HandsOn Technology

http://www.handsontec.com creativity for tomorrow's better living...

HandsOn Tech provides a multimedia and interactive platform for everyone interested in electronics. From beginner to diehard, from student to lecturer… Information, education, inspiration and entertainment. Analog and digital; practical and theoretical; software and hardware…

open source hardware

HandsOn Technology support Open Source Hardware(OSHW) Development Platform.

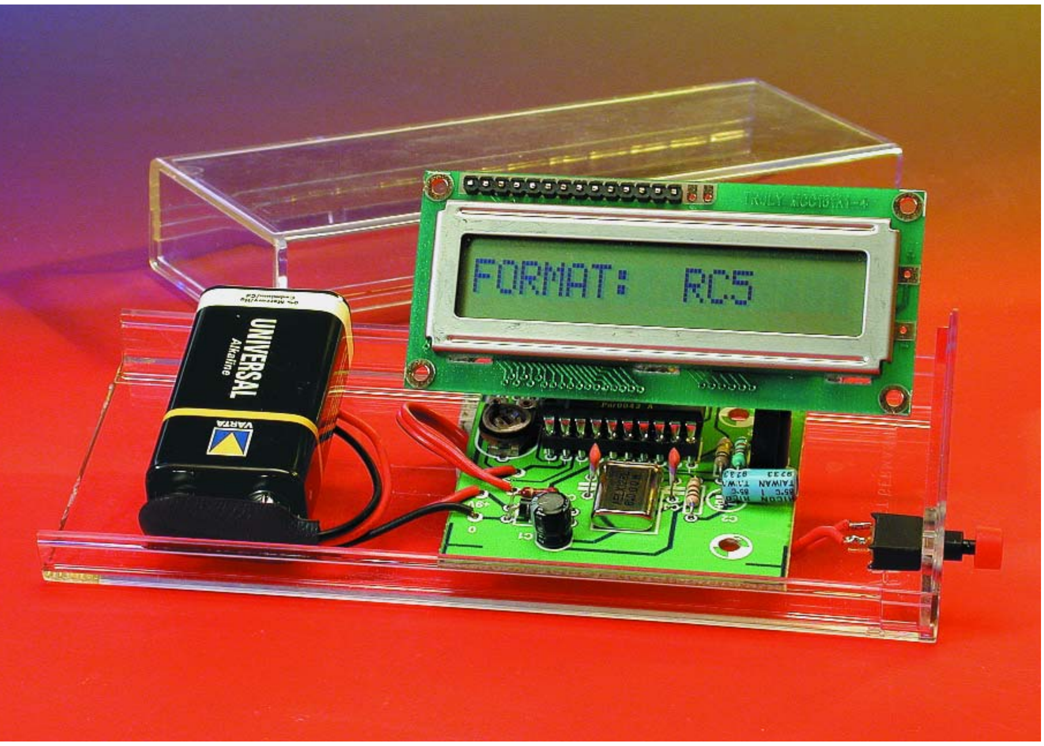
Learn : Design : Share

www.handsontec.com

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

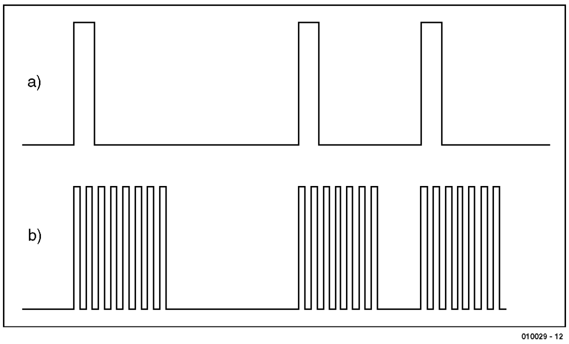


Figure 1. Flash mode signal (a) modulated mode signal at 36 kHz (b).

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

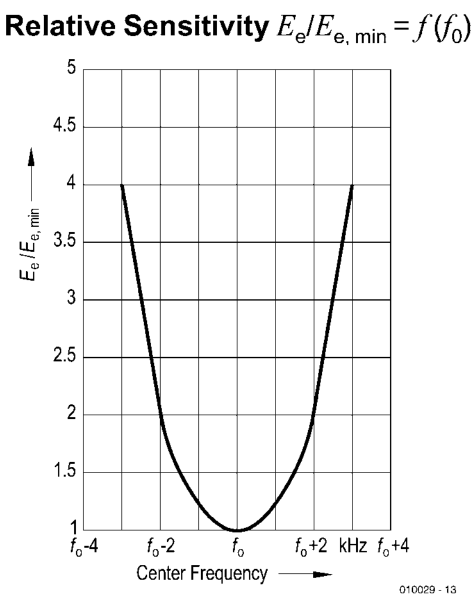


Figure 2. The sensitivity of the IR receiver falls off sharply either side of its centre frequency.

