Rolling Code Keyless Entry System

Versatile IR unit also functions as an alarm

Part.1: By JOHN CLARKE

Ideal for keyless entry for doors in cars, homes and industry, this Keyless Entry System features a rolling code to ensure high security. It also has two door-strike outputs, an alarm system and provision to use up to 16 separate keyfob transmitters with the same receiver.

LOTS OF DIFFERENT electronic systems have now been developed for keyless entry. These include systems that require a coded electronic key, such as RF and infrared transmitters, RFIDs (Radio Frequency Identification Devices), keypads and swipe cards. There are also units that do not require a coded electronic key and these include fingerprint, face and iris recognition.

Regardless of format, electronic keys usually comprise a small keyfob-style transmitter and a receiver that goes with the door lock mechanism. The transmitter sends a string of data that is unique to each individual lock and this data must match the data stored in the receiver before the lock will be released. The concept is roughly similar to a standard metal key, which has a pattern of peaks and valleys along its length. These peaks and valleys must match the tumblers within the lock in order for the lock to open.

With any type of lock, there is always a problem of security. Keys can easily be copied, while many conventional RF and infrared transmitters are far from tamper-proof. One technique is to use a special receiver to intercept and copy the transmitted code. Once copied, the signal can then be re-transmitted to the door lock to gain unauthorised entry.

In fact, this technique was commonly used by car thieves in carparks and proved very effective against early electronic locking systems. It could also be used to open automatic garage doors and gain access to buildings.

Rolling code security

Modern transmitters now circumvent this problem by changing their code each time they send a signal. So if an unauthorised person captures the transmitted code, re-sending this code will not unlock the door. This is because the door lock is now expecting a new code based on an algorithm that both the transmitter and receiver have in common.

This code changing technique is commonly called a 'rolling code', although it is sometimes also called 'code hopping'. It renders copying useless and thus provides a very high level of security. It is also virtually impossible to send a correct code without having a valid rolling code transmitter. This is because of the huge number of code variations possible.

Because of its security advantages, a rolling code transmitter forms the basis of the Rolling Code Keyless Entry System described here. In fact, the odds of picking a correct code at random for our rolling code transmitter are one in 1.4 trillion or roughly one in 10^{12}.

If you want to know more about rolling code transmissions, refer to the separate panel elsewhere in this article.

Features

Transmitter

- Rolling code infrared transmission
- Small keyfob style case
- Dual function buttons
- Randomisation of code parameters feature
- Synchronising of parameters feature
- Up to 16 identifications

Receiver

- 12V operation
- Up to 16 separate transmitters can be synchronised
- Dual function with an independent output
- Two alarm inputs with exit and entry delays
- Two door strike outputs
- Alarm output
- Arm/disarm output and LED indicator
- IR receive acknowledge LED
- Strike 1 operates on arm, disarm or both
- Strike 2 operates independently with momentary operation or toggle output
- Arm output invert option
- Adjustable door strike, entry/exit delay and alarm periods
- 200-code look ahead feature
- Transmitter lockout feature

Main features

Our new Rolling Code Keyless Entry System comprises a small keyfob-style transmitter and a separate receiver. The transmitter is small enough to be attached to a keyring, and has two pushbutton switches, each capable of sending a separate code. Each time one of the switches is pressed, a small indicator LED flashes to indicate that the transmitter has sent its code.

The larger of the two switches activates the alarm functions of the receiver. It arms the unit so that it will sound an alarm should there be unauthorised access.

The alarm functions include an electric door strike control (this allows the door to be opened), two alarm inputs (eg, to monitor doors, windows or other sensors), and an arm/disarm output. The door strike can optionally be set to operate on arming, on disarming or both.

In addition, an alarm output is provided to sound a siren if required.

The second, smaller pushbutton switch on the transmitter is independent of the alarm. It can be used to operate a separate door strike or some other device connected to the receiver. Such devices can include a light or a siren that can be used as a panic alarm. This can be optionally set to operate momentarily or can be toggled on and off with each switch pressing.

The door strike outputs can be set to operate from between 0 to 64 seconds, while the inputs can include delayed operation from 0 to 64 seconds. These delayed inputs allow the alarm to be armed while giving the user enough time to exit the door without setting off the alarm. An identical delay period allows the alarm to be disarmed on entry.

During the exit delay period, the ARM indicator LED in the receiver unit flashes on and off at a one second rate. At the end of the exit delay, this LED indicates that the unit is armed by flashing briefly once every second. This conserves power and increases its effectiveness when it comes to attracting attention.

An Acknowledge/Power LED is also included in the receiver. This normally flashes with a very short duty cycle. However, when the receiver picks up a signal from the transmitter, the Ack/Power LED flashes at a very high rate. It also shows if the received code is invalid by momentarily blinking off and on.

If the code is correct, the receiver responds to the signal. The transmission range is about 4m, which should be sufficient for most purposes. **Note, however, that it will not work if the IR receiver is in direct sunlight.**

Setting it up

Before using the Infrared Rolling Code Alarm, both the transmitter and the receiver must be set up correctly. First, each transmitter must be given a separate identity ranging from 1 to 16. This is selected using link options on the transmitter board, but note that no two transmitters should be given the same identity.

Second, the transmitter must be randomised. This changes the initial rolling code and algorithm parameters to ensure that the transmitter code is going to be unique.

The third step involves synchronising the transmitter and receiver. This process involves sending the rolling code parameters to the receiver, as described next month. You can synchronise from 1 to 16 transmitters, provided each has a different identity.

Also included is a facility to prevent any or all transmitters from operating the receiver once they have been synchronised. This 'lockout' feature can be useful if a transmitter has been lost and you no longer want it to work with your alarm system.

A transmitter identity can be locked out individually, but if you don't know the identity of a lost transmitter, all identities can be locked out. The transmitters that are to be used with the receiver are then re-synchronised.

Parts List – Rolling Code Keyless Entry System

Receiver

- ★1 PC board, code 721, size 61mm × 122mm
- 1 UB3-type plastic box, size 130mm × 68mm × 44mm
- 5 2-way PC-mount screw terminal blocks (5mm or 5.08mm pin spacing)
- 1 SPST vertical mount micro tactile switch, with 0.7mm actuator (S1)
- 3 3-way pin header terminal strips (2.54mm spacing)
- 4 2.54mm jumper shunts
- 3 PC stakes
- 1 25mm length of 0.8mm tinned copper wire

Semiconductors

- 1 PIC16F88-I/P microcontroller programmed with irrcroll.hex (IC1)
- 1 78L05 low-power 5V regulator (REG1)
- 1 38kHz infrared receiver (IRD1)
- 2 BD681 Darlington *NPN* transistors (Q1,Q2)
- 2 BC337 *NPN* transistors (Q3,Q4)
- 1 16V 1W Zener diode (ZD1)
- 4 1N4004 1A diodes (D1-D4)
- 1 1N5404 3A diode (D5)
- 2 3mm red LEDs (LED1,LED2)

Capacitors

- 5 100µF 16V PC electrolytic
- 3 100nF MKT polyester
- 3 10nF MKT polyester
- 1 1nF MKT polyester

Resistors (0.25W, 1%)

- 4 10kΩ
- 2 2.2kΩ
- 2 1kΩ
- 2 680Ω
- 1 220Ω
- 2 100Ω
- 1 10Ω

Test Components

- 4 red LEDs
- 4 2.2kΩ 0.25W 1% resistors

Transmitter

- ★1 PC board, code 722, measuring 30 × 36mm
- 1 keyfob remote control case (Jaycar HB-5605 or equivalent)
- 1 12V A23 car alarm battery (9.5 diameter × 27mm)
- 2 SPST SMD tactile switches 6 × 6 × 3.85mm (S1,S2)
- 1 TO-3P transistor silicone insulating washer, cut to 20 × 24mm
- 5 PC stakes
- 1 25mm length of 0.8mm tinned copper wire
- 1 ICSP 5-pin connector (CON1)

Semiconductors

- 1 PIC16F628A-20/SO 18-lead SOIC microcontroller, programmed with irxmroll.hex (IC1)
- 1 MC78M05 DPAK 5V regulator (REG1)
- 1 MMBT100 SOT-23 SMD *NPN* transistor (Q1)
- 1 MMBT200 SOT-23 SMD *PNP* transistor (Q2)
- 2 1N4148 diodes (D1,D2)
- 1 3mm infrared emitting LED (LED1)
- 1 green gull wing style surface mount LED (2.2 × 2.2mm) (LED2)

Capacitors

- 2 1µF monolithic ceramic
- 1 100nF monolithic ceramic

Resistors (0.25W, 1%)

- 2 10kΩ
- 3 1kΩ
- 2 22Ω

- 2 10kΩ horizontal trimpots (VR1,VR2)

★ Printed circuit boards available from the *EPE PCB Service*

Transmitter circuit

OK, so much for the background details. Let's now take a look at how the circuit works, starting with the transmitter – see Fig.1.

