ns | |
| (9) | TCLR1X | RDY Hold Time into 82C84A (Notes 7, 8) | 0 | | 0 | | ns | |
| (10) | TRYHCH | READY Setup Time into 80C86 | 118 | | 68 | | ns | |
| (11) | TCHRYX | READY Hold Time into 80C86 | 30 | | 20 | | ns | |
| (12) | TRYLCL | READY Inactive to CLK (Note 9) | -8 | | -8 | | ns | |
| (13) | THVCH | HOLD Setup Time | 35 | | 20 | | ns | |
| (14) | TINVCH | INTR, NMI, $\overline{\text{TEST}}$ Setup Time (Note 8) | 30 | | 15 | | ns | |
| (15) | TILIH | Input Rise Time (Except CLK) | | 15 | | 15 | ns | From 0.8V to 2.0V |
| (16) | TIHIL | Input Fall Time (Except CLK) | | 15 | | 15 | ns | From 2.0V to 0.8V |
| **TIMING RESPONSES** | | | | | | | | |
| (17) | TCLAV | Address Valid Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF |
| (18) | TCLAX | Address Hold Time | 10 | | 10 | | ns | $C_L$ = 100pF |
| (19) | TCLAZ | Address Float Delay | TCLAX | 80 | TCLAX | 50 | ns | $C_L$ = 100pF |
| (20) | TCHSZ | Status Float Delay | | 80 | | 50 | ns | $C_L$ = 100pF |
| (21) | TCHSV | Status Active Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF |
| (22) | TLHLL | ALE Width | TCLCH-20 | | TCLCH-10 | | ns | $C_L$ = 100pF |
| (23) | TCLLH | ALE Active Delay | | 80 | | 50 | ns | $C_L$ = 100pF |
| (24) | TCHLL | ALE Inactive Delay | | 85 | | 55 | ns | $C_L$ = 100pF |

## AC Electrical Specifications

$V_{CC}$ = 5.0V ±10%; $T_A$ = 0°C to +70°C (C80C86, C80C86-2)
$V_{CC}$ = 5.0V ±100%; $T_A$ = -40°C to +85°C (I80C86, I80C86-2)
$V_{CC}$ = 5.0V ±100%; $T_A$ = -55°C to +125°C (M80C86)
$V_{CC}$ = 5.0V ±5%; $T_A$ = -55°C to +125°C (M80C86-2) **(Continued)**

**MINIMUM COMPLEXITY SYSTEM**

| SYMBOL | | PARAMETER | 80C86 | | 80C86-2 | | UNITS | TEST CONDITIONS |
|---|---|---|---|---|---|---|---|---|
| | | | MIN | MAX | MIN | MAX | | |
| (25) | TLLAX | Address Hold Time to ALE Inactive | TCHCL-10 | | TCHCL-10 | | ns | $C_L$ = 100pF |
| (26) | TCLDV | Data Valid Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF |
| (27) | TCLDX2 | Data Hold Time | 10 | | 10 | | ns | $C_L$ = 100pF |
| (28) | TWHDX | Data Hold Time After $\overline{WR}$ | TCLCL-30 | | TCLCL-30 | | ns | $C_L$ = 100pF |
| (29) | TCVCTV | Control Active Delay 1 | 10 | 110 | 10 | 70 | ns | $C_L$ = 100pF |
| (30) | TCHCTV | Control Active Delay 2 | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF |
| (31) | TCVCTX | Control Inactive Delay | 10 | 110 | 10 | 70 | ns | $C_L$ = 100pF |
| (32) | TAZRL | Address Float to READ Active | 0 | | 0 | | ns | $C_L$ = 100pF |
| (33) | TCLRL | $\overline{RD}$ Active Delay | 10 | 165 | 10 | 100 | ns | $C_L$ = 100pF |
| (34) | TCLRH | $\overline{RD}$ Inactive Delay | 10 | 150 | 10 | 80 | ns | $C_L$ = 100pF |
| (35) | TRHAV | $\overline{RD}$ Inactive to Next Address Active | TCLCL-45 | | TCLCL-40 | | ns | $C_L$ = 100pF |
| (36) | TCLHAV | HLDA Valid Delay | 10 | 160 | 10 | 100 | ns | $C_L$ = 100pF |
| (37) | TRLRH | $\overline{RD}$ Width | 2TCLCL-75 | | 2TCLCL-50 | | ns | $C_L$ = 100pF |
| (38) | TWLWH | $\overline{WR}$ Width | 2TCLCL-60 | | 2TCLCL-40 | | ns | $C_L$ = 100pF |
| (39) | TAVAL | Address Valid to ALE Low | TCLCH-60 | | TCLCH-40 | | ns | $C_L$ = 100pF |
| (40) | TOLOH | Output Rise Time | | 20 | | 15 | ns | From 0.8V to 2.0V |
| (41) | TOHOL | Output Fall Time | | 20 | | 15 | ns | From 2.0V to 0.8V |

NOTES:

7. Signal at 82C84A shown for reference only.
8. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
9. Applies only to T2 state (8ns into T3).

## Waveforms

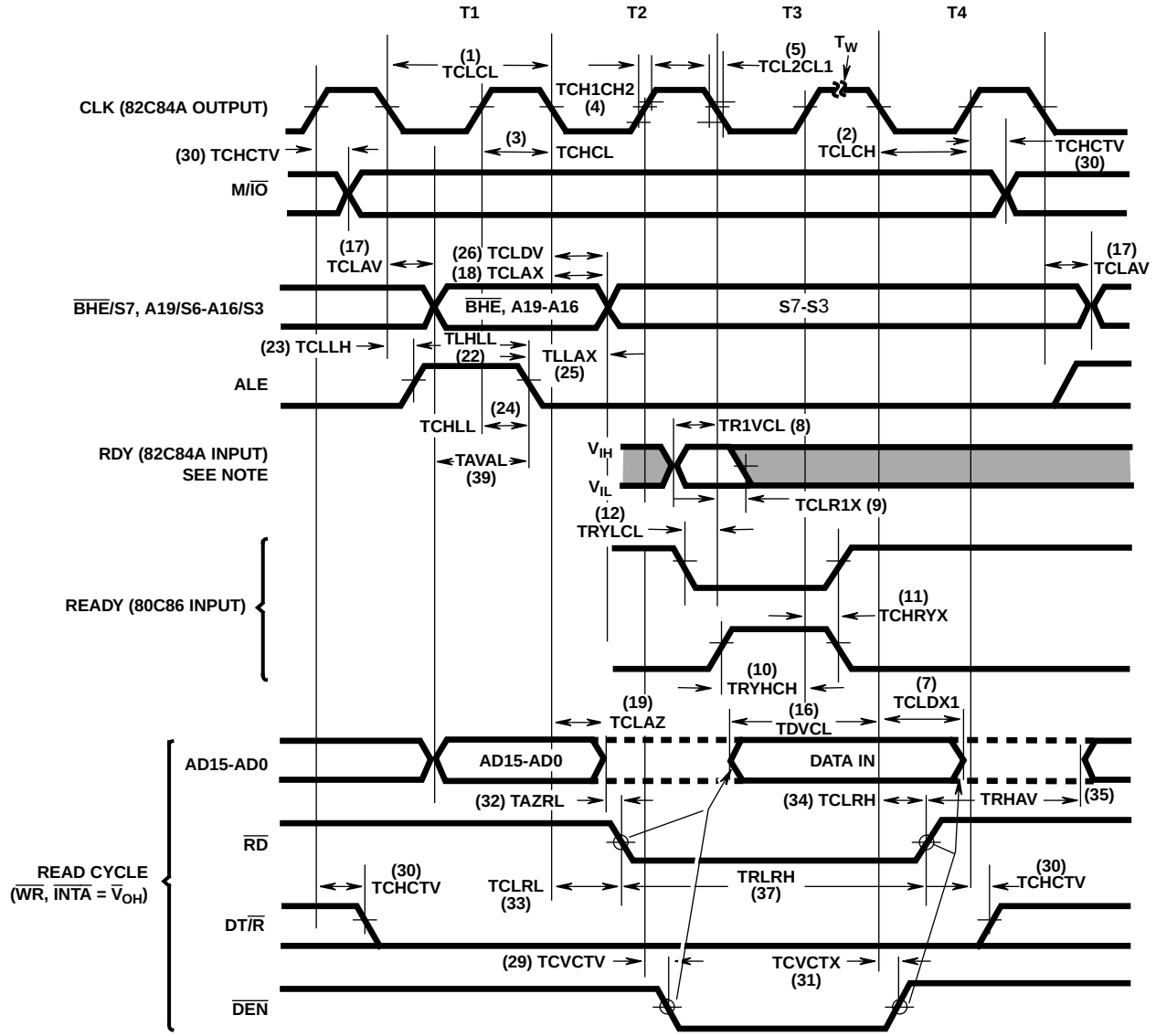



**FIGURE 7A. BUS TIMING - MINIMUM MODE SYSTEM**

NOTE: Signals at 82C84A are shown for reference only. RDY is sampled near the end of T2, T3, TW to determine if TW machine states are to be inserted.

## Waveforms (Continued)

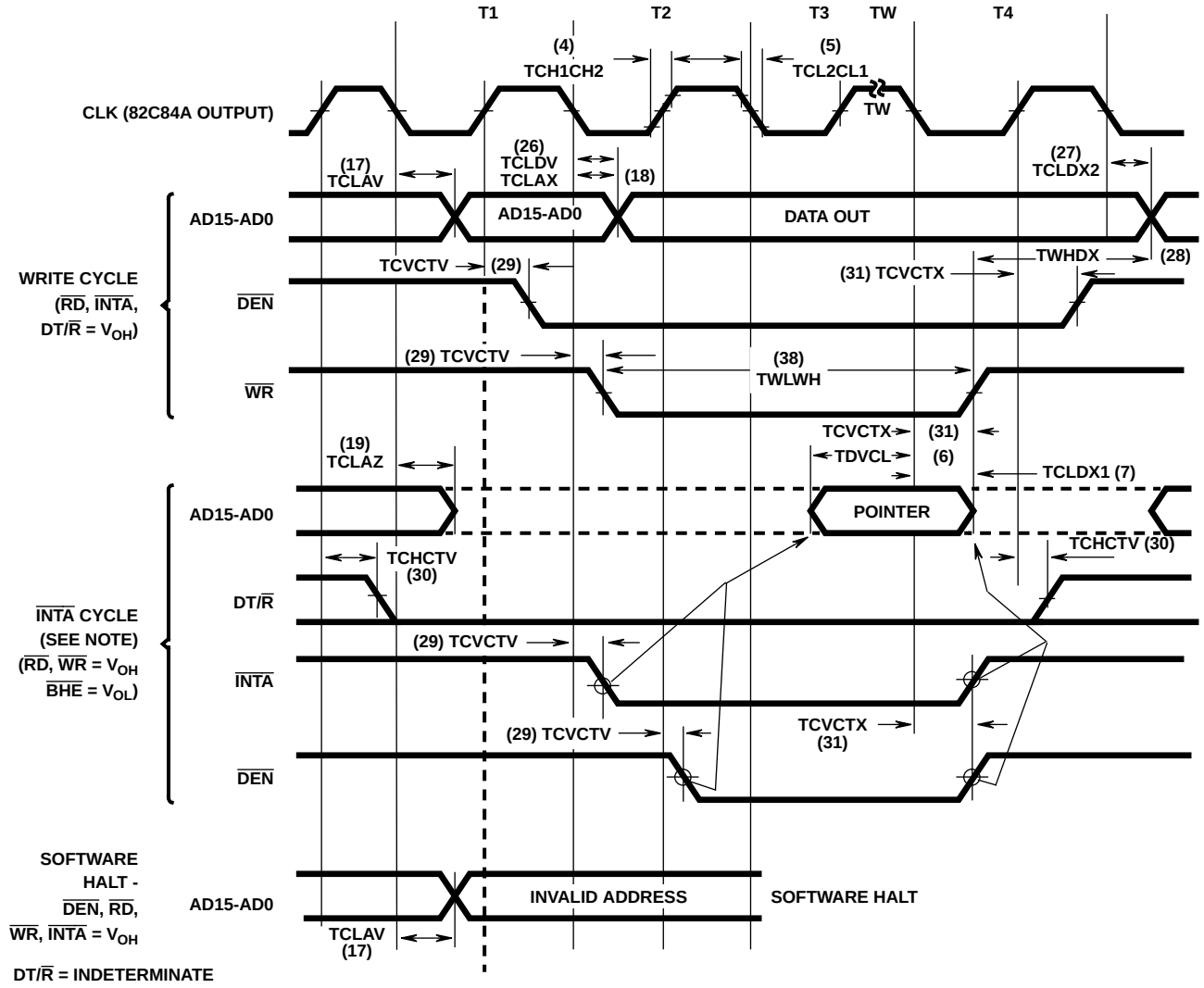



**FIGURE 7B. BUS TIMING - MINIMUM MODE SYSTEM**

NOTE: Two $\overline{\text{INTA}}$ cycles run back-to-back. The 80C86 local ADDR/DATA bus is floating during both $\overline{\text{INTA}}$ cycles. Control signals are shown for the second $\overline{\text{INTA}}$ cycle.

## AC Electrical Specifications

$V_{CC}$ = 5.0V ±10% $T_A$ = 0°C to +70°C (C80C86, C80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -40°C to +85°C (I80C86, I80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -55°C to +125°C (M80C86)
$V_{CC}$ = 5.0V ±5%; $T_A$ = -55°C to +125°C (M80C86-2)

**MAX MODE SYSTEM (USING 82C88 BUS CONTROLLER)**

| TIMING REQUIREMENTS | | | 80C86 | | 80C86-2 | | | |
|---|---|---|---|---|---|---|---|---|
| SYMBOL | | PARAMETER | MIN | MAX | MIN | MAX | UNITS | TEST CONDITIONS |
| (1) | TCLCL | CLK Cycle Period | 200 | | 125 | | ns | |
| (2) | TCLCH | CLK Low Time | 118 | | 68 | | ns | |
| (3) | TCHCL | CLK High Time | 69 | | 44 | | ns | |
| (4) | TCH1CH2 | CLK Rise Time | | 10 | | 10 | ns | From 1.0V to 3.5V |
| (5) | TCL2CL1 | CLK Fall Time | | 10 | | 10 | ns | From 3.5V to 1.0V |
| (6) | TDVCL | Data in Setup Time | 30 | | 20 | | ns | |
| (7) | TCLDX1 | Data In Hold Time | 10 | | 10 | | ns | |
| (8) | TR1VCL | RDY Setup Time into 82C84A (Notes 10, 11) | 35 | | 35 | | ns | |
| (9) | TCLR1X | RDY Hold Time into 82C84A (Notes 10, 11) | 0 | | 0 | | ns | |
| (10) | TRYHCH | READY Setup Time into 80C86 | 118 | | 68 | | ns | |
| (11) | TCHRYX | READY Hold Time into 80C86 | 30 | | 20 | | ns | |
| (12) | TRYLCL | READY Inactive to CLK (Note 12) | -8 | | -8 | | ns | |
| (13) | TINVCH | Setup Time for Recognition (INTR, NMI, $\overline{TEST}$) (Note 11) | 30 | | 15 | | ns | |
| (14) | TGVCH | $\overline{RQ}$/$\overline{GT}$ Setup Time | 30 | | 15 | | ns | |
| (15) | TCHGX | $\overline{RQ}$ Hold Time into 80C86 (Note 13) | 40 | TCHCL+ 10 | 30 | TCHCL+ 10 | ns | |
| (16) | TILIH | Input Rise Time (Except CLK) | | 15 | | 15 | ns | From 0.8V to 2.0V |
| (17) | TIHIL | Input Fall Time (Except CLK) | | 15 | | 15 | ns | From 2.0V to 0.8V |
| **TIMING RESPONSES** | | | | | | | | |
| (18) | TCLML | Command Active Delay (Note 10) | 5 | 35 | 5 | 35 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (19) | TCLMH | Command Inactive (Note 10) | 5 | 35 | 5 | 35 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (20) | TRYHSH | READY Active to Status Passive (Notes 12, 14) | | 110 | | 65 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (21) | TCHSV | Status Active Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (22) | TCLSH | Status Inactive Delay (Note 14) | 10 | 130 | 10 | 70 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |

## AC Electrical Specifications

$V_{CC}$ = 5.0V ±10% $T_A$ = 0°C to +70°C (C80C86, C80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -40°C to +85°C (I80C86, I80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -55°C to +125°C (M80C86)
$V_{CC}$ = 5.0V ±5%; $T_A$ = -55°C to +125°C (M80C86-2) **(Continued)**

**MAX MODE SYSTEM (USING 82C88 BUS CONTROLLER)**

| TIMING REQUIREMENTS | | | 80C86 | | 80C86-2 | | | |
|---|---|---|---|---|---|---|---|---|
| SYMBOL | | PARAMETER | MIN | MAX | MIN | MAX | UNITS | TEST CONDITIONS |
| (23) | TCLAV | Address Valid Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (24) | TCLAX | Address Hold Time | 10 | | 10 | | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (25) | TCLAZ | Address Float Delay | TCLAX | 80 | TCLAX | 50 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (26) | TCHSZ | Status Float Delay | | 80 | | 50 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (27) | TSVLH | Status Valid to ALE High (Note 10) | | 20 | | 20 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (28) | TSVMCH | Status Valid to MCE High (Note 10) | | 30 | | 30 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (29) | TCLLH | CLK low to ALE Valid (Note 10) | | 20 | | 20 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (30) | TCLMCH | CLK low to MCE High (Note 10) | | 25 | | 25 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (31) | TCHLL | ALE Inactive Delay (Note 10) | 4 | 18 | 4 | 18 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (32) | TCLMCL | MCE Inactive Delay (Note 10) | | 15 | | 15 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (33) | TCLDV | Data Valid Delay | 10 | 110 | 10 | 60 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (34) | TCLDX2 | Data Hold Time | 10 | | 10 | | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |

## AC Electrical Specifications

$V_{CC}$ = 5.0V ±10% $T_A$ = 0°C to +70°C (C80C86, C80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -40°C to +85°C (I80C86, I80C86-2)
$V_{CC}$ = 5.0V ±10%; $T_A$ = -55°C to +125°C (M80C86)
$V_{CC}$ = 5.0V ±5%; $T_A$ = -55°C to +125°C (M80C86-2) **(Continued)**

**MAX MODE SYSTEM (USING 82C88 BUS CONTROLLER)**

| TIMING REQUIREMENTS | | | 80C86 | | 80C86-2 | | | |
|---|---|---|---|---|---|---|---|---|
| SYMBOL | | PARAMETER | MIN | MAX | MIN | MAX | UNITS | TEST CONDITIONS |
| (35) | TCVNV | Control Active Delay (Note 10) | 5 | 45 | 5 | 45 | ns | $C_L$ = 100pF for All 80C86 Outputs (In Addition to 80C86 Self Load) |
| (36) | TCVNX | Control Inactive Delay (Note 10) | 10 | 45 | 10 | 45 | ns | $C_L$ = 100pF |
| (37) | TAZRL | Address Float to Read Active | 0 | | 0 | | ns | $C_L$ = 100pF |
| (38) | TCLRL | $\overline{RD}$ Active Delay | 10 | 165 | 10 | 100 | ns | $C_L$ = 100pF |
| (39) | TCLRH | $\overline{RD}$ Inactive Delay | 10 | 150 | 10 | 80 | ns | $C_L$ = 100pF |
| (40) | TRHAV | $\overline{RD}$ Inactive to Next Address Active | TCLCL -45 | | TCLCL -40 | | ns | $C_L$ = 100pF |
| (41) | TCHDTL | Direction Control Active Delay (Note 10) | | 50 | | 50 | ns | $C_L$ = 100pF |
| (42) | TCHDTH | Direction Control Inactive Delay (Note 10) | | 30 | | 30 | ns | $C_L$ = 100pF |
| (43) | TCLGL | $\overline{GT}$ Active Delay | 10 | 85 | 0 | 50 | ns | $C_L$ = 100pF |
| (44) | TCLGH | $\overline{GT}$ Inactive Delay | 10 | 85 | 0 | 50 | ns | $C_L$ = 100pF |
| (45) | TRLRH | $\overline{RD}$ Width | 2TCLCL -75 | | 2TCLCL -50 | | ns | $C_L$ = 100pF |
| (46) | TOLOH | Output Rise Time | | 20 | | 15 | ns | From 0.8V to 2.0V |
| (47) | TOHOL | Output Fall Time | | 20 | | 15 | ns | From 2.0V to 0.8V |

NOTES:

10. Signal at 82C84A or 82C88 shown for reference only.
11. Setup requirement for asynchronous signal only to guarantee recognition at next CLK.
12. Applies only to T2 state (8ns into T3).
13. The 80C86 actively pulls the $\overline{RQ}/\overline{GT}$ pin to a logic one on the following clock low time.
14. Status lines return to their inactive (logic one) state after CLK goes low and READY goes high.

## Waveforms

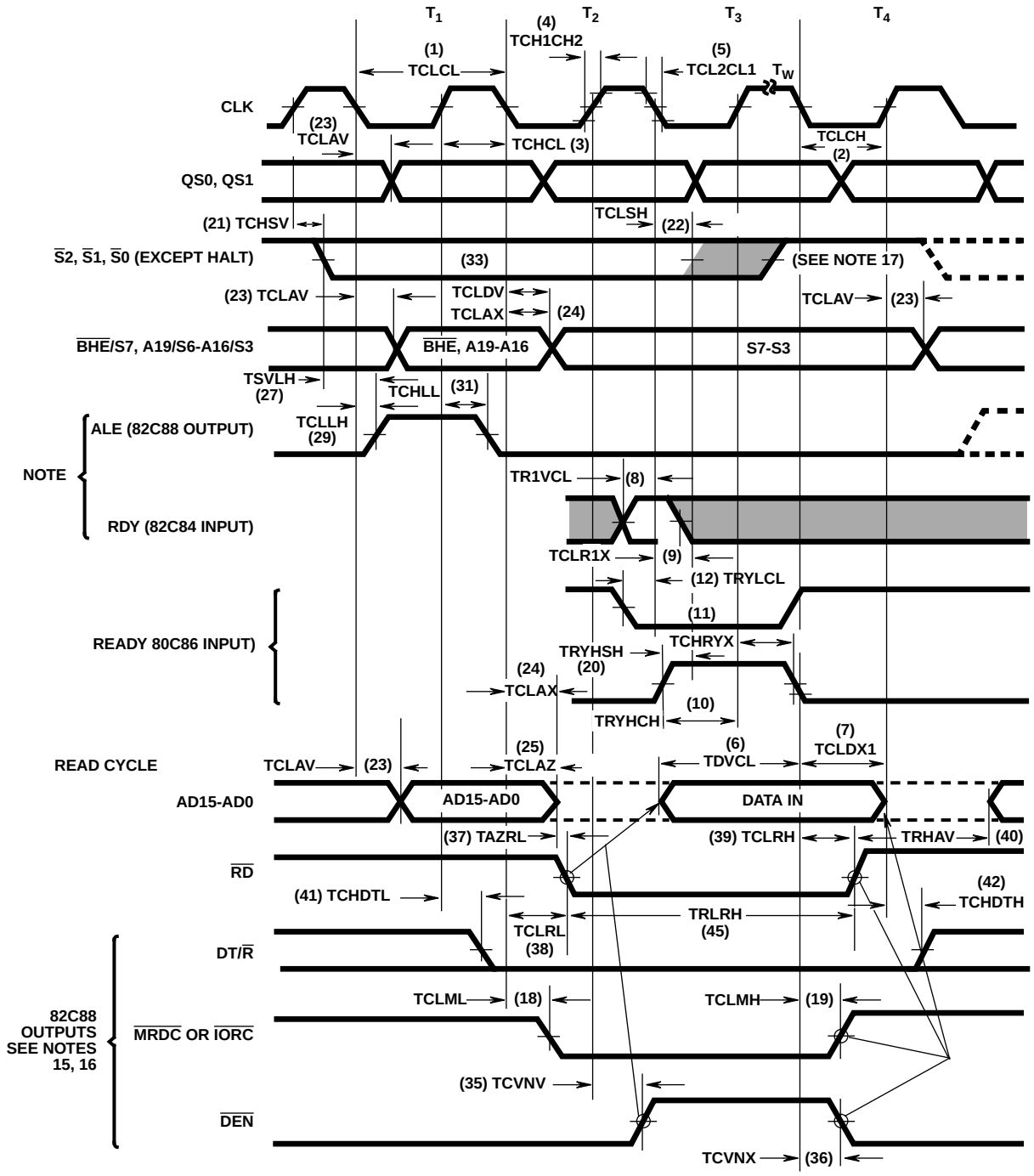



**FIGURE 8A. BUS TIMING - MAXIMUM MODE (USING 82C88)**

NOTES:

15. Signals at 82C84A or 82C88 are shown for reference only. RDY is sampled near the end of T2, T3, TW to determine if TW machine states are to be inserted.
16. The issuance of the 82C88 command and control signals ($\overline{\text{MRDC}}$, $\overline{\text{MWTC}}$, $\overline{\text{AMWC}}$, $\overline{\text{IORC}}$, $\overline{\text{IOWC}}$, $\overline{\text{AIOWC}}$, $\overline{\text{INTA}}$, and $\overline{\text{DEN}}$) lags the active high 82C88 CEN.
17. Status inactive in state just prior to T4.