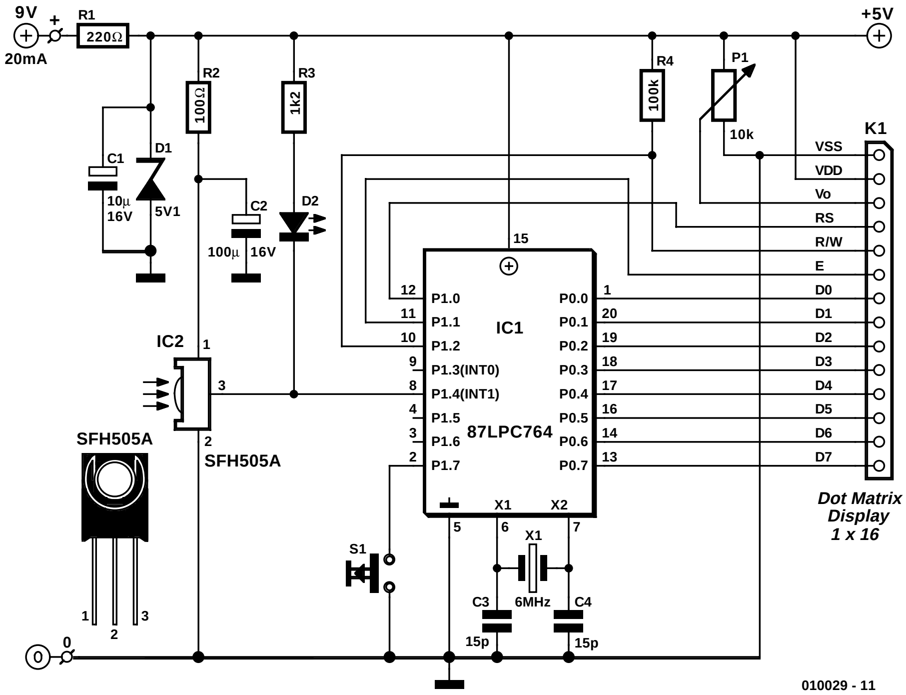


Figure 3. Only two IC's are used in the analyser circuit diagram.

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'.

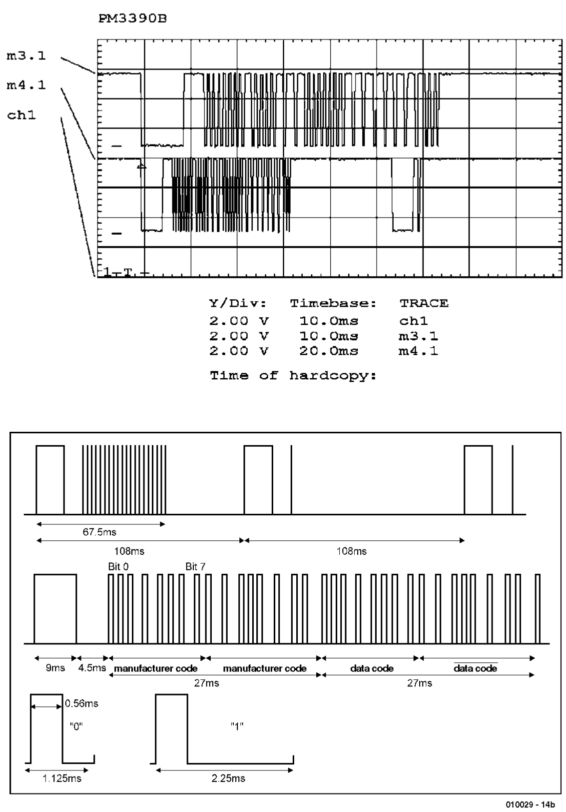


Figure 4. An NEC coded signal (used by Harman/Kardon or Yamaha).

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.

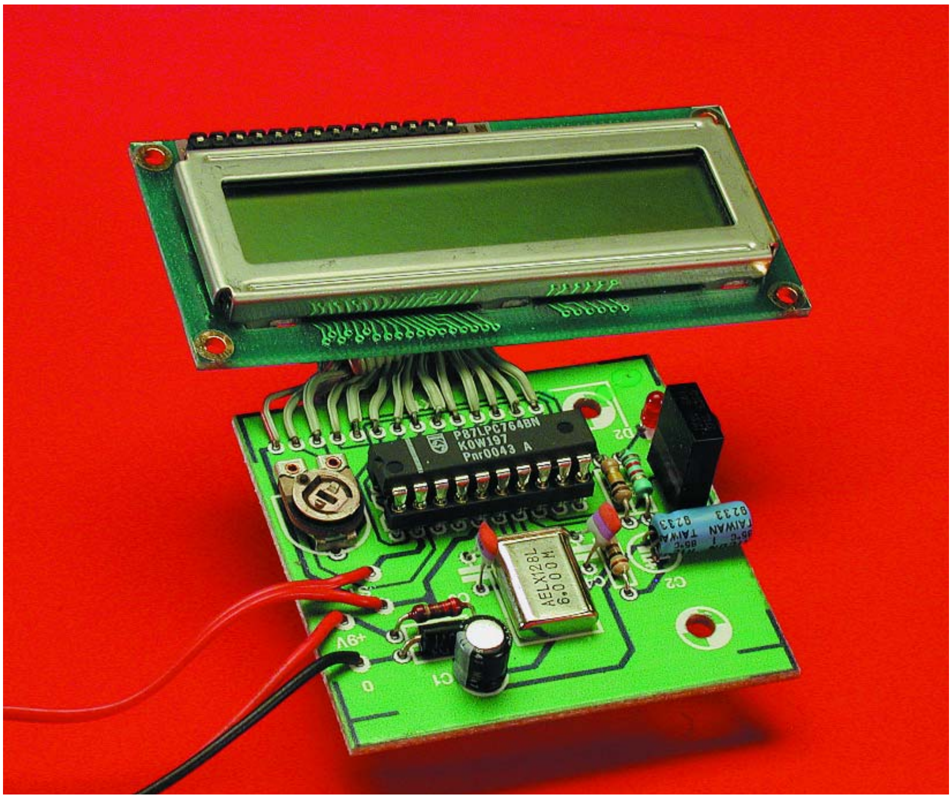
COMPONENTS LIST

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)

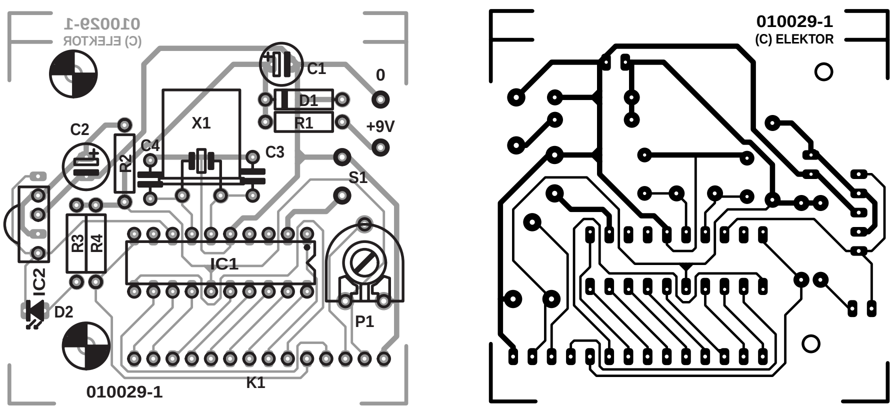


Figure 5. The crystal needs to be mounted flat on the PCB (board not available ready-made).