IC1, a PIC16F628A microcontroller, forms the heart of the transmitter circuit. The circuit might look quite simple, but there are a lot of 'smarts' hidden inside the PIC micro, including the software necessary to generate the rolling code.

Under normal conditions, switches S1 and S2 are open circuit and transistor Q2 is off, so no power is applied. This is done to ensure long battery life. If power were continuously applied, the current drawn from the battery would be around 4mA because of the quiescent current of the 5V regulator.

Conversely, pressing either S1 or S2 connects the 12V battery to the input of regulator REG1 via diode D1 or D2. A 22Ω resistor is included in series between the battery and the switches to limit the initial charging current into the 1µF bypass capacitor at REG1's input. This minimises wear on the switch contacts.

When power is applied to REG1's input, its output delivers a regulated +5V rail to IC1. As a result, the micro powers up and runs its internal software program.

Switch check

One of the first things the program does is check which switch was pressed (this happens after a short delay to make sure the switch is fully closed). In operation, the program can decide if S1 or S2 is pressed because of the 10kΩ resistor connected between S2 and the micro's RA4 input.

It works like this. Initially, RA4 is set low by the program. This pin is then made open circuit so that it can be pulled high if switch S2 was closed. However, if S1 was closed instead, the RA4 pin will stay at 0V. By checking the voltage on RA4, the program can thus determine which switch was pressed and initiate the correct function codes for that switch.

The 10kΩ resistor is necessary to limit the current into the internal clamping diodes at RA4 when S2 is closed. In practice, the positive clamp diode will conduct, clamping the RA4 input to 0.6V above the +5V supply. This protects the input from damage.

Diodes D1 and D2 protect the regulator from reverse polarity should the battery be inserted the wrong way around. These diodes also isolate the switch outputs from each other, so that the RA4 input will only go high if S2 is pressed. If S1 is pressed, the 12V at REG1's input reverse biases D2, and so is blocked from reaching RA4.

Next, the program sets RA2 at pin 1 of the micro high. This output drives the base (B) of *NPN* transistor Q1 via a 10kΩ resistor. As a result, Q1 switches on and this in turn switches on transistor Q2.

This action latches the supply to regulator REG1, even if switch S1 or S2 is released. This is necessary to allow time for the rolling code calculations to

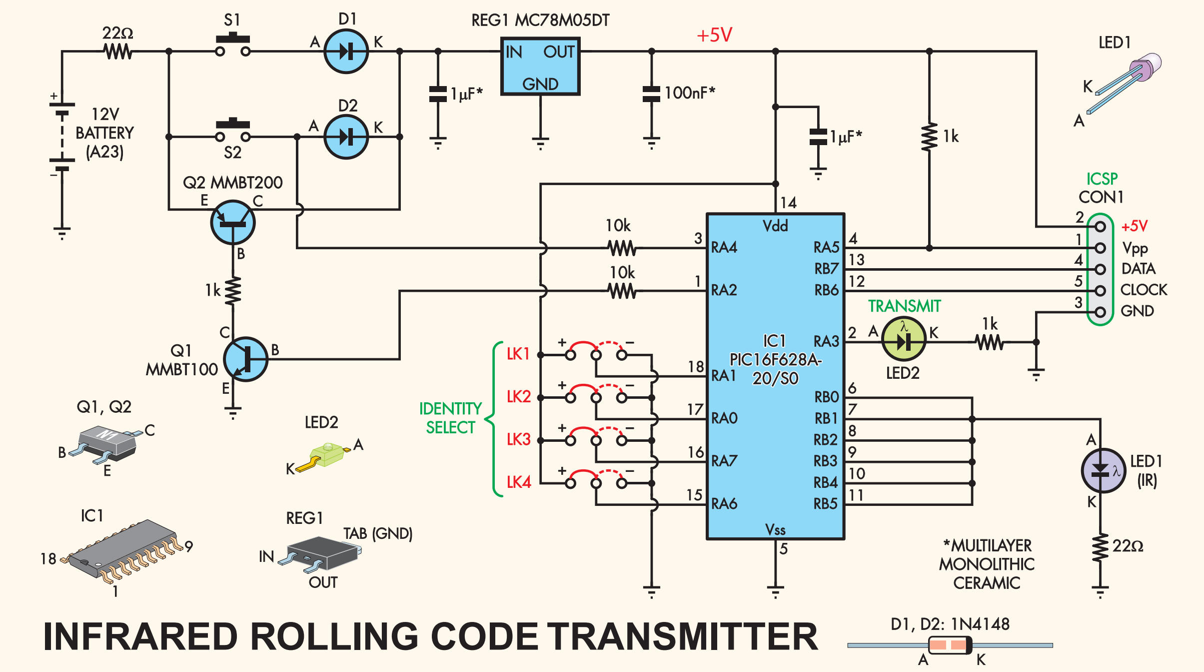


Fig.1: a PIC16F628A microcontroller forms the heart of the transmitter circuit. It contains all the software necessary to generate the rolling code and drives an infrared LED (LED1).

be made and stored without interruption, otherwise the code may become corrupted. It also ensures that the rolling code is transmitted in its entirety.

The next stage in the program involves calculating the code and storing the values. This calculation is based on the previously transmitted code and uses an internal algorithm. Once calculated, the new code appears at outputs RB0 to RB5, which in turn drive an infrared LED (LED1). The 22Ω resistor in series with LED1 limits the current to a safe value.

In operation, LED1 is driven using 100mA pulses at a rate of 38.46kHz. A high (or a '1') is transmitted as a 512µs burst of 38.46kHz signal, followed by 512µs of no transmission. Conversely, a low (or a '0') consists of a 512µs period of no transmission followed by a 512µs burst of 38.46kHz signal.

LED2 is the Transmit LED and is driven by output RA3 during code transmission. Basically, RA3 goes high each time there is a '1' in the transmitted code, and low each time there is a '0'. As a result, LED2 flashes to mimic the transmission code.

Transmitter identity

Transmitter identity is selected using the LK1 to LK4 link connections to RA1, RA0, RA7 and RA6. As shown, each individual input can be connected to either the +5V supply or the ground (0V) supply, but **NOT** to both or the supply will be shorted. The number of possible combinations is 16.

Each of these inputs is initially tied to +5V on the PC board (via thin PC tracks) and this selection is identity 1. The other 15 identities are selected by breaking one or more of these connections to the +5V rail and connecting them instead to an adjacent 0V rail.

We'll talk more about this in the construction.

In-circuit programming

Five-pin header CON1 is provided on the circuit to allow for In-Circuit Serial Programming (ICSP) of IC1 using a PIC programmer. Alternatively, we have developed a surface-mount converter board that will allow IC1 to be programmed directly using a PIC programmer. We'll publish the details on this next month.

The ICSP connections on the transmitter are also used to run the randomisation and synchronisation functions using a bridge between pins 3 and 5 and 3 and 4 respectively.

IC1 runs at a nominal 4MHz, as provided by an internal oscillator. This oscillator has a 1% tolerance and its accuracy is sufficient for this application (ie, there's no need for a crystal oscillator). However, because the oscillator frequency can vary with temperature, we have included a means for the receiver to lock onto the transmitter's clock rate, so that variations over a long time period do not matter.

By the way, the transmitter uses several surface-mount components so that the circuit will fit into a small keyfob case. These surface-mount parts include IC1, REG1, Q1, Q2, LED2, S1 and S2. The remaining parts are standard through-hole component types that are small enough to fit onto the PC board.

Receiver circuit

Refer now to Fig.2, which shows the receiver circuit. It's built around infrared receiver IRD1 and PIC microcontroller IC1, the latter operating at 4MHz to match the transmitter's frequency. Once again, much of the complexity is hidden by the software programmed into the microcontroller.

IRD1 only has three leads, but inside it comprises a complete infrared detector and processor. First, it receives the 38kHz infrared pulse signal from

Specifications

Transmitter

Standby current: 0mA

Total transmit current: rolling code transmission = 35mA for 80ms; synchronise = 35mA for 100ms; randomisation = 10mA.

Infrared transmit frequency: 38.46kHz

Code transmission rate: 1.024ms

Encoding: a high (or a 1 bit) is transmitted as a 512µs burst of 38.46kHz infrared signal, followed by 512µs of no transmission. A low (or 0 bit) is transmitted by a 512µs period of no transmission, followed by a 512µs burst of 38.46kHz infrared signal.