## Waveforms (Continued)

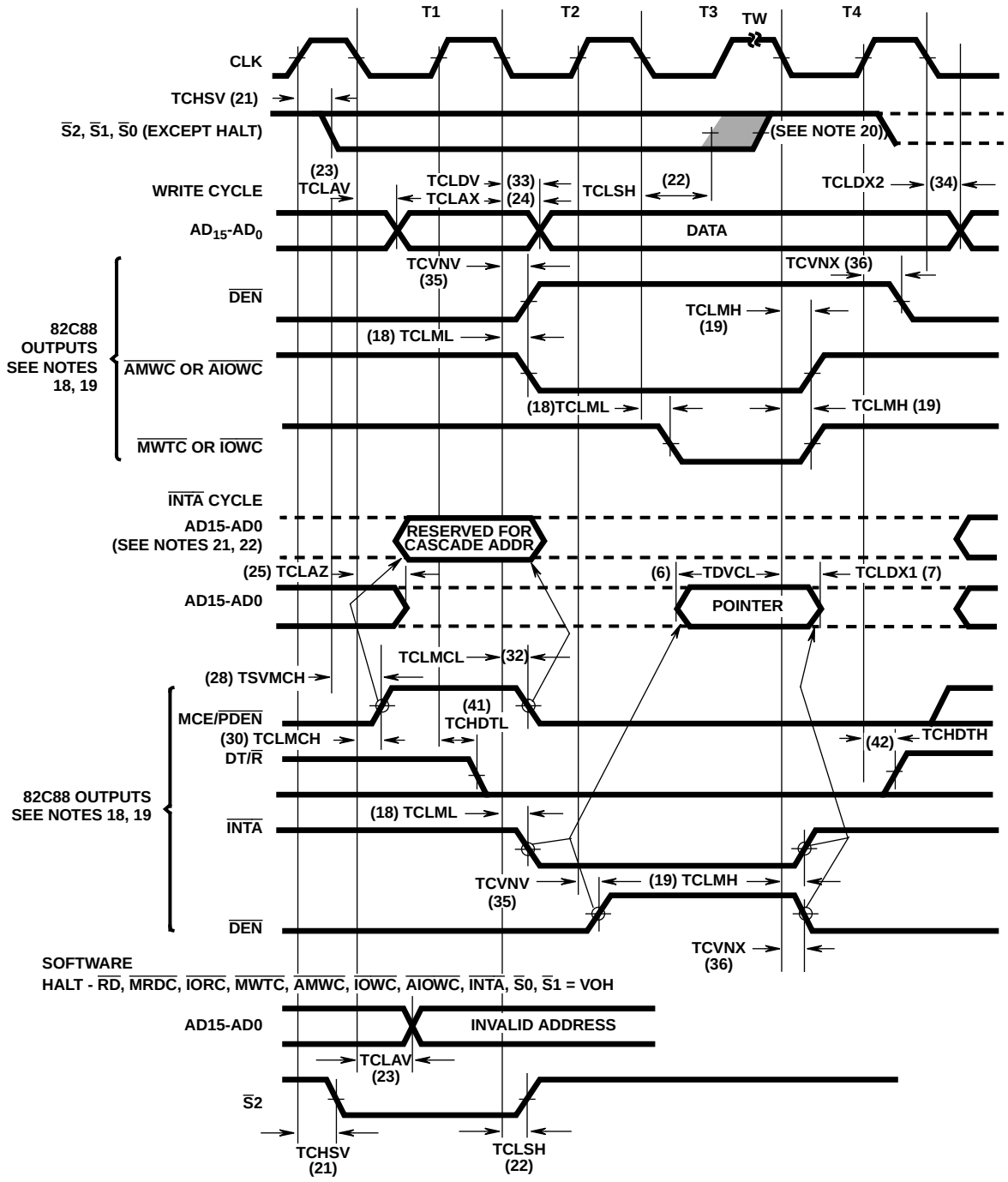
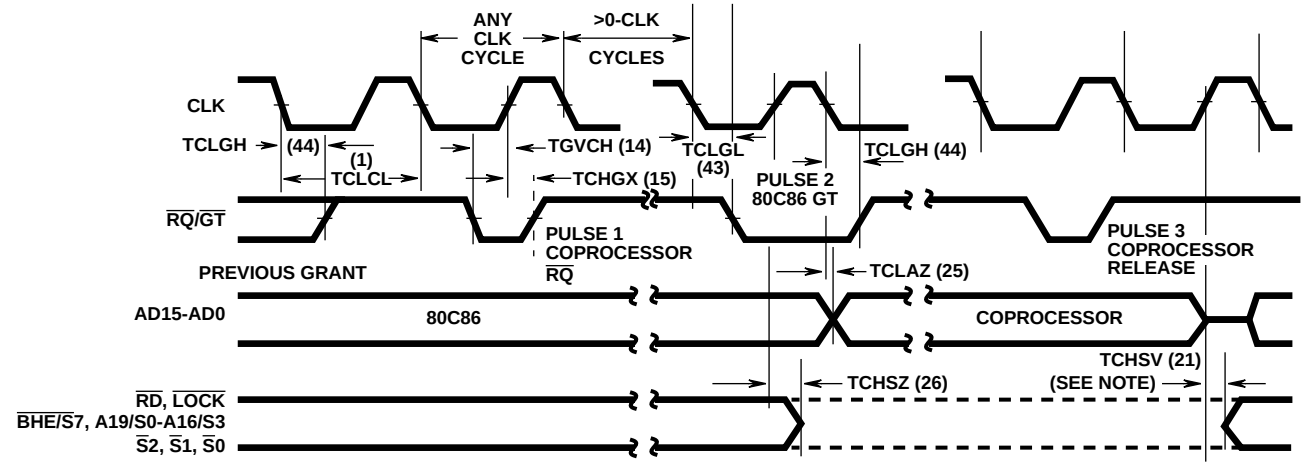



**FIGURE 8B. BUS TIMING - MAXIMUM MODE (USING 82C88)**

NOTES:

18. Signals at 82C84A or 82C86 are shown for reference only.
19. The issuance of the 82C88 command and control signals ($\overline{MRDC}$, $\overline{MWTC}$, $\overline{AMWC}$, $\overline{IORC}$, $\overline{IOWC}$, $\overline{AIOWC}$, $\overline{INTA}$ and $\overline{DEN}$) lags the active high 82C88 CEN.
20. Status inactive in state just prior to T4.
21. Cascade address is valid between first and second $\overline{INTA}$ cycles.
22. Two $\overline{INTA}$ cycles run back-to-back. The 80C86 local ADDR/DATA bus is floating during both $\overline{INTA}$ cycles. Control for pointer address is shown for second $\overline{INTA}$ cycle.

## Waveforms (Continued)




NOTE: The coprocessor may not drive the busses outside the region shown without risking contention.

**FIGURE 9. REQUEST/GRANT SEQUENCE TIMING (MAXIMUM MODE ONLY)**

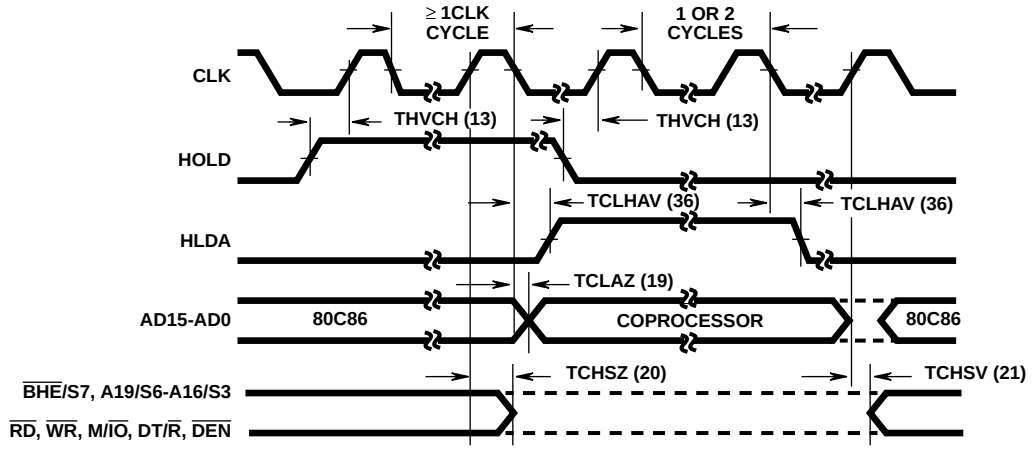
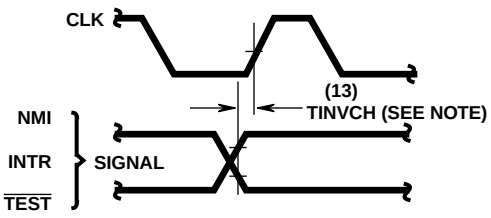



**FIGURE 10. HOLD/HOLD ACKNOWLEDGE TIMING (MINIMUM MODE ONLY)**




NOTE: Setup requirements for asynchronous signals only to guarantee recognition at next CLK.

**FIGURE 11. ASYNCHRONOUS SIGNAL RECOGNITION**

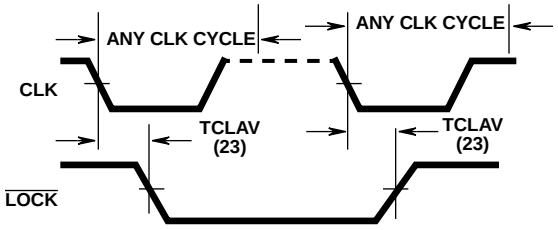



**FIGURE 12. BUS LOCK SIGNAL TIMING (MAXIMUM MODE ONLY)**

## Waveforms (Continued)

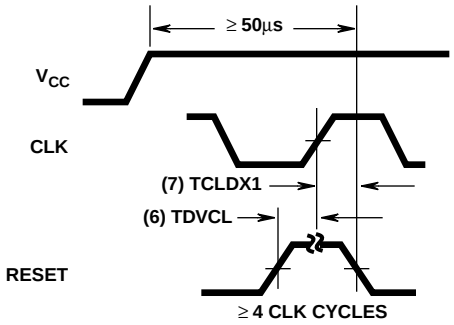
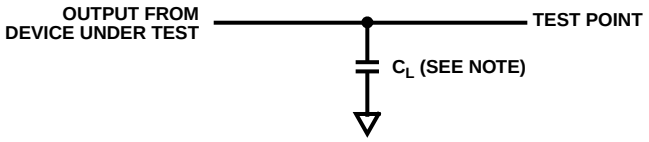


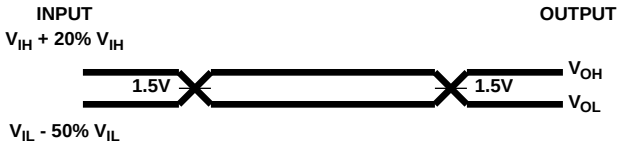

**FIGURE 13. RESET TIMING**

## AC Test Circuit




NOTE: Includes stay and jig capacitance.

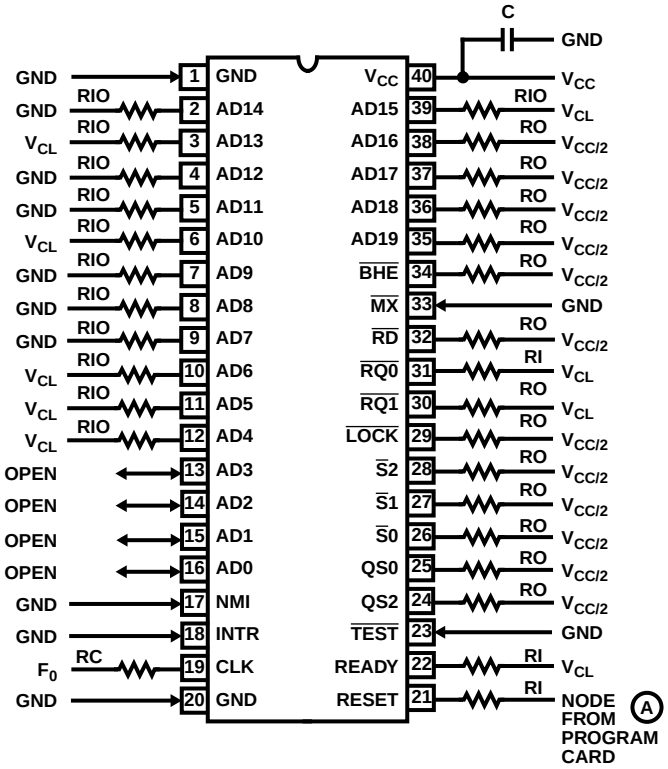
## AC Testing Input, Output Waveform




NOTE: AC Testing: All input signals (other than CLK) must switch between $V_{ILMAX}$ -50% $V_{IL}$ and $V_{IHMIN}$ +20% $V_{IH}$. CLK must switch between 0.4V and $V_{CC}$-0.4 Input rise and fall times are driven at 1ns/V.

## Burn-In Circuits

MD80C86 CERDIP




NOTES:

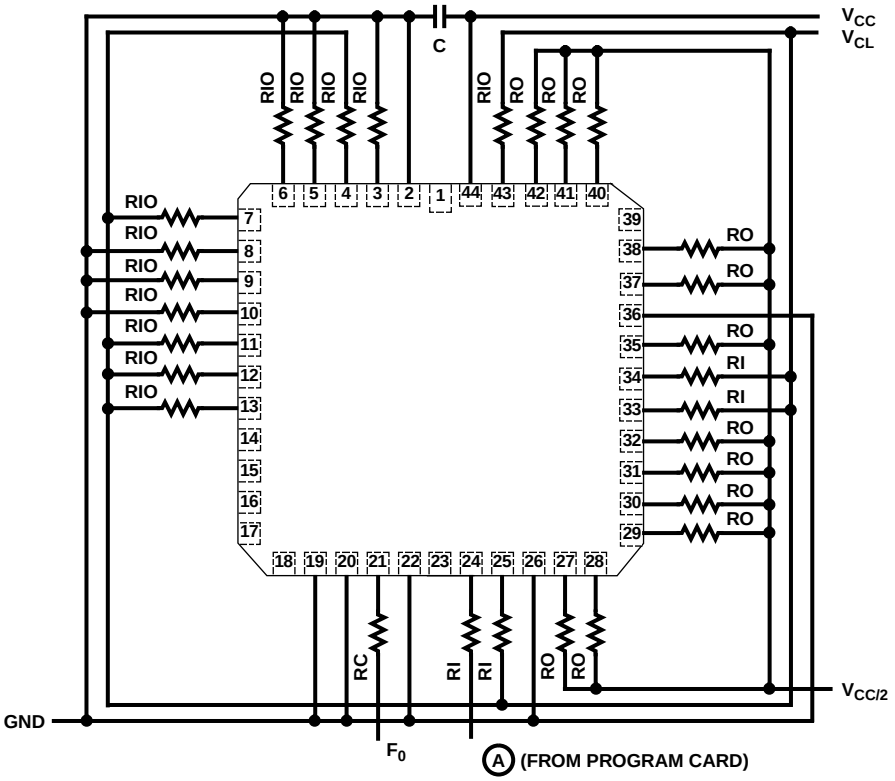
$V_{CC}$ = 5.5V ±0.5V, GND = 0V.
Input voltage limits (except clock):
$V_{IL}$ (maximum) = 0.4V
$V_{IH}$ (minimum) = 2.6V, $V_{IH}$ (clock) = ($V_{CC}$ -0.4V) minimum.
$V_{CC/2}$ is external supply set to 2.7V ±10%.
$V_{CL}$ is generated on program card ($V_{CC}$ - 0.65V).
Pins 13 - 16 input sequenced instructions from internal hold devices.
$F_0$ = 100kHz ±10%.
Node (A) = a 40µs pulse every 2.56ms.

COMPONENTS:

1. RI = 10kΩ ±5%, 1/4W
2. RO = 1.2kΩ ±5%, 1/4W
3. RIO = 2.7kΩ ±5%, 1/4W
4. RC = 1kΩ ± 5%, 1/4W
5. C = 0.01µF (Minimum)

## Burn-In Circuits (Continued)

**MR80C86 CLCC**




NOTES:

$V_{CC}$ = 5.5V ±0.5V, GND = 0V.
Input voltage limits (except clock):
$V_{IL}$ (maximum) = 0.4V
$V_{IH}$ (minimum) = 2.6V, $V_{IH}$ (clock) = ($V_{CC}$ -0.4V) minimum.
$V_{CC/2}$ is external supply set to 2.7V ±10%.
$V_{CL}$ is generated on program card ($V_{CC}$ - 0.65V).
Pins 13 - 16 input sequenced instructions from internal hold devices.
$F_0$ = 100kHz ±10%.
Node (A) = a 40µs pulse every 2.56ms.

COMPONENTS:

1. RI = 10kΩ ±5%, 1/4W
2. RO = 1.2kΩ ±5%, 1/4W
3. RIO = 2.7kΩ ±5%, 1/4W
4. RC = 1kΩ ± 5%, 1/4W
5. C = 0.01µF (Minimum)

## Metallization Topology

**DIE DIMENSIONS:**
249.2 x 290.9 x 19

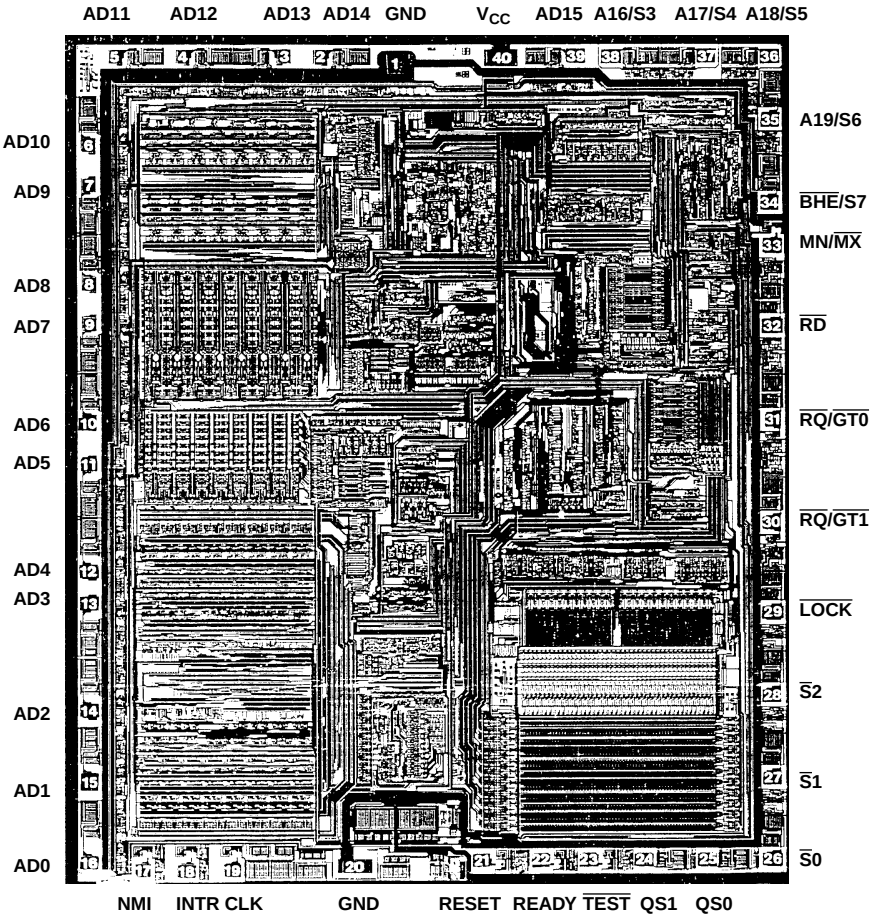
**METALLIZATION:**
Type: Silicon - Aluminum
Thickness: 11kÅ ±2kÅ

**GLASSIVATION:**
Type: Nitrox
Thickness: 10kÅ ±2kÅ

**WORST CASE CURRENT DENSITY:**
$1.5 \times 10^5$ A/cm$^2$

## Metallization Mask Layout

**80C86**



## Instruction Set Summary

| MNEMONIC AND DESCRIPTION | INSTRUCTION CODE | | | |
|---|---|---|---|---|
| | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 |
| **DATA TRANSFER** | | | | |
| **MOV = MOVE:** | | | | |
| Register/Memory to/from Register | 1 0 0 0 1 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 1 0 0 0 1 1 w | mod 0 0 0 r/m | data | data if w 1 |
| Immediate to Register | 1 0 1 1 w reg | data | data if w 1 | |
| Memory to Accumulator | 1 0 1 0 0 0 0 w | addr-low | addr-high | |
| Accumulator to Memory | 1 0 1 0 0 0 1 w | addr-low | addr-high | |
| Register/Memory to Segment Register †† | 1 0 0 0 1 1 1 0 | mod 0 reg r/m | | |
| Segment Register to Register/Memory | 1 0 0 0 1 1 0 0 | mod 0 reg r/m | | |
| **PUSH = Push:** | | | | |
| Register/Memory | 1 1 1 1 1 1 1 1 | mod 1 1 0 r/m | | |
| Register | 0 1 0 1 0 reg | | | |
| Segment Register | 0 0 0 reg 1 1 0 | | | |
| **POP = Pop:** | | | | |
| Register/Memory | 1 0 0 0 1 1 1 1 | mod 0 0 0 r/m | | |
| Register | 0 1 0 1 1 reg | | | |
| Segment Register | 0 0 0 reg 1 1 1 | | | |
| **XCHG = Exchange:** | | | | |
| Register/Memory with Register | 1 0 0 0 0 1 1 w | mod reg r/m | | |
| Register with Accumulator | 1 0 0 1 0 reg | | | |
| **IN = Input from:** | | | | |
| Fixed Port | 1 1 1 0 0 1 0 w | port | | |
| Variable Port | 1 1 1 0 1 1 0 w | | | |
| **OUT = Output to:** | | | | |
| Fixed Port | 1 1 1 0 0 1 1 w | port | | |
| Variable Port | 1 1 1 0 1 1 1 w | | | |
| **XLAT =** Translate Byte to AL | 1 1 0 1 0 1 1 1 | | | |
| **LEA =** Load EA to Register2 | 1 0 0 0 1 1 0 1 | mod reg r/m | | |
| **LDS =** Load Pointer to DS | 1 1 0 0 0 1 0 1 | mod reg r/m | | |
| **LES =** Load Pointer to ES | 1 1 0 0 0 1 0 0 | mod reg r/m | | |
| **LAHF =** Load AH with Flags | 1 0 0 1 1 1 1 1 | | | |
| **SAHF =** Store AH into Flags | 1 0 0 1 1 1 1 0 | | | |
| **PUSHF =** Push Flags | 1 0 0 1 1 1 0 0 | | | |
| **POPF =** Pop Flags | 1 0 0 1 1 1 0 1 | | | |
| **ARITHMETIC** | | | | |
| **ADD = Add:** | | | | |
| Register/Memory with Register to Either | 0 0 0 0 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 s w | mod 0 0 0 r/m | data | data if s:w = 01 |
| Immediate to Accumulator | 0 0 0 0 0 1 0 w | data | data if w = 1 | |
| **ADC = Add with Carry:** | | | | |
| Register/Memory with Register to Either | 0 0 0 1 0 0 d w | mod reg r/m | | |

