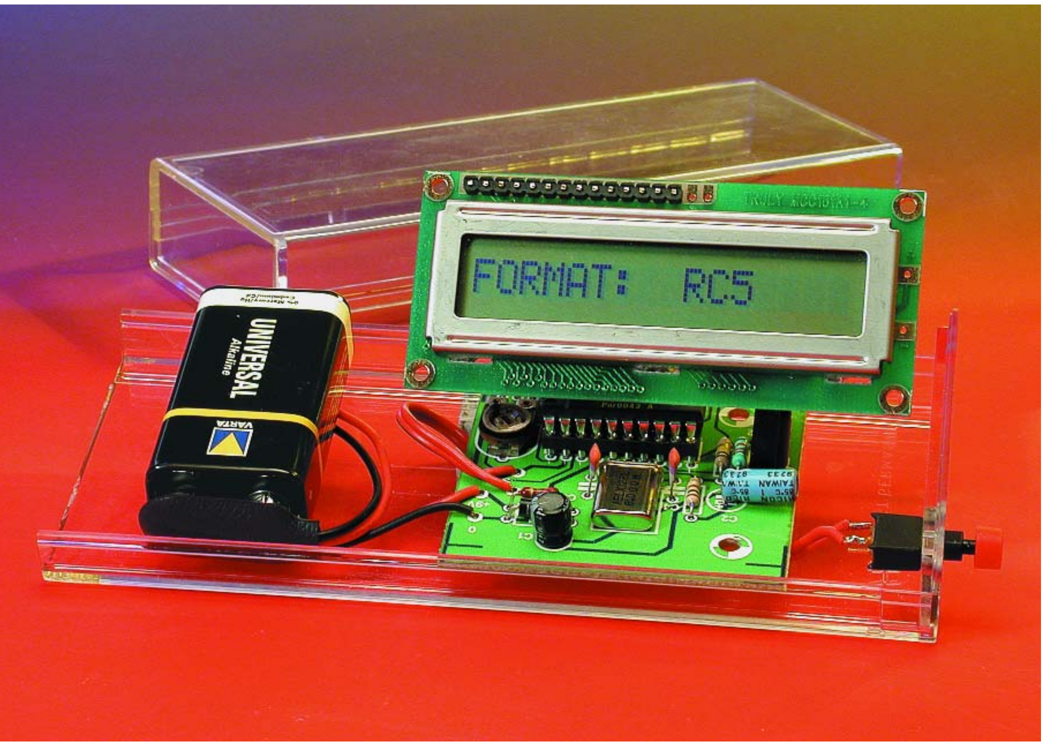
IR Code Analyser

identify remote control codes!

Point any IR remote controller at this nifty device and its in-built microcontroller will quickly analyse the signal and reveal which of eight standard control protocols the controller is using.



Microcontroller

- 4 KByte ROM
- 128 Byte RAM
- 32 Byte EEPROM for user code
- 2.7 to 6 V Supply voltage
- Two 16 Bit Timer/Counter
- Integrated reset
- Internal RC Oscillator (selectable)
- 20 mA drive current from all the port pins
- Maximum 18 I/O-Pins, when the internal reset and RC oscillator is used.
- 2 analogue comparators
- I^2C interface
- Full duplex UART
- Serial In-circuit Programmable

It's a rare piece of entertainment equipment that does not come with its own remote controller. After a few years it's easy to accumulate a drawer-full of (functional) controllers belonging to equipment long since consigned to showrooms at your local household amenity tip.

Equipment manufacturers tend to use different controller protocols so it is rare to find by chance a controller from one manufacturer able to control equipment from another. All of the protocols used should however comply with one of the eight industry standards.

Many electronics enthusiasts can probably think of lots of applications for a discarded remote controller. For example using the well-documented RC5 or RECS80 protocol from Philips a simple receiver/decoder can be built using the SAA3009 or SAA3049 IC from Philips if only you knew details of the controller coding. Similarly if you are looking for a second controller for a piece of equipment its not practical to lug a portable oscilloscope to the local car boot sale to analyse each controller on offer.