Rolling code: sends four start bits, an 8-bit identifier, the 48-bit code plus four stop bits. The start bits include a 16.4ms gap between the second start bit and the third start bit.

Synchronise code: sent as two blocks. Block 1 sends four start bits, the 8-bit identifier, a 32-bit seed code and four stop bits. Block 2 sends four start bits, a 24-bit multiplier, the 8-bit increment and 8-bit scramble values, and four stop bits. The start bits include a 16.4ms gap between the second start bit and the third start bit.

Code randomisation: alters the multiplier values, the increment value, the scramble value and the seed code at a 40µs rate.

Infrared transmission range: 4m

Receiver

Supply Current: 7.6mA typical when armed and with no external devices powered.

Strike 1 period: adjustable from 0-64 seconds in 0.25s steps approximately.

Strike 2 period: adjustable from 0-64 seconds in 0.25s steps approximately.

Input 1 delay: adjustable from 0-64 seconds in 0.25s steps approximately for exit and entry delays.

Input 2 delay: adjustable from 0-64 seconds in 0.25s steps approximately for exit and entry delays.

Alarm period: adjustable from 0-128 seconds in 0.50s steps approximately

the transmitter and amplifies this to a constant level. This signal is then fed to a 38kHz bandpass filter to remove any 50Hz or 100Hz mains signal and other noise. It then demodulates the signal to produce a serial data burst at IRD1's pin 1 output.

This serial data signal from IRD1 is fed to the RB4 input of IC1 via a 100Ω resistor. A 1nF capacitor filters out any transients.

IRD1 is powered from the receiver's +5V supply rail. A 100Ω resistor and a 100µF capacitor provide supply decoupling and filtering, to prevent the receiver from producing false signals due to power line changes.

As well as the IR receiver, there are two other inputs to the PIC microcontroller. These are alarm sensor inputs – Input 1 and Input 2 – and these connect to the RB5 and RB6 inputs of IC1 via 2.2kΩ resistors. Each input is also bypassed using a 100nF capacitor to filter out transients and thus prevent false triggering of the alarm.

When these inputs are open, both RB5 and RB6 are held high (ie, at +5V) via internal pull-up resistors. In practice, this means that you can use normally-open (NO) or normally-closed reed switch and magnet assemblies to trigger the inputs.

If you use an NO switch, the input will normally be high and the system will trigger if a switch is closed. Conversely, if an NC switch is used, the input will normally be pulled low but will go high if the switch is opened.

Basically, any change in level when a reed switch opens or closes will be detected and sound the alarm at the end of the entry period – provided that the receiver is in its armed state. Note, however, that the alarm will not sound if the receiver is still within its exit delay period.

Door strike outputs

When an IR signal transmission is received, the output from IRD1 is processed by IC1. This then drives Darlington transistors Q1 and Q2 as appropriate to control the door strike outputs (ie, Strike1 and Strike2).

As shown, Q1 and Q2 are driven via 680Ω resistors from IC1's RB0 and RA2 outputs respectively. Diodes D1 and D2 clamp the voltage produced by the door strike solenoid to the supply rail when the transistor is switched off.

Transistors Q1 and Q2 are both BD681 Darlington types and can be used to drive loads up to 1.5A. A typical electric door strike only draws about 800mA at 12V.

The other two outputs are the Alarm and Arm outputs and these are controlled by transistors Q4 and Q3 (both BC337) respectively. Q4 is driven by IC1's RB1 output via a 220Ω current limiting resistor. However, the base current is sufficient for the transistor to remain fully saturated for a 200mA load and this is ideal for many piezo sirens.

Similarly, transistor Q3 is driven via a 10kΩ resistor from IC1's RB2 output. Q3's collector provides the Arm output and this can be used as a toggle output to set a second alarm system.

Typically, you would use a 1kΩ pull-up resistor between the Arm output and the +12V rail, so that the level can swing between 0V and 12V. Alternatively, Q3's collector could be used to drive a relay coil. In this case, the 10kΩ base resistor will need to be reduced to 1kΩ so that the transistor can remain in saturation while driving a 285Ω 12V relay coil.

The unit can be optionally configured with Q3 either on or off when armed. This is set using link LK4.

When LK4 is in the '+' position, Q3 is on when the unit is armed and off when disarmed. In this case, the RB3 input is held at +5V via an internal pull-up resistor within IC1.

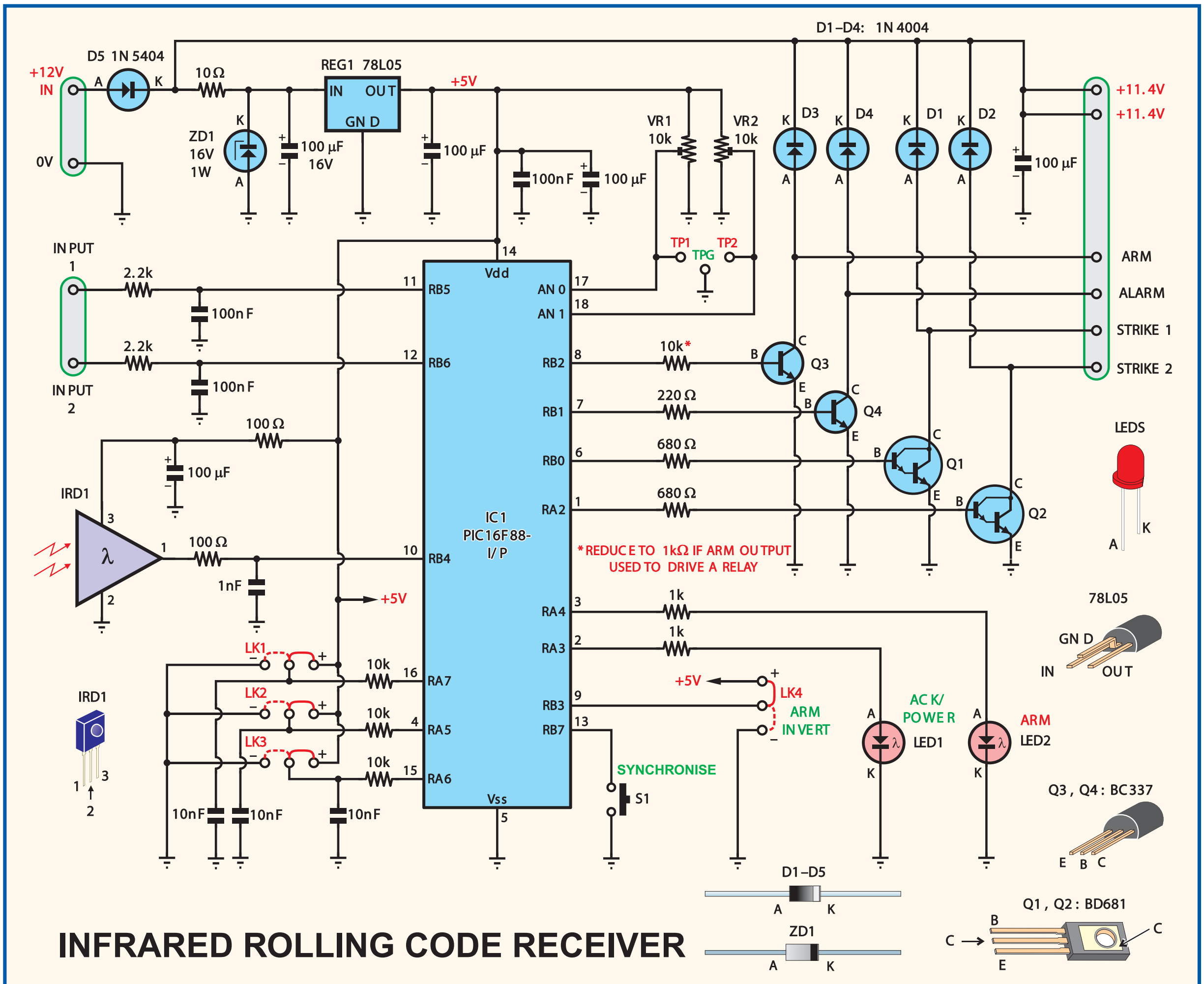


Fig.2: infrared receiver IRD1 and PIC microcontroller IC1 are the main parts in the receiver. IRD1 picks up and demodulates the infrared transmissions, while IC1 decodes the data and drives the various outputs.

Moving LK4 to the '–' position pulls RB3 to ground and changes the sense of the Arm output. In this case, Q3 is off when the unit is armed and on when disarmed.

LED 2 indicates the state of the unit. It's driven from the RA4 output of IC1 via a 1kΩ resistor and flashes when the unit is armed.

