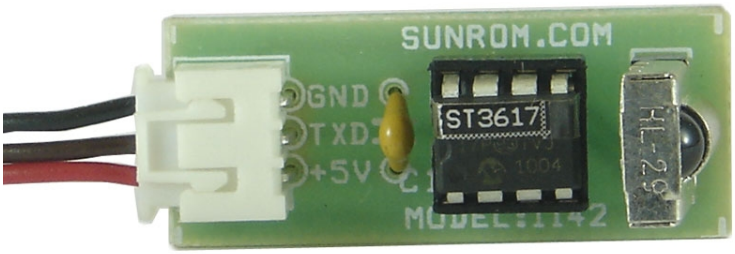
SUNROM TECHNOLOGIES
Your Source for Embedded Systems

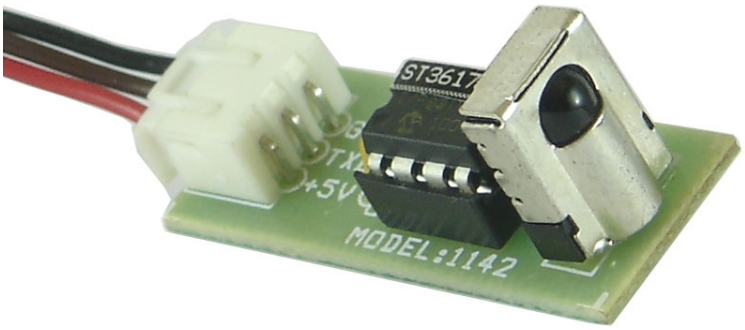
Email: info@sunrom.com
Visit us at http://www.sunrom.com

Document: Datasheet Date: 2-Jan-11 Model #: 1142 Product's Page: www.sunrom.com/p-1007.html

RC5 Remote Decoder Serial Out

The RC5 remote decoder board is based on the IC ST3617 which decodes the received remote control data and output 16 bit serial data output. The decoded data contains various information like Toggle Bit, Address of Remote and Command Key Pressed. This decoded information from transmitter can be used in various ways to make any remote controlled application. The transmitter should be a RC5(Philips) protocol type normally found in house hold TV remote controls. Custom RC5 transmitter can be designed using SAA3010 IC.





Features

- Simple serial data output
- TTL level output compatible with any microcontrollers pins directly
- Decodes any RC5(Philips) type remote control

Specification

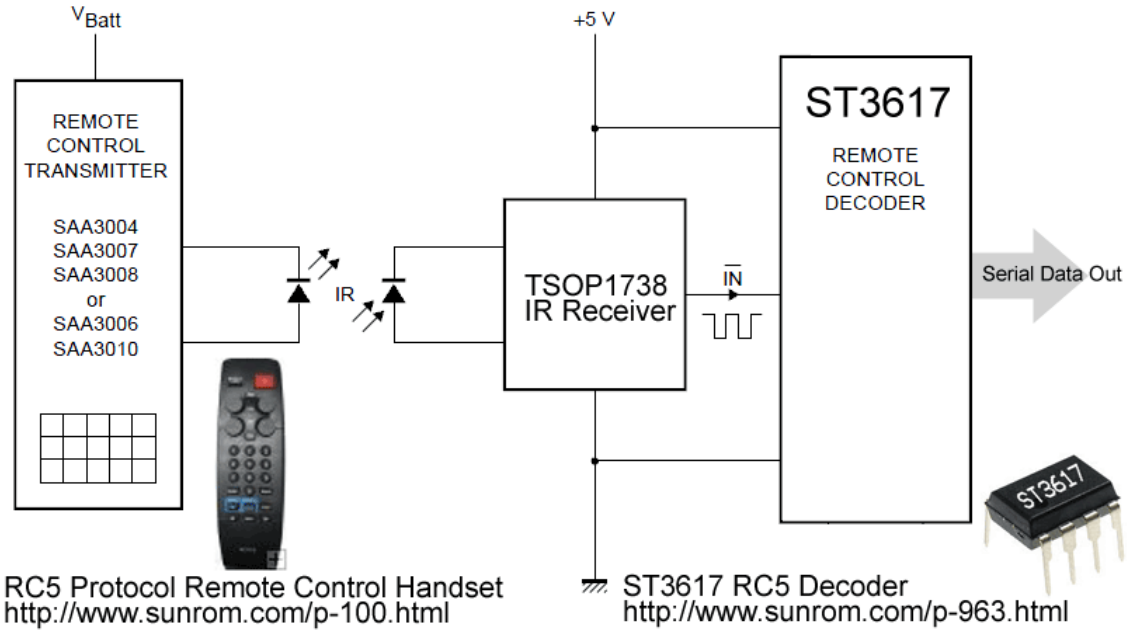
Parameter	Value
Working Voltage	5V DC
Current	10mA
Serial UART Interface	9600 bps, 8 bit data, no parity, 1 stop bit at 5V (For 3V systems insert a 1K resistor in series to TXD to drop the voltage at 3V)
Board Dimensions	36 mm x 15 mm