IR Remote Control Codes (1)

formats, protocols and (in)compatibility

By A.N. Other

There are so many different remote control message formats currently in use that it can all be a bit confusing. If you are experimenting with remote controller IC's then its important to know the different control protocols used by each manufacturer. This article seeks to describe the most popular protocols in current use.

Its almost impossible to buy a TV today that doesn't have a IR remote controller and its only when we temporarily mislay this device that we realise how useful they are. Back in 1975 when the first remote controllers appeared they used ultra sonic signals to send the control information, these were later superseded by the controllers that we are familiar with today using infra red. The infra red devices offer lower production costs, wide operating range and good communication security. A look inside a typical remote controller will show that it consists of only one IC. The IC interprets each key press and sends a coded data signal to the transmitter IR diode. A simple resonator is also used to supply a stable clock. On the receiving side in the equipment being controlled we find a IR detector and demodulator which is normally integrated into the same device. The SFH505-xx family of devices from Siemens contains the receiver, demodulator and output driver so that the received data can be connected directly to a micro controller or control decoder. Unfortunately the actual control protocol used by each manufacturer are mostly incompatible.

The IR message transmitted by the controller is subject to interference from other IR sources in the vicinity. These include heaters, incandescent lamps and other heat generators. One standard method of rejecting this unwanted interference is to modulate each transmitted bit with a stable carrier frequency in the range of 30 to 40 kHz. Another method is the so-called flash mode, this technique is employed by the Plessey MV500 chip (described in the February 1991 edition of *Elektor Electronics*). This method outputs data in the form of 17 µs short flashes of IR light followed by different off periods. Nokia also use this method with their IRT1250 IC. This system has however not gained wide acceptance and the vast majority of remote controllers use the modulation technique.

The accompanying oscilloscope pictures show each of the described transmission formats received by the Temic TFMS5360 receiver IC. This device is optimised for reception of a signal modulated at 36 KHz but can also detect other frequencies albeit with a reduced range. In the upper half of the picture a single telegram is shown and in the lower half a continuous key-press is shown. The output of the IC goes low when the modulated signal is detected.

It is important to note that the equipment manufacturer is entirely at liberty to choose a transmitter clock frequency and as such the timing given here may not be accurate under all conditions. The timings of pulse lengths may also be affected by the sample clock in the TFMS5360 and could have an error of ±160 µs (Temic data sheet).

The communications formats described are the most popular but it does not represent all the possible formats that you are likely to find. Many firms have devised their own control format, sometimes in order to reduce costs or sometimes to incorporate different control features that are not catered for with the existing standards. If you use a mask programmable micro controller for coding and decoding you will be completely at liberty to devise your own protocol which may be more suited to your own particular hardware of software. This method also ensures that a manufacturer will not need to worry about licencing fees or possible patent infringement.

Some modern remote IR controllers transmit the message a number of times using different message formats. For example the controller will first send out the Japanese code and then 50 ms later sends out the same command but this time using RC 5 code. The advantage here for the equipment manufacturer is that for future equipment development you need not wait for a chip manufacturer to produce a controller using a particular IR coding standard. It is now possible to select the best or cheapest integrated equipment controller and be sure that the IR remote controller will produce compatible control signals.

So which manufacturer and which coding system? This article describes some of the most popular IR coding standards currently in use.

(010023-1)

Code	Manufacturer
RECS80	Thomson, Nordmende
NEC	Harman/Kardon, Yamaha, Canon
DENON	Denon
SIRCS	Sony
RC5	Loewe, Philips, Grundig, Marantz
MOTOROLA	Grundig, Kathrein
JAPAN	Panasonic, Loewe
SAMSUNG	Samsung
DAEWOO	Daewoo

RC5 Code

The most widely used coding method for IR control in Europe is the RC 5 code. This was originally developed by Philips and has the capacity to send 2048 different commands. 32 addressable groups each with 64 commands. Each piece of equipment has its own address so that for example adjusting the volume of your audio system will not affect the sound level of your TV. One complete message has a length of 14 bits and is composed of the following bits:

– 2 Start bits to control the AGC levels (Auto gain control) in the receiver IC.
– 1 Toggle bit indicates that a new key is pressed.
– 5 System address bits
– 6 command bits

The toggle bit changes its value every time a new key is pressed and is used to tell the difference between pressing the key again and holding the key down. The five address bits follow the toggle bit and indicate which piece of equipment is being controlled. Lastly the six command bits contain the control information.

RC5 code employs biphase encoding, One bit of data is represented by two half bits. A Low/High combination of these bits indicates a data '1' whereas a High/Low combination indicates a data '0' The length of each bit is 1.778 ms, and a complete message is 24.889 ms long.

The RC5 code is probably the best documented protocol for IR control and particularly interesting are the two free system addresses 7 and 13 these are not allocated to any particular equipment type but are reserved for experimental purposes. Typical IC's used for this message format are:

Transmitter:
SAA3006, SAA3010 (Philips)
HT6230 (Holtek)

Receiver:
SAA3009, SAA3049 (Philips)

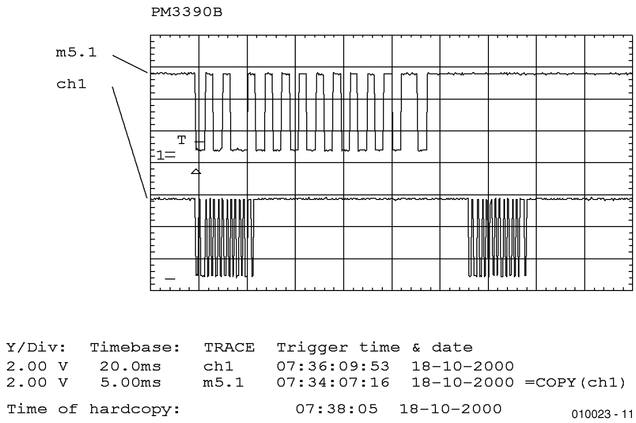


Figure 1. RC5 code at the output of the receiver IC TFMS5360.