There are two different flash styles. During the entry and exit delay periods, the LED flashes with a 50% duty cycle (ie, it is on for half the time and off for half the time). However, at the end of the delay period, it flashes on for only 4% of the duty cycle (ie, each flash is very brief).

Other link options

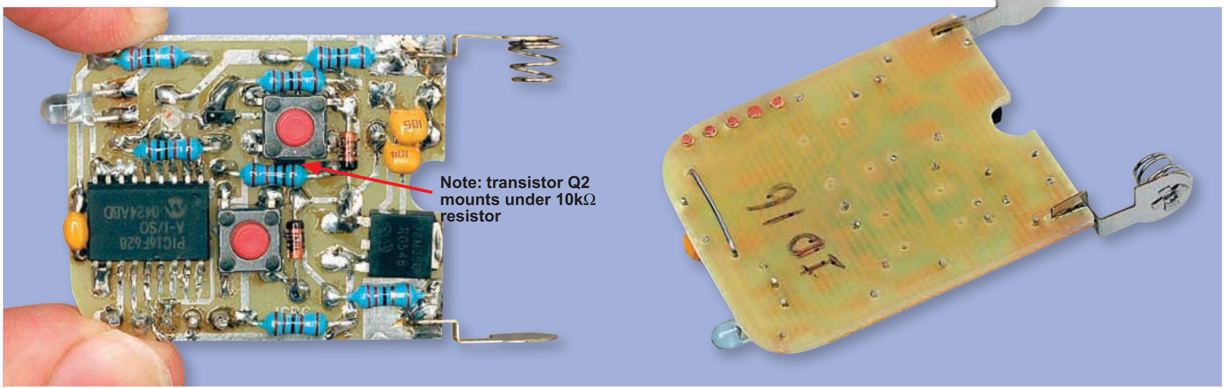
Links LK1, LK2 and LK3 are included to provide further options. For example, LK1 can be tied to either the +5V rail or to 0V, or it can be left open. These three options determine how the Strike1 output operates. Basically, Strike1 can be set to operate when the unit is armed, when it is disarmed or on both arming and disarming.

In operation, the software programmed into the PIC micro decides where the link is inserted by running a few tests. First, it takes the RA7 output high (5V) and then sets the RA7 pin as an input to read the voltage. If the voltage is now low, then the link must be in the '–' position. However, if the input remains high, then the link is either in the '+' position or is open circuit (it remains high when the link is open because of the charge on the associated 10nF capacitor to ground).

To test if the link is in the '+' position or open, the RA7 pin is made an output again and is driven low (to 0V). The RA7 pin is then changed to an input and the level checked again. If the voltage is now high, then the link must be in the '+' position. Conversely, if the voltage is low, then the link is open.

The 10kΩ resistor in series with RA7 is there to prevent shorting when this pin is taken high and low with a link in position.

LK2 sets Strike2's operation for either momentary operation or for toggle operation. This link pulls the RA5 input either to +5V when it is in the '+' position (momentary) or to 0V when it is in the '–' position (toggle). Note that this link cannot be left open because the RA5 pin can only be used as an input.



These two larger-than-life-size photos clearly show how the parts are mounted on the transmitter board. You will need a fine-tipped soldering iron (2mm diameter or less) and a magnifying glass to do the assembly.

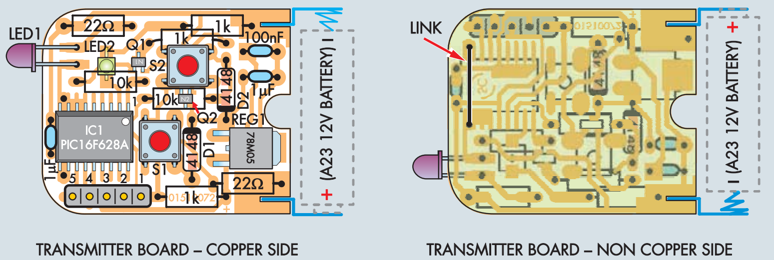


Fig.3: follow these parts layout diagrams to build the transmitter board. Note that you have to set the transmitter's identity before installing IC1 (see text) and don't forget transistor Q2 – it goes under a 10kΩ resistor, just below S2.

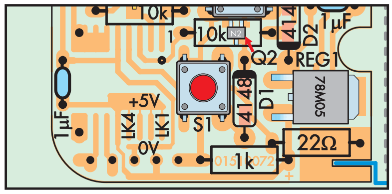


Fig.4: this enlarged track section shows the locations of links LK1-LK4 on the transmitter board. The transmitter identity is changed by breaking one or more of the thinned link connections to the +5V rail and bridging them (with solder) to the adjacent 0V rail instead.

LK3 is used in conjunction with trimpots VR1 and VR2 to set the various time periods. These include the Strike1 and Strike2 momentary on periods, the entry and exit delays for Input1 and Input2, and the alarm period.

As shown in Fig.2, trimpots VR1 and VR2 are connected across the 5V supply and their wipers (moving contact) connect to analogue inputs AN0 and AN1 respectively. The voltage applied to each analogue input is converted to a digital value within the software and it is these values that determine the timeout periods.

Synchronise switch

Switch S1 is the Synchronise switch, and this connects to the RB7 input. This input is normally held high via an internal pull-up resistor, but when S1 is closed, it pulls RB7 to 0V.

Basically, S1 is used to synchronise the receiver with the transmitter. It is also used when setting the time periods. In addition, if S1 is closed during power-up, it selects the transmitter identity lockout function.

Power supply

Power for the circuit is from a 12V supply such as a battery or DC plugpack. Diode D5 provides reverse polarity protection and is rated at 3A so that it can handle the currents that may be drawn by an electric door strike and siren.

The 10Ω resistor and Zener diode ZD1 provide transient protection, with the Zener clamping voltages over 16V. The 10Ω resistor limits the current through ZD1 to a safe level.

Following ZD1, the supply is filtered using a 100µF capacitor and applied to the 3-terminal regulator REG1. The resulting regulated +5V rail is then

Capacitor Codes

Value	µF Code	IEC Code	EIA Code
100nF	0.1µF	100n	104
10nF	.01µF	10n	103
1nF	.001µF	1n0	102

Resistor Colour Codes (Transmitter)

❏	No.	Value	4-Band Code (1%)	5-Band Code (1%)
❏	2	10kΩ	brown black orange brown	brown black black red brown
❏	3	1kΩ	brown black red brown	brown black black brown brown
❏	2	22Ω	red red black brown	red red black gold brown

Table 1: Transmitter Identity

Iden-tity	LK1	LK2	LK3	LK4
1	+	+	+	+
2	+	+	+	-
3	+	+	-	+
4	+	+	-	-
5	+	-	+	+
6	+	-	+	-
7	+	-	-	+
8	+	-	-	-
9	-	+	+	+
10	-	+	+	-
11	-	+	-	+
12	-	+	-	
13	-	-	+	+
14	-	-	+	-
15	-	-	-	+
16	-	-	-	-

used to power IC1 and the infrared receiver (IRD1).

Power on/off indication is provided by LED1, which also acknowledges the infrared signal. Normally, LED1 flashes with a 4% duty cycle about twice per second. However, when an infrared signal is received, it flashes at the infrared reception rate.

LED1 also flashes with an even duty cycle for a short time at the end of synchronisation and if the infrared signal is incorrect.

Construction

The Rolling Code Keyless Entry System is built on two PC boards: a Receiver board, code 721, and a Transmitter board, code 722. Both boards are available from the *EPE PCB Service*.

We'll start with the transmitter assembly, which is the trickier of the two. In order to fit in the keyfob case, the transmitter board measures just 30 × 36mm and uses lots of surface-mount components.

However, these are not too difficult to solder in, provided you have a soldering iron tip that is just 2mm in diameter or finer. A magnifying glass (or, preferably, a 'maggie lamp') is also required to check your soldering, while a length of 1.5mm de-soldering braid (Solderwick) would also be useful for cleaning up any excess solder that may flow between connections.

Fig.3 shows the component layout on the PC board. The first step is to check the PC board carefully for any breaks in the copper or shorts between tracks. Repair any faults that you do find (rare these days), then check the shape of the board. It should have a curved front edge and a small circular cut out at the other end. In addition, there should be two slots for the battery clips.

The keyfob case is supplied with the key switch covers mounted as shown here. This assembly must be removed.