Data Interface for RC5 Decoder

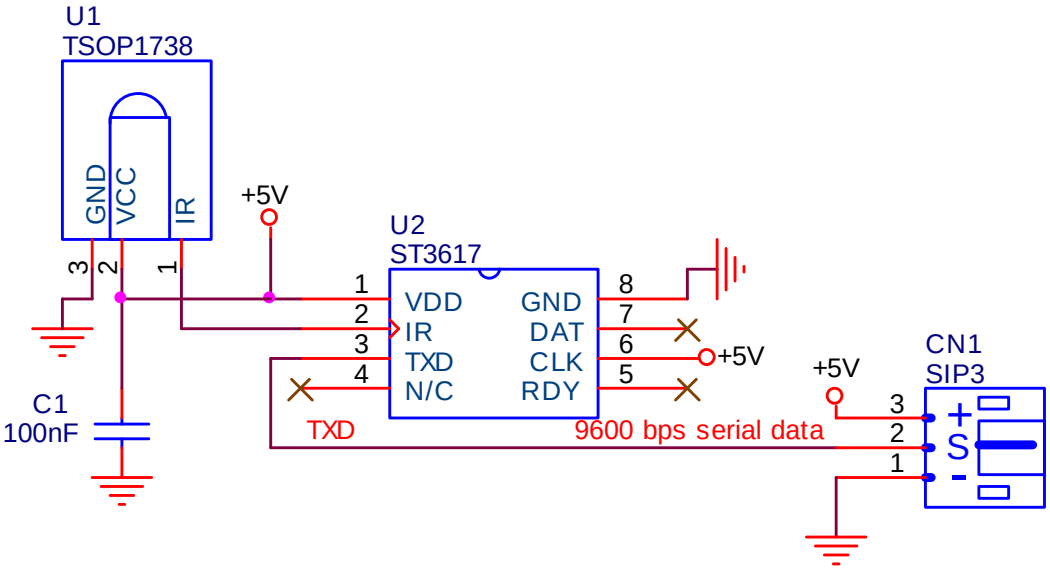
For each output the data is two bytes long containing total 16 bits RC5 data. Output of serial data at 9600 bps is particular useful when you have a dedicated serial input pin available on your application microcontroller to get the 2 bytes of data. You can also use the serial data to interface to PC using MAX232 level convertor for serial port or use USB-TTL chip to get a virtual serial port on PC to which many software like Hyperterminal can be connected. Custom software can also be made to monitor the incoming data and develop applications further.

Block Diagram

ST3617 needs only one component to work that is IR receiver like TSOP1738 or similar. The data is output as simple 2 bytes of serial data consisting of 16 bits of information for each key press on the remote.