Table 1 shows in decimal the correspondence between the equipment and command codes used for this format.

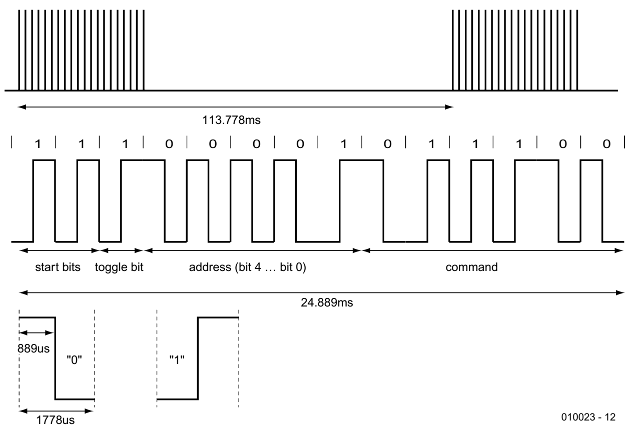


Figure 2. RC5 code message format (Address 1, command 28 shown).

The RC5 codes:

Address	Equipment
0	TV1
1	TV2
2	Videotext
3	Expansion for TV1 and TV2
4	Laser Vision Player
5	Video recorder1 (VCR1)
6	Video recorder2 (VCR2)
7	Reserved
8	SAT1
9	Expansion for VCR1 and VCR2
10	SAT2
11	Reserved
12	CD Video
13	Reserved
14	CD Photo
15	Reserved
16	Audio preamplifier1
17	Tuner
18	Analogue cassette recorder
19	Audio preamplifier2
20	CD
21	Audio Rack or Aufnahmegerät
22	Audio Satellite receiver
23	DCC Recorder
24	Reserved
25	Reserved
26	writable CD
27...31	Reserved

keycodes:

Code	Key Function
0	0
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
16	Volume +
17	Volume –
18	Brightness +
19	Brightness –
20	Colour saturation +
21	Colour saturation –
22	Bass +
23	Bass –
24	Treble +
25	Treble –
26	Balance right
27	Balance left
63	System select
71	Dim local display
77	Linear function increment
78	Linear function decrement
80	Step up
81	Step down
82	Menu on
83	Menu off
84	Display A/V system status
85	Step left
86	Step right
87	Acknowledge
88	PIP on/off (Pay TV channel + for system 3)
89	PIP shift (Pay TV channel - for system 3)
90	PIP / main swap (Radio channel + for system 3)
91	Strobe on/off (Radio system – for channel 3)
92	Multi strobe (Date + for system 9)
93	Main frozen (Date – for system 9)
94	3/9 multi-scan (Start time + for system 9)
95	PIP select (Start time – for system 9)
96	Mosaic/multi-PIP (Record program + for system 9)
97	Picture DNR (Record program – for system 9)
98	Main stored (Alternate channel for system 9)
99	PIP strobe (Stop time + for system 9)
100	Recall main picture (Stop time – for system 9)
101	PIP freeze
102	PIP step up +
103	PIP step down –
118	Sub mode
119	Options sub mode
123	Connect
124	Disconnect

Special commands for equipment addresses 0 und 1 (TV1 / TV2):

Code	Key Function
10	1/2/3 digits / 10
11	Freq./prog./ch./11
12	Standby
13	Mute/de-mute
14	Personal pref.
15	Display
28	Contrast +
29	Contrast –
30	Search +
31	Tint/hue –
32	Ch./prog. +
33	Ch./prog. –
34	Altern./ch.
35	? language
36	Spatial stereo
37	Stereo/mono
38	Sleep timer
39	Tint/hue. +
40	RF switch
41	Store/execute/vote
42	Time
43	Scan fwd./increm.
44	Decrement
46	Sec con/menu
47	Show clock
48	Pause
49	Erase/correct
50	Rewind
51	Go to
52	Wind
53	Play
54	Stop
55	Record
56	External 1
57	External 2
59	Advance
60	TXT sub-mode/12
61	Sys. Standby
62	Crispener
70	Speech/music
79	Sound scroll
104	PIP size
105	Pic. Scroll
106	Act. On/off
107	Red
108	Green
109	Yellow
110	Cyan
111	Index/white
112	Next
113	Previous
122	Store open/close
126	Movie expand
127	Parental access

SIRCS/Control S Code

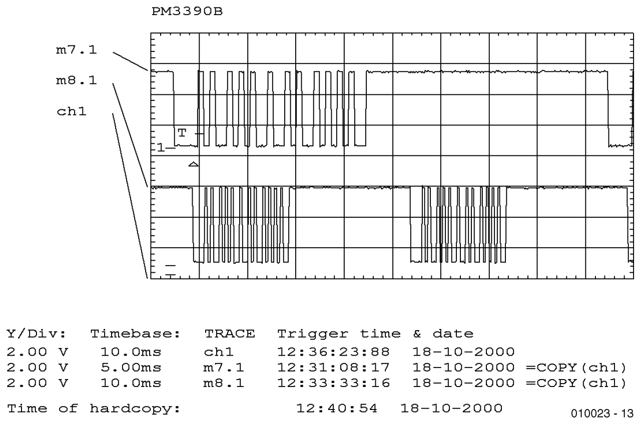


Figure 3. SIRCS code at the output of the TFMS5360 receiver IC.

A message sent using the SIRCS or CNTRL S protocol from Sony consists of twelve to twenty bits. Five to thirteen of these bits is used for the address field and seven bits for the key code.