This device offers the ideal solution. The chief design criteria for the IR analyser was to produce a neat, portable unit that would quickly enable a check to be made of the output signal of an unknown remote controller. A microcontroller decodes the IR message and displays the important information on an LCD. In use the unknown IR controller should be placed close to the analyser's receiver IC before a command key is pressed. The microcontroller will decode the message and display information indicating the

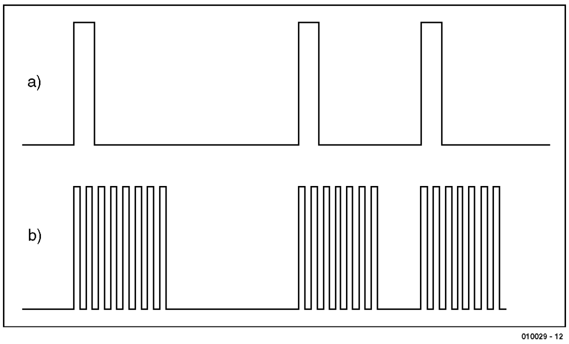


Figure 1. Flash mode signal (a) modulated mode signal at 36 kHz (b).

protocol used and the command sent. A good source of detailed information about the different IR protocol formats can be found in the March and April 2001 edition of *Elektor Electronics*.

This analyser is only suitable for decoding modulated signals in the range of 30 - 40 kHz. Fortunately all the common control protocols use this frequency range.

Some (older) controllers use flash mode where the message information is conveyed by switching on the sender diode for a short period rather than modulating it at 36 kHz (see **Figure 1**). The analyser's IR receiver chip will only respond to modulated signals so that flash mode messages will be ignored.

Operation

Once the analyser is switched on the message 'CODE ANALYSER VER 1' will be displayed and then 'WAITING...' indicating that the analyser is ready and waiting to receive an IR controller message. As soon as a message is received the analyser will display the type of coding sent by the controller and pressing push-button S1 will display further information about the received message.

1. Key press: Address
This hexadecimal value indicates the type of equipment that the controller was originally designed to control. As an example, if the controller uses RC5 protocol format and the address 0 is displayed this indicates that controller will control a TV. Some manufacturers do not use the address field in the message so in this case two lines (—) will be shown on the display.

2. Key press: Command
This hexadecimal value indicates the type of command that was sent in the message. A 10_{hex} (16_{dec}) for example in RC5 coding indicates that the command will increase volume of the controlled equipment.

3. Key press: Complete code
The contents of the entire message are displayed in hexadecimal. The hex value assumes that the data is sent in the order Bit 7 to Bit 0. Some manufacturers (e.g. Sony) however reverse the bit order. In this case the values are corrected before display.

4. Key press: Type of Code
This displays the protocol type of the last received message e.g. RC5, SIRCS or RECS80.

When the display shows UNKNOWN this indicates that the received message does not conform to any of the eight standard protocols. It could also indicate however that the signal was too weak or too distorted. The IR receiver IC2 is optimised for reception of a 36 kHz modulated signal so its sensitivity to some signal protocols at the extremes of the 30-40 kHz band will be greatly reduced (see **Figure 2**). Always place the output diode of the handheld controller as close as possible to IC2 of the analyser to ensure a good signal. LED D2 will blink to indicate signal reception irrespective of the type of protocol so it gives a good indication that the controller is sending out a signal. If the diode does not blink try a new set of batteries in the controller. Items at car boot sales rarely have batteries fitted so it's a good idea to carry a few spares with you.

Some of the more modern remote controllers will send out two messages each time a key is pressed. The first message is in one format while the second message contains the same information in a different format. The analyser will only display the last message format sent.

Hardware

There are no surprises in the circuit diagram of the IC code analyser shown in **Figure 3**.

The infra red signal is detected by IC2. Infineon have ceased production of their receiver IC but any one of the compatible devices specified in the parts list would be suitable. This device contains a highly sensitive infra red receiver, amplifier, filter and demodulator tuned to an IR carrier frequency of 36 kHz. A data sheet for this device can be down loaded from:
(www.infineon.com/cmc_upload/0/000/008/562/sfh5110.pdf).
The inverted demodulated received signal is connected directly to an input port of the microcontroller IC1. IC2 is relatively sensitive to supply rail disturbances so R2 and C2 are used to form a low pass filter, decoupling any interference.