## Instruction Set Summary (Continued)

| MNEMONIC AND DESCRIPTION | INSTRUCTION CODE | | | |
|---|---|---|---|---|
| | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 |
| Immediate to Register/Memory | 1 0 0 0 0 0 s w | mod 0 1 0 r/m | data | data if s:w = 01 |
| Immediate to Accumulator | 0 0 0 1 0 1 0 w | data | data if w = 1 | |
| **INC = Increment:** | | | | |
| Register/Memory | 1 1 1 1 1 1 1 w | mod 0 0 0 r/m | | |
| Register | 0 1 0 0 0 reg | | | |
| **AAA =** ASCII Adjust for Add | 0 0 1 1 0 1 1 1 | | | |
| **DAA =** Decimal Adjust for Add | 0 0 1 0 0 1 1 1 | | | |
| **SUB =** Subtract: | | | | |
| Register/Memory and Register to Either | 0 0 1 0 1 0 d w | mod reg r/m | | |
| Immediate from Register/Memory | 1 0 0 0 0 0 s w | mod 1 0 1 r/m | data | data if s:w = 01 |
| Immediate from Accumulator | 0 0 1 0 1 1 0 w | data | data if w = 1 | |
| **SBB = Subtract with Borrow** | | | | |
| Register/Memory and Register to Either | 0 0 0 1 1 0 d w | mod reg r/m | | |
| Immediate from Register/Memory | 1 0 0 0 0 0 s w | mod 0 1 1 r/m | data | data if s:w = 01 |
| Immediate from Accumulator | 0 0 0 1 1 1 0 w | data | data if w = 1 | |
| **DEC = Decrement:** | | | | |
| Register/Memory | 1 1 1 1 1 1 1 w | mod 0 0 1 r/m | | |
| Register | 0 1 0 0 1 reg | | | |
| **NEG = Change Sign** | 1 1 1 1 0 1 1 w | mod 0 1 1 r/m | | |
| **CMP = Compare:** | | | | |
| Register/Memory and Register | 0 0 1 1 1 0 d w | mod reg r/m | | |
| Immediate with Register/Memory | 1 0 0 0 0 0 s w | mod 1 1 1 r/m | data | data if s:w = 01 |
| Immediate with Accumulator | 0 0 1 1 1 1 0 w | data | data if w = 1 | |
| **AAS =** ASCII Adjust for Subtract | 0 0 1 1 1 1 1 1 | | | |
| **DAS =** Decimal Adjust for Subtract | 0 0 1 0 1 1 1 1 | | | |
| **MUL =** Multiply (Unsigned) | 1 1 1 1 0 1 1 w | mod 1 0 0 r/m | | |
| **IMUL =** Integer Multiply (Signed) | 1 1 1 1 0 1 1 w | mod 1 0 1 r/m | | |
| **AAM =** ASCII Adjust for Multiply | 1 1 0 1 0 1 0 0 | 0 0 0 0 1 0 1 0 | | |
| **DIV =** Divide (Unsigned) | 1 1 1 1 0 1 1 w | mod 1 1 0 r/m | | |
| **IDIV =** Integer Divide (Signed) | 1 1 1 1 0 1 1 w | mod 1 1 1 r/m | | |
| **AAD =** ASCII Adjust for Divide | 1 1 0 1 0 1 0 1 | 0 0 0 0 1 0 1 0 | | |
| **CBW =** Convert Byte to Word | 1 0 0 1 1 0 0 0 | | | |
| **CWD =** Convert Word to Double Word | 1 0 0 1 1 0 0 1 | | | |
| **LOGIC** | | | | |
| **NOT = Invert** | 1 1 1 1 0 1 1 w | mod 0 1 0 r/m | | |
| **SHL/SAL =** Shift Logical/Arithmetic Left | 1 1 0 1 0 0 v w | mod 1 0 0 r/m | | |
| **SHR =** Shift Logical Right | 1 1 0 1 0 0 v w | mod 1 0 1 r/m | | |
| **SAR =** Shift Arithmetic Right | 1 1 0 1 0 0 v w | mod 1 1 1 r/m | | |
| **ROL =** Rotate Left | 1 1 0 1 0 0 v w | mod 0 0 0 r/m | | |
| **ROR =** Rotate Right | 1 1 0 1 0 0 v w | mod 0 0 1 r/m | | |
| **RCL =** Rotate Through Carry Flag Left | 1 1 0 1 0 0 v w | mod 0 1 0 r/m | | |

## Instruction Set Summary (Continued)

| MNEMONIC AND DESCRIPTION | INSTRUCTION CODE | | | |
|---|---|---|---|---|
| | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 | 7 6 5 4 3 2 1 0 |
| **RCR =** Rotate Through Carry Right | 1 1 0 1 0 0 v w | mod 0 1 1 r/m | | |
| **AND = And:** | | | | |
| Reg./Memory and Register to Either | 0 0 1 0 0 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 1 0 0 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 1 0 0 1 0 w | data | data if w = 1 | |
| **TEST = And Function to Flags, No Result:** | | | | |
| Register/Memory and Register | 1 0 0 0 0 1 0 w | mod reg r/m | | |
| Immediate Data and Register/Memory | 1 1 1 1 0 1 1 w | mod 0 0 0 r/m | data | data if w = 1 |
| Immediate Data and Accumulator | 1 0 1 0 1 0 0 w | data | data if w = 1 | |
| **OR = Or:** | | | | |
| Register/Memory and Register to Either | 0 0 0 0 1 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 1 0 1 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 0 0 1 1 0 w | data | data if w = 1 | |
| **XOR = Exclusive or:** | | | | |
| Register/Memory and Register to Either | 0 0 1 1 0 0 d w | mod reg r/m | | |
| Immediate to Register/Memory | 1 0 0 0 0 0 0 w | mod 1 1 0 r/m | data | data if w = 1 |
| Immediate to Accumulator | 0 0 1 1 0 1 0 w | data | data if w = 1 | |
| **STRING MANIPULATION** | | | | |
| **REP =** Repeat | 1 1 1 1 0 0 1 z | | | |
| **MOVS =** Move Byte/Word | 1 0 1 0 0 1 0 w | | | |
| **CMPS =** Compare Byte/Word | 1 0 1 0 0 1 1 w | | | |
| **SCAS =** Scan Byte/Word | 1 0 1 0 1 1 1 w | | | |
| **LODS =** Load Byte/Word to AL/AX | 1 0 1 0 1 1 0 w | | | |
| **STOS =** Stor Byte/Word from AL/A | 1 0 1 0 1 0 1 w | | | |
| **CONTROL TRANSFER** | | | | |
| **CALL = Call:** | | | | |
| Direct Within Segment | 1 1 1 0 1 0 0 0 | disp-low | disp-high | |
| Indirect Within Segment | 1 1 1 1 1 1 1 1 | mod 0 1 0 r/m | | |
| Direct Intersegment | 1 0 0 1 1 0 1 0 | offset-low | offset-high | |
| | | seg-low | seg-high | |
| Indirect Intersegment | 1 1 1 1 1 1 1 1 | mod 0 1 1 r/m | | |
| **JMP = Unconditional Jump:** | | | | |
| Direct Within Segment | 1 1 1 0 1 0 0 1 | disp-low | disp-high | |
| Direct Within Segment-Short | 1 1 1 0 1 0 1 1 | disp | | |
| Indirect Within Segment | 1 1 1 1 1 1 1 1 | mod 1 0 0 r/m | | |
| Direct Intersegment | 1 1 1 0 1 0 1 0 | offset-low | offset-high | |
| | | seg-low | seg-high | |
| Indirect Intersegment | 1 1 1 1 1 1 1 1 | mod 1 0 1 r/m | | |
| **RET = Return from CALL:** | | | | |
| Within Segment | 1 1 0 0 0 0 1 1 | | | |
| Within Seg Adding Immed to SP | 1 1 0 0 0 0 1 0 | data-low | data-high | |

## Instruction Set Summary (Continued)

| MNEMONIC AND DESCRIPTION | INSTRUCTION CODE | | | |
|---|---|---|---|---|
| | 76543210 | 76543210 | 76543210 | 76543210 |
| Intersegment | 11001011 | | | |
| Intersegment Adding Immediate to SP | 11001010 | data-low | data-high | |
| **JE/JZ =** Jump on Equal/Zero | 01110100 | disp | | |
| **JL/JNGE =** Jump on Less/Not Greater or Equal | 01111100 | disp | | |
| **JLE/JNG =** Jump on Less or Equal/ Not Greater | 01111110 | disp | | |
| **JB/JNAE =** Jump on Below/Not Above or Equal | 01110010 | disp | | |
| **JBE/JNA =** Jump on Below or Equal/Not Above | 01110110 | disp | | |
| **JP/JPE =** Jump on Parity/Parity Even | 01111010 | disp | | |
| **JO =** Jump on Overflow | 01110000 | disp | | |
| **JS =** Jump on Sign | 01111000 | disp | | |
| **JNE/JNZ =** Jump on Not Equal/Not Zero | 01110101 | disp | | |
| **JNL/JGE =** Jump on Not Less/Greater or Equal | 01111101 | disp | | |
| **JNLE/JG =** Jump on Not Less or Equal/Greater | 01111111 | disp | | |
| **JNB/JAE =** Jump on Not Below/Above or Equal | 01110011 | disp | | |
| **JNBE/JA =** Jump on Not Below or Equal/Above | 01110111 | disp | | |
| **JNP/JPO =** Jump on Not Par/Par Odd | 01111011 | disp | | |
| **JNO =** Jump on Not Overflow | 01110001 | disp | | |
| **JNS =** Jump on Not Sign | 01111001 | disp | | |
| **LOOP =** Loop CX Times | 11100010 | disp | | |
| **LOOPZ/LOOPE =** Loop While Zero/Equal | 11100001 | disp | | |
| **LOOPNZ/LOOPNE =** Loop While Not Zero/Equal | 11100000 | disp | | |
| **JCXZ =** Jump on CX Zero | 11100011 | disp | | |
| **INT =** Interrupt | | | | |
| Type Specified | 11001101 | type | | |
| Type 3 | 11001100 | | | |
| **INTO =** Interrupt on Overflow | 11001110 | | | |
| **IRET =** Interrupt Return | 11001111 | | | |
| **PROCESSOR CONTROL** | | | | |
| **CLC =** Clear Carry | 11111000 | | | |
| **CMC =** Complement Carry | 11110101 | | | |
| **STC =** Set Carry | 11111001 | | | |
| **CLD =** Clear Direction | 11111100 | | | |
| **STD =** Set Direction | 11111101 | | | |
| **CLI =** Clear Interrupt | 11111010 | | | |
| **ST =** Set Interrupt | 11111011 | | | |
| **HLT =** Halt | 11110100 | | | |
| **WAIT =** Wait | 10011011 | | | |
| **ESC =** Escape (to External Device) | 11011xxx | mod x x x r/m | | |
| **LOCK =** Bus Lock Prefix | 11110000 | | | |

## Instruction Set Summary (Continued)

| MNEMONIC AND DESCRIPTION | INSTRUCTION CODE | | | |
|---|---|---|---|---|
| | 76543210 | 76543210 | 76543210 | 76543210 |

NOTES:
AL = 8-bit accumulator
AX = 16-bit accumulator
CX = Count register
DS= Data segment
ES = Extra segment
Above/below refers to unsigned value.
Greater = more positive;
Less = less positive (more negative) signed values
if d = 1 then “to” reg; if d = 0 then “from” reg
if w = 1 then word instruction; if w = 0 then byte instruction
if mod = 11 then r/m is treated as a REG field
if mod = 00 then DISP = O†, disp-low and disp-high are absent
if mod = 01 then DISP = disp-low sign-extended 16-bits, disp-high is absent
if mod = 10 then DISP = disp-high:disp-low
if r/m = 000 then EA = (BX) + (SI) + DISP
if r/m = 001 then EA = (BX) + (DI) + DISP
if r/m = 010 then EA = (BP) + (SI) + DISP
if r/m = 011 then EA = (BP) + (DI) + DISP
if r/m = 100 then EA = (SI) + DISP
if r/m = 101 then EA = (DI) + DISP
if r/m = 110 then EA = (BP) + DISP †
if r/m = 111 then EA = (BX) + DISP
DISP follows 2nd byte of instruction (before data if required)
† except if mod = 00 and r/m = 110 then EA = disp-high: disp-low.
†† MOV CS, REG/MEMORY not allowed.

if s:w = 01 then 16-bits of immediate data form the operand.
if s:w. = 11 then an immediate data byte is sign extended to form the 16-bit operand.
if v = 0 then “count” = 1; if v = 1 then “count” in ($C_L$)
x = don't care
z is used for string primitives for comparison with ZF FLAG.

SEGMENT OVERRIDE PREFIX

| 001 reg 11 0 |
|---|

REG is assigned according to the following table:

| 16-BIT (w = 1) | 8-BIT (w = 0) | SEGMENT |
|---|---|---|
| 000 AX | 000 AL | 00 ES |
| 001 CX | 001 CL | 01 CS |
| 010 DX | 010 DL | 10 SS |
| 011 BX | 011 BL | 11 DS |
| 100 SP | 100 AH | 00 ES |
| 101 BP | 101 CH | 00 ES |
| 110 SI | 110 DH | 00 ES |
| 111 DI | 111 BH | 00 ES |

Instructions which reference the flag register file as a 16-bit object use the symbol FLAGS to represent the file:

FLAGS =

X:X:X:X:(OF):(DF):(IF):(TF):(SF):(ZF):X:(AF):X:(PF):X:(CF)

The Micro TX is a Type Approved AM radio transmitter module operating at UHF frequencies. It is compatible with both low cost super-regenerative and AM superhet receivers. By providing excellent RF performance in a Type Approved module, the Micro TX minimises design costs and delays. The sub-miniature two-pin package ensures that the module can be fitted into any convenient space on the user's board. This makes it ideally suited to keyfob designs, where space is often limited due to the demand for ever more compact designs.

The unique design (Patent Pending) of this module allows operation on any supply voltage between 2.5 and 13V, simply by changing one external resistor. Users requiring high performance from a compact transmitter will appreciate the efficient operation of the module when driving a tuned loop or short whip antenna. Up to -6 dBm radiated power can be achieved with a 90 mm whip, just over half the length of the usual 1/4 wave antenna. It is compatible with most encoding ICs operating from 3V to 12V.

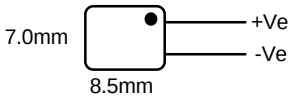
**Features**

Type Approved to MPT 1340
Ultra-compact two pin package
Wide supply voltage 2.5 to 13 V

**Applications**

Low cost key-fob designs
Car alarm “blippers”
Garage door openers
Lighting controls

## Package Dimensions




## Technical Specification

Ambient Temperature 20 ° C

| Parameter | Min | Typical | Max | Units |
|---|---|---|---|---|
| | | | | |
| Frequency (UK) | 417.925 | 418.000 | 418.075 | MHz |
| Frequency (Europe) | 433.845 | 433.920 | 433.995 | MHz |
| Module voltage | 2.2 | | 3.0 | Volts |
| Supply voltage (RD = 100 ohms ) | 2.5 | | 3.5 | Volts |
| Supply voltage (RD = 2200 ohms) | 8.8 | | 13.1 | Volts |
| Input current (mark ) | 3.0 | | 4.6 | mA |
| Input current (space ) | | 0 mA | | |
| Effective Radiated Power (ERP) | | -6dBm | | |
| Maximum baud rate | | | 1200 | bps |
| Range (with suitable receiver ) | | 100 | | Metres |
| Dimensions | | 8.5 x 7.0 x 4.2 +/- 10% | | mm |
| Pin Pitch | | 5.08 | | mm |
| Operating temperature | -10 | | +40 | deg. C |
| Storage Temperature | -40 | | +85 | deg. C |
| Patent Pending | | | | |

## Antenna

Fig. 1 shows a typical circuit for a design using a printed circuit board (PCB) loop antenna. A 1 to 5pF ceramic trimmer is used to tune the loop for maximum output. Fig. 2 shows a typical board layout for a key fob style transmitter. The dimensions of the loop are not critical, but excessively small or large loops should be avoided, as these will affect antenna tuning and efficiency. The loop should enclose an area of about 700mm$^2$ using a PCB track width of 1.5-2.5mm. Alternatively 1-1.5mm tinned copper wire may be used to form the loop.

The module can also drive a short whip antenna by using the matching network as shown in the circuit diagrams. The whip can be a wire rod or PCB track of about 90mm length. The inductor can be a 15nH surface mount inductor or an air cored solenoid of 0.8mm ID, close wound with 9 turns of 0.56mm enamel covered wire. A trimming capacitor of 1.5-5pF will generally be adequate.

When using a printed antenna on the PCB, always specify a good quality fibreglass base material. Lower cost materials such as SRBP (paper) will cause excessive losses at UHF. For printed circuit board aerials the variable trimmer capacitor can be replaced with two fixed value capacitors in series allowing intermediate values of capacitance to be obtained. The aerial should then be tuned for resonance to determine the values of the capacitors. However the variable capacitor is preferable if maximum output power and therefore range is required for all production transmitters.

## Circuit Board Layout and Decoupling

In order to achieve satisfactory RF performance, good PCB layout practice should be observed. The loop antenna should be free of any components or tracks except for the module and the tuning capacitor. All aerials radiate more efficiently against ground planes and the PCB should therefore be “flooded” with copper in the areas not being used for the aerial. Double sided PCBs can provide extra area for ground planes and the top and bottom ground plane layers should be generously connected with vias. Always use ceramic capacitors to decouple the supply at RF.

CMOS ICs can be susceptible to local RF fields and the use of the above techniques minimises this possibility. Do not use stripboard for prototypes as results may be misleading.

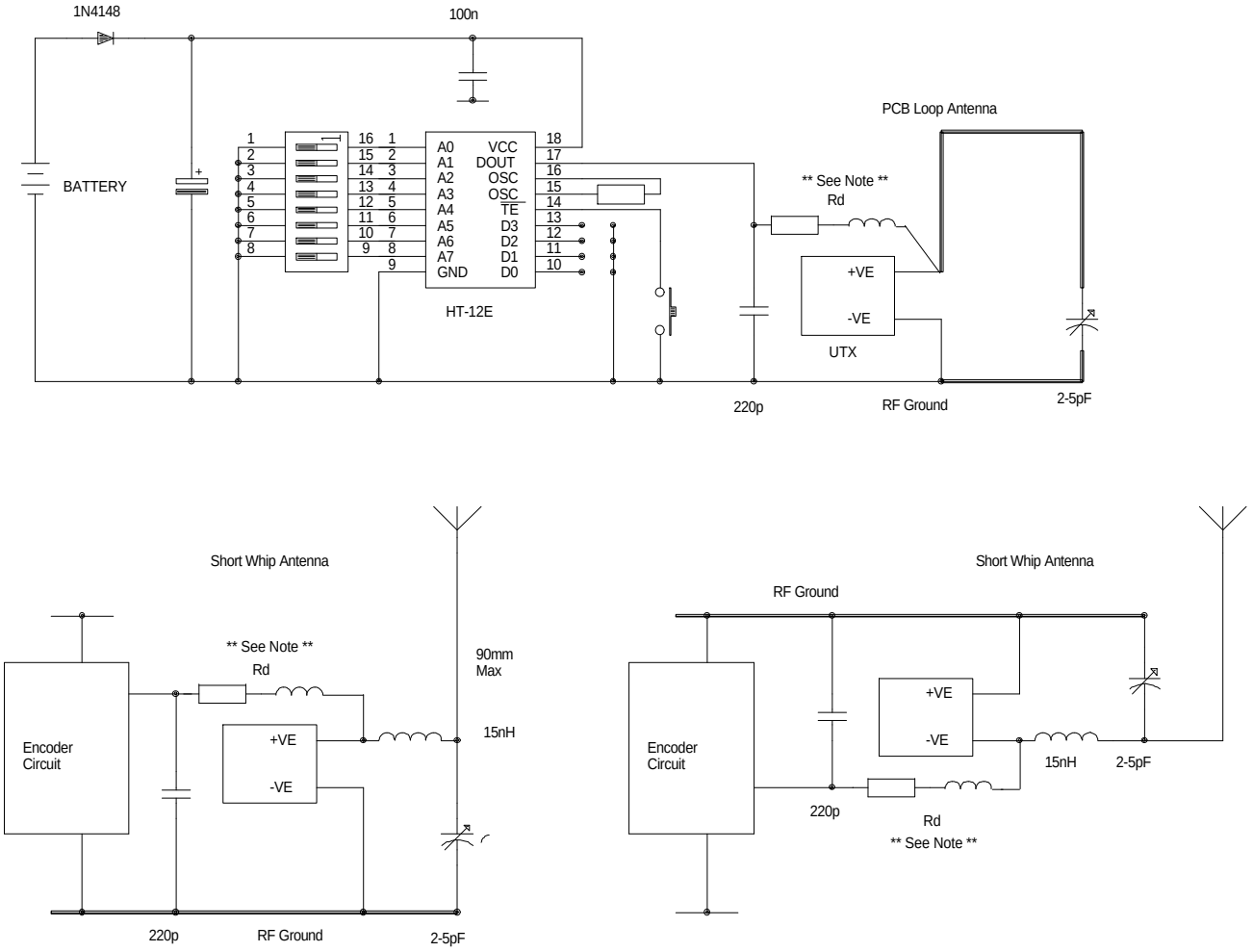
## Power Supply

The Micro Tx can be matched to any power supply voltage by varying the resistor in series with the module. The module typically requires an operating current of 4mA and drops 2.5V across it. The required series resistor can thus be easily calculated by Ohms Law, alternatively the following table can be used. As it is a two pin device it can be driven from either a current source or a current sink. However, ensure that the encoding device can source or sink sufficient current. and be aware that if “sinking” current the transmitter is “on” when the output is low, thus inverting the data if using standard encoders.

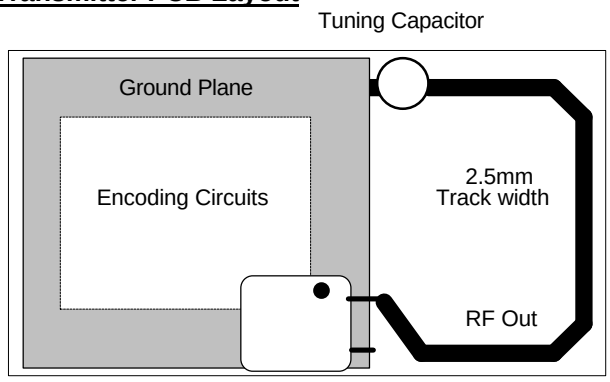
| Supply Voltage | | Resistor Value |
|---|---|---|
| **Min** | **Max** | |
| -1.0 | 0.5 | Off |
| 2.5 | 3.5 | 100 |
| 2.8 | 4.0 | 220 |
| 3.2 | 4.5 | 330 |
| 3.6 | 5.2 | 470 |
| 4.2 | 6.1 | 680 |
| 5.2 | 7.6 | 1k |
| 6.7 | 9.9 | 1k5 |
| 8.8 | 13.1 | 2k2 |

For operation at low voltages (<3V) a 100uH RF choke should be placed in series with Rd. This minimises RF energy being absorbed by low impedance drive circuitry.

## Fig.1 Application Circuits




## Fig.2 Typical Tuned Loop Transmitter PCB Layout




## Notes

The Transmitter module should ideally be placed over the ground plane as shown. The loop aerial should be completely clear of the ground plane and all other components. Do not place other components within the loop area.

For operation at low voltages (<3V) a 100uH RF choke should be placed in series with Rd. This minimises RF energy being absorbed by low impedance drive circuitry.

A variable tuning capacitor is preferable to one, or two fixed capacitors in series, if maximum output power is required for all production transmitters. This allows for various tolerances in the device.

**MPT 1340 Requirements (UK only)**

MPT 1340 is the appropriate Type Approval specification issued by the Radio Agency (DTI) and copies may be obtained from the RA's library service on 0171 211 0211.

The Type Approval number for this device is : **12321**

Users should be aware of the following requirement:

*"The equipment in which the module is used must carry an inspection mark located on the outside of the equipment and be clearly visible. The minimum dimensions of the inspection mark shall be 10 x 15 mm and the letter and figure height must be no less than 2mm. The wording shall read " MPT 1340 W.T. LICENCE EXEMPT ".*

**Example Label:**

**MPT 1340**
**W.T. Licence Exempt**
**Type Approval No:**
**12321**

MPT1340 also states that: *"All transmitters shall use integral antennas only. In this specification an integral antenna is defined as one which is designed to be connected **permanently** to the transmitter or receiver without the use of an external feeder. Receivers may use an external antenna or an integral antenna.*

## Product Order Codes

| Part No | Description |
|---|---|
| | |
| LQ-TX418A-S | Transmitter Module 418MHz |
| LQ-TX433A-S | Transmitter Module 433MHz |

## Document History

| Issue | Date | Revision |
|---|---|---|
| | | |
| 1.0 | Nov 96 | Preliminary |
| 1.1 | Jan 97 | Minor corrections |
| 1.2 | Mar 97 | Package dimensions modified |
| 1.3 | Aug 97 | Application circuits clarified |
| 1.4 | Apr 98 | Change of address, font size increased. |
| 1.5 | Apr 98 | Series choke & notes added |

## Copyright



## Disclaimer

The Quantelec Group Ltd has an on going policy to improve the performance and reliability of their products, we therefore reserve the right to make changes without notice. The information contained in this data sheet is believed to be accurate however we do not assume any responsibility for errors nor any liability arising from the application or use of any product or circuit described herein. This data sheet neither states nor implies warranty of any kind, including fitness for any particular application.