Board Schematic



Sunrom Technologies	http://www.sunrom.com	
Title	RC5 Remote Code Receiver/Analyser	
Code	1142	Rev 1
Date:	Tuesday, November 23, 2010	Sheet 1 of 1

Philips RC5 Protocol

Let us review the RC5 protocol to understand the RC5 decoder IC ST3617 better. The RC5 code from Philips is possibly the most used protocol by hobbyists, probably because of the wide availability of cheap remote controls.

The protocol is well defined for different device types ensuring compatibility with your whole entertainment system.

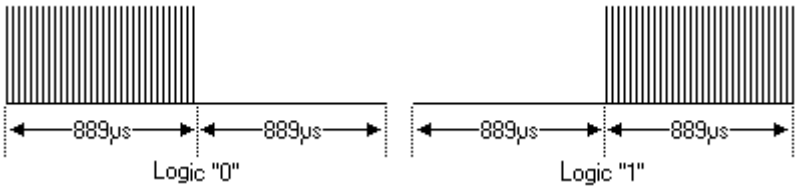
RC5 Protocol Remote Control Handset
http://www.sunrom.com/p-100.html

Features of Protocol

- 5 bit address and 6 bit command length
- Bi-phase coding (aka Manchester coding)
- Carrier frequency of 36kHz or 38kHz
- Constant bit time of 1.778ms (64 cycles of 36 kHz), Different timing for 38Khz, Should be adjusted in decoder part by monitoring first two bits.
- Manufacturer Philips

Modulation

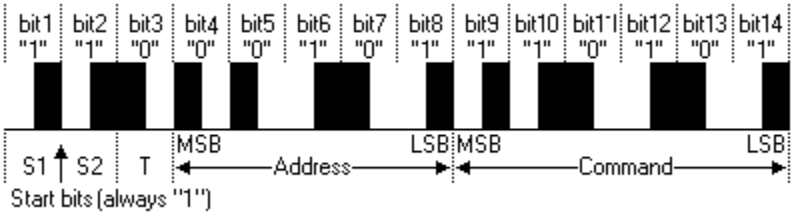
RC5 Modulation The protocol uses bi-phase modulation (or so-called Manchester coding) of a 36kHz IR carrier frequency. All bits are of equal length of 1.778ms in this protocol, with half of the bit time filled with a burst of the 36kHz carrier and the other half being idle. A logical zero is represented by a burst in the first half of the bit time. A logical one is represented by a burst in the second half of the bit time. The pulse/pause ratio of the 36kHz carrier frequency is 1/3 or 1/4 which reduces power consumption.



Protocol

The drawing below shows a typical pulse train of an RC-5 message. This example transmits command $35 to address $05.

RC-5 Pulse Train



The first two pulses are the start pulses, and are both logical "1". Please note that half a bit time is elapsed before the receiver will notice the real start of the message.

The 3rd bit is a toggle bit. This bit is inverted every time a key is released and pressed again. This way the receiver can distinguish between a key that remains down, or is pressed repeatedly.

The next 5 bits represent the IR device address, which is sent with MSB first. The address is followed by a 6 bit command, again sent with MSB first.

A message consists of a total of 14 bits, which adds up to a total duration of 25 ms. Sometimes a message may appear to be shorter because the first half of the start bit S1 remains idle. And if the last bit of the message is a logic "0" the last half bit of the message is idle too.

As long as a key remains down the message will be repeated every 114ms. The toggle bit will retain the same logical level during all of these repeated messages. It is up to the receiver software to interpret this auto repeat feature.

Output Data format

Output from ST3617 is in two bytes, thus making total 16 bits of data, let us see meaning of each bit

High Byte - First

Bit Position->	15	14	13	12	11	10	9	8
Value	0	0	1	1	T	A4	A3	A2
	Always 0	Always 0	Always 1	Always 1	Toggle Bit 0 or 1	Address	Address	Address

Low Byte - Second

Bit ->	7	6	5	4	3	2	1	0
Value	A1	A0	D5	D4	D3	D2	D1	D0

A4-A0 = RC5 address of remote control, For TV remote this is zero.
D5-D0 = RC5 command for each keypress at remote control. For Key 1 its 1, Key 2 = 2 and such, Find a table on last of this datasheet showing key value for each key press.