The core of the microcontroller is based on the Intel 51 architecture that lends itself to simple low-cost program development by using any of the available Shareware development tools. The 87LPC764 is produced by Philips and is described as a *Low power, low price, low pin count microcontroller.* Decoding

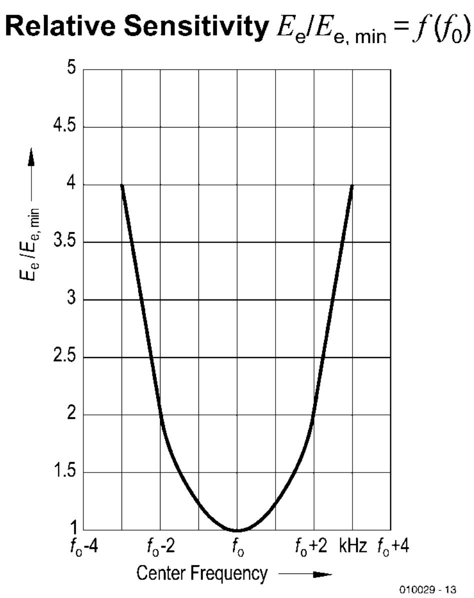


Figure 2. The sensitivity of the IR receiver falls off sharply either side of its centre frequency.

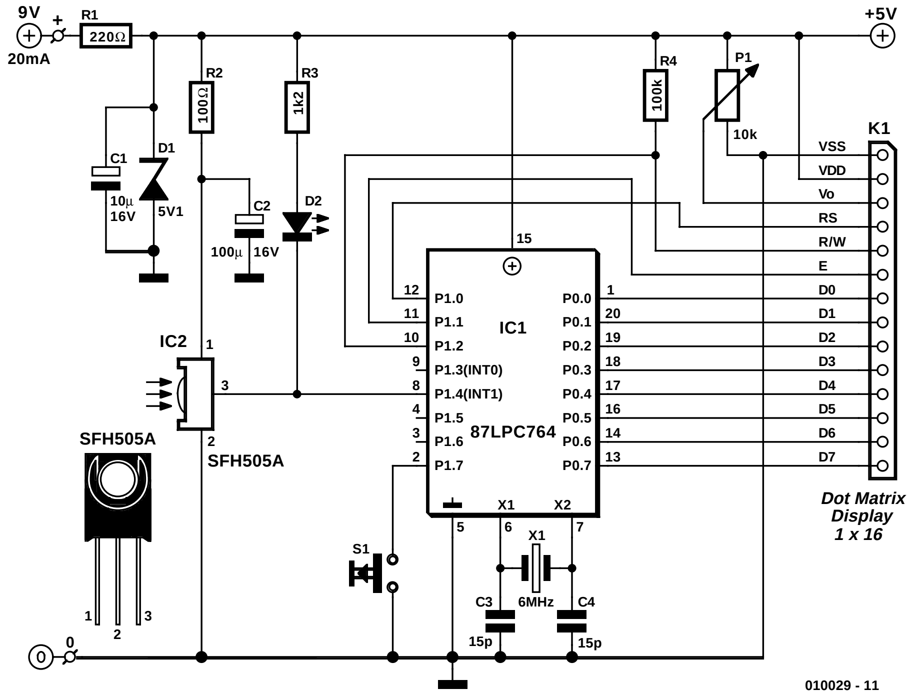


Figure 3. Only two IC's are used in the analyser circuit diagram.

the IR message is relatively processor intensive so a clock rate of 6 MHz is used for the microcontroller giving a machine cycle of 1 µs. The main features of this processor are summarised under the heading 'microcontroller'.