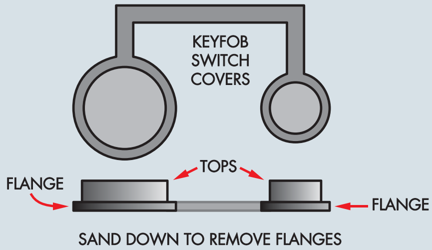


Fig.5: once the switch covers have been removed, the flanges are ground down using 180-grit sandpaper, so that only the tops remain (see text).

Next, check that the PC board fits neatly into the base of the keyfob case. If it does not fit, it's just a matter of filing it neatly along the edges until it does.

Setting the identity

Before mounting any of the parts, it's first necessary to set the transmitter's identity – but only if more than one transmitter is to be used. If more than one transmitter is used, then each will require a unique identity.

As supplied, the PC board initially ties links LK1 to LK4 to the +5V supply rail. This is Identity1, or ID1. If only one transmitter is to be used, then you don't have to do a thing – just leave it at the default identity (ID1).

If you do wish to change the identity, it's just a matter of altering one or more of the links as shown in Table 1. You do that by breaking the link's thinned connection to the +5V track

Above: the finished transmitter board inside its keyfob-style plastic case. Power comes from a 12V A23 car alarm battery. Note how the keyswitch covers are mounted on the lid, using a 20 × 24mm silicone washer – see text.

Right: the two keyswitch covers are attached to the 20 × 24mm silicone washer as shown here. Use silicone sealant to 'glue' them in place. The keyfob lid can be used as a template to position them correctly.

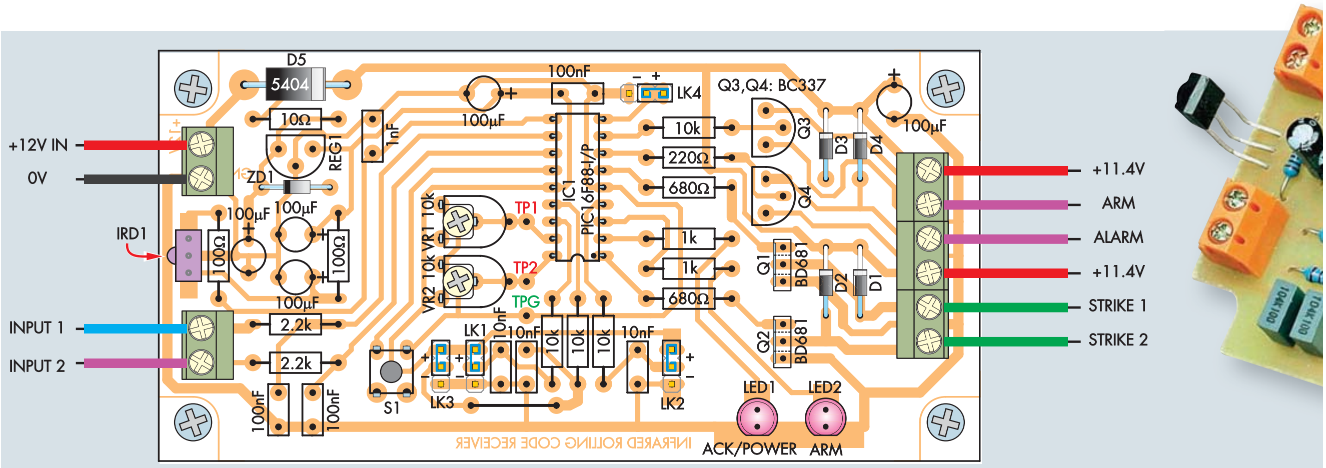


Fig.6: follow this diagram to mount the parts on the receiver board. Use a socket for the PIC microcontroller and take care to ensure that all polarised parts are correctly orientated. The infrared receiver module (IRD1) can either be mounted on the board (as in the prototype) or connected via shielded cable (see diagram next month).

and connecting it to the adjacent 0V track instead via a small solder bridge.

Make sure, however, that a link connection is not made to both the +5V and 0V tracks. We have labelled the +5V connection with a plus (+) sign and the 0V connection with a minus (–) sign.

It is important to select the identity now because the +5V track section cannot be accessed when IC1 is in place. The +5V connections should only be broken with a sharp craft knife and, once broken, should not be resoldered. That's because IC1 would no longer sit properly on the board, making it difficult to solder its pins.

The selected identity should be marked on the back of the PC board using a marker pen. For example, if the identity is 2, write ID2 on the PC board. This number can also be written on the back of the keyfob transmitter case, in the indentation provided.

Software

If you are building the unit from a kit, then IC1 will be supplied pre-programmed. If not, you will have to program the PIC yourself using a suitable programmer. As previously mentioned, we have provided two programming options, the first of which is to use the in-circuit programming connector on the PC board.

Alternatively, you can build and use the surface mount adaptor board to be described next month, so that IC1 can be programmed out of circuit.

The sofware files will be available via the *EPE* Library site, accessed via **www.epemag.com**. Pre-programmed PICs will also be available from Magenta Electronics – see their advert in the issue for contact details.

Parts assembly

Except for a single wire link, all parts for the transmitter mount on the copper side of the PC board. Don't install the link yet though – that step comes after you install IC1.

To install IC1, position it on the board with its pin 1 at top right – see Fig.3 (pin 1 is indicated by a small adjacent dot in the body of the IC). Carefully adjust it so that its pins line up with the tracks and use a clothes peg (or some other small spring-clamp) to hold it in position.

That done, solder a couple of diagonally opposite pins, check that everything is correct, then remove the peg and carefully solder the remaining pins.

The main thing to watch out for here is unwanted solder bridges between adjacent copper tracks. If this does happen, use some solder wick to draw up the excess solder to clear the short. A magnifying glass will be handy here to inspect your work.

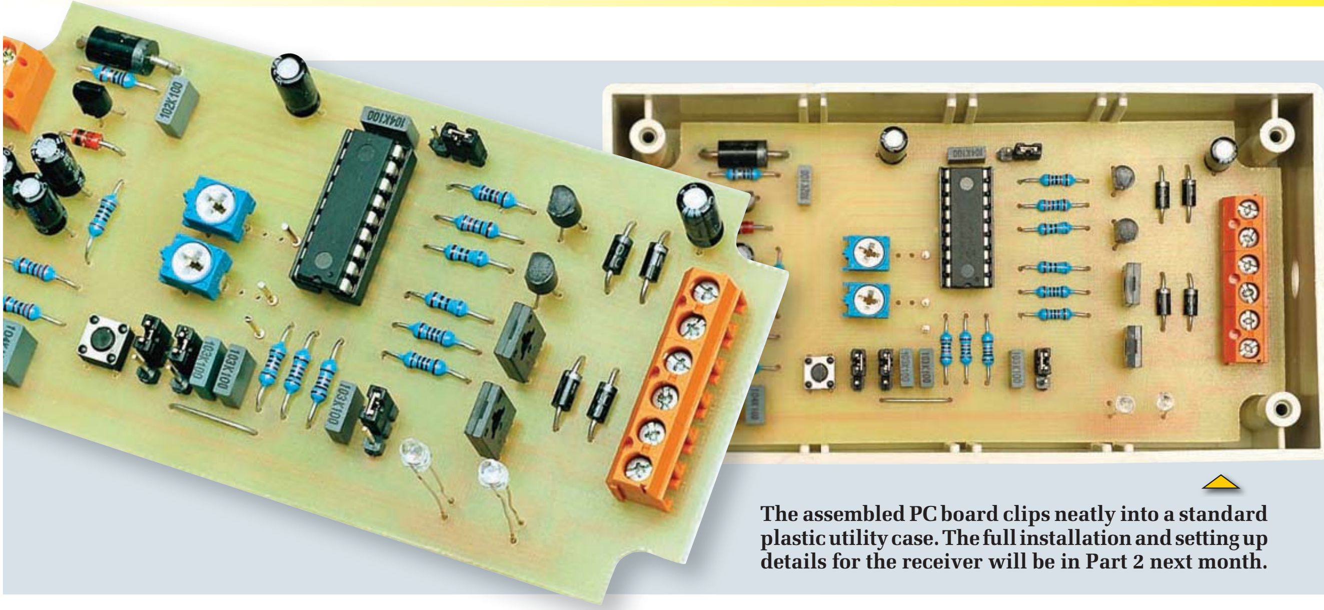
Note that pins 6 to 9 and 10 and 11 are connected together anyway, so solder between these pins is OK.

Once the IC is in, you can install the link beneath it on the other side of the board. This link must sit flat against the board, otherwise the board will not sit down in the case correctly.