A Start bit (2.4 ms) is sent followed by a 0.6 ms space or pause. Next comes the data. A '1' is represented by a 1.2 ms ON or mark followed by a 0.6 ms OFF or pause. A '0' is represented by a 0.6 ms ON and a 0.6 ms OFF. A typical message is shown in **Figure 4**. The message is sent a minimum of twice (five times for a camcorder). The message is discarded if an error is detected. SIRCS message coding is identical to CNTRL S, but SIRCS modulates the code at 40 kHz ready to be sent to an IR diode. CNTRL-S is the baseband (unmodulated) signal and is used between equipment where a communications cable is fitted. Sony produce the following IC:

Transmitter:
KIE RA275 S42

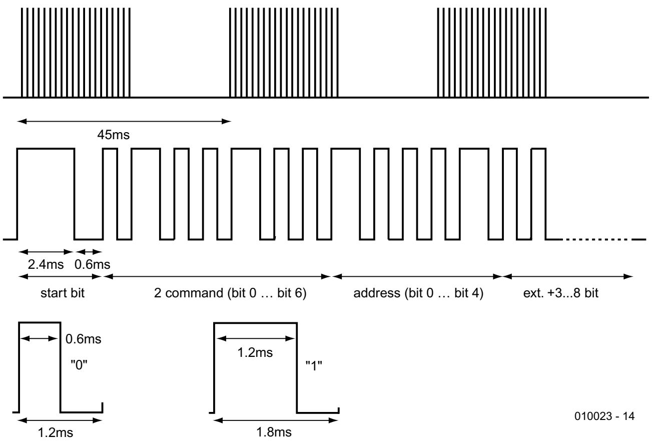


Figure 4. CNTRL-S and SIRCS message format.

The Sony codes:

Equipment address codes (decimal):

Address	Equipment type
1	TV
2	VTR1
4	VTR2
6	Laserdisc
7	VTR2
11	VTR3
12	Surround sound processor
16	Cassette deck, tuner
17	CD Player
18	Equaliser
164	TV digital effects (8 bit device code)

keycodes:

code	key Function
000	1 button
001	2 button
002	3 button
003	4 button
004	5 button
005	6 button
006	7 button
007	8 button
008	9 button
009	10 button/0 button
011	Enter
016	Channel up
017	Channel down
018	Volume up
019	Volume down
020	Mute
021	Power
022	Reset TV
023	Audio mode: mono/SAP/stereo
024	Picture up
025	Picture down
026	Colour up
027	Colour down
030	Brightness up
031	Brightness down
032	Hue up
033	Hue down
034	Sharpness up
035	Sharpness down
036	Select TV tuner
038	Balance left
039	Balance right
041	Surround on/off
042	Aux/Ant
047	Power off
048	Time display
054	Sleep timer
058	Channel display
059	Channel jump
064	Select input video1
065	Select input video2
066	Select input video3
074	Noise reduction on/off
078	Cable/broadcast
079	Notch filter on/off
088	PIP channel up
089	PIP channel down
091	PIP on
092	Freeze screen
094	PIP position
095	PIP swap
096	Guide
097	Video setup
098	Audio setup
099	Exit setup
107	Auto program
112	Treble up
113	Treble down
114	Bass up
115	Bass down
116	+ key
117	– key
120	Add channel
121	Delete channel
125	Trinitone on/off
127	Displays a red RtestS on the screen

RECS80 Code

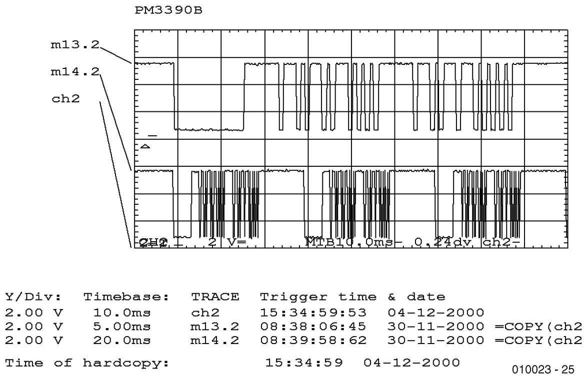


Figure 5. RECS80 code at the output of the TFMS5360 receiver IC.

The RECS80 code from Philips a pulse position modulation technique. With this system a fixed length pulse of light is followed by a variable length space. The space timing conveys the data. There are 1280 possible codes divided into 64 commands and 20 subsystems. A subsystem is simply the type of equipment being controlled i.e. a TV or a VCR. A message is composed of 11 bits. The first two bits are the toggle bits followed by three sub-system address bits and six data bits, these indicate which key was pressed. The toggle bits are incremented if a key is released for a minimum time but will remain unchanged within a multiple key-stroke sequence. If the transmitter is configured to operate in modulated mode the first toggle bit is replaced by a REF bit of fixed duration.

In the lower trace of Figure 15 not all the data is shown because of the low sampling rate of the scope some of the bits have been missed. The RECS80 protocol encodes the data by variable length spaces between constant width ON pulses (140.8 µs). If the transmitter is configured to modulation mode this time period will be represented by a burst of carrier frequency. If configured to flash mode the IR transmitter will be flashed on at this time. A '0' has a space of 5.06 ms while a '1' has a space of 7.60 ms (derived from a 455 kHz resonator in the remote control transmitter). Although the length of the data packet is dependent on the commands sent, the time between two messages is fixed at 121 ms. The modulation frequency is 38 kHz.