For further information or technical assistance please contact:

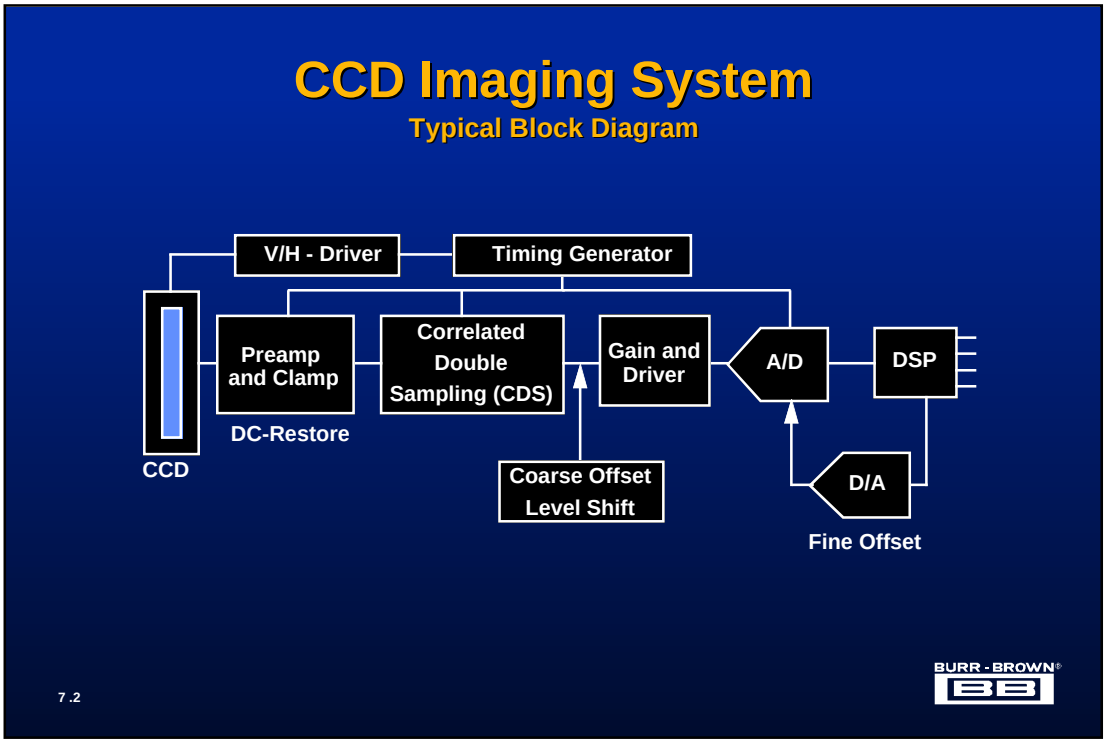
**LOW POWER RADIO SOLUTIONS**
**A Division of The Quantelec Group Ltd.**
Two Rivers Industrial Estate
Station Lane
Witney
Oxon. OX8 6BH
England
.

Tel: +44 (0)1993 709418
Fax: +44 (0)1993 708575
Web: http://www.lprs.co.uk
Email: info@lprs.co.uk

# CCD Imaging Systems



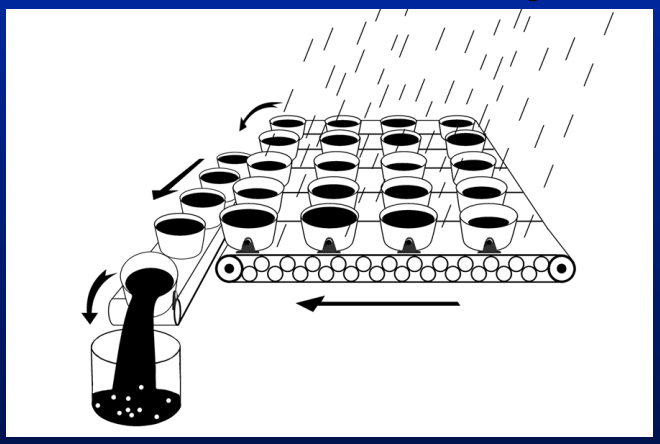

Contributing Author: Stephan Baier




Technology advances with CCDs (Charge Coupled Device), like increased resolution at lower manufacturing costs, have fueled the growth in the electronic imaging industry. However, some of the typical constraints of CCDs remain unchanged, such as the very low output signal level and the inherent noise sources. Furthermore, increased resolution generally equals higher read-out speed, which in turn dictates the requirements for the subsequent electronics.

The CCD is the central element in an imaging system. Designers need to be aware of the special requirements for the signal conditioning of the CCD in order to achieve the maximum performance. The output signal of the CCD is a constant stream of the individual pixel “charges” and this results in the typical form of stepped DC voltage levels. This output signal also contains a DC-bias voltage, which is in the order of several volts. The signal is then passed through a capacitor to block the DC voltage before going into the preamplifier. To maintain the necessary relationship between the pixel information and the baseline, a clamp or DC-restore circuit is usually situated in the first processing stage. The next stage is used as a noise reduction circuit specific to CCD based systems: the correlated double sampler (CDS). Following is another gain stage, which could be a automatic gain control amplifier (AGC), or a fixed gain stage with offset adjustment. Before going into the A/D converter it usually passes through a dedicated buffer or driver circuit optimized for the selected converter type. Further baseline stabilization can be achieved by having a D/A converter in a digital control loop. In the following discussion the CCD itself is looked at and design techniques are explored.

## Basic CCD Theory

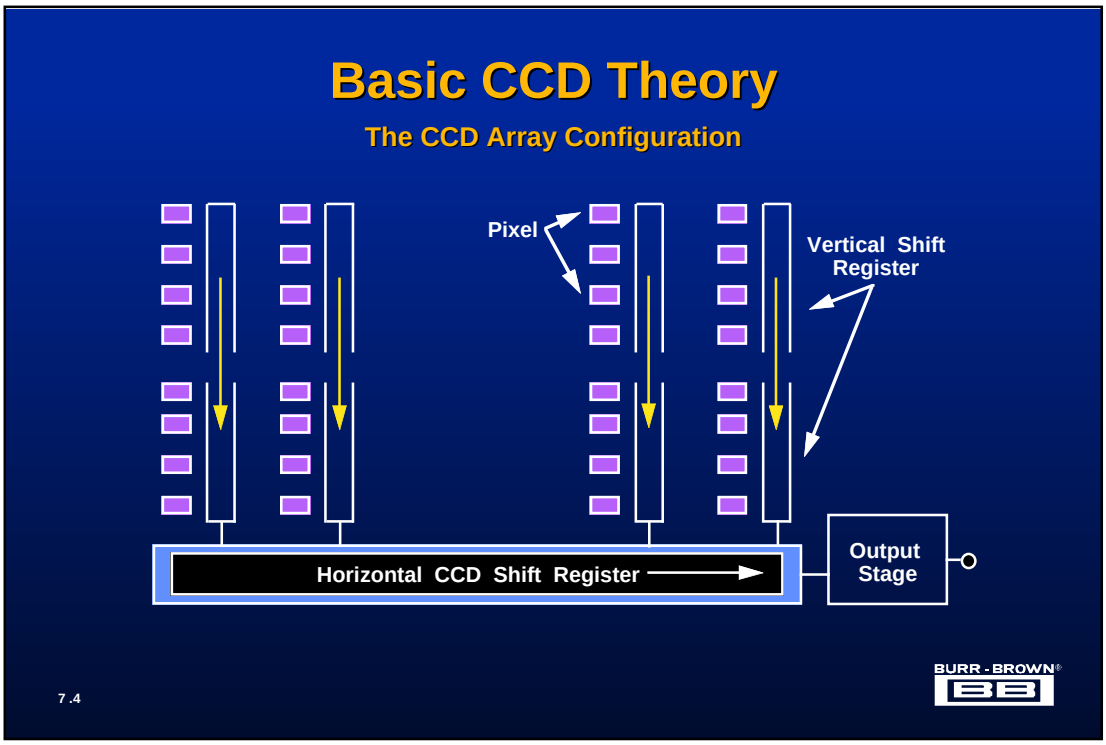


- Raindrops = Photons
- Buckets = Pixel
- Conveyor Belts = CCD Shift Register
- Metering Glass = Sense Capacitor

7.3

BURR-BROWN®

In its principle, the operation of a CCD array is quiet simple. A common analogy[1] is shown here, using an array of buckets on conveyor belts. During a rain shower the raindrops will fill the lined up buckets more or less. Then the conveyor belts transport the buckets to the front belt and dump their content into another row of buckets. As they move forward the rainwater is spilled into the metering glass. The scale on the metering glass indicates how much water was collected in the individual bucket. When relating this model to a real CCD element, the “raindrops” are the light (photons) falling onto the CCD surface, the buckets are the many pixels of a CCD array and the “conveyor belts” are the shift registers that transport the pixel charge to the output stage. This output stage is mainly the sense capacitor, here the “metering glass”, and an output source follower is used to buffer this sense capacitor.



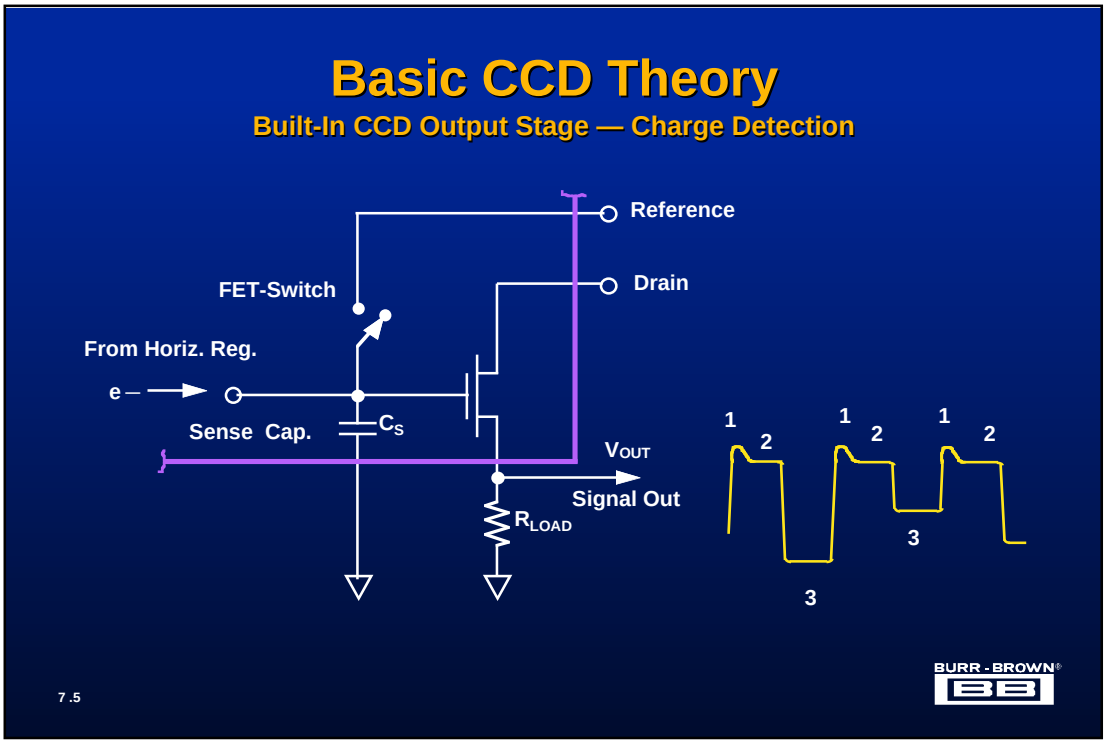

The CCD array is configured into multiple vertical shift registers and usually one horizontal shift register, both requiring different clock patterns. The flow is as follows: the pixel converts the light (incoming photons) into electrons which are stored as electrical charge. Then the charge is transferred down the vertical register in a conveyor-belt fashion to the horizontal shift register. This register collects one line at a time and transports the pixel charges in a serial manner to the on-chip output stage. The on-chip output converts the charge into a voltage. This voltage is then available at the output in the typical CCD pulse form.

With the standard CCD, most of the pixels can detect the light. The CCD also has small sections at the beginning and at the end of each vertical segment that are covered and therefore “optically black”. Those pixels will always have the voltage level representing black. Some image circuits use those as reference pixels to adjust the signal offset.

Some numbers:

The horizontal read-out speed for systems with up to 12-bit resolution is up to 10MHz. For higher resolutions (≤16-bit) the clock speed is around 1MHz.

Typical pixel dimensions are: ~27µm$^2$ for a 512x512 array or ~12µm$^2$ for a 1024x1024 array.

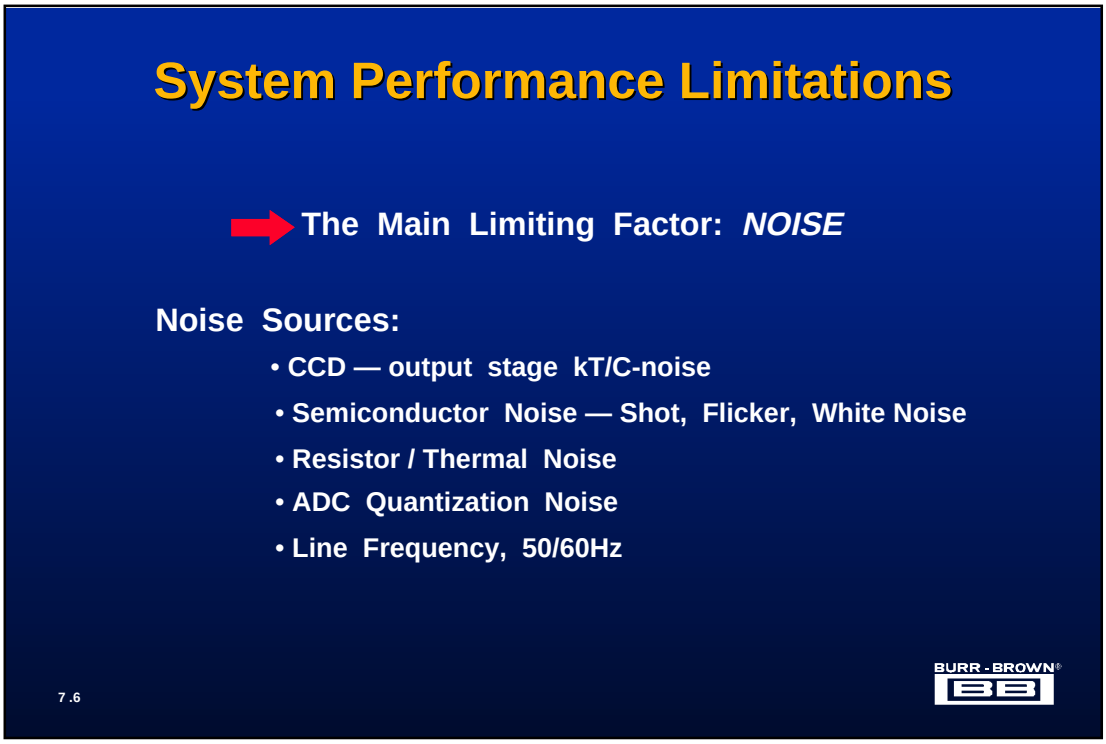



Shown here is the conceptual schematic of the output stage inside the CCD element. This stage is responsible for the so called 'charge detection'. As discussed earlier, the charge $e^-$ generated is moved into the horizontal shift register. The charge of each individual pixel is controlled by the horizontal clock and stored onto the Sense Capacitor ($C_S$). A typical value for such a capacitor is 0.1pF to 0.5pF. According to V = Q/C, the charge will develop a voltage across capacitor $C_S$, representing the light intensity for the particular pixel. A MOSFET transistor configured as a source follower buffers the capacitor from the output node, which connects to the load resistor, $R_{LOAD}$. At this point, the image (video) signal becomes available at $V_{OUT}$ for further signal processing.

As indicated in the figure above, the output voltage is a series of stepped DC voltages. One pixel period is composed of three different levels: (1) the "reset feedthrough", (2) the "reference level", (3) and the "pixel level".

A readout sequence begins with the reset. Where the FET-switch is closed, set the sense capacitor to the initial reference voltage. The reference voltage can be relatively high, up to +12V. The closing of the switch causes the reset feedthrough, a result of capacitive coupling through the MOSFET. After the decay of this feedthrough the capacitor will reflect the reference voltage level (2). Once the capacitor has been reset, the switch opens and pixel charge is transferred to the capacitor, altering its voltage.

An important specification for CCD elements is the sensitivity. This is a measure of the achievable output voltage per electron, $S_V = V_{OUT}/e^-$. With a 0.1pF capacitor, the output voltage would be -1.6µV per electron. Unfortunately, the source follower has a gain of less than 1 (~0.8).



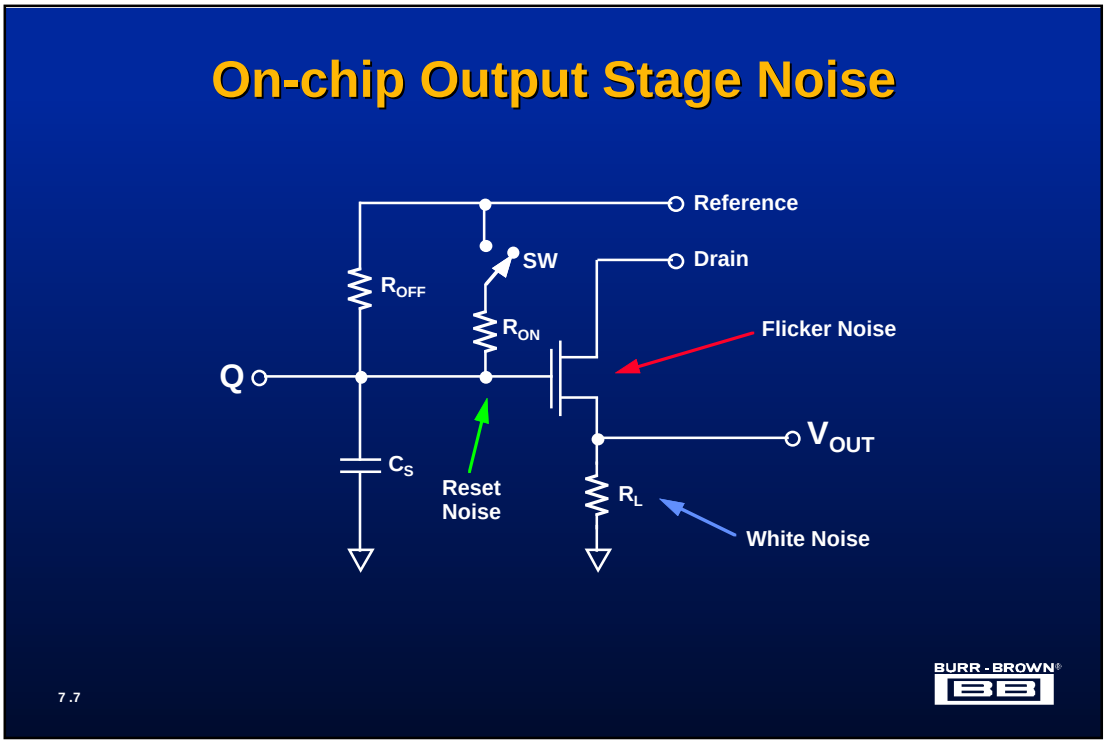

The lower limit of the dynamic range in an image system is set by the noise floor. Different techniques are available to maximize the dynamic range and optimize for the input range of the A/D converter, but a thorough understanding about the noise sources is crucial. The main noise source, besides digital feedthrough, is the so called kT/C-noise of the FET reset switch caused by its channel resistance.

The MOSFET also contributes noise — the flicker (1/f ) noise and some white (thermal) noise. Additionally, each resistor is a source of thermal/white noise.

Another limit is set by the quantization noise of the A/D converter. The rms quantization noise is expressed by the equation $q/\sqrt{12}$, with q being the bit size or LSB weight of the converter. For example, a 10-bit converter with a full-scale input range of 2V has a bit size of 2.0V/1024 = 1.953mV. Hence, the quantization noise is 564µVrms. Assuming a 0.1pF sense capacitor the detection limit would be at about 350 electrons due to the quantization noise.

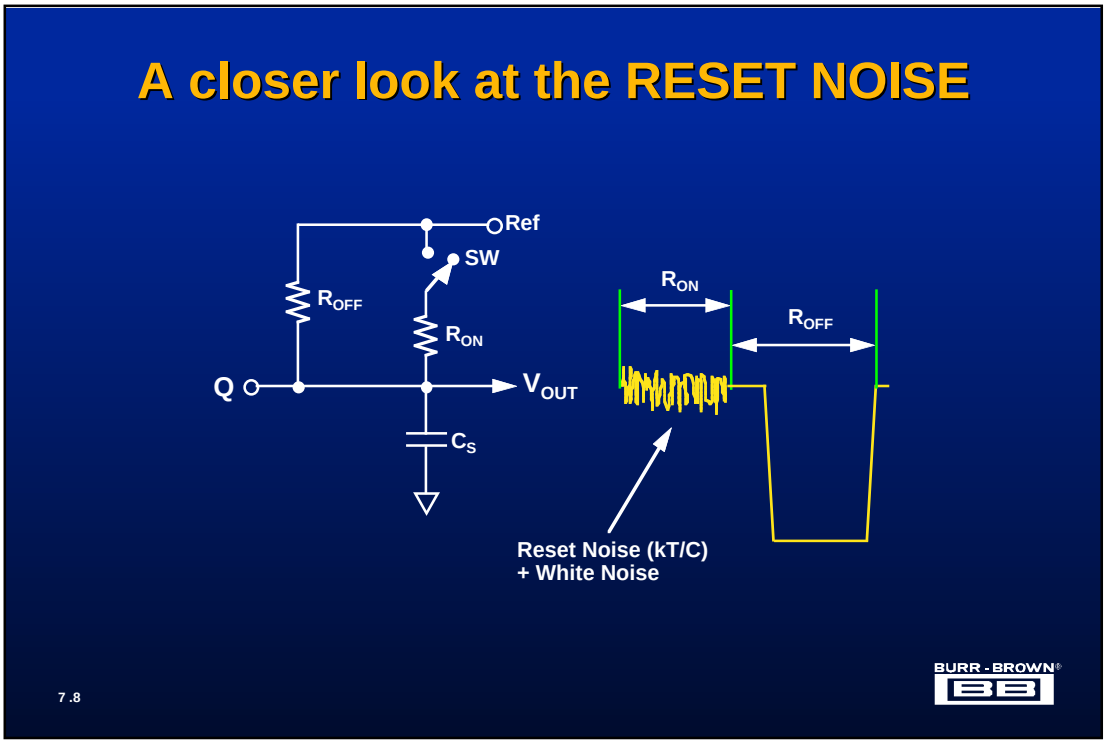
One obvious way to reduce this limitation is to use an A/D converter with a higher resolution, e.g., 12 bits.

Another example for a noise source would be the line frequency with 50Hz or 60Hz.




Looking again at the built-in output stage of the CCD, we can identify the different noise sources previously discussed.

- RESET NOISE: The thermal noise of the channel resistance ($R_{ON}$) of the FET switch (SW). This noise is often termed as kT/C-noise. With a typical value of 100 to 300 electrons (rms), this is the dominant limitation for the detection of small signals.
- FLICKER NOISE: Also 1/f-noise. Originates in the MOSFET, and relates to the presence of traps associated with contamination and crystal defects in the semiconductor. Its magnitude is therefore process dependent.
- WHITE NOISE also known as resistor noise: It is temperature dependent and equal to $\sqrt{(4kTRB)}$. White Noise has several origins. For example, the noise of the load resistor ($R_L$).




The main components of reset-noise come from the sense capacitor $C_S$ and the switch (SW), represented by its On-resistance, $R_{ON}$. Capacitors are usually thought of as noise-free devices. In the case of sampling systems, however, they exhibit a theoretical noise because the capacitor is periodically reset. The finite resistance of the reset switch has an associated thermal noise. This noise is transferred to the capacitor when the reset switch opens. As the resistor, $R_{ON}$, is made smaller, its noise will decrease, but the RC bandwidth will increase at the same time. The noise can be calculated as follows:

The thermal noise of a resistor is given by: $e_n = \sqrt{4kTRB}$ (Vrms).
Where:

k = Boltzmann's constant = 1.38054 E -23
T = absolute temperature in Kelvin, (298°K = +25°C)
R = on resistance of switch in Ω
B = Noise Bandwidth in Hz

Since the CCD model has a single-pole response, B is the -3dB bandwidth times π/2, which will be called NBW indicating the noise bandwidth for a single pole response. Therefore:

$NBW = 1/(2\pi R C) \bullet \pi/2 = 1/(4RC)$

$\Longrightarrow e_n = \sqrt{(4kTR)/(4RC)} = \sqrt{\mathbf{(kT/C)}}$

Using an example relevant to CCDs: $C_S$ = 0.1pF, $R_{ON}$ = 2kΩ

The Noise Bandwidth would be: NBW = 1.25GHz

The Reset Noise would be: $e_{nr}$ = 0.203mVrms

Taking the lowest detectable charge, one electron ($e^-$ = 1.6 E-19A-sec), the reset noise relates to 126$e^-$ of charge. With a maximum video signal amplitude of about 0.3V (187500$e^-$), the signal-to-noise ratio would be 1500 or 63dB. The dynamic range is about 11.5 bits.