Example
For example pressing Key 1 on remote control can output 0x3001 where 0x30 is high byte and 0x01 is low byte.

If we interpret, in terms of RC5 data we get below

High: 0x30 in binary is

Bit ->	15	14	13	12	11	10	9	8
Data->	0	0	1	1	0	0	0	0
					Toggle	A4	A3	A2

Low: 0x01 in binary is

Bit ->	7	6	5	4	3	2	1	0
Data->	0	0	0	0	0	0	0	1
Value	A1	A0	D5	D4	D3	D2	D1	D0

Note the Toggle is zero in this example, It can also be one, in this case you can get 0x3801 for key1 value. If you keep the Key1 pressed, the next output will have toggle value same as zero. If you leave Key1 and then press again, the toggle value will be one. Therefore toggle tells you if user is keeping the key press or left the key once and pressed again. This is particular useful if you are implementing Toggle output like Relay ON and OFF logic.

The Address bits A0 to A4 are zero since TV remote has zero RC5 address.
The Command bits D0 to D5 are 0x01 since User has pressed Key1 and value of key1 is 1.

Serial Data Output Format

When you see data output from chip in serial at 9600 baud rate, you will get total six bytes output as each key press in ASCII format so you can view it on screen.

Let us see what data output you will get in serial mode. The last two bytes in serial mode are new line characters so that when you see this data in terminal you can see each new data in new line. If you press Key1 on remote, you will get following output

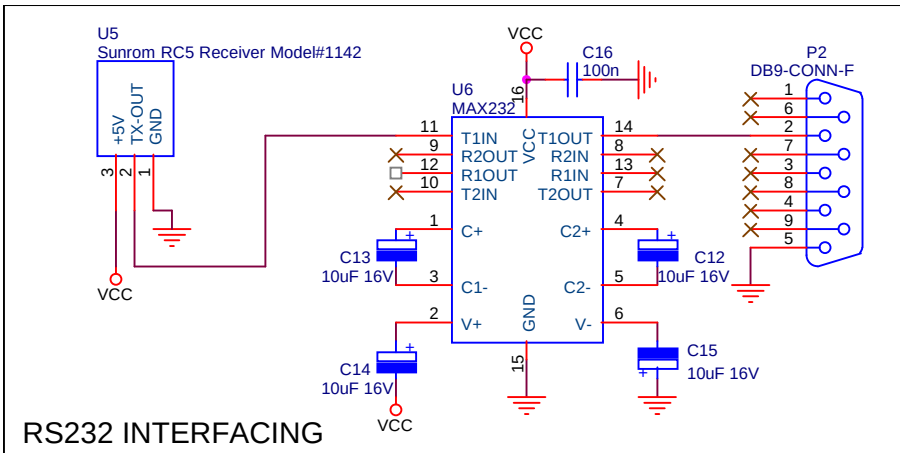
Example Output of Serial in terminal software for Key1:
3001<CR><LF>
Interpreting above result in below table

BYTE COUNT	HEX	DECIMAL	CHARACTER DISPLAYED	Details
1	0x33	51	'3'	RC5 Data High Byte
2	0x30	48	'0'	RC5 Data High Byte
3	0x30	48	'0'	RC5 Data Low Byte
4	0x31	49	'1'	RC5 Data Low Byte
5	0x0D	13	'\r' = CR	New Line Character
6	0x0A	10	'\n' = LF	New Line Character

The above values in serial data are ASCII characters. You can convert the value to binary to use in your program by deducting 0x30 from ASCII value. Our sample code given on next page uses this technique to convert this ASCII buffer of four digit to single integer of RC5 data variable containing 16 bits.