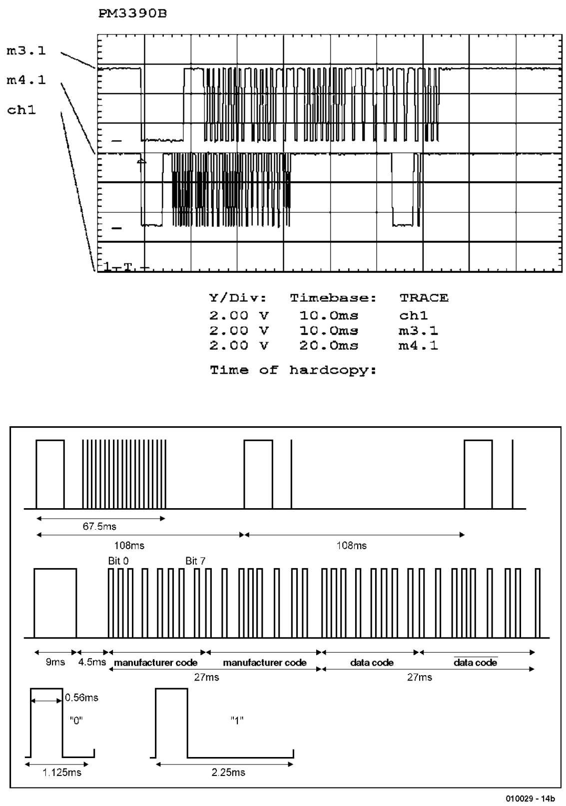


Figure 4. An NEC coded signal (used by Harman/Kardon or Yamaha).

Power for the analyser is provided by a 9 V battery and zener diode D1 regulates this on-board to 5 V. Resistor R1 limits the supply current and overall consumption is approximately 20 mA.

Display

The display is a standard 1x16 LCD (indicating 1 row of 16 characters). This display is a standard item from many manufacturers. It has a built-in controller using a standard command set. All of the compatible displays mentioned in the parts list have identical pin-outs, RAM addresses and multiplex rate. The information generated by this analyser could be displayed on a four line LCD but to reduce costs it was decided to employ a single line display and use a pushbutton to toggle through the information. The display control software is simplified by using eight controller port pins. P1 is a variable resistor, allowing adjustment of the display contrast.

Software

All the relevant data for the following protocols are integrated into the software so that they can be matched to the incoming signal:

JAPANESE NEC RC5 RECS80
SIRCS DENON DAEWOO
MOTOROLA

The software starts its analysis of the incoming signal by measuring the first low-phase. Almost all of the protocol formats have different length start bits so this simplifies the identification process. All of the incoming signal time measurements are made using the microcontrollers timer, this gives a maximum resolution of 1 ms with the 6 MHz clock frequency. After the start bit that usually consists of a low followed by a high of predefined length the microcontroller decides how the incoming data will be stored in the internal registers. The message length will depend on the type of protocol and can be from 11 bits for RCS80 protocol up to 48 bits for Japanese coding. The timing tolerance of the message length is taken into account and the length of each bit is measured and compared to its limit values.

The Software, including source

code is available free of charge from the *Elektor Electronics* website or alternatively can be ordered (at nominal cost) on diskette **010029-11**. If you are planing to modify or customise the software it should be noted that the pulse width of the received IR signals varies quite substantially with received signal strength so the decoding software should be correspondingly tolerant. The following table gives a practical example of pulse tolerances for the signal shown in **Figure** 4 (NEC code from Harman/Kardon or Yamaha):

1. Low phase:	8700 - 9200 ms
1. High phase:	4352 - 4607 ms
Second low phase:	400 - 767ms
Gap between low phases:	
	0-Bit: 400 - 767 ms
	1-Bit: >767 ms

The relatively large tolerances often correspond to the high byte value of the 16 bit timing counter so that the hex calculations can be simplified by ignoring the least significant eight bits of the timing counter.