Typical ICs for remote control:

Transmitter:
SAA3004, SAA3007 and SAA3008 (Philips)
M3004, M3005, M3006 (ST Microelectronics)

Receiver:
SAA3009, SAA3049 (Philips)

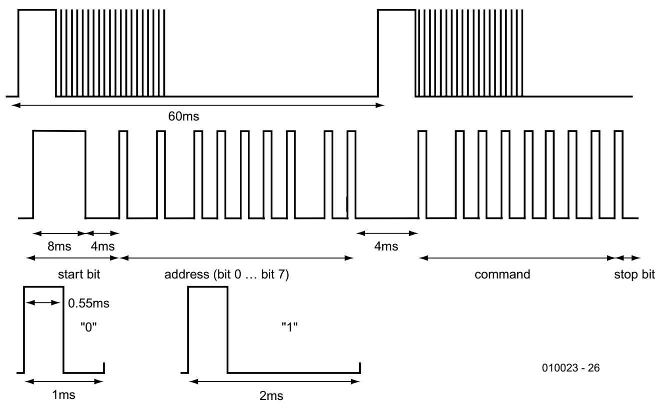


Figure 6. RECS80 code message format

Internet Links

Nec Format:
http://www.princeton.com.tw/spechtml/remote/2221.htm

Philips semiconductor:
http://www.semiconductors.com
www-us.semiconductors.com/pip/SAA3049AP

Motorola Home-Page:
http://motorola.com
Motorola-Format:
http://holtek.com

Samsung Home-Page:

http://www.intl.samsungsemi.com/System_LSI/Microcontroller/Product_Guide/Microcontroller/product_guide.html

Next month:

– NEC
– DENON
– MOTOROLA
– JAPAN
– SAMSUNG
– DAEWOO

IR Remote Control Codes (2)

Part 2

By A.N. Other

In this second and concluding instalment we look at the structure of the infra-red remote control code systems proposed by Denon, NEC, Motorola, Samsung and Daewoo, as well as a format which is generally referred to as 'Japanese Code'.

Correction to Part 1 (March 2001)

In last month's instalment, the illustrations with the RECS80 format description show the Daewoo format. The correct drawings are shown below.

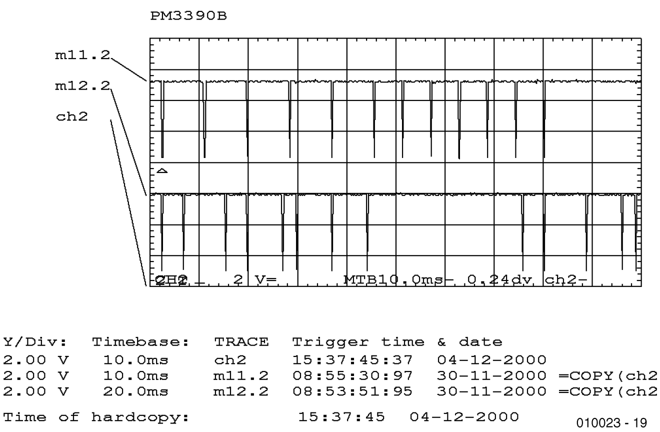


Figure 5. RECS80 code at the output of the TFMS5360 receiver IC.

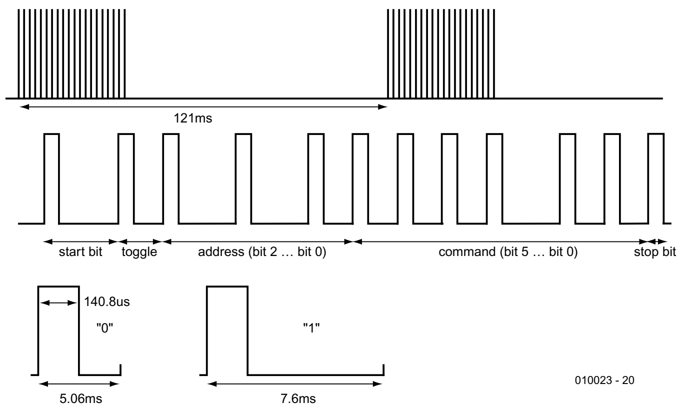


Figure 6. RECS80 code message format

Denon Code

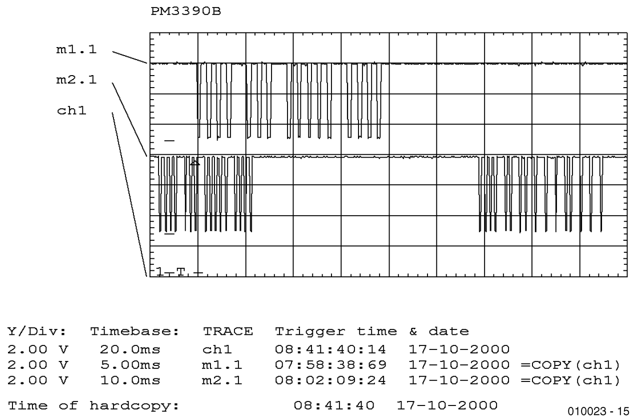


Figure 7. Denon code at the output of the TFMS5360 receiver IC (CD Player)

A Denon coded message consists of 15 information bits. The first five bits are the equipment address field while the remaining ten bits contain the command.
The modulation frequency is 32 kHz and the bit coding is:

1: 275 µs mark, 1900 µs space
0: 275 µs mark, 775 µs space

To reduce the effects of interference the message is sent twice, the second time 65 ms after the first. During the second time the command field bits are inverted. The receiver will only accept commands when the second message is identical to the first after the command field bits are inverted. The address field is always sent uninverted. Bit 16 is a stop bit.

There are currently no dedicated chips to implement this code. Transmitters and receivers can be built from mask programmable micro controllers e.g. the Mitsubishi M50560.

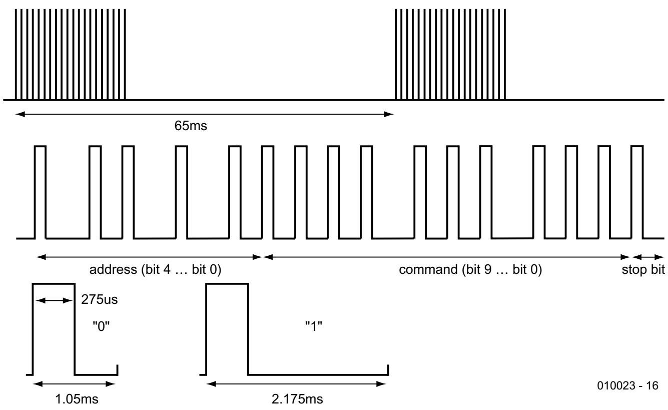


Figure 8. Denon code message format

NEC Code

The NEC code operates with a carrier frequency of 38 kHz and uses pulse position modulation (PPM). Transmission begins with a 9 ms long start bit, followed by a 4.5 ms space. The message information is contained in the following 32 bits which consists of a 16 bit manufacturer field and a 16 bit command field. The 8 bit wide data is sent twice, the second time inverted. A complete message is 67.5 ms long. Each bit is sent using the following format:
1: 0.56 ms Pulse, 1.69 ms space
0: 0.56 ms Pulse, 0.565 ms space
A new message is sent 108 ms after the start of the preceding message. A special current saving feature is implemented if a key is held down on the controller. In this case the message consists of a 9 ms start bit followed by a 2.25 ms space and a 0.56 ms pulse.