# A Closer Look at the White Noise

**White Noise:** $e_{nw} = \sqrt{4kTBR_O}$

**With:** $R_O$ = output impedance of CCD output stage
typical $R_O$ = 200Ω to 20kΩ

**Example:** B = 1MHz, $R_O$ = 2000Ω

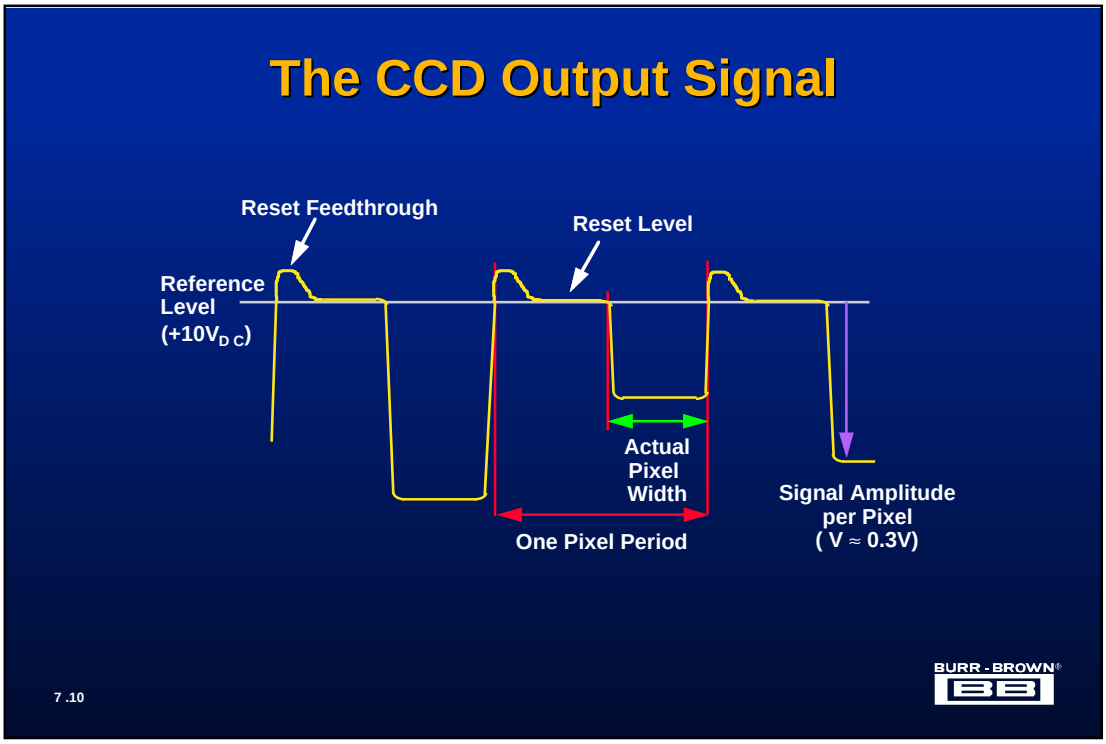
$e_{nw}$ = 5.75μVrms

7.9

BURR-BROWN®

White noise or thermal noise is present in any resistor or conductor. The rms-voltage noise is proportional to the square root of the temperature, T, the bandwidth, B, and the resistance, R, as given by the equation $e_{nw} = \sqrt{4kTRB}$.

The typical output resistance of a CCD, which includes the external load resistor, is in the range of 200Ω to 20kΩ. The output impedance of the CCD output stage ($R_O$) is formed by the load resistor ($R_L$) and the channel resistance of the output MOSFET.



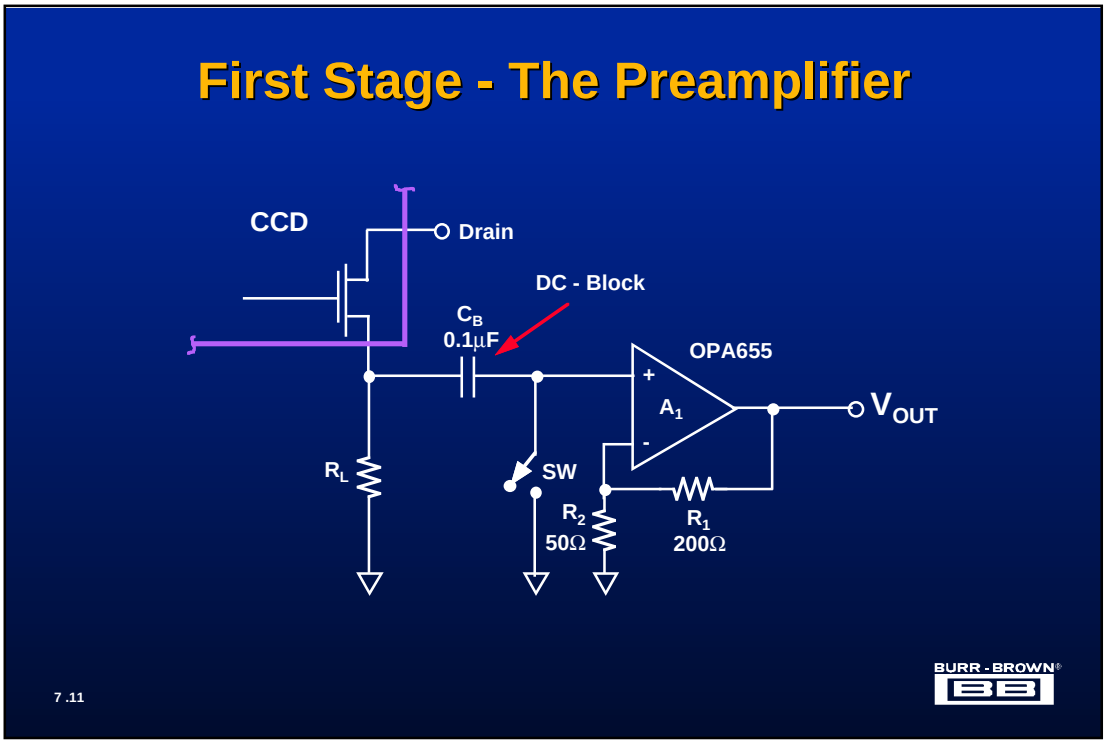

This is a typical output voltage waveform from a CCD element. The signal can be described by five characteristics: the Reset Feedthrough, the Reset Level, the Signal Amplitude, the Pixel Period and the actual Pixel Width.

As mentioned before, this CCD signal is not a continuous sinusoidal waveform, but rather is a sequence of stepped DC levels.

The sequence for one pixel is as follows:

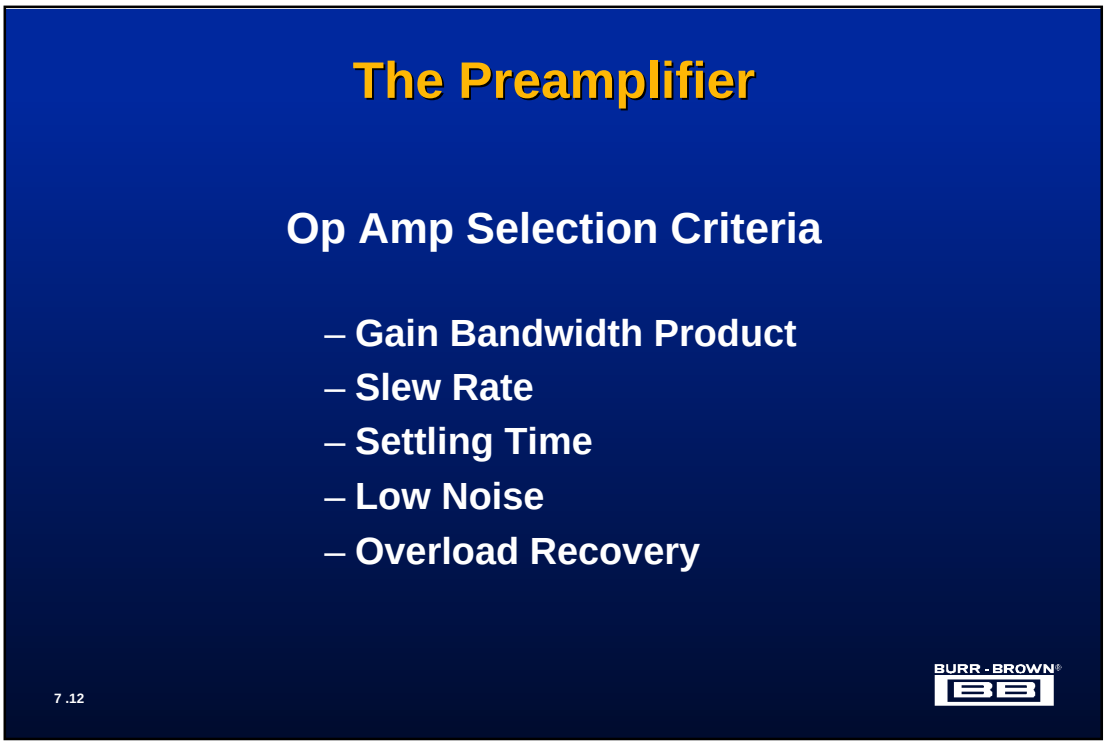
- Reset Feedthrough: This can be a relatively large pulse, as a result of capacitive coupling through the FET.
- Reset Level: The “Sense Capacitor” will be charged to this final reset voltage. This level can be in the order of +10V or more, creating the requirement for a DC-decoupling capacitor at the output of the CCD element.
- Pixel Level: After the reset period, the pixel is transferred. The amplitude corresponds to the charge representing the incident light level of the addressed pixel. Because of the electron charge ($e^-$) the CCD output signal is inherently unipolar (negative).

  Typical CCD pixel rates can vary between 1Mpixel/sec up to 20Mpixel/sec, depending on the application.



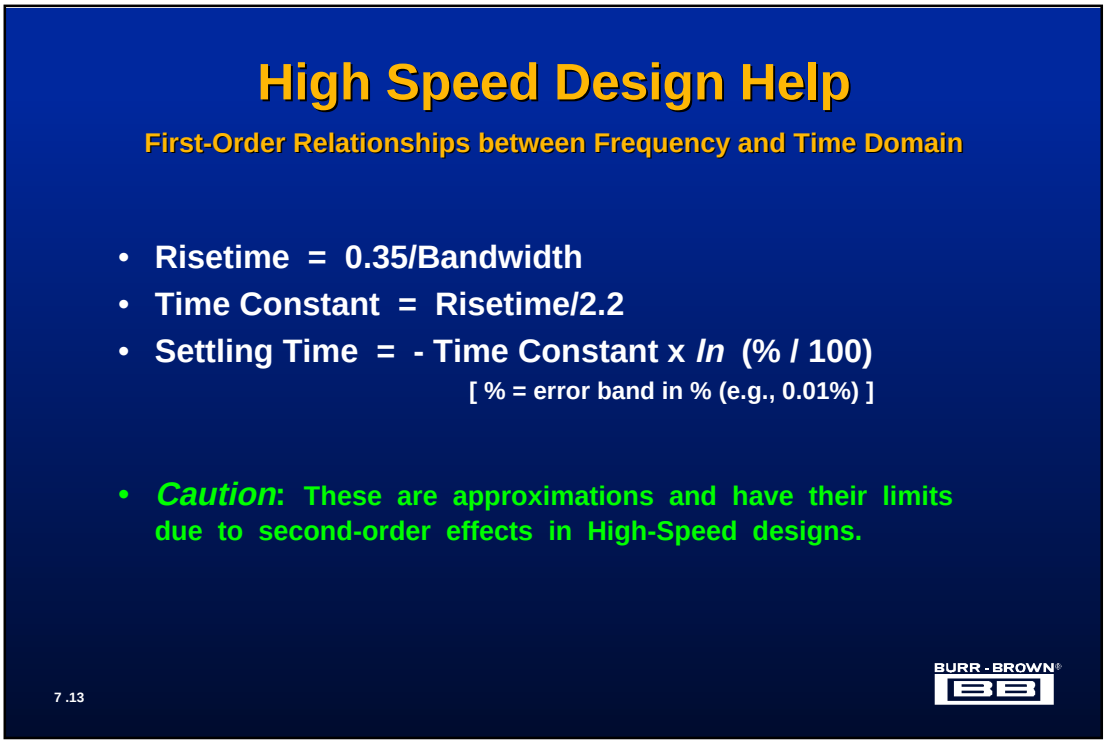

The CCD output signal is immediately gained up by the preamplifier, as shown in this circuit schematic. The amplifier itself uses the wideband FET input op amp OPA655, set in a gain of +5V/V. With a -3dB bandwidth of 400MHz for a gain of +1, the bandwidth of the OPA655, in a gain of +5V/V, is 75MHz. The specified 12-bit settling time for this part is about 16ns. To estimate the total response time, the slewing time for the 1Vp-p output needs to be added to the 16ns settling time. With a slewing time of 3.3ns, this adds up to a total of about 20ns. Considering a system with a 5MHz readout frequency, one pixel period takes 200ns. The actual pixel width will be approximately half of that time, or 100ns. The OPA655 will take only 1/5 of the pixel time and still be accurate to 12 bits. The fast response of the OPA655 leaves sufficient time for the subsequent stages and for the acquisition time of the A/D converter.

As discussed previously, the pixel information rides on the reference voltage, which can be +10V or more. This could cause unwanted common-mode effects or even saturation. The series capacitor, $C_B$, blocks this DC component from the video signal and the reference of the baseline is lost. A new baseline can be established with the switch, SW, to ground. For each reset period the switch closes and grounds one side of the capacitor, setting its charge to a defined potential, ground or 0V, in this case.




Because of the nature of the CCD output voltage, the performance requirements for the processing components, like the preamplifier, focus on the time domain specifications of the ICs.

For the operational amplifier specifications like slew-rate, settling time and overload, recovery time is important. Of course, the components should have low noise specifications and not add too much noise to the CCD signal, reducing the dynamic range.




When designing a CCD imaging system, time domain parameters of the ICs are the specifications that the designer will look at. A designer may not always be able to find all of them in the product data sheets and a good first-order approximation of the missing specification can be obtained using the relationships between the frequency and time domain shown above. For example, the rise time may be calculated by using the simple relationship between the -3dB bandwidth and the rise time of a first-order system: Risetime = 0.35/Bandwidth.

This rise time may then be used to obtain the system's time constant, which will lead to the settling time: Time Constant = Risetime/2.2.

Another familiar relationship for a single pole system is its pulse response: $V_{OUT}(t) = V_{IN}(1 - e^{-t/\tau})$. Here, $\tau$ is the time constant formed by RC. Solving this equation for t results in $t = -\tau \times \ln(\%/100)$. This provides the settling time for any desired settling accuracy, usually used as a % term.

Again, keep in mind that all these design equations are based on a single-pole response. The actual values may differ due to second-order effects of the ICs.

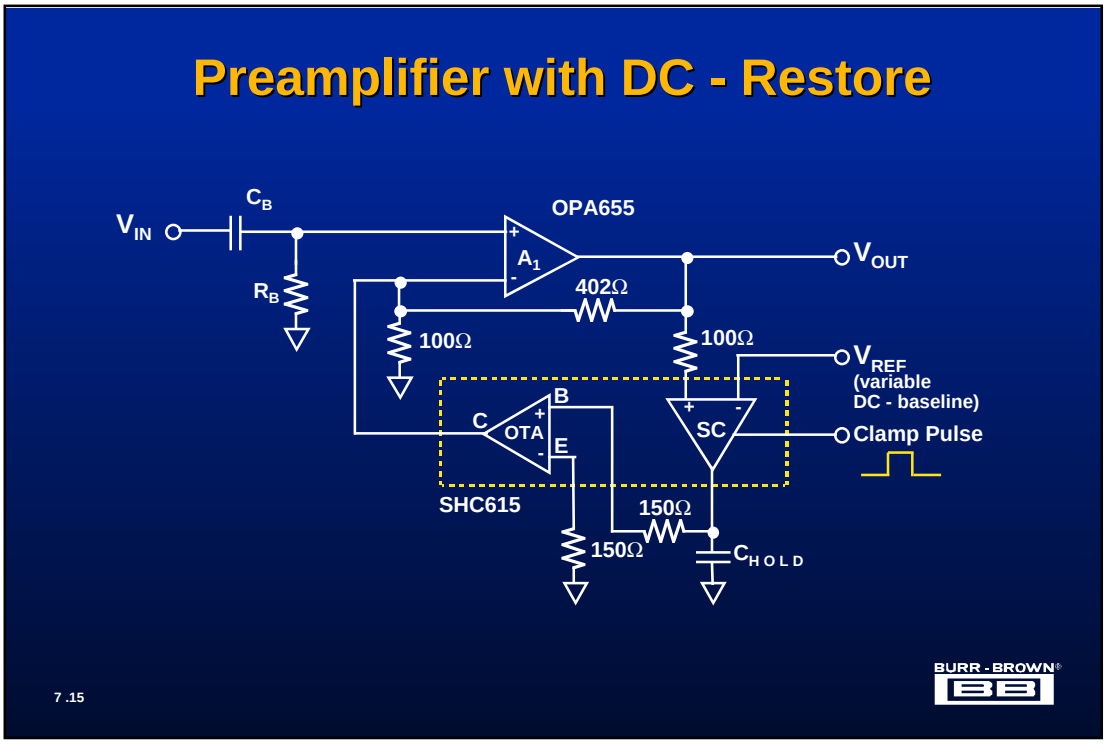
# High Speed FET Op Amps

| | BANDWIDTH | SLEW RATE | SETTLING TIME | NOISE |
|---|---|---|---|---|
| OPA655 | 400MHz | 290V/μs | 17ns | 6nV/√Hz |
| OPA637 | 80MHZ | 135V/μs | 450ns | 4nV/√Hz |
| OPA671 | 35MHz | 107V/μs | 240ns | 9nV/√Hz |
| OPA627 | 16MHz | 55V/μs | 550ns | 4nV/√Hz |

7 .14

BURR - BROWN®

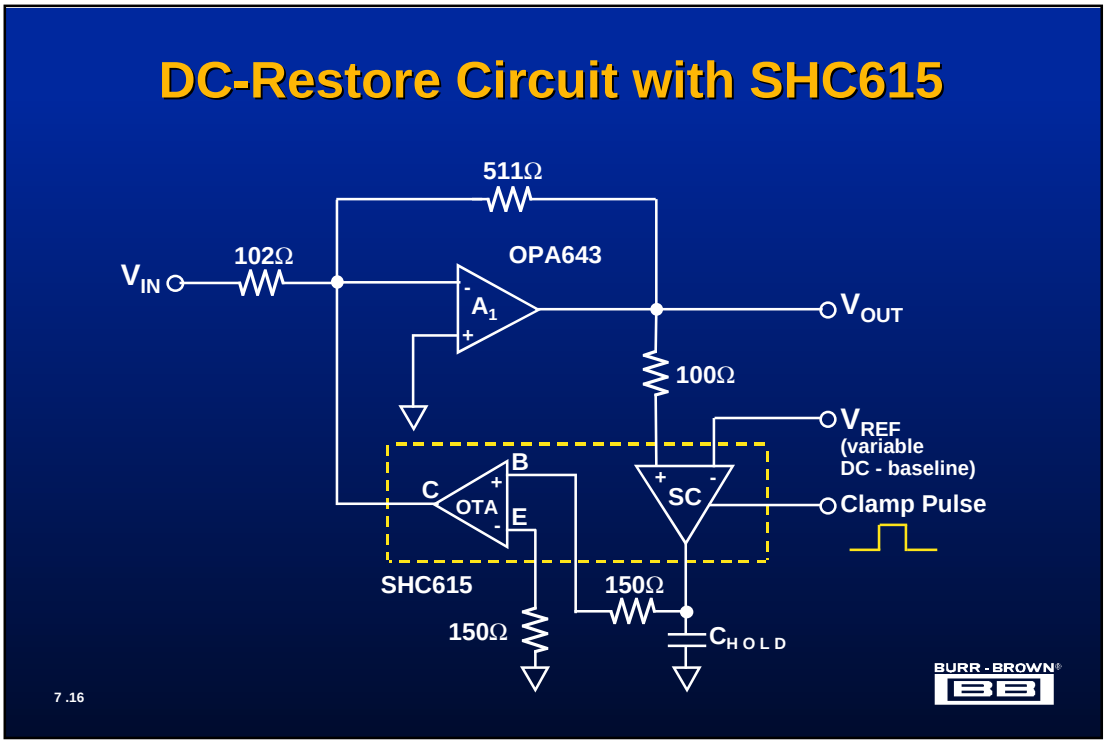
This table provides a selection of high speed op amps that feature a FET input. The advantage over bipolar input stages is the very low bias current. The bias current becomes important in applications where a capacitor is connected to the input, or where the source impedance is high. The typical input bias current of all three listed op amps is 5pA or better. Input bias currents on high speed bipolar op amps are in the range of 1 to 30μA. All of the listed op amps operate on ±5V supplies, except for the OPA671, which requires ±15V supplies. The OPA637 is a decompensated version of the OPA627 and therefore only stable in gains of 5 or higher. The decision whether to use a FET input or bipolar input device will always depend on the requirements of the individual application.



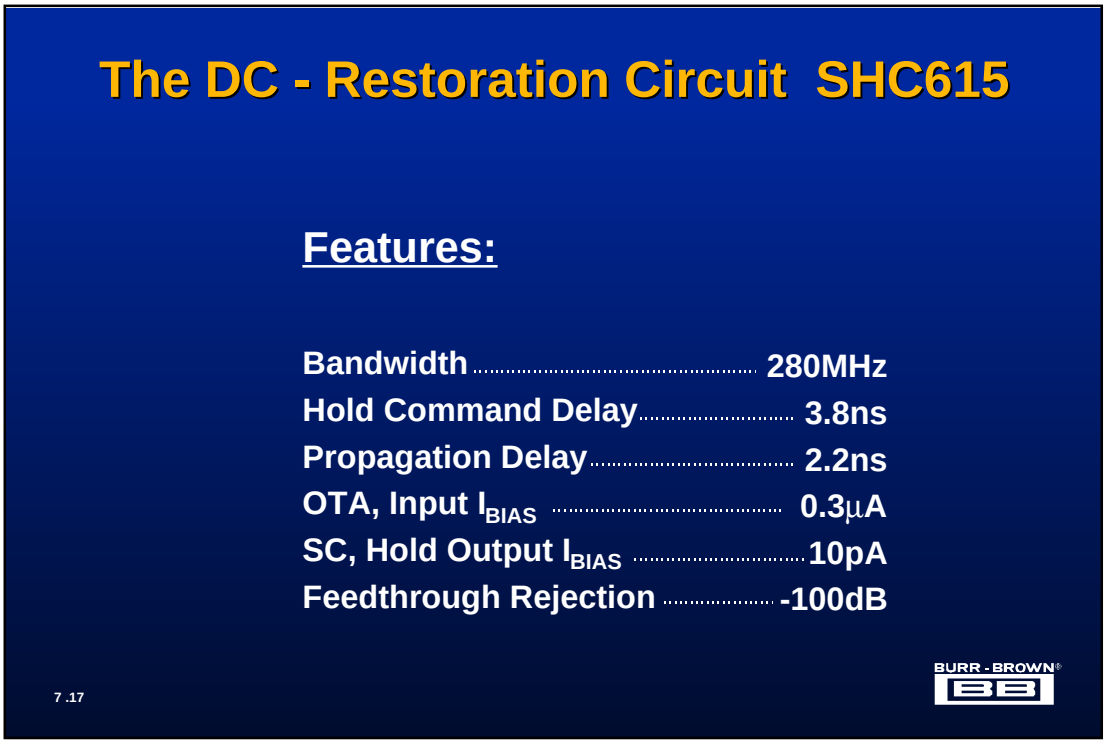

Another circuit example for the preamplifier and the clamp circuit is shown here. The preamplifier uses the wideband, low noise OPA655, again configured in a gain of +5V/V. Here, the OPA655 has a typical bandwidth of 57MHz with a 12-bit settling time of about 30ns (0.01%).

The video signal passes through the capacitor $C_B$, blocking the DC component. To restore the DC level to the desired baseline, the DC-restoration IC SHC615 is used. The SHC615 basically incorporates two OTAs, or voltage controlled current sources. One of the OTAs can be digitally controlled and resembles a switched difference amplifier. The inverting input is connected to a reference voltage. During the high time of the clamp pulse the switching comparator (SC) will compare the output of the op amp to the reference level. Any voltage difference between those pins will result in an output current that either charges or discharges the hold capacitor, $C_{HOLD}$. This charge creates a voltage across the capacitor, which is buffered by the OTA. Multiplied by the transconductance the voltage will cause a current flow in the collector, C, terminal of the OTA. This current will level shift the OPA655 up to the point where its output voltage is equal to the reference voltage. This also closes the control loop. Because of the buffer, the voltage across $C_{HOLD}$ stays constant and maintains the baseline correction during the off - time of the clamp pulse.