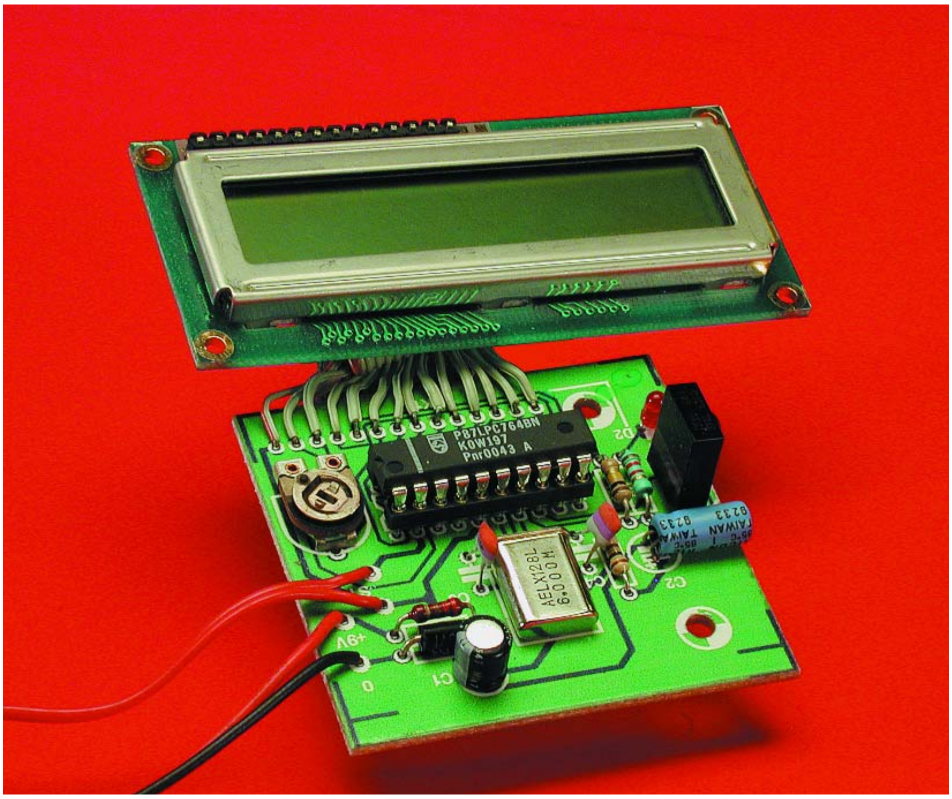
COMPONENTS LIST

Resistors:
R1,R2 = 220Ω
R3 = 1kΩ2
R4 = 100kΩ
P1 = 10kΩ preset H

Capacitors:
C1 = 10µF 16V radial
C2 = 100µF 16V radial
C3,C4 = 15pF

Semiconductors:
D1 = zener diode 5V1, 1 W
D2 = LED, high-efficiency, red
IC1 = 87LPC764 (programmed, order code **010029-41**)
IC2 = SFH505A (TSOP1736, SFH5110-36, PIC26043SM, IS1U60, TFMS5360)

Miscellaneous:
K1 = 14-way flatcable
S1 = push-button, 1 make contact
X1 = 6MHz quartz crystal
Dot matrix display, 1 x 16 characters (MCC161A1-4, MCC161A2-3(Truly), LM161556 (Sharp)
9V battery with clip
Enclosure (Heddic 222)
On/off switch



Construction

The PCB layout is shown in **Figure 5.** The board shown is unfortunately not available ready-made through our Readers Services, so you have to make it yourself. Microcontroller IC1 is fitted to the PCB using a 20 pin DIP socket. The IR receiver (IC2) shown is the Infineon type but other compatible types given in the parts list can be fitted. The compatible types have similar performance but not identical pin-outs so extra mounting pads on the PCB enable these other types to be accommodated (just double-check the pin-outs).

The LCD is connected to the PCB via a short length of ribbon cable. The displays given in the parts list have a 1:1 pin to pin compatibility with the PCB layout. Other types of displays can be used but you will need to study the corresponding data sheet to ensure correct signal connections. Once the circuit has been given a functional test it can be mounted together with the 9 V battery in a suitable case. If a transparent case is used it will only be necessary to drill two holes in the case, one for the on/off switch and one for the mode switch. The display will be visible through the case.

(010029-1)

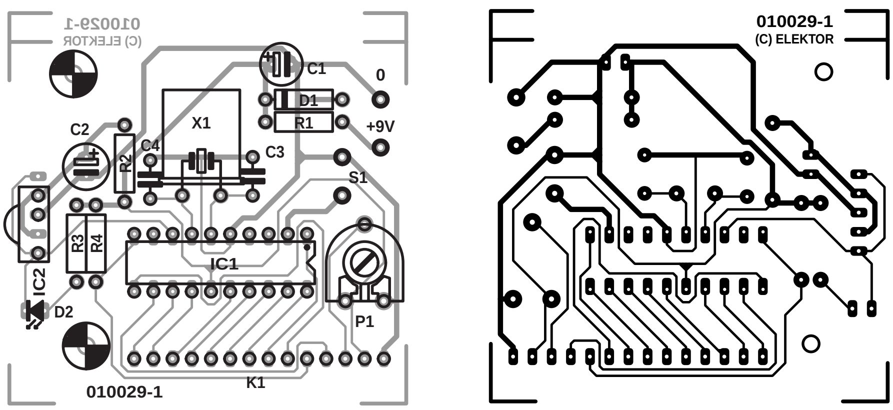


Figure 5. The crystal needs to be mounted flat on the PCB (board not available ready-made).