Sanyo supply ICs that generate codes using this format but have a 13 bit manufacturers code.

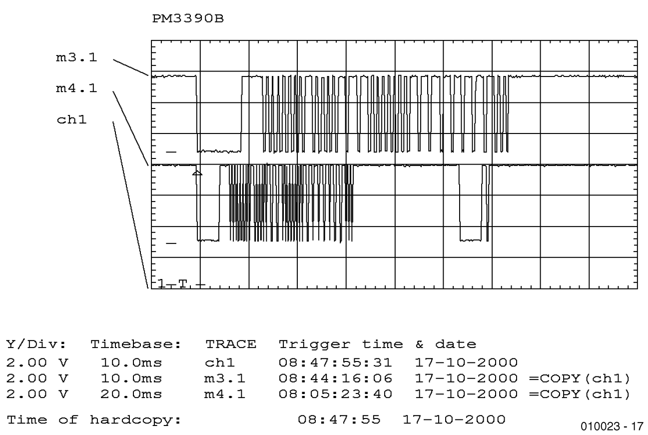


Figure 9. NEC code at the output of the TFMS5360 receiver IC

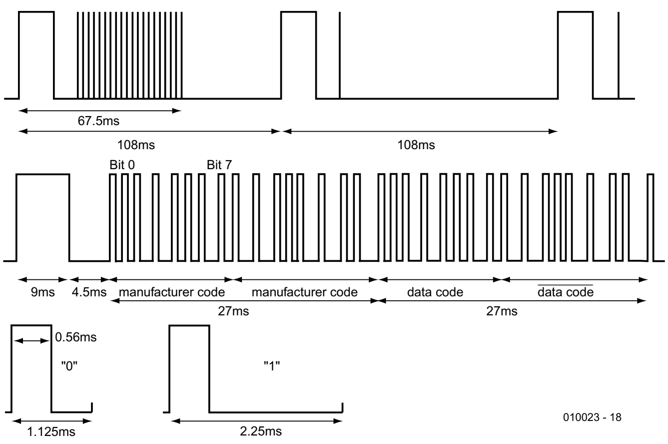


Figure 10. NEC code message format

Typical transmitter ICs remote controls:
PTPT2221, PT2222 (Princeton)
uPD6120, uPD6121 (NEC)
LC7461M, LC7462M (Sanyo)

Motorola Code

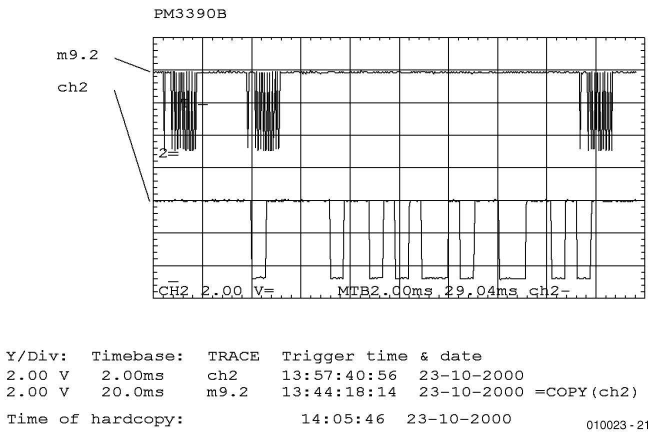


Figure 11. Motorola code at the output of the TFMS5360 receiver IC.

The Motorola code consists of a 9 bit data word. Biphase modulation is used similar to the RC5 standard but with Motorola a ‘0‘ is represented by a 512 µs pause followed by a 512 µs long high and a ‘1‘ is represented by a 512 µs high followed by a 512 µs long pause. This is opposite to the data representation found in the RC5 code. A 32 KHz carrier frequency is used.

A typical telegram has a message start header consisting of nine consecutive ‘1‘s followed by the key code of the pressed key (repeated for as long as the key is pressed) and ended by sending nine consecutive ‘1‘s. A brief key press at the controller will cause three messages to be transmitted.

Each message consists of a pre-bit, a pre-bit-pause, a start-bit and nine data bits. The pre-bit and the start-bit are always a logical 1. The pre-bit is used by the receiver to set the AGC gain level of the IR receiver.

The IC MC144105 chip is typically used to implement a remote control system using the Motorola code.

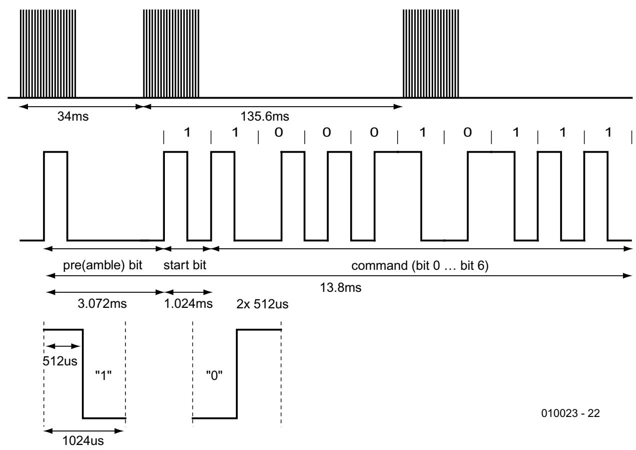


Figure 12. Motorola code message format.

Japanese Code

Similar to the way in which the RC5 code has been standardised in Europe, the Japanese Association for Electric Home Appliances has produced a standard for IR control. This system is called (uninspiringly) the ‘recommended standard for infrared remote controls‘.