Interfacing with RS232

If you wish to interface the module with RS232 level like a PC serial port or any other device you need a level convertor such as MAX232 as shown below.

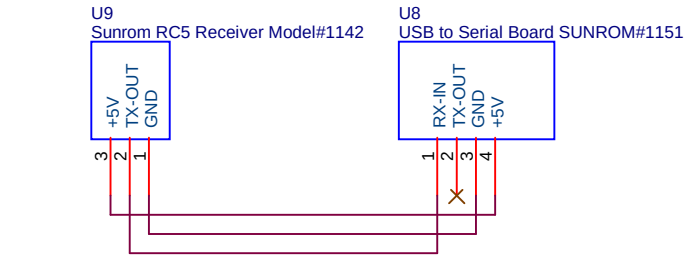


RS232 INTERFACING

You can also use our Max232 Board Model 1104

http://www.sunrom.com/p-245.html

Interfacing to USB Port and Powering from USB Port

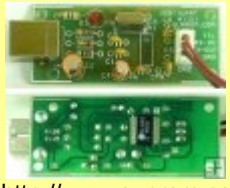


USB INTERFACING

It will appear as virtual serial port on PC to which you can communicate through any software which can transmit receive by this serial port like hyperterminal or custom made software.

To get +5V power for RC5 Decoder from USB port, solder +5V wire of RF module to +ve pin of this capacitor.