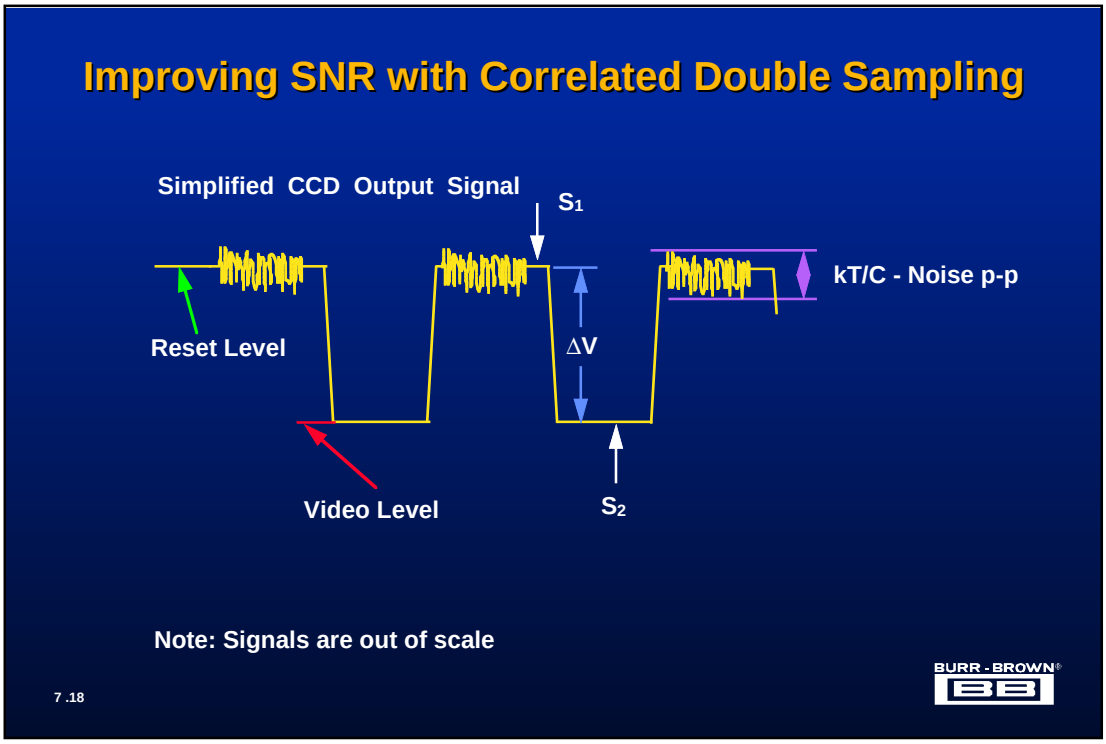
The external capacitor ($C_{HOLD}$) allows for a wide range of flexibility. By choosing small values, the circuit can be optimized for a short clamping period or with higher values for a low droop rate. Another advantage of this circuit is that small clamp peaks at the output of the switching comparator are integrated and do not cause problems in the signal path.




This circuit shows a slight alteration of the previous restoration circuit. Here, the FET op amp, OPA655, is replaced with the wideband, low noise amplifier, OPA643. The OPA643 is configured in an inverting gain of 5V/V. The typical bandwidth of the OPA643 is 250MHz, making the op amp ideal for CCD systems with high pixel rates. The common mode voltage problem encountered in the previous circuit is avoided with this configuration by selecting the inverting topology for the signal processing op amp. Depending on the individual application, the resistor values may need to be changed in order not to load the previous stage.



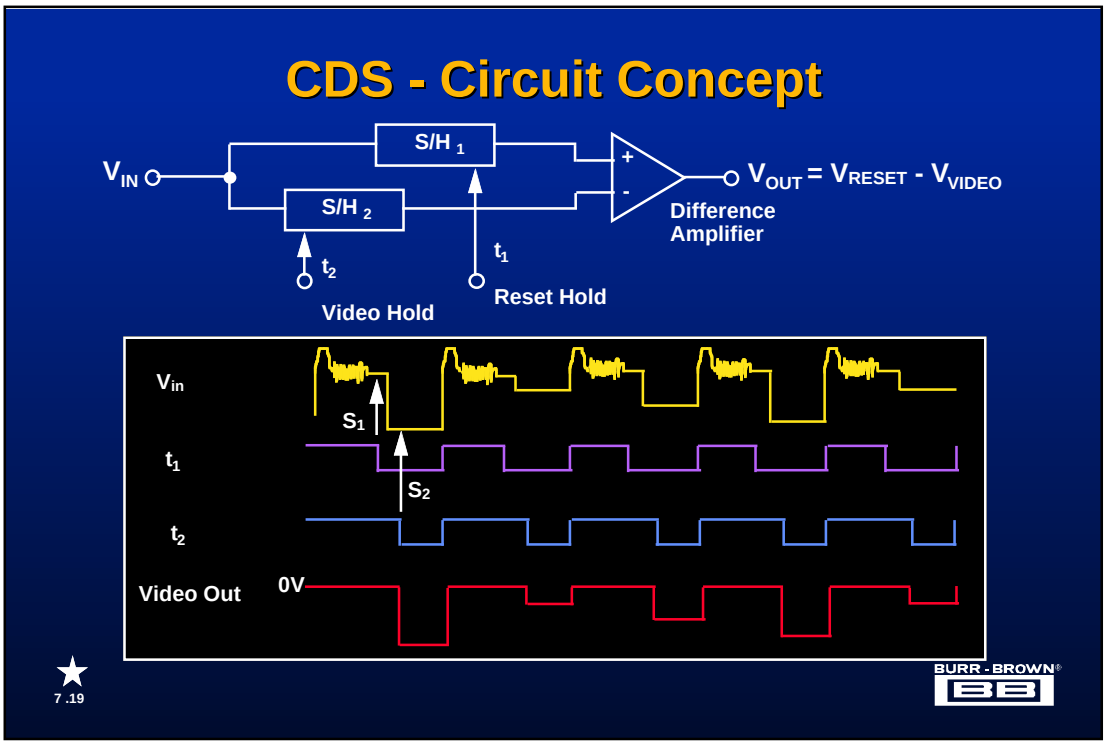

The SHC615 is a complete monolithic IC for very fast and precise DC-restoration, offset clamping, and low frequency hum suppression. This function can be realized with just the SHC615 alone or having the SHC615 alongside a video op amp for out-of-path restoration. The application list for the SHC615 also includes high speed sample and holds, high speed integrators, or peak detectors for fast pulses. The functions inside the SHC615 are a sampling comparator (SC) and an operational transconductance amplifier (OTA) for buffering the external hold capacitor. The transconductance of the OTA and the sampling comparator can be adjusted by an external resistor, allowing bandwidth, quiescent current, and gain tradeoffs to be optimized.




As shown before, the noise is the limiting factor for the resolution in a CCD system, where the kT/C noise is dominant. To reduce this noise, imaging systems use a circuit called a “Correlated Double Sampler” (CDS). The name comes from the double sampling technique of the CCD charge signal. The first sample ($S_1$) is taken at the end of the reset period. When the reset switch opens again, the effective noise bandwidth changes because of the large difference in the switch’s $R_{ON}$ and $R_{OFF}$ resistance. This causes the dominating kT/C noise essentially to “freeze” in its last point.

The other sample ($S_2$) is taken during the video portion of the signal. Ideally, the two samples differ only by a voltage corresponding to the transferred charge signal. This is the video level minus the noise ($\Delta V$).

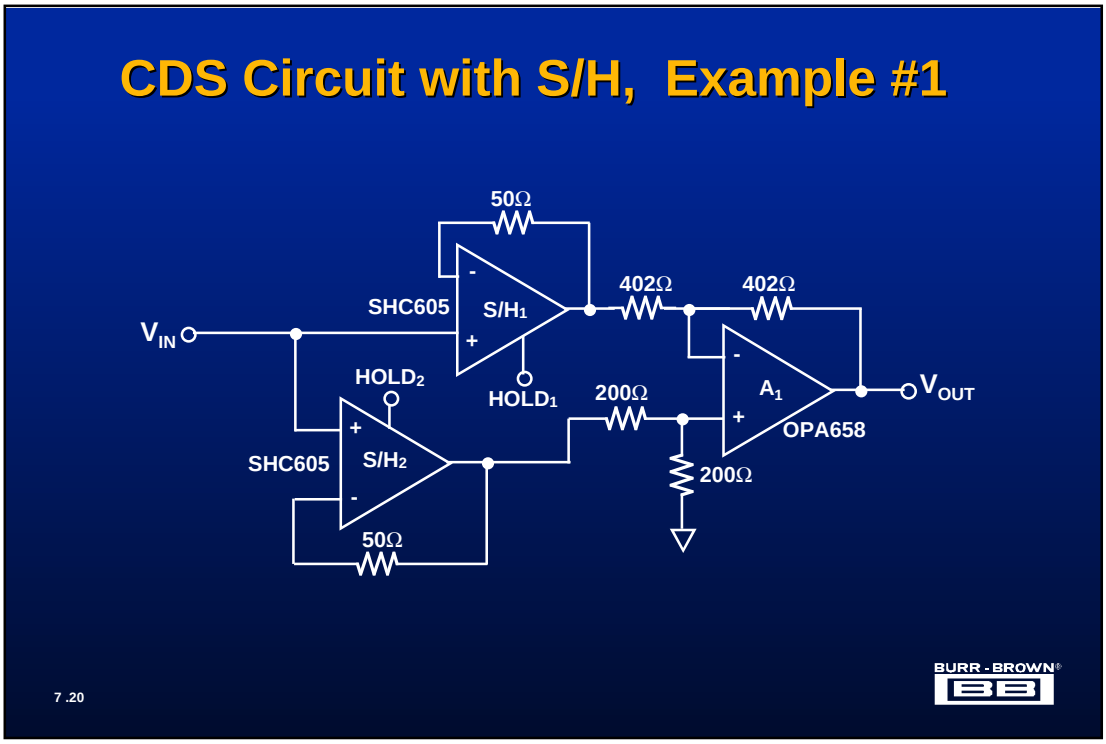
The CDS function will eliminate the kT/C noise as well as much of the 1/f and white noise.




Here is a block diagram of a CDS circuit. Two sample and hold amplifiers and one difference amplifier constitute the correlated double sampler.

The signal coming from the CCD is applied to the two sample and hold, with their outputs connected to the difference amplifier. The timing diagram will clarify the operation. At time $t_1$, the sample & hold ($S/H_1$) goes into the hold mode, taking a sample of the reset level including the noise. This voltage ($V_{RESET}$) is applied to the non-inverting input of the difference amplifier. At time $t_2$, the sample and hold ($S/H_2$) will take a sample of the video level, which is $V_{RESET} - V_{VIDEO}$ . The output voltage of the difference amplifier is defined by the equation $V_{OUT} = V_{IN+} - V_{IN-}$. The sample of the reset voltage contains the kT/C noise, which is eliminated by the subtraction of the difference amplifier.

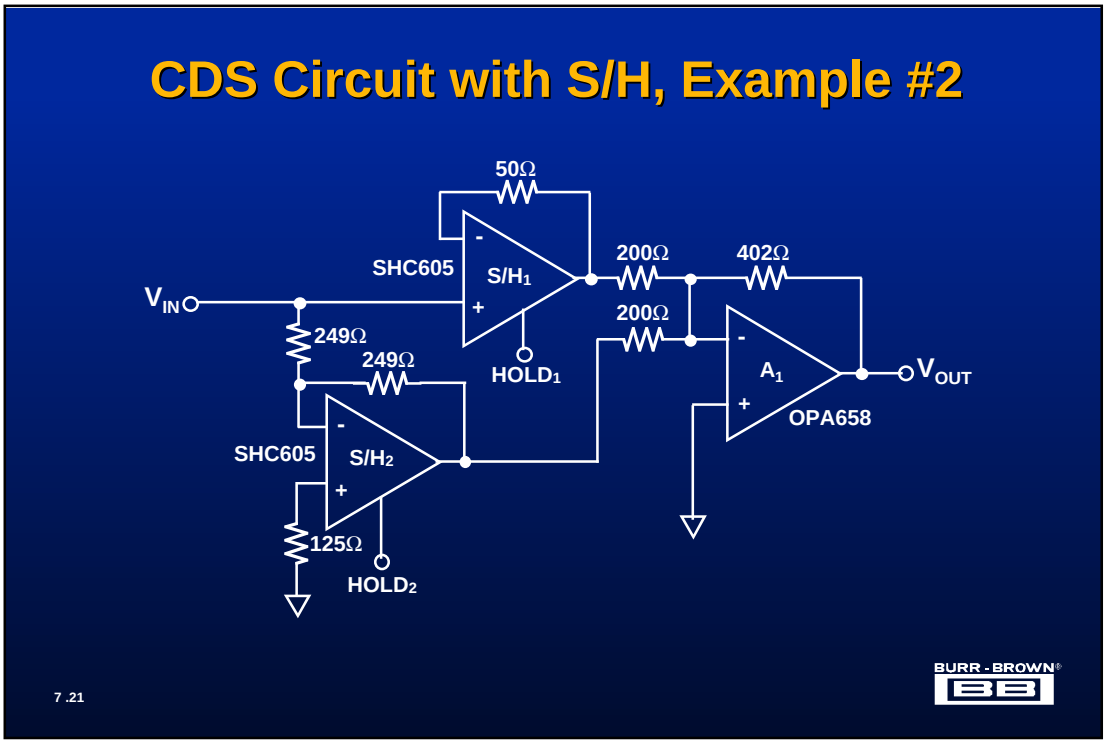
The double sampling technique also reduces the white noise. The white noise is part of the reset voltage ($V_{RESET}$) as well as of the video amplitude ($V_{RESET} - V_{VIDEO}$). With the assumption that the noise of the second sample was unchanged from the instant of the first sample, the noise amplitudes are the same and are correlated in time. Therefore, the noise can be reduced by the CDS function.



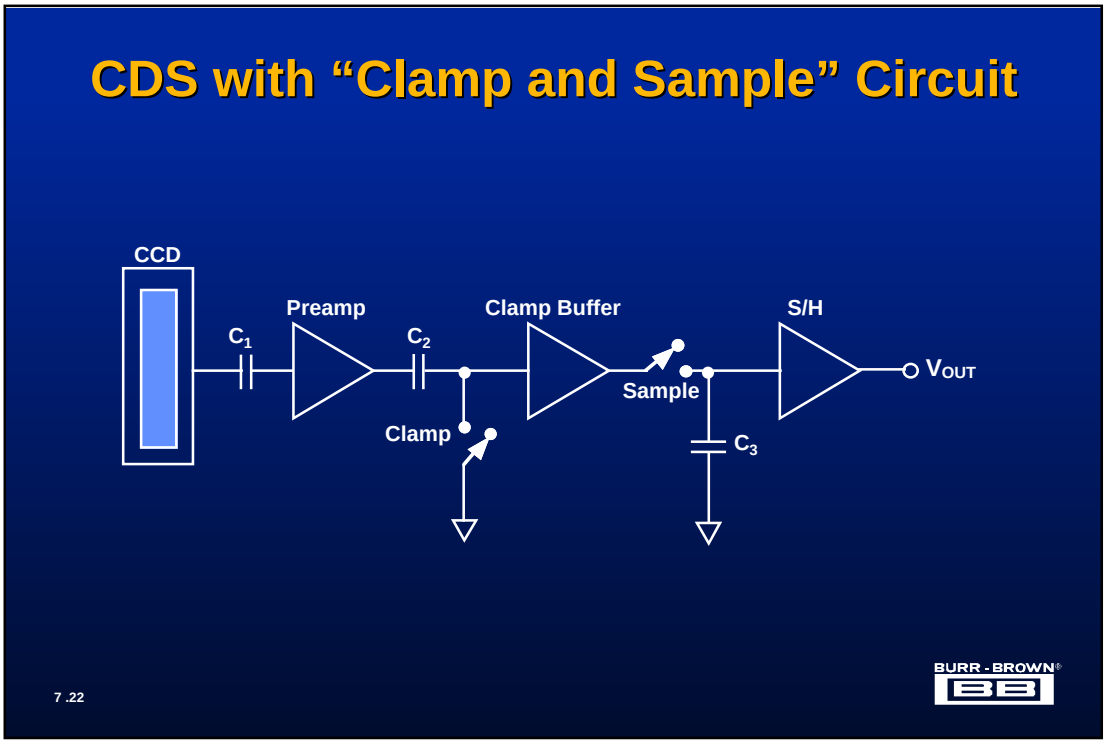

This is a circuit implementation of a correlated double sampler, CDS. It uses two SHC605 sample and hold ICs and the current-feedback op amp OPA658. The SHC605 has a differential input and can be connected in inverting or non-inverting configurations. In this design both sample and holds are connected as unity gain buffers, with a 50Ω resistor in the feedback. $S/H_1$ will capture a sample from the reset level. The other sample and hold, $S/H_2$, takes the sample of the video signal. The held output signals drive the positive and negative inputs of the difference amplifier, $A_1$. Its function is described by the equation $V_{OUT} = V_{IN+} - V_{IN-}$. As previously shown, the result will be the charge signal reduced by the noise component.

To match the output load of the two sample and holds, $S/H_2$ drives into two 200Ω resistors. This matches the load $S/H_1$ sees driving the one 402Ω resistor to the inverting input of the amplifier $A_1$.

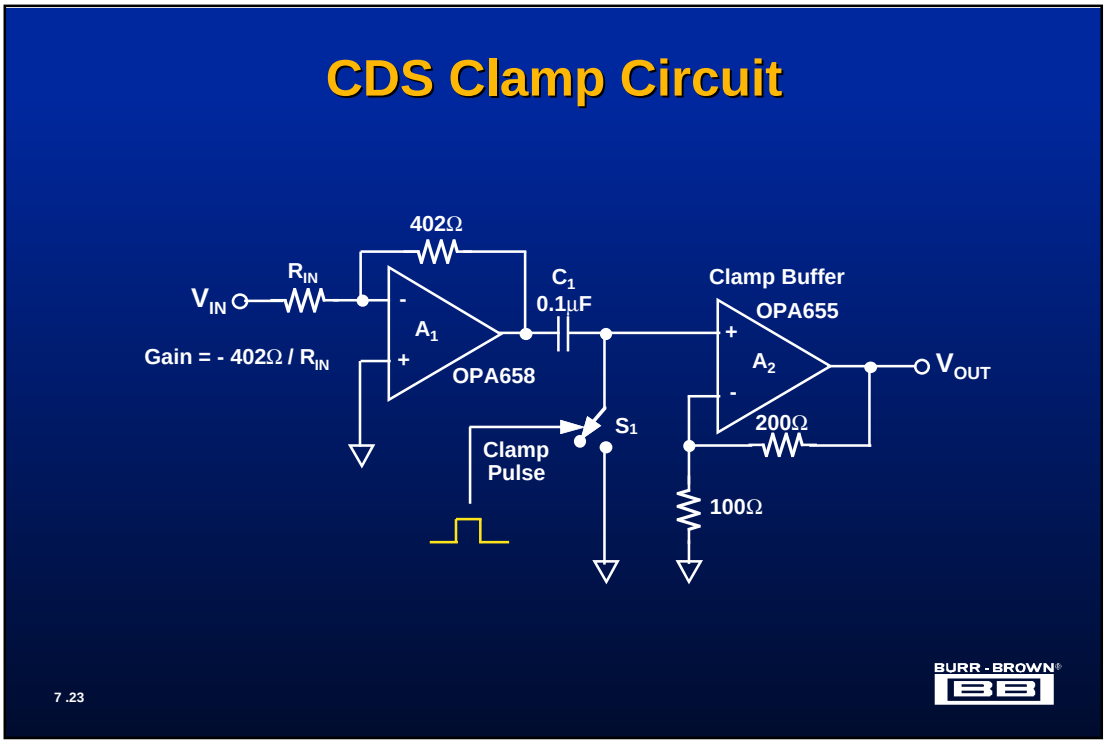
For more details on the difference amplifier employing a current feedback amplifier and optimizing it, see section 6 (High Speed Amplifiers).




This circuit shows a slight modification from the previous one. It also uses two sample and hold circuits (SHC605) to take the two samples needed for the correlated double sampling function. Again, sample and hold amplifier $S/H_1$ is set in a unity gain configuration. To implement the subtract function now, sample and hold amplifier $S/H_2$ is used for the signal inversion. This is done by configuring it in a gain of -1V/V. The difference amplifier can now be replaced by a simple summing amplifier, $A_1$. A current feedback amplifier, like the OPA658, is the preferred choice for this function because of its wide bandwidth and the independence of the bandwidth to the input resistors.




Another implementation of correlated double sampling is the “clamp and sample” technique as shown in this circuit. Capacitor $C_1$ blocks the high DC level from the first amplifier stage leaving only the signal amplitude. As already discussed, the preamplifier will gain up this signal. The capacitor $C_2$ and the clamp switch are the main elements within the CDS function. During the reset period of the CCD, the clamp switch is closed, grounding one side of the capacitor. At that time, the noise will charge the capacitor. In the next phase the clamp switch opens and the signal charge is processed through the amplifiers. By passing through $C_2$ the signal level gets subtracted by the amount of charge previously stored, thus eliminating the noise. The “noise free” signal level gets sampled in the subsequent sample and hold stage. This stage could be implemented by a monolithic S/H IC, like the SHC605.

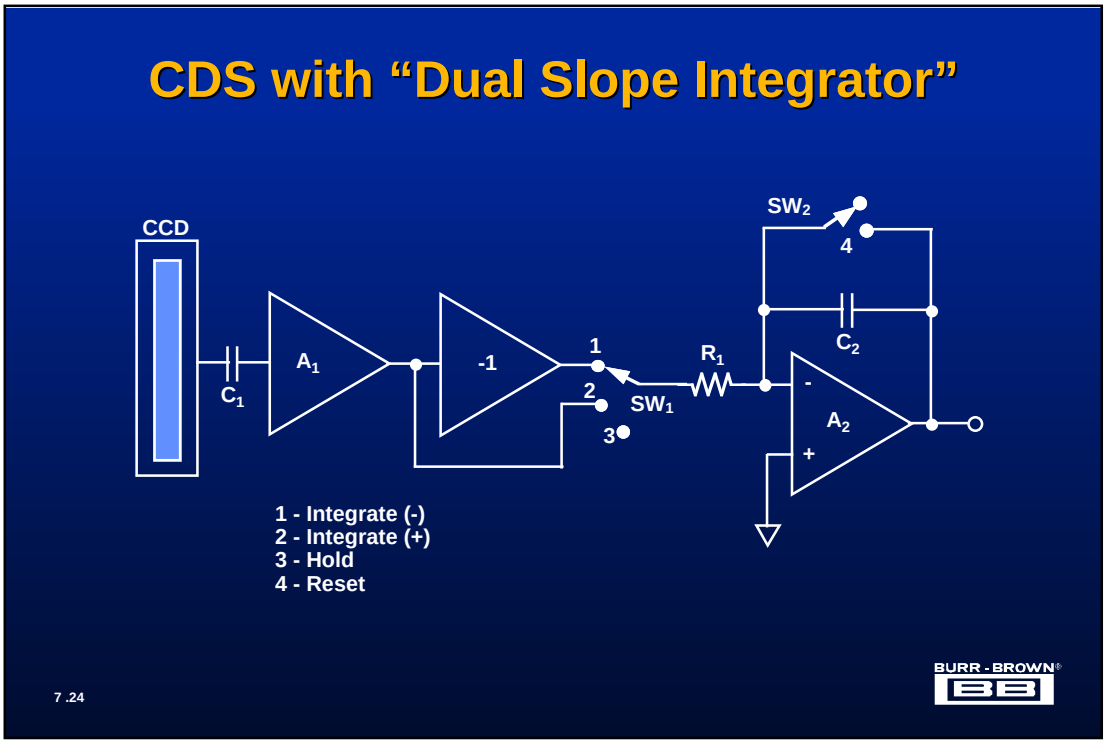



To further improve the signal-to-noise ratio, the first amplifier, $A_1$, gains the input signal by G = -402Ω/$R_{IN}$. This also gains the noise, but it will be removed from the signal with the CDS function. During the CCD reset period the clamp pulse is applied to switch $S_1$, grounding the capacitor $C_1$. The voltage on the ungrounded side of this clamp capacitor remains stationary, leaving the noise sample stored on the capacitor. The CCD reset period ends and switch $S_1$ opens. Now the pixel information, which is also contaminated with the noise, is transferred. Gained up by amplifier $A_1$ the signal passes through the clamp capacitor. Here a subtraction takes place, eliminating the noise component from the signal. Again, this is because the capacitor carries a charge (voltage) representing a sample of the rms-noise. As the signal passes the capacitor $C_1$, it is reduced by the noise sample stored on the capacitor. This effectively eliminates the noise. The “on” and “off” state of the switch implements the correlated double sampling.

The clamp buffer insures a decoupling of the sensitive node. The wideband FET op amp OPA655 is used here, since the FET input assures a very low bias current (10pA, typ). Bias current causes droop on the clamp capacitor, which could create a significant error depending on the timing.