The remaining surface mount components – Q1, Q2, LED2 and REG1 – can now be soldered in place. **Transistor Q1 has an N1 label on its top, while Q2 has an N2 label instead.**

Resistor Colour Codes (Receiver)

❏	No.	Value	4-Band Code (1%)	5-Band Code (1%)
❏	4	10kΩ	brown black orange brown	brown black black red brown
❏	2	2.2kΩ	red red red brown	red red black brown brown
❏	2	1kΩ	brown black red brown	brown black black brown brown
❏	2	680Ω	blue grey brown brown	blue grey black black brown
❏	1	220Ω	red red brown brown	red red black black brown
❏	2	100Ω	brown black brown brown	brown black black black brown
❏	1	10Ω	brown black black brown	brown black black gold brown



The assembled PC board clips neatly into a standard plastic utility case. The full installation and setting up details for the receiver will be in Part 2 next month.

These numbers relate to the MMBT100 and MMBT200 types respectively. Don't get them mixed up.

Installing the semiconductors

The orientation of the two transistors is obvious – they have one pin on one side of the body and two on the other side. REG1 has a GND tab plus IN and OUT pins that must be soldered to the PC board. The central pin between the IN and OUT pins is left unconnected.

Be careful with the orientation of LED2 – its cathode lead is the longer of the two.

Next, solder in switches S1 and S2, then install five PC stakes for the ICSP header. These pins are inserted from the non-copper side of the PC board and soldered in position. The pins are then trimmed on the copper side to 3mm in height. On the underside, they are trimmed and filed to 0.5mm.

The standard components can now be installed. These must be mounted flat against the PC board or as close to it as possible in the case of the 10kΩ resistor that straddles Q2. Take care with the orientation of diodes D1 and D2 and note that the tops of the three monolithic capacitors must be no more than 4mm above the PC board.

In particular, the two capacitors near REG1 can be laid over at about 45°, while the one adjacent to IC1 needs to have its leads adjusted so it can be pushed down onto the PC board far enough to meet the 4mm height requirement.

Cut all the leads beneath the PC board (ie, on the non-copper side) flush with the surface.

LED1 can go in next. Its anode lead is the longer of the two (unlike LED2) and this lead must go towards IC1. To mount it, first bend its leads down by 90° exactly 2mm from its body, then insert the leads into the PC board. Finally, push the LED all the way down onto the PC board, solder the leads and cut them flush with the underside.

Note that a small circular notch is required in the rim of the keyfob base for the LED to sit in. This can be made using a small rat-tail file. When this notch has been made, file a matching notch in the top half of the keyfob case.

Battery terminals

The battery terminals are installed by first placing the PC board in the base of the case. That done, the terminals are slid into position and soldered. Make sure that the terminal with the spring is located as shown in Fig.3.

Switch cover modifications

The key switch covers that are supplied with the keyfob case have to be modified to suit the two switches on the PC board.

As supplied, the two switch covers are already secured in place in the keyfob lid. This assembly must be removed and the covers carefully ground down to 1.5mm thick – see Fig.5. This is done by placing some 180-grit sandpaper onto a flat bench and sanding the switch covers until they are flat on their base.

That done, cut out a 20 × 24mm rectangular piece from a silicone TO-3 washer (20 × 24mm) to make a new switch cover assembly. It's then simply a matter of attaching the switch covers to this washer using silicone sealant – see photo. Use the keyfob lid as a template to position the covers correctly.

Receiver assembly

Now for the receiver – see Fig.6. As usual, start by checking the PC board for any defects. Check also that the hole sizes for the screw terminal blocks are correct and enlarge them if necessary.

That done, check that the PC board fits inside the specified utility case. File the board edges to get it to fit if necessary but don't file them too much, otherwise the board will not lock correctly into the wall slots.

Fig.6 shows the assembly details. Install the wire link first, then install the resistors. The accompanying table shows the resistor colour codes, but you should also check them using a digital multimeter.

The diodes and the IC socket can go in next, taking care to orient each with the correct polarity. Follow these with the capacitors, again making sure that the electrolytics go in correctly. The three PC stakes for TP1, TP2 and TPG can then be installed.

Depending on your requirements, LEDs 1 and 2 can either be mounted directly on the PC board or mounted externally and connected using wire leads. Be sure to mount each LED with its cathode lead (the shorter of the two) towards the lower edge of the PC board.

Similarly, IRD1 can either be mounted directly on the PC board or connected

Frequently Asked Questions

One question that's often asked about rolling code systems is what happens if the transmitter is out of range and one of the transmit switches is pressed? Will the receiver still work when the transmitter is later brought within range and the button pressed again?

This question is asked because the code the receiver was expecting has already been sent and the transmitter has rolled over to a new code. So how does the system get around this problem?

The answer to this is that the receiver will acknowledge a signal that is the correct length and data rate, but it will not trigger unless it receives the correct code. So if the signal format is correct but the code is incorrect, the receiver then calculates the next code that it would expect, and checks this against the received code. If the code is now correct the receiver will unlock the door.

If the code is still incorrect, the receiver calculates the next expected code and will do this up to 200 times. If none of these are correct, the receiver keeps its original code, but it will not trigger. In fact, the only way to trigger the receiver after this is to re-synchronise it to the transmitter.

Of course, a second transmitter will still operate the receiver (provided they have been synchronised in the first place). That's because this transmitter has a different identity and a different code to the other transmitter.

Automatic synchronisation

Some rolling code transmitters systems offer automatic synchronisation if the transmitter and receiver lose sync. In these systems, the receiver includes a code 'look-ahead' feature, as described above, but the number of look-ahead codes is usually limited to fewer than 200. What happens is that if the code is not recognised after all the look-ahead calculations have been made, the receiver changes its synchronisation method.

Basically, the receiver requires two separate transmission codes before restoring correct operation. On the first transmission, it calculates the next code it should receive using this received code as the basis for calculation. If the second code sent by the transmitter is the same as the code that was calculated, the receiver operates.

The drawback of this latter scheme is somewhat less security since, in theory, two successive transmission codes could be intercepted and recorded. These codes could then be re-transmitted to synchronise and thus trigger the receiver.

Calculating the code

Another question that's often asked is how does the receiver know which code to expect from the transmitter, since this changes each time?

The answer to this is that the transmitter and the receiver both use the same calculation to determine the next code. They also both use the same variables in the calculation and these variables tend to be unique values that no other transmitter uses.

For example, if the calculation for consecutive codes requires the original calculated code to be multiplied by 100 and the number 7 added to it, then both the transmitter and receiver will use these numbers to perform the calculation.

Without knowing both the multiplier and the increment value, it would be very difficult to predict the next code. This is particularly true because of the very large numbers involved. The values quoted for the multiplier and increment value are not as simple as 100 and 7 but are 24 bits and eight bits respectively in length.

In addition, the code length is 48 bits with as many as 2.8×10^{14} combinations. This reduces by a factor of 200 because of the look ahead feature to a 1 in 1.4×10^{12} chance of striking the correct code – still impossibly long odds.

Code scrambling

A further complication with the transmitted code is that the code is not necessarily sent in sequence. There are also 32 possible scrambling variations that can be applied to the code.

What if the transmitter sends two consecutive codes that are the same and the code is intercepted and re-transmitted to open the lock? This is highly improbable and our rolling code transmitter has safeguards to prevent the same code appearing twice in succession. For each code calculation, a comparison is made between the current and last code. If the code is the same, the code is recalculated after an increment of the code value to ensure successive code calculations diverge. It is this new code that is transmitted.

The receiver performs the same recalculation so that the new code will be accepted.

Another question concerns the use of different transmitters. Does each transmitter use the same rolling code calculation and if so, wouldn't the receiver lose its synchronisation if several transmitters were used? The answer is that the receiver will not lose synchronisation, even if one of the transmitters is not generally used. This is because each transmitter operates independently from the others.

Only 16 transmitters can be used with a given receiver and each must have its own different identity from 1 to 16. The identity is built into each transmitter and synchronisation is required for each transmitter.

The codes sent by each transmitter are different and the code includes the transmitter identity value. The receiver has 16 different rolling code and calculation parameters, and so each transmitter is treated independently.

---

using twin-core shielded cable (see diagram in Part 2 next month).

Trimpots VR1 and VR2 and the 3-way and 2-way pin headers for LK1-LK4 are next on the list. That done, install REG1 and transistors Q1 to-Q4. Q1 and Q2 must be installed with their metal faces towards IC1.

Finally, complete the board assembly by installing switch S1 and the screw terminal blocks. Note that the 6-way terminals at the righthand edge of the PC board are made up using three 2-way blocks. These are joined by sliding their dovetail joints together before installing them on the PC board.

That's all we have space for this month. Next month, we'll complete the construction and describe the installation and setting-up procedures, including setting the entry and exit delays. We'll also describe the optional SOIC adapator board, so that you can program the PIC microcontroller out of circuit.