You can use our USB to Serial Board Model 1151

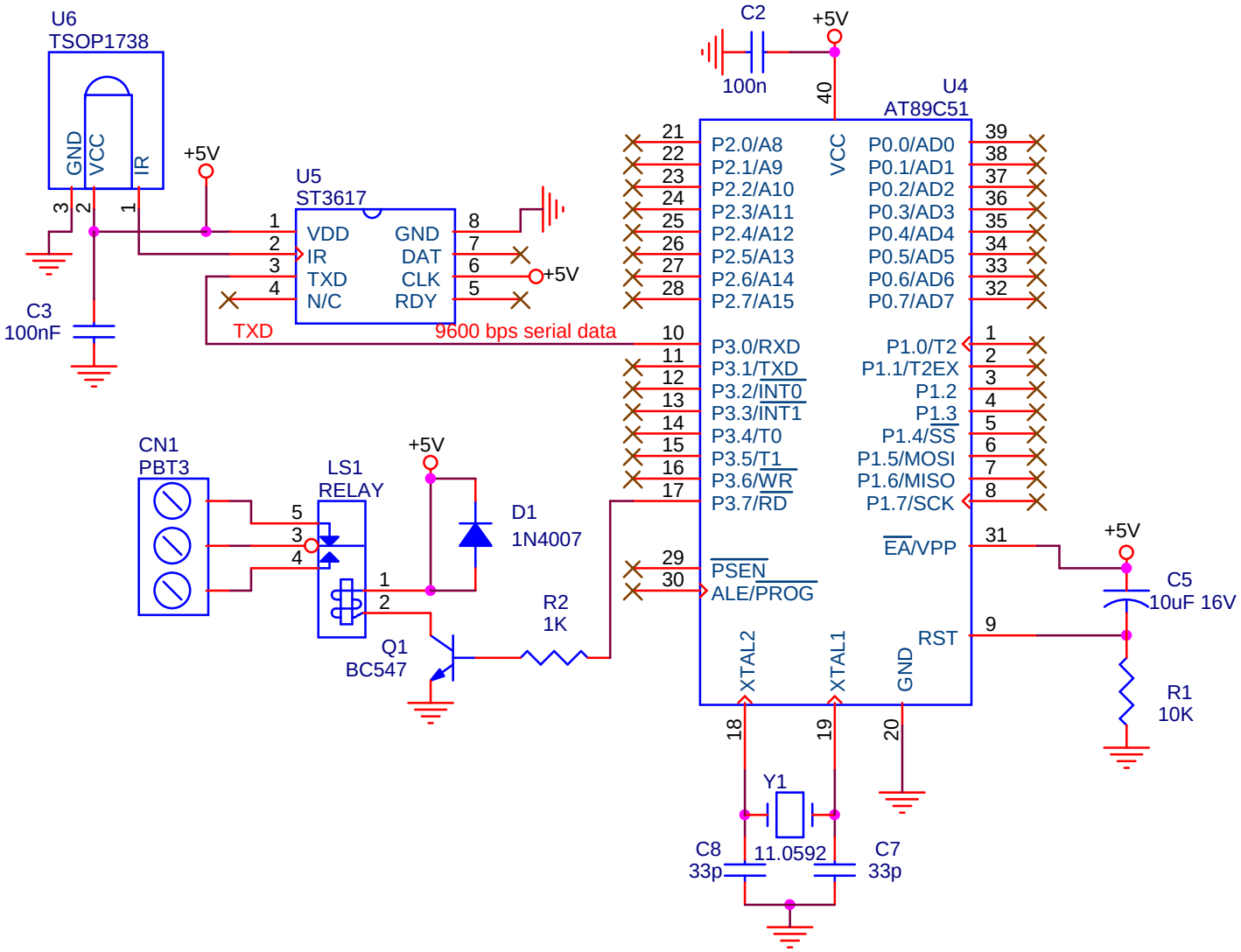


http://www.sunrom.com/p-244.html

Application example using serial data

We have used AT89C51's RXD pin to receive serial data from ST3617 and we switch on Relay ON or OFF for Key1 pressed on Remote. Key1 when first pressed will turn relay ON and Key1 pressed again will make Relay OFF. Relay used is 5V type. But can be any voltage if you have higher voltage available on your application. You can use any microcontroller to interface using this interface. We have chosen AT89C51 to show since it is more widely used. The sample code we have given can be adapted to any C compiler or any microcontrollers like AVR or PIC since with minor changes.

Source code can be downloaded from
http://www.sunrom.com/files/3617-samplecode.zip
Code is compiled using keil compiler



RC5 address and command Table

Commonly used Address and Command data are as per table below
TV remotes are Address 0 only, If you want to make a RC5 remote that will not respond to TV then customize it for a different address.

RC5 ADDRESS of Remote Control	EQUIPMENT
0	TV SET 1
1	TV SET 2
2	VIDEOTEXT
3	EXPANSION FOR TV 1 AND 2
4	LASER VIDEO PLAYER
5	VIDEO RECORDER 1 (VCR 1)
6	VIDEO RECORDER 2 (VCR 2)
7	RESERVED
8	SAT 1
9	EXPANSION FOR VCR 1 OR 2
10	SAT 2
11	RESERVED
12	CD VIDEO
13	RESERVED
14	CD PHOTO
15	RESERVED
16	AUDIO PREAMPLIFIER 1
17	RECEIVER / TUNER
18	TAPE / CASSETE RECORDER
19	AUDIO PREAMPLIFIER 2
20	CD
21	AUDIO RACK
22	AUDIO SAT RECEIVER
23	DCC RECORDER
24	RESERVED
25	RESERVED
26	WRITABLE CD
26-31	RESERVED

Keypress in Remote Control are RC5 commands

RC5 COMMAND (in decimal)	DESCRIPTION of FUNCTION
0-9	NUMERIC KEYS 0 - 9
12	STANDBY
13	MUTE
14	PRESETS
16	VOLUME UP
17	VOLUME DOWN
18	BRIGHTNESS +
19	BRIGHTNESS -
20	COLOR SATURATION +
21	COLOR SATURATION -
22	BASS UP
23	BASS DOWN
24	TREBLE +
25	TREBLE -
26	BALANCE RIGHT
27	BALANCE LEFT
48	PAUSE
50	FAST REVERSE
52	FAST FORWARD-
53	PLAY
54	STOP
55	RECORD
63	SYSTEM SELECT