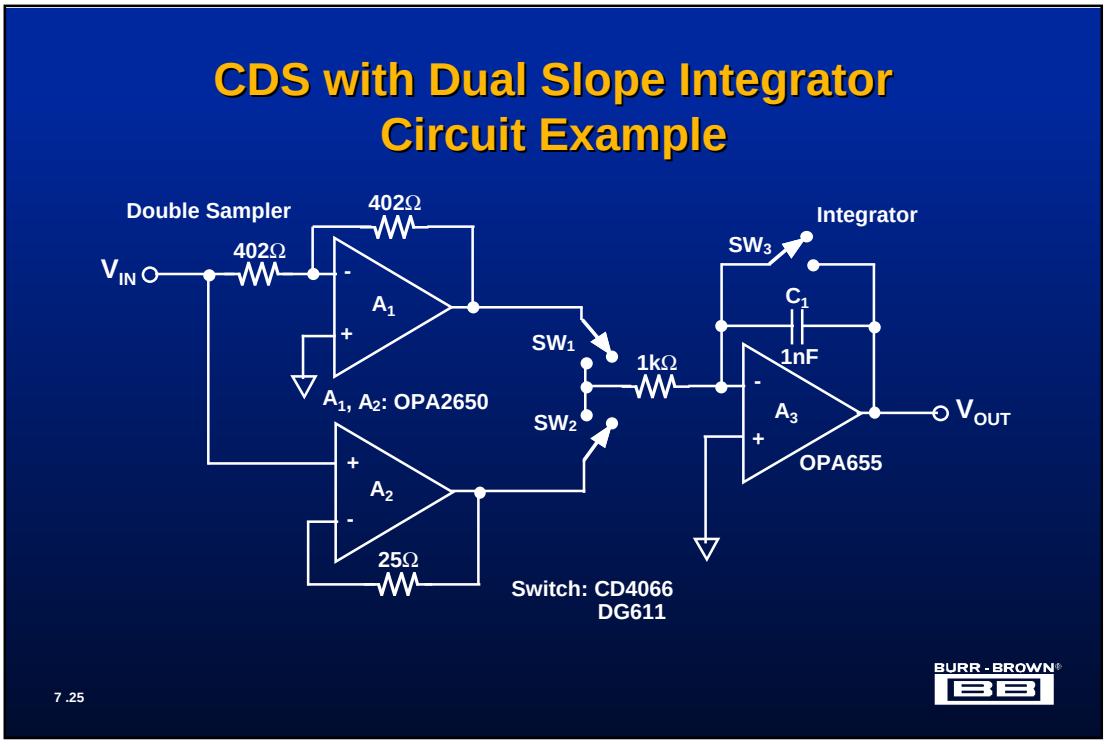
The output of amplifier $A_2$ would be sampled by a sample and hold IC.

The drawback of this circuit is that the used electronic switches can introduce kT/C-noise again. A standard CD4066 switch could be used for slow speed systems, whereas switches like the DG611 from Siliconix could be employed for high speed systems.




Another example of a CDS circuit is known as the dual slope integrator, named after the method used for its implementation. A basic design schematic is shown here. The principle function is as follows:

- During the reset time the first sample is taken with the switch ($SW_1$) in position 1. The sample is inverted before stored on the capacitor of the integrator.
- The second sample is taken shortly after the signal charge is transferred from the CCD output. Now the inverting stage is bypassed and the integrator sees the signal without the sign inversion.
- The switch goes into position 3, which is the HOLD position. Because of the sign inversion of the first sample, this charge will be subtracted from the signal charge of the second sample. Because of its correlation, the portion of the signal that gets eliminated is the noise. The signal sample is now available for further processing.
- Before the above procedure is repeated, the charge on the integration capacitor is zeroed by closing the reset switch ($SW_2$) across it.
- This CDS method is often used in low-light detecting systems. The integration time can range from ms up to several hours. The integrator CDS is found in applications for Astronomy, Spectroscopy, Microscopy, and Photometry. Those applications demand a wide range and high resolution.



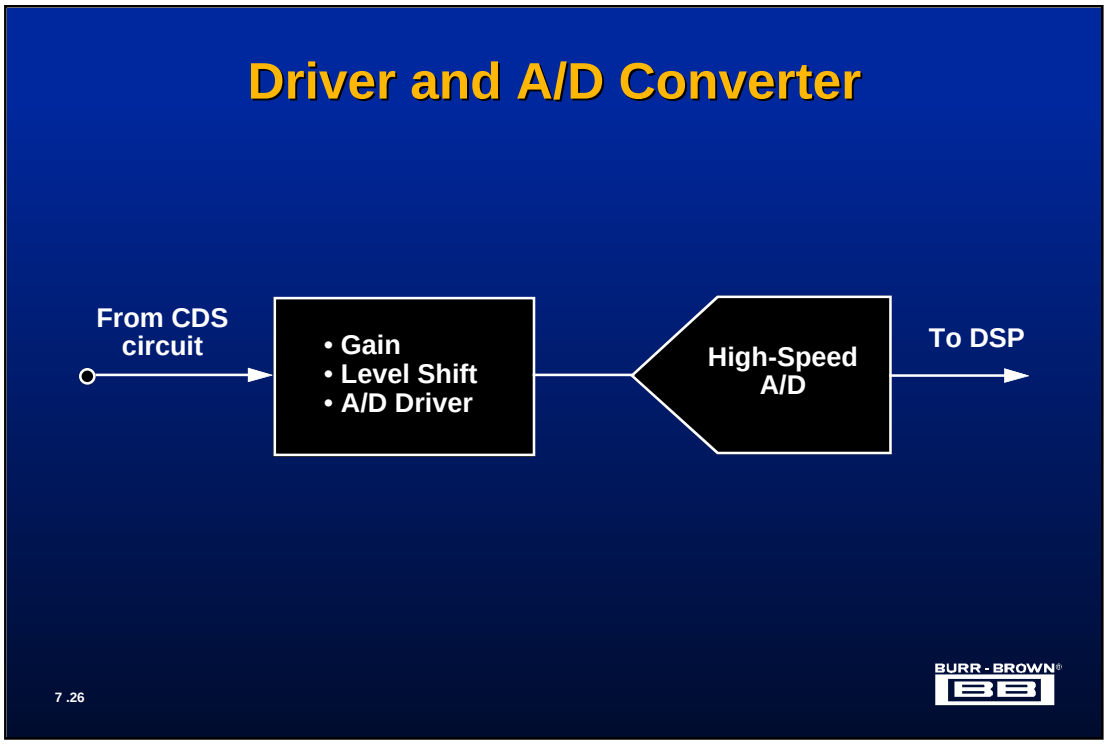

Another way of implementing the CDS function is by using a dual slope integrator. The simplified circuit shown here realizes this function. The requirement for an integrating CDS is that the video signal needs to be preconditioned in a way that it is available as a straight signal (0° phase) and as an inverted signal (180° phase). Here, amplifier $A_1$ and $A_2$ respectively, perform this function. A dual op amp, like the OPA2650, is well suited for this, since both amplifiers are closely matched in bandwidth and phase characteristics. During the CCD reset phase switch $SW_1$ closes, allowing the integrator $A_3$ to store the noise sample (kT/C-noise) for a finite time period.

In the next phase switch $SW_1$ opens and switch $SW_2$ closes, transferring the pixel information, which also includes the noise, to the integrator. Because this signal is now inverted, the common signal contents, which is the noise, will cancel out leaving only the video signal.

Note that a 25Ω feedback resistor, rather than a direct short, is used for the unity gain follower, $A_2$. This effectively reduces the Q of what would otherwise be a parasitic inductance (the feedback wire) into the parasitic capacitance at the inverting input.

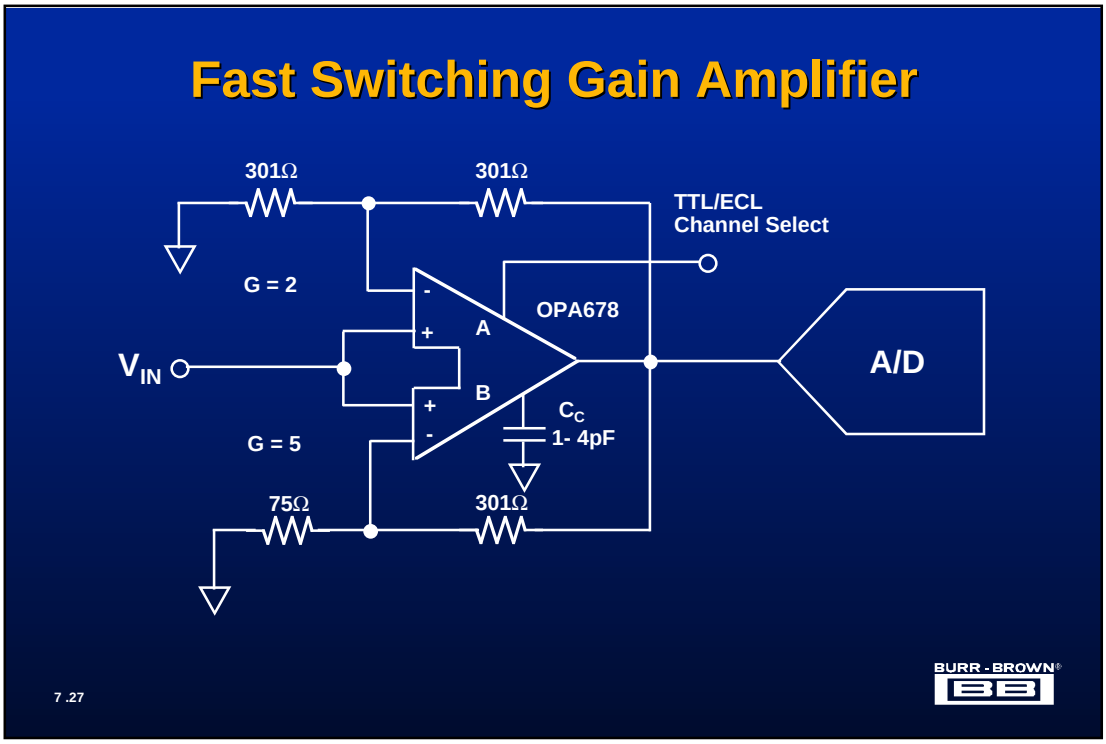
Depending on the application and the desired speed, the current feedback equivalent to the OPA2650 can be used, which is the OPA2658.

The integrator circuit uses the FET input wideband amplifier OPA655.



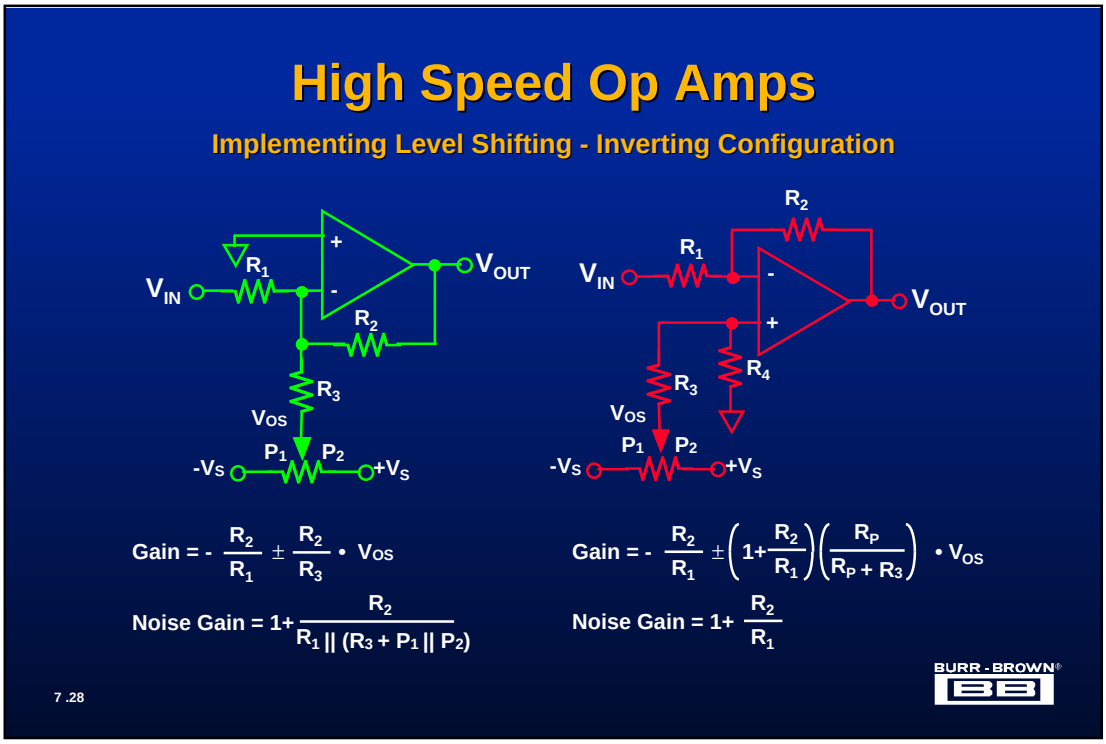

The last two analog processing stages within this CCD image system are the A/D converter and its driving circuit. This circuit could also be used to make the last span or offset adjustments to the signal to accommodate for the maximum input range of the converter. Compared to their industrial counterparts, converters in the imaging arena usually have a much smaller input range, for example ±1V. One reason for this is that a ±10V input range would require the driving amplifier to slew ten times faster for a fixed time interval. On the other hand, the limited ±1V input range sets tighter limits on the signal-to-noise ratio, and designers always strive to optimize the signal to the full scale range.

On the next few pages some general design ideas will be discussed.




As the signal magnitude becomes smaller, the signal-to-noise ratio decreases. Small signals might need to be gained in order to keep SNR at the required level. Of course, many techniques of implementing gain are available. Besides the continuous level adjust performed by AGCs the signal can also be adjusted in discrete gain steps.

One circuit example that switches between two different fixed gains is given here. The main component in this circuit is the switched op amp, or SWOP AMP, OPA678. The OPA678 is a wideband (200MHz) op amp with two independent differential inputs. Either one of the inputs can be selected by the TTL or ECL logic channel select pin, which takes only 4ns. The settling time to 0.01% for a 1V step takes about 30ns. Therefore, the gain could be switched between channel A and channel B immediately after a weak signal has been detected. To optimize the frequency response, an external compensation capacitor, $C_C$, in the range of 1 to 4pF is used.



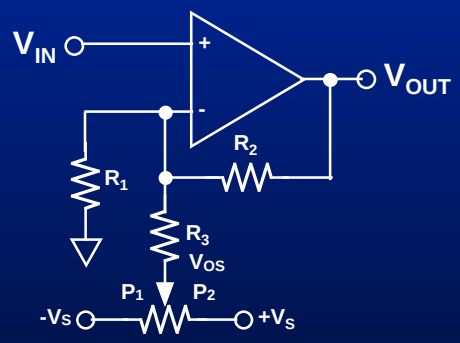

The driver circuit for the A/D converter can be configured for the final gain offset adjustment. Because high speed op amps usually do not have offset adjust capabilities directly on the IC, designers are required to implement the appropriate level shifting externally.

Two inverting level shifting circuits are shown here. In the left figure, a current is injected into the summing junction through $R_1$. Even though it's a simple method it has the disadvantage of an increased noise gain due to $R_3$ and the potentiometer resistance. If $R_3$ can be made much greater than $R_1$ and $R_2$, the increase in noise gain can be limited.

The circuit on the right avoids this noise gain problem. Here, the offset voltage is added to the signal through the non-inverting input. However, another resistor, $R_4$, is needed. The current noise of the non-inverting input creates, together with $R_4$, another noise source besides $R_4$ itself. To reduce this noise $R_4$ could be bypassed with a small capacitor in parallel.

# High Speed Op Amps

## Implementing Level Shifting - Noninverting Configuration




$$\text{Gain} \approx \left(1 + \frac{R_2}{R_1}\right) \pm \frac{R_2}{R_3} \bullet V_{OS}$$

$$\text{Noise Gain} = 1 + \frac{R_2}{R_1 \,||\, (R_3 + P_1 \,||\, P_2)}$$

7 .29

BURR - BROWN®

This circuit shows the level shifting scheme for the non-inverting op amp configuration. Similar to the first circuit, it adds a current to the summing junction. This works well for a small adjustment range, when $R_3$ could have a much higher value than $R_1$. Because the offset adjustment is in parallel to $R_1$ it affects the signal gain. For example, if $R_3$ plus the potentiometer resistance is 100 times higher than $R_1$, the gain will be altered by about 1%.

A common practice is to place the potentiometer between the two supply rails. The disadvantage of this can be easily seen; any variations in the power supplies will have an immediate effect to the signal.

## The A/D Converter

| | $f_{SAMPLING}$ | DNL | Power |
|---|---|---|---|
| *10-Bit* | | | |
| - ADS820 | 20MHz | ± 0.2LSB | 195mW |
| - ADS821 | 40MHz | ± 0.5LSB | 380mW |
| *12-Bit* | | | |
| - ADS800 | 40MHz | ± 0.4LSB | 390mW |
| - ADS801 | 25MHz | ± 0.3LSB | 270mW |
| - ADS802 | 10MHz | ± 0.4LSB | 250mW |
| *14-Bit* | | | |
| - ADC614 | 5.12MHz | ± 0.8LSB | 6.0W, Hybrid |
| *16-Bit* | | | |
| - ADC701 | 0.5MHz | ± 0.4LSB | 2.8W, Hybrid |

7 .30

BURR - BROWN®

CCDs sensors come in a wide range of capabilities which make them well-suited for a variety of industrial and commercial applications. For example, CCDs are largely used in commercial products like video cameras, fax machines or document scanners. Here, the requirements are mainly focused on the price of using an A/D converter with 8- to 12-bit resolution. Suitable A/D converters for this market segment are the ADS8xx family, offering different speeds and resolutions. The ADS8xxs are monolithic ICs using the pipeline topology, allowing fast clock speeds. Built on a 0.6µm CMOS process they come in a small 28-pin SOIC package.

In the following “High Speed A/D Converter” section we will take a closer look into the pipeline technology and its driving circuitry.

Scientific or medical applications often need higher resolution. Using larger CCD arrays to gather more light they usually operate with longer integration times. This means that the throughput rate of the A/D converter does not need to be that high. In this segment, converters with more than 12 bits of resolution have their place. Converters like the ADC614 (14-bit) or the ADC701(16-bit) could be used here.

# KIT 48. INTRODUCTION TO POWER AUDIO AMPLIFIERS

Using complete amplifier chips like the LM386, TDA2004, etc. ARE modern, quick and easy but they teach nothing about how power amplification actually work. Here is a Class A & AB amplifier made from individual components. It will deliver several watts of power into an ordinary 8 ohm speaker. It provides hands-on learning about audio topics such as cross-over distortion, bootstrapping, complementary pairs, and push-pull. We have given a description of these topics.

**ASSEMBLY INSTRUCTIONS**

Solder in the lowest height components first, then work up to the highest. The 10K pot is connected to the PCB by some 3 strand cable which you have to supply to the length you require.

Add the power transistors BD139 & BD140 last. Make sure to get the BD139 & BD140 in the correct way around and in their correct positions - the metal back of the package goes in above the bar marked on the overlay. Screw on the heat sinks **before** you solder them into the board. You will find that both transistors must be soldered some distance **above** the board so the heat sinks do not touch any of the other components. Join the two 9V battery snaps together in series to make the 18V DC. (Solder together one red lead to the black lead of the other snap.) The metal plates of both power transistors are joined to their collectors so there is no need to insulate the heat sinks.

**SETUP**

First remove the heat sink of Q6. This will allow easy access to the trimpot. The first setup step is to adjust the trimpot to remove any crossover distortion in the power transistors. Because the amount of bias applied to transistors increases as the resistance increases, start by rotating the pot fully clockwise. This should give a resistance of zero. Do this before turning on the power to avoid giving too much bias.

Ideally to do this adjustment you should apply a 1 kHz sine wave input. (If you have our Kit 23 you can use that.) As you listen to the speaker gradually rotate the pot until the distortion disappears. The increase in sound quality should be quite obvious. However check that you have not set the bias too high which will cause quiescent current to flow. If the power transistors feel quite hot to touch without any input signal then the bias is set too high.

If you do not have a sine wave input then a second method is to monitor the current drawn by the amplifier as the resistance is increased. Put an ammeter into the circuit. Increasing the pot resistance will increase the current, but the rate of increase will become much more rapid as Q5 & Q6 turn on more. The elbow in the rate of current drawn should occur about 14mA.

As we will discuss below the pot resistance for no crossover distortion should be about 22 ohms. If the battery voltage falls below 18V the resistance will increase. If you use 9V power supply the resistance needed will be about 180 ohms. This is why a 200R pot has been used. If you will always use an 18V supply then you can use a lower value pot for greater accuracy.

**WHAT TO DO IF IT DOES NOT WORK**

Poor soldering is the most likely reason that the circuit does not work. Check all solder joints carefully under a good light. Next check that all components are in their correct position on the PCB.

**CIRCUIT DESCRIPTION**

The power module has been designed for a maximum input signal of about 50mV. With a signal larger than this the 10K volume control acts as a potential divider to reduce the signal and prevent distortion. For smaller signals you will need to use a pre-amplifier. Or you could increase the value of the 10K feedback resistor R5 and so increase the amplifier gain above 11 (the ratio of (R3+R5) to R3.) Do not reduce R3 which will also affect the frequency response of the amplifier.

There are various classes of amplifiers which can be used to drive a loud speaker. The text books list them: class A, B, AB & C. The classes are defined in terms of the amount of bias which is applied to the amplifier input. Class A is permanently and fully biased on, while Class C only conducts on the peaks of the signals. In this design we have used Classes A and AB.

**First Stage.** Q1 & Q2 are arranged as a complementary pair in a common emitter mode. This gives a high voltage amplification. However, it has a high output impedence which means it is not suitable for driving an 8 ohm speaker directly. R1 & R2 apply sufficient bias to keep the transistor pair permanently turned on. The amount of bias places them in the middle of their conduction range so that they react to both positive and negative swings of the input signal. There is continuous current flowing whether or not a signal is actually present and being amplified.

To reduce the wastage of quiescent current another class of amplifier, Class B, uses a pair of complementary transistors which are not quite biased on. The signal is amplified by using the NPN transistor to react to the positive voltage side of the input signal while the PNP reacts to the negative side. At any instant only one transistor is turned on. This is called push-pull; when the NPN transistor (like out Q5) turns on the output voltage is pulled up towards the positive supply rail, while turning on the PNP (Q6) pushes it towards ground. Class B solves the quiescent current problem of Class A amplifiers, but introduces another - crossover distortion. Because the first part of the signal is used to complete the turn-on bias for each transistor you introduce distortion in the output whenever the input signal swings between positive and negative, or negative to positive. Enter Class AB Amplifiers.

**Second Stage.** Class AB also uses a pair of complementary transistors acting in the same push-pull arrangement, but it

adds sufficient bias between the base-emitter junctions to just turn on both transistors when there is no signal flowing. When a signal is applied one of the transistors will turn on more and the other turned off. A diode in series with an adjustable resistance (as in our circuit), or the controlled output voltage of another transistor is normally used to apply precise bias. If the bias is too small then there will still be some distortion; but if it is too large an increased quiescent current will flow which will waste power.

In the schematic you can see that the second stage of our amplifier is set up as an emitter follower with the load connected to the emitter of the transistor rather than the collector as in the first stage. The voltage at the output 'follows' the voltage at the input. They are the same value except from a 0.65V drop across the base-emitter junction. The advantage of the arrangement is that it produces a large output current at a low output impedence. This is ideal for driving a speaker.

Now let us look at the bias needed to remove the crossover distortion which is the special feature of Class AB amplifiers. Because our circuit uses two Darlington pairs in the final amplifier stage, the bias needed to just turn on Q3 - Q6 should be about 2.6V (4 x 0.65V.) This is supplied by the voltage drop across the green LED and the trimpot. After adjusting the pot to remove crossover distortion you you should have the following circuit values: the bias voltage is about 2.05V - 1.93V across the LED and 0.12V across the pot; a collector current of 5.44 mA and a pot resistance of 22 ohm.

**Overall Description.** Q1 & Q2 form a complementary pair for the audio voltage amplification. The advantage of using two transistors for this stage is that the arrangement has a very high input impedence and hence does not load down the input signal. That means it draws very little current from the previous stage. The load for transistor Q2 is the green LED, the pot resistance plus R4. Note that R4 is not joined directly to the ground rail (as you might expect) but indirectly via the speaker. This is called bootstrapping, a topic we will shortly return to.

The pairs of transistors Q3/Q5 & Q4/Q6 form a current amplifier. The small valued R6 & R7 help to stabalize the circuit. Voltage changes across these resistors act as negative feedback and help to counteract any alterations in current caused by temperature, or differential gains in different transistors. There is also a negative feedback loop provided by R5.

**Bootstrapping.** This is a technique which allows you to unlock or separate the AC & DC operations of an amplifier in order to get an increased power output. In some ways it is like the operation of an inductor which has a low DC resistance but a high AC impedence. The smaller DC resistance of the bootstrapped load does not restrict the current flow, while the higher AC impedence results in a large voltage being generated across the load. And combining high current with high voltage gives high power (P=E x I).

In general, bootstrapping provides positive feedback from the output to the input of a unity gain amplifier in such a way that a particular point of the circuit is 'pulled up by its own bootstraps'. The signal voltages at the opposite ends of the bootstrap rise & fall together, with virtually the same AC signal appearing on both side, providing a higher impedence load for the driver transistor than its ohmic value would indicate. Let us look at how this works for our circuit.

Resistor R4 is the bootstrapped load. Because it is connected after the capacitor C3, the bootstrap is effective only for the AC signal and the extent of the multiplying effect will be determined by the true voltage gain of the nominally unity gain amplifier. Suppose the amp has true unity gain. Then the voltage gain at both ends of resistor R4 would be exactly the same. With no AC voltage drop across R4 no current would flow through it so it would have an effectively infinite resistance. Now suppose that the voltage gain is a more realistic 0.9. This means that the voltage at the top & bottom of R4 would be 1V and 0.9V respectively making the voltage drop across it 0.1V. This is only one-tenth of the full V which would occur across R4 if it were connected directly to ground. Hence, R4's 'bootstrapped' resistance is 10 times greater than its 1K5 value. So to the driver transistor Q2, R4 now provides an AC load impedence of 15K. The closer the gain approaches unity the greater the effective impedence provided by the bootstrap.