Rolling Code Keyless Entry System

Versatile IR unit also functions as an alarm

Part.2: By JOHN CLARKE

Last month, we described the circuitry and gave the PC board assembly details for our Rolling Code Keyless Entry System. This month, we cover the installation and setting-up procedures and describe an optional SOIC adaptor board, so that you can program the PIC micro out of circuit.

HAVING completed the receiver board assembly, as described last month, it can be housed in a UB3-sized plastic box. As shown in the photo last month, it simply clips into place, but first you will need to drill a hole in one end for IRD1, plus a hole in the other end for the external wiring.

You will also have to drill matching holes in the lid for the Ack/Power and Arm LEDs (LED1 and LED2).

Now for the initial set-up. First, install a jumper link in the minus (–) position for LK2. This will set the Strike2 output to toggle mode (note: LK2 must always have a jumper connection, either to the '+' or '–' position). Leave jumpers LK1, LK3 and LK4 out for now.

Next, set trimpots VR1 and VR2 to mid-range. These trimpots are later used to set the various time periods.

Transmitter set-up

At this stage, the transmitter is already partially set up because its identity is selected during construction. If the transmitter's PIC microcontroller has not been programmed, then program it now via the ICSP connection. This connection can be made by soldering five leads to the transmitter's ICSP pins and then connecting the other ends of these leads to a 5-way ICSP socket to plug into the PIC programmer.

After the IC has been programmed, clip in the 12V battery and check that the green acknowledge LED lights when a switch is pressed.

Of course, if you buy a complete kit, the PIC microcontroller (and the PIC in the receiver) will be supplied pre-programmed, so you won't have to worry about that last step.

Testing the receiver

The receiver can now be tested. First, with IC1 out of its socket, connect a 12V power source that can supply at least 60mA. That done, switch on and check that there is 5V between pins 14 and 5

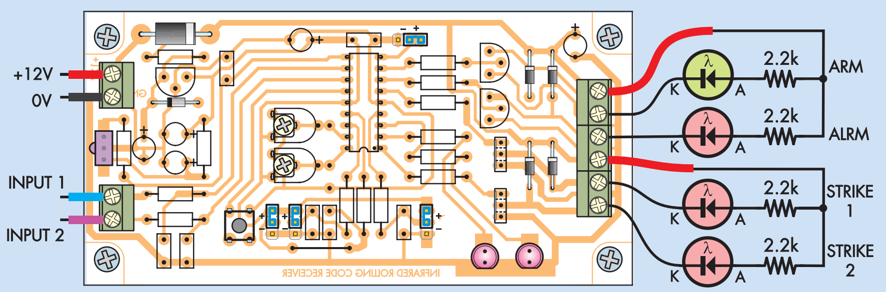


Fig.6: the test LEDs are connected to the receiver as shown here. Follow the procedure in the text to synchronise the transmitters and test the receiver.

of the IC socket. If this is within 10% of 5V (4.5V to 5.5V), switch off and plug IC1 into its socket, making sure that it is correctly orientated.

Next, wire up the test LEDs as shown in Fig.6. These are all wired in series with 2.2kΩ current-limiting resistors. Once the LEDs are wired up, apply power and check that the receiver's power LED flashes briefly at about once per second. If it does, then so far so good.

The transmitter must now be randomised and then synchronised with the receiver. Let's now take a look at these two procedures.

Randomising

Randomisation of the transmitter ensures that it uses a unique set of parameters to calculate the rolling code. This procedure is important because the original parameters programmed in are the same for every transmitter.

Basically, you need to personalise the parameters to prevent another transmitter that has the same identity from operating your receiver. If randomisation is not done, there is the real risk that someone else's transmitter that has also not been randomised will operate your receiver.

To randomise a transmitter, simply connect pins 3 and 5 of its ICSP connector together and then press switch S2. The transmit LED will flash at a one-second rate for the duration. Release the switch when you are ready (after between several seconds and several minutes).

The parameters are all altered every 40µs (that's 25,000 times a second), so they will be different for each transmitter after even short presses.

Synchronising

After randomising, the transmitter must then be synchronised with the receiver. To do this, disconnect pins 3 and 5 of the ICSP header and connect pins 3 and 4 together instead. That done, press and hold down S1 on the receiver and then press one of the switches on the transmitter.

The transmit LED will now flash twice momentarily and the receiver's acknowledge LED will flash on and off at a one-second rate until switch S1 on the receiver is released.

Rolling Code Protection: Keeping It Secret

As previously noted, the Rolling Code Keyless Entry System provides a high level of security because the transmitted code changes each time it is sent. However, to further improve security, we have also included code protection for both the transmitter and receiver.

Basically, code protection prevents the program and data within the PIC microcontrollers from being read by a PIC programmer. As a result, the parameters used to calculate successive rolling codes are kept safe within the microcontrollers. In particular, this effectively prevents a transmitter from being 'interrogated', in order to make a duplicate transmitter that will operate the door lock.

So, while the hex files can be used to program the microcontrollers, they cannot be read back once programming has been verified. The parameters used for calculating the rolling code are then randomised in the transmitter using the set-up procedure already described. It is these parameter and the rolling code seed values that are hidden by the code protection.

Now remove the link between pins 3 and 4 on the transmitter's ICSP header. Once that's done, you should now find that the transmitter operates the receiver. If it doesn't, try synchronising again and make sure that the IR receiver has a clear 'view' of the transmitting LED.

The above randomisation and synchronisation procedures must be done for each new transmitter. Note that a transmitter that has not been synchronised will not be able to operate its receiver, even if their rolling codes are the same. Note also that synchronising a new transmitter prevents the use of a previously synchronised transmitter that has the same identity.

Next, press the main switch on the transmitter and check that the receiver's Strike1 LED lights for about five seconds. The external Arm LED should also light, while the receiver's on-board Arm LED should flash with an even on-off duty cycle. This flashing shows the exit delay.

After about 20s, the exit delay should expire and the Arm LED should then flash briefly once per second.

Now check the operation of the second (smaller) switch which is on the transmitter. This switch should toggle the Strike2 LED on and off with successive pressings.

Testing the alarm

To test the alarm, arm the unit and short Input1 on the receiver to ground (0V) using a clip lead. The external alarm (ALRM) LED should light after 20s and should then stay on for 60s.

You can check the operation of the delayed exit by arming the unit and momentarily shorting Input1 or Input2 to 0V during the exit period. The alarm LED should not light after the exit period has expired.

Receiver options

The receiver can be powered from a 12V DC plugpack or a 12V battery. When powered by a plugpack, make sure it can supply the necessary current for the electric striker and an alarm siren if fitted. Many electric strikes draw around 800mA, so a 1A plugpack will be required.

Note that the armed status is stored in case the power goes off; the armed or disarmed mode will be returned when power is reconnected. So, if the receiver was armed when power was lost, then the armed mode will be restored when power is returned.

Table 1: Strike1 operation (LK1)			
LK1	+	–	Open
Strike1 operates on	Arm Only	Disarm Only	Arm and Disarm

Table 2: Strike2 operation (LK2)			
LK2	+	–	Open
Strike2 operation	Momentary	Toggle	Not valid

Table 3: LK3, VR1 and VR2 settings			
LK3	+	–	Open
Operates when S1 pressed	VR1 sets Strike1 period VR2 sets Strike2 period	VR1 sets Input1 delay VR2 sets Input2 delay	VR1 sets alarm period
Notes	5V sets 64s 2.5V sets 32s 1.25V sets 16s 0.625V sets 8s 0.313V sets 4s 0.156Vsets 2s	5V sets 64s 2.5V sets 32s 1.25V sets 16s 0.625V sets 8s 0.313V sets 4s 0.156Vsets 2s	5V sets 128s 2.5V sets 64s 1.25V sets 32s 0.625V sets 18s 0.313V sets 8s 0.156Vsets 4s

When powering from a 12V battery, a charger should also be connected to maintain battery charge – see Fig.7. A 12V 350mA charger for sealed lead-acid batteries would be suitable. These chargers are fully automatic – they charge the battery when required and maintain full charge with a trickle current.

Depending on your application, Strike1 can be optioned to operate on arming, on disarming or on both arming and disarming. These options are selected using link LK1. Table 1 shows what each link connection does. You may also wish to place a small buzzer across the door strike connection to give an audible indication of door strike operation.

The Strike2 output can be momentarily activated whenever the secondary switch on the transmitter is pressed. Alternatively, it can be toggled on or off with each switch pressing. Link LK2 selects these options.