The code is used by a number of manufacturers and has a message length of 48 Bits split into the following fields:

Manufacturers Code (16 bit)
These 16 Bits comprise the unique code for each manufacturer and are registered by the standards organisation. This code is programmed into the IC mask.

Parity Code (4 bit)
These four bits detect data corruption in the message.

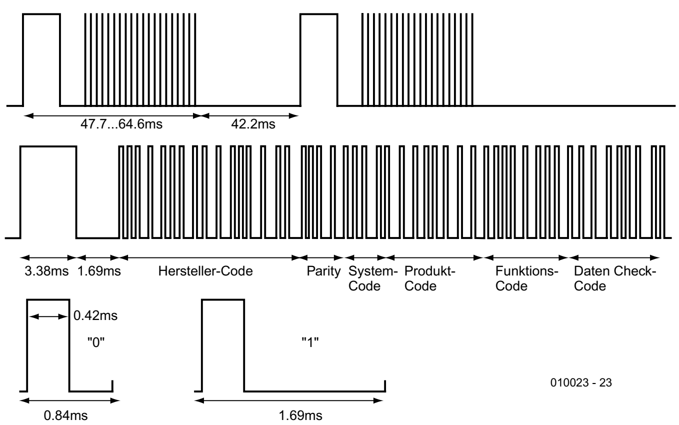


Figure 13. Japanese code message format.

System Code (4 bit)
The four system code bits are pre-programmed into the IC during manufacture.

Product Code (8 bit)
The 8 Bit Product code is made up of two mask programmed bits and 6 user wired bits. The six hardwired bits determine the equipment address.

Functions Code (8 bit)
The 8 bit function code is the value of the key pressed.

Data check code (8 bit)
These eight bits are used to detect data corruption. The system, product and function codes are passed through an algorithm to generate this check code.

A digital zero is represented by a '1' of 0.42 ms followed by a '0' of 1.27 ms. A digital one is represented by a '1' of 0.42 ms followed by a '0' of 0.42 ms.

Sanyo produce the LC7465M this device interfaces to a keypad and sends IR messages using this format.

SAMSUNG Code

The Samsung code consists of a start bit followed by a 12 Bit manufacturers code and an 8 bit command code. The message is always sent a minimum of twice.

A digital zero is represented by a '1' of 0.56 ms followed by a '0' of 0.56 ms. A digital one is represented by a '1' of 0.56 ms followed by a '0' of 1.69 ms. A continuous key press will cause the message to be repeated every 60 mS. The carrier frequency used for this code is 38 kHz.

There are no dedicated transmitter IC's for the Samsung system. A microcontroller is generally used to generate the code e.g. the data sheet of the KS51840 from Samsung gives an application for remote control use.

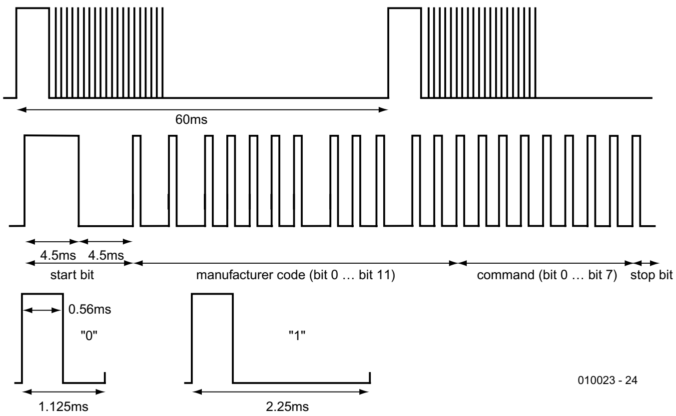


Figure 14. Samsung code message format.

Daewoo Code

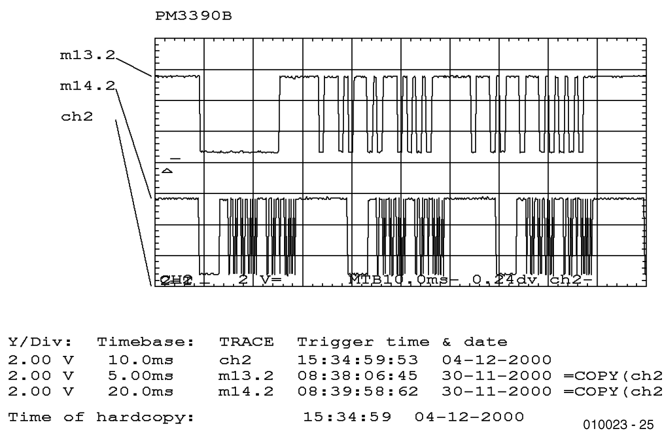


Figure 15. Daewoo code at the output of the TFMS5360 receiver IC

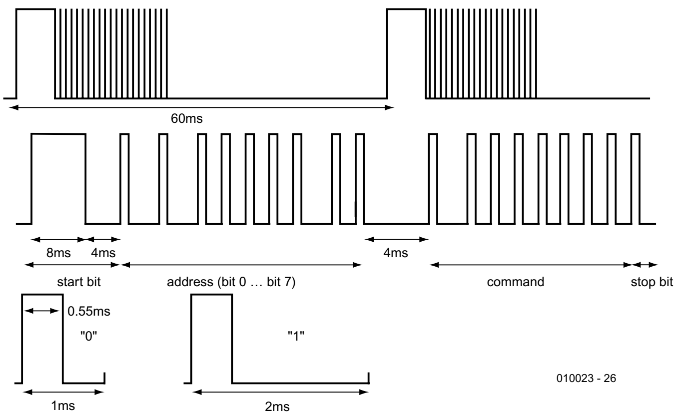


Figure 16. Daewoo code message format.

A digital zero is represented by a '1' of 0.55 ms followed by a '0' of 0.45 ms. A digital one is represented by a '1' of 0.55 ms followed by a '0' of 1.45 ms. This uses a carrier frequency of 38 kHz. A 4 ms space separates the address field from the command field.

(010023-2)