The advantage of such a setup is that Q1 & Q2 achieve a higher AC voltage gain because their output is developed across a higher value load resistor. If R4 were not bootstrapped but simply connected to ground the value of R4 would have to be increased to 15K to achieve the same effect. But such an increase would result in the base currents of Q4 & Q6 (which also flow through R4) developing a far larger voltage across the resistor. This voltage could easily be large enough to turn off Q2 on the negative sections of the signal preventing its voltage swing going anywhere near ground.

**Calculation.** An 8 ohm speaker dissipating 1W of power draws an RMS current of 354 mA ($P=I^2R$), or a peak current of 500mA (354 x sqrt2). If the current gain of each transistor Q4 & Q6 is 30 (typical) then the base current of Q4 is 556 uA. This small current still produces a voltage drop across 15K of 8.3V, restricting the negative output voltage swing to only 0.7V (9 - 8.3V). Clearly the driver transistor Q2 will shut down long before the amplifier delivers a peak load to the speaker.

Hence the advantages of bootstrapping - the multiplied resistance of the load resistor - gives us an increased AC voltage gain from the first stage of the amplifier without decreasing the AC current from the second stage. This results in higher output power.

# KIT 48. INTRODUCTION TO POWER AUDIO AMPLIFIERS

**Supply Voltage.** This calculation also explains why increasing the supply voltage will increase the volume of the output. A higher supply voltage means that the input signal to the power stage can have a greater voltage swing before clipping which results in a greater swing for the output. If the same current still flows then doubling the voltage (RMS) will double the output power. To show this the power output of the module was measured at various supply voltages. To do this the speaker was replaced by a 5W, 8 ohm resistance, a 1kHz signal was the input and the output viewed on a CRO. The pot was adjusted for each voltage.

Next the amplitude of the input signal was adjusted to achieve the maximum output signal without distortion (by looking at the output on the CRO.) Then the AC voltage across the load resistor was measured. This gave an approximate RMS value. The power dissipated in the 8 ohm load was calculated from $P = V_{rms}^2/R$. This gave the following results:

| VCC | VRMS | Power |
|---|---|---|
| 9V | 1.70V | 361mW |
| 12V | 2.34V | 684mW |
| 15V | 3.13V | 1.22W |
| 18V | 3.67V | 1.68W |

So it can be seen that any increase in supply voltage gives a substantial power increase. Doubling its value gives four times the power. This is why very large power amplifiers (hundreds of watts) need very high supply rail voltages.

**Web Address & Email.** You can email us at **peter@kitsrus** if you have any problems or requests. See our Web page at:

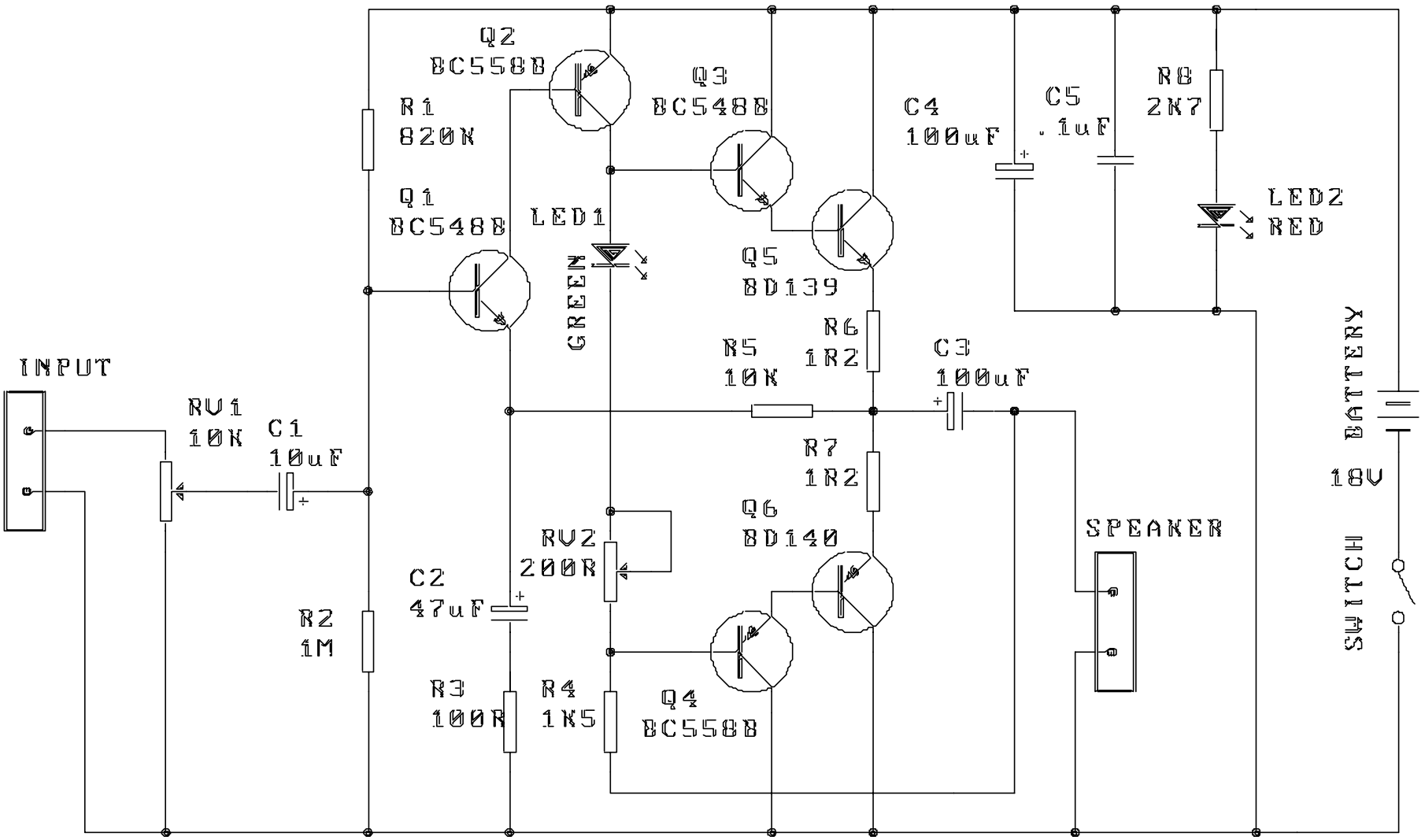
**http://kitsrus.com**

**Acknowledgements.** This module was first published in ElectronicsAustralia December, 1993, and is adapted here with permission. The PCB has been redesigned by DIY Electronics.

**COMPONENTS**

| | | |
|---|---|---|
| Resistors, 5% carbon: | | |
| 1R2 | R6 R7 | 2 |
| 100R | R3 | 1 |
| 1K5 | R4 | 1 |
| 1M | R2 | 1 |
| 2K7 | R8 | 1 |
| 820K | R1 | 1 |
| 10K | R5 | 1 |
| 10K pot | | 1 |
| 200R Koa trimpot | | 1 |
| 0.1 uF capacitor | C5 | 1 |
| Electrolytic capacitors: | | |
| 10uF | C1 | 1 |
| 47uF | C2 | 1 |
| 100uF | C3 C4 | 2 |
| BC548B | Q1 Q3 | 2 |
| BC558B | Q2 Q4 | 2 |
| BD139 | Q5 | 1 |
| BD140 | Q6 | 1 |
| 5mm Green LED | | 1 |
| 5mm Red LED | | 1 |
| Speaker 8 ohm, 3" | | 1 |
| 9V battery snap | | 2 |
| SPDT PCB switch | | 1 |
| 2 Pole Terminal blocks | | 3 |
| Heat sinks HS103 | | 2 |
| Nut & bolt set | | 2 |
| Kit 48 PCB | | 1 |

# KIT 48. INTRODUCTION TO POWER AUDIO AMPLIFIERS



# KIT 53. CODEPAD/ALARM USING 68HC705P9

This project started out to use the K1 micro-controller (as used in some of our other Kits) for a codepad - entering a four digit code would operate a relay. The most common use of such a codepad is to open a security door using a door relay (also called a door strike.) Then we looked at other codepads on the market. We tried to incorporate all their good features into our software to produce the best codepad possible. (And at the same time learn what not to do from some of the very poor features we came across.) The project outgrew the K1 and we ended up using a Motorola 68HC705P9.

We found some codepads which already used a micro-controller (uC) did not use it to its full advantage. Other codepads were so complicated to program that you almost needed a PhD in electronics! We have put together a combined codepad & independent burglar alarm into a hopefully easy to understand package. It has all the normal features of a modern codepad (EEPROM storage, programmable alarm times, master & user access codes, panic, tamper alarm etc.) But we have added other features such as an independent burglar alarm with its own relay, a second relay on board with its own user access code and direct triggering of the main door relay (used for the door strike) in software if required. We have tried to use the power & capability of the P9 uC to make all features logical & easy to use. The 3x4 matrix keypad is located remote from the main board. The software code fully commented is supplied on 3 1/2" floppy disk so you can learn how to program these uC's for yourself.

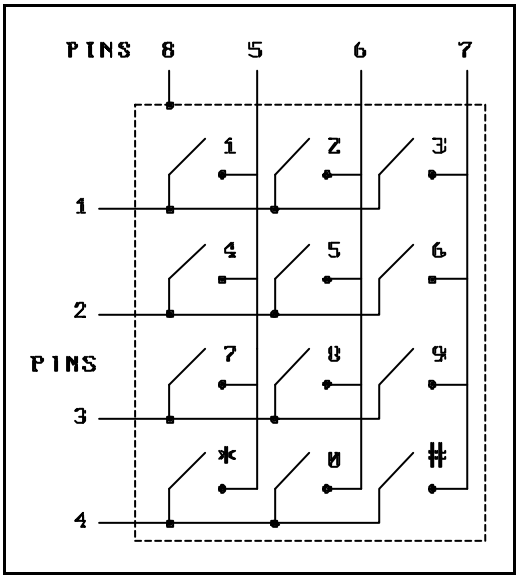



**Figure 1. Keypad Connections**

**Features**

3 relay outputs:

- door strike relay (with green LED indicator)
- auxiliary relay (with yellow LED indicator)
- alarm relay (flashing red LED)

3 programmable User Access Codes (UAC):

- UAC1 operates door strike relay only
- UAC2 operates auxiliary relay only
- UAC3 operates both door & auxiliary relays

Programmable Master Access Code. If you forget the Master Access Code, you can enter programming mode and make another one if you have access to the PCB.

EEPROM for data retention and security.

Momentary or latch output for either door strike or auxiliary relay.

Programmable access time from 1 to 9 seconds. Normally Open (NO) and Normally Closed (NC) alarm inputs which operate an alarm relay. Alarm inputs are turned off after entering a valid User Access Code 1. (Why? Because the alarm may be connected to a device - reed switch on the door, PIR - inside the room you are just about to enter.) Alarm inputs are re-enabled when you leave the room by entering '*' followed by a valid user access code.

Alarm inputs may be completely disabled.

Programmable alarm time from 0001 to 9999 seconds (0000 disables alarms). Alarm turned off by entering a valid user access code. Visual indication that alarm was set off.

Keypad Tampering. The alarm is triggered after 4 invalid access codes are entered consecutively over any length of time. The invalid count is cleared by entering a valid access code.

Panic Alarm. The alarm is triggered by pressing '*' and '#' keys together. This feature can be disabled.

Hardware test point to check software operation.

**Construction**

First check the tracks on the top and bottom layers of the PCB under a strong light. Look for any defects in the track work. Next check the components against the listing on the next page. The 68HC705P9P, or (P9CP) comes preprogrammed. However, we have supplied a copy of the source code with full comments on the floppy disk so you can learn how to program these uC's.

It is generally good practice to insert & solder all the lowest height components first. Note the orientation for the electrolytic capacitors, the ICs, the piezo buzzer and the LED's. Bend the legs of the 7805 using needle-nosed pliers. Do not just solder it into place then push it over; you may break the case where the legs enter it. Screw down the body using the nut & bolt provided.

Follow the overlay as you place the components and check them against the reference designator in the Components list. Note that the terminal blocks slide-fit together.

# KIT 53. CODEPAD/ALARM USING 68HC705P9

**Keypad**. Although most commercial codepads place the keypad & the main PCB together as a single unit we felt the kit was more versatile by allowing them to be separate. For example, to have the codepad unit outside a door but the expensive PCB board to be on the other side of the wall. We have supplied some flat rainbow cable with the kit. It is up to the user to attach the keypad to the code pad to suit their own application. The GND pad of the keypad, KP1 on the PCB goes to pin 8 of the keypad unit. Then all the wires connect in turn to pin 1 on the keypad unit which goes to the top pad of KP1 in the box marked on the overlay by a box.

Looking at the back of the keypad, pin 1 is on the right and pin 8 is on the left.

The internal matrix arrangement of the keypad can be seen on the previous page. A '1' for example is indicated to the microcontroller by shorting together of pins 1 & 5.

**PCB I/O.** At the top of the PCB is terminal block X1. A 12V unregulated supply is satisfactory but a regulated supply would be preferred. Note that the voltage output for the door strike at X2 will draw well over 1A when operating (typically 1.4A) so the power supply must be able to deliver this. A 1A supply is not enough. X2 supplies 12V when a correct UAC1 or UAC3 is entered. X3 outputs the auxilliary relay triggered by UAC2. The Common, NO & NC terminals are available. UAC2 allows you to use the codepad to operate another device from it

X4 makes available the Common, NC & NO outputs from the alarm relay.

X5 accepts NC & NO external alarm loops. Eg, the NC loop could go to a tamper switch; if someone forces open the container & the switch opens then the alarm is triggered. Or it could go to a reed switch that the door strike opens. If not used put a wire connection between GND & NC (otherwise the alarm will go off continuously.)

Closing the terminals of X6 triggers the door strike. In other codepads this is usually done in the power supply to the door relay itself. But it is easier & neater to incorporate the function into software.

**Software Test Point.** Test point TP1 has been provided on the PCB to test that the software is working. The software generates a continuous 250 Hz square wave at pin 24 of the P9. You can use a CRO or frequency meter to test for the presence of this signal.

**First Time Start.** First, did you put a wire connection between the GND & NC terminals of the terminal block X5, Alarm Inputs? This is the NC burglar alarm input. If there is no connection between NC and GND then the alarm will be triggered continuously.

**Second,** there are no preprogrammed defaults. When you power-up for the first time hold down the Lost Code button - the zippy switch - in the centre of the PCB. This will take you straight into program mode. This is indicated by the rapidly flashing red & green LEDs. Press 0 (the yellow LED will go on) to set the Master Access Code of 4 digits. The yellow LED will go out when you have entered 4 digits & the red & green LEDs will keep flashing. You have 30 seconds to continue with programming. Press 1 to set the user access code 1; the yellow LED come on then goes off after 4 digits are entered. Then press 2 to set user access code 2. (After 30 seconds of inactivity the program jumps out of program mode back to normal mode.)

If you press an option then decide you do not want to go on then press #. Pressing # a second time will take you out of program mode. Now play with it. When you have a good idea of what is going on go into program mode again & set the other program options:

- 3. Set User Access Code 3 (4 digits)
- 4. Set Door Strike relay ON time from 1 to 9 seconds (0= latch mode). (1 digit)
- 5. Set Auxiliary relay ON time from 1 to 9 seconds (0 = latch mode). (1 digit)
- 6. Set alarm ON time from 0001 to 9999 seconds (0000 = disable). (4 digits). See below for how this option also affects the red LED L1.
- 7. Enable/disable panic alarm (0 = disable, 1 = enable). (1 digit)
- 8. Enable/disable user access code (1, 2 or 3) followed by 0 (disable) or 1 (enable). (2 digits.)

All these entered options are now remembered by the 93LC46. Even when the power is turned off they will be loaded automatically into the P9 upon power-up.

**Operation Guide**

**Overview.** We have tried to make this Codepad & Burglar Alarm as simple & logical to use as possible. We have tried to include different options.

The Codepad has two modes of operation - **NORMAL** mode and **PROGRAM** mode. Normal mode is the day to day running mode of the codepad. Program mode is used to set or alter the various options available.

**NORMAL mode.** The red LED L1 can be on or off. If Option 6 is set to '0000' then the alarm inputs are peranently disabled & the red LED is off. If Option 6 is set between '0001' to '9999' seconds then the alarm inputs are enabled & the L1 LED is continuously on. See "Alarm Conditions" for further information. Let us assume for the following discussion that the Alarm Option is on & the red LED L1 is on.

Entering User Access Code 1 operates the door strike relay and turns on the green LED. The red LED starts to flash. The door strike relay remains operated for the user-

# KIT 53. CODEPAD/ALARM USING 68HC705P9

programmed entry time (1 - 9 seconds). If the door strike relay is set up for latch mode operation (0 seconds), it will remain on until UAC 1 is entered again. After the relay entry time the green LED goes but the red LED L1 keeps flashing. For the reason see 3 paragraphs below.

The door strike relay may also be operated via the External Strike Control input. (For example, a guard on the inside of the door sees you, recognises you as allowed to enter and presses a button to open the door for you to save you entering the UAC.)

Entering User Access Code 2 operates the auxiliary relay and turns on the yellow LED. It remains operated for a user-programmed time (1 - 9 seconds). This may be a different time to that programmed for the door strike relay. Latch mode is also available if 0 seconds is set. Note that this Auxilliary Relay is only rated at 1A and must not be used to directly control a second door strike relay.

Entering User Access Code 3 operates both the door strike and auxiliary relays. Both the green and yellow LEDs will turn on for their respectivelyprogrammed entry times.

**Flashing Red LED.** Entering a valid user access code automatically disables the alarm inputs. The red LED L1 starts to flash, indicating that the alarm inputs have been temporarily disabled. Why do this? Because the alarm may be connected to a device - reed switch on the door, PIR - inside the room you are just about to enter. When
you leave the room or you want to re-enable the alarm inputs enter '*' followed by a valid user access code. This reactivation of the alarm input - by putting * in front of a valid UAC will not activate a relay. It will just stop the LED flashing as the alarm input is re-enabled. All UAC's still work and operate the relays. (If you want to completely disable the alarm inputs and at the same time stop the LED flashing enter '0000' for Option 6.)

Each UAC may be disabled (see Program Mode).

Once started, each digit of an access code must be entered within 5 seconds of the previous one. If not, any entered digits are "thrown away" and the sequence automatically re-started. If you make a mistake while entering the access code, press ' * ' or ' # ' to start again.

**Alarm Conditions.** The alarm relay may be triggered in threeways:

- Burglar alarm input (provided it is enabled)
- Entering 4 successive invalid access codes
- Panic alarm (provided it is enabled)

If any of the above conditions are met, the alarm relay will operate and remain operated for the programmed alarm time (0001 - 9999 seconds). The alarm LED L4 will also start flashing, giving visual indication that an alarm has been triggered. This LED L4 will continue to flash even after the alarm time has elapsed. (This is to let you know that the alarm has been triggered.)

The alarm (and flashing LED) may be turned off by entering any valid User Access Code either with or without a * before it.

If both L1 & L4 are flashing then the one entry of a UAC preceeded by a '*' will turn of both of them.

The burglar alarm can be either the closing of a normally open loop, or the opening of a normally closed loop. It operates independently of the Codepad. It could be the input to a tamper switch if the PCB is mounted inside a box. The burglar alarm inputs may be permanently disabled by setting the alarm ON time to "0000" (programming option 6). This turns the red LED off completely. Keyboard tamper and panic alarms (if enabled) are not affected except that the alarm ON time
is now automatically set to 10 minutes.

Alarm conditions are continually monitored, even if the codepad is in Program mode. To turn off the alarm while in Program mode, first exit Program mode then enter a valid user access code.

The panic alarm feature can be disabled if not required (Program Mode Option 7).

**PROGRAM Mode.** Program mode can be accessed at any time by entering the Master Access Code. The codepad will beep 4 times and the red and green LEDs will flash.

When using the codepad for the first time or if you forget the Master Access Code, hold down the Lost Code button when you turn on the power. This will take you directly into Program mode. There are no pre-programmed default codes or conditions in the codepad. When first used, all access codes and time settings must be entered by the user.

The flashing LEDs indicate that the codepad is waiting for a valid programming option to be entered. The codepad will automatically exit Program mode after 30 seconds if a valid programming option is not entered at this point. The codepad will beep 4 times and normal mode re-entered.

Program mode may be exited by pressing ' # '.

There are 9 valid programming options, 0 - 8:

- 0. Set Master Access Code (4 digits)
- 1. Set User Access Code 1 (4 digits)
- 2. Set User Access Code 2 (4 digits)
- 3. Set User Access Code 3 (4 digits)
- 4. Set Door Strike relay ON time from 1 to 9 seconds (0 = latch mode). (1 digit)
- 5. Set Auxiliary relay ON time from 1 to 9 seconds (0 = latch mode). (1 digit)

# KIT 53. CODEPAD/ALARM USING 68HC705P9

- 6. Set alarm ON time from 0001 to 9999 seconds (0000 = disable). (4 digits)
- 7. Enable/disable panic alarm (0 = disable, 1 = enable). (1 digit)
- 8. Enable/disable user access code (1, 2 or 3) followed by 0 (disable) or 1 (enable). (2 digits.)

Once a programming option has been entered, the yellow LED will turn on. This indicates that the codepad is waiting for new programming data to be entered.

Data entry must commence within 5 seconds of the programming option being selected. Where more than 1 digit of data is required (e.g. access codes), each digit must be entered within 5 seconds of the previous one. If not, the codepad will automatically exit the programming option selected. Pressing '#' will abort the option selected without any change taking place. In either case, a long beep will sound to indicate the "error" condition.

Four beeps will sound when the correct data is entered and the codepad will return to Program mode entry (flashing red/green LEDs and yellow LED off). There is no "enter" key.

Options 0,1,2,3 and 6 require 4 digits, options 4, 5 and 7 require 1 digit and option 8 requires 2 digits.

**External Control Input.** This normally open input allows a pushbutton switch or remote control device to be connected to the codepad. Shorting the two input pins together operates the door strike relay for the programmed ON time. This input is disabled when the codepad is in program mode.

**Access Code Priorities.** The Master Access Code has priority over all the User Access Codes. This means that any User Access Code that is the same as the Master Access Code will be ignored. The User Access Codes also have a priority order, 1-2-3. Setting 2 or more User Access Codes the same will cause the lower priority code to be ignored.

**Some Technical Information.** The MC68HC705P9P (or P9CP) is a member of the HC05 family of 8-bit microcontrollers from Motorola. The P9 contains 2112 bytes of one-time programmable ROM, and 128 bytes of user RAM. There are 20 bi-directional I/O port pins and one input-only port pin. There is a 4 channel, 8 bit ADC. Single 3.3V - 5.0V supply. The P suffix means that it is a One Time Programmable device. There is no window in it to allow the program to be erased. See the MC68HC705P9 Data Book from Motorola for full information or download it from their website.

**93LC46/93C46.** This IC can be a trap. There are 8 bit & 16 bit versions & not all manufacturers support both versions. Some manufacturers pull a pin high or low to switch between versions. Others like Microchip use different parts completely. We now use the Microchip 93LC46A (8 bit) NOT the B (16 bit.)

## COMPONENTS

| | | |
|---|---|---|
| Resistors 1/4W, 5%: | | |
| 10K | R5 to R15, R17 | 12 |
| 560R | R2 R3 R4 R16 | 4 |
| 10M | R1 | 1 |
| | | |
| Capacitors: | | |
| 10nF mono cap | C7 | 1 |
| 100nF mono caps | C1 C4 | 2 |
| 10uF mini elcap | C5 | 1 |
| 1uF mini elcap | C6 | 1 |
| 27pF ceramic | C2 C3 | 2 |
| | | |
| Regulator 7805 | | 1 |
| EEPROM 93LC46A | IC2 | 1 |
| Programmed 68HC705P9CP | | 1 |
| BC547 | Q1 Q2 Q3 Q4 | 4 |
| Piezo Buzzer | B1 | 1 |
| Zippy tact switch | SW1 | 1 |
| 1N4004 diode | D1 D2 D3 D4 | 4 |
| 4.7V zener diode | D5 | 1 |
| 12V Relay RWH-SH-112D / RUDH-SS-112D | | 1 |
| Mini 12V relay UA-SH-112D | | 2 |
| 4.000 mHz crystal (49U/S) | Y1 | 1 |
| 3mm green LED | L2 | 1 |
| 3mm red LED | L1 L4 | 2 |
| 3mm yellow LED | L3 | 1 |
| 3x4 key X-Y Codepad | | 1 |
| 2 pole terminal block | | 3 |
| 3 pole terminal block | | 3 |
| Nut & bolt for 7805 | | 1 set |
| K53 PCB | | 1 |
| 8 pin IC socket | | 1 |
| 28 pin IC socket | | 1 |
| Rainbow ribbon cable, 8 strand | | 12" |
| Floppy disk | | 1 |
| Documentation | | |

For questions email the kit designer Frank Crivelli at

**frank@ozitronics.com**

- - - - - - - - - - - - - - - - - - -

# KIT 53. CODEPAD/ALARM USING 68HC705P9