Receiver time periods

Trimpots VR1 and VR2 are used to set the time periods for Strike1 and Strike2, the exit and entry delays for Input1 and Input2, and the alarm period. Link LK3 provides the means to set each time period – see Table 3.

With LK3 in the '+' position, VR1 and VR2 set the strike period for Strike1 and Strike2 respectively. Table 3 shows the various voltages that VR1 and VR2 can provide to set the strike periods. These voltages can be measured at TP1 for VR1 and at TP2 for VR2.

To set the strike periods, simply adjust VR1 and VR2 to the voltage settings required and press the synchronise switch (S1) on the receiver board.

The delayed inputs (ie, the entry delays for Input1 and Input2) are set when LK3 is in the '–' position. Once again, it's simply a matter of setting the voltages at TP1 and TP2 and pressing S1 to set the values.

Finally, when LK3 is out, VR1 sets the alarm period (VR2's setting is ignored). Just set the required voltage at TP1 and press S1 to program the period in.

Note that because pressing switch S1 programs in the timing adjustments, synchronisation will also alter the timing. This means that if you synchronise a transmitter to the receiver at a later date, you will have to make sure that VR1 and VR2 are in the correct positions for the LK3 option selected before pressing S1.

In practice, this just means leaving VR1, VR2 and LK3 in their final positions after you finish the timing adjustments. That way, if you synchronise a transmitter later on, the last set timing values are simply reset to the same values.

Arm output option

Link LK4 sets the arm output option – see Table 4. When LK4 is in the '+' position, the Arm output is low on arm and open on disarm. Conversely, when LK4 is in the '–' position, the Arm output is open on arm and low on disarm. It all depends on how you intend to use this output as to which option you choose.

Receiver lockout

Any transmitter that has been synchronised can later be locked out from operating the receiver. This is done by setting links LK1, LK2, LK3 and LK4 in the receiver and pressing switch S1 during power up.

Table 5 shows the link options for each transmitter identity. Note that these link settings correspond exactly to the links used in the transmitter to set the transmitter identity

When lockout is performed, the power LED flashes the identity number to indicate that the procedure has been successfully completed. So, for example, if you lock-out an identity 3 transmitter, the power LED will flash three times at a nominal 1s rate before a 4s break until S1 is released.

When S1 is released, the receiver then operates normally, but with the selected transmitter now locked out.

If S1 is held closed, the cycle of LED flashing continues. At the end of the third cycle, all identities will be locked out and the power LED will stay lit until S1 is released. This feature is included as a short cut to locking out all identities.

If one transmitter is locked out and a second one also needs to be locked out, then the power will have to be switched off and links LK1-LK4 repositioned for that transmitter identity. The power must then be re-applied with S1 pressed.

Once the lockout procedure has been completed, you must relocate links LK1-LK4 to their correct positions for the receiver functions that you wish to select. It is then best to test that everything is correct by pressing the switches on another (non-locked-out) transmitter and verifying that the receiver operates as expected.

Where To Get The Bits

Suitable reed switch assemblies, door strikes and sirens are available from Jaycar electronics. They can also supply kits for this project.

The parts available from Jaycar include: (1) the LA-5072 normally closed (NC) reed switch magnet assembly; (2) the LA-5078 door strike; and (3) the LA-5255 and LA-5256 piezo sirens.

Above right: door strike available from Jaycar.

Calculating The Rolling Code

The rolling code for the infrared transmitter comprises four start bits, a 48-bit code and four stop bits.

A calculation comprising a multiplier and an increment value is used to generate the 48-bit code. First, you start with a number (called the seed), then you multiply this seed by the multiplier and then add the increment. The result becomes the next value for random code.

Normally, if the calculation is continued, the random code will become larger and larger as we multiply and then add the increment value. However, this is prevented by limiting the seed value used in the calculation to a certain width; 32 bits in this case.

In practice then, the 24-bit multiplier multiplies the 32-bit seed. The 8-bit increment value is then added and the result is limited to 48-bits by eliminating the more significant bits. This resulting 48-bit code is the code used for the rolling code transmission. In addition, the order of transmission for these bits is jumbled using an 8-bit scramble code with 32 possible combinations.

The calculations do not necessarily produce random numbers, but they do produce variations from one transmission to the next. However, in some cases, the result could converge to settle at the same value, so it is important to check this and make sure the calculations do give diverging values each time.

To do this, the result of each calculation is compared to the last value to ensure it is not repeated. If the result is the same as before, the duplicate code is not transmitted and a new calculation is made after incrementing the result. Subsequent calculations will then begin to diverge again.

Randomisation

To avoid conflict, each transmitter must have a unique set of parameters for making the rolling code calculations. As a result, we have included a 'randomisation' function, whereby the multiplier value, the increment value, the scramble value and the seed value are all changed in a relatively random way.

There are 16.7 million multipliers available and 54 possible increment values. Together with the 32 scramble variations, these provide 29 billion different combinations. In addition, the minimum multiplier value is 8192 to ensure a significant change in value with each calculation.

Even if two transmitters do end up with the same parameter values, the fact that the seed value is a part of the calculation means that you need to be within 200 values of the correct value in order to unlock someone else's lock. The probability of this is 2^{24} divided by 200, or one in 83,000. This is in addition to the one in 29 billion chance of having the same parameter values!

There are up to 16 different transmitters that can be used with the one receiver, and each transmitter uses a different set of seed, multiplier, increment and scramble values. The transmitter sends out its identification code that is embedded in the rolling code, so the receiver knows which set of values it must use in the calculation for each transmitter.

When the transmitter is sending synchronising code to the receiver, it sends the 8-bit identifier, the 24-bit seed, the 24-bit multiplier, the 8-bit increment value and the 8-bit scramble values. The identifier value is also stored, so that the receiver knows that this identity has been synchronised. An identity that has not been synchronised will not operate the receiver.

Once the receiver has these parameters, the transmitter and receiver will remain in lock because they use the same calculation values.

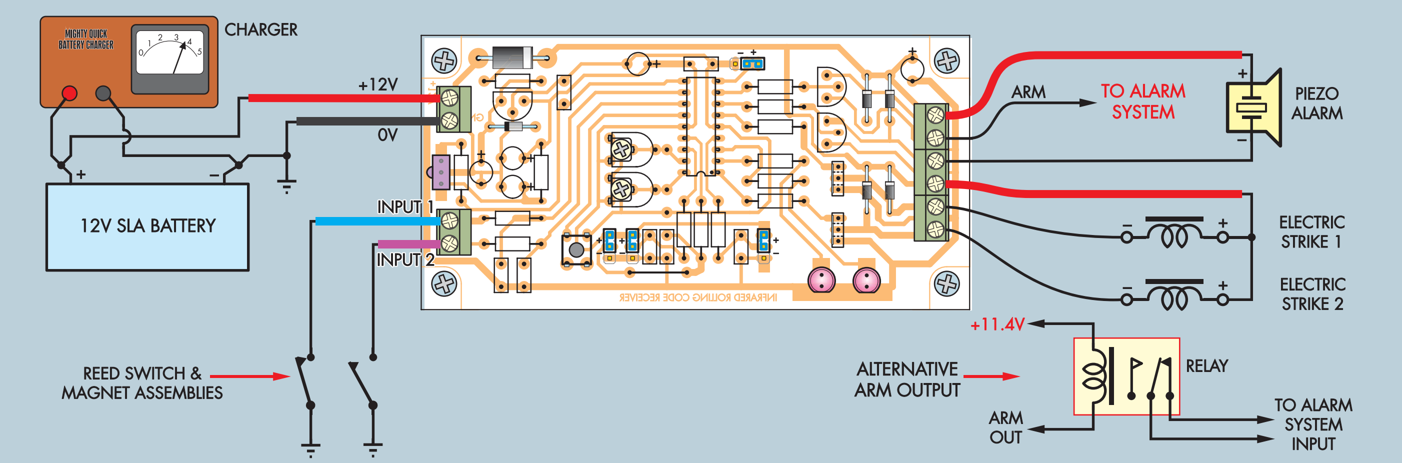


Fig.7: here's how to connect the receiver in a typical installation. Note that you can use both NO (normally open) and NC (normally closed) sensors on the alarm inputs (Input1 and Input2). The battery charger keeps the battery topped up.

Undoing lockout

It's easy to get a locked out transmitter to operate the receiver again (ie, to unlock it). Just synchronise the transmitter with the receiver and all will be back to normal.

Installation

The Rolling Code Keyless Entry System is suitable for use in homes, factories and cars. Fig.7 shows how to wire the unit for a typical installation. **Note that IRD1 must be shielded from direct sunlight, otherwise the reception range will be severely affected.**

In some cases, it may be necessary to connect the infrared receiver (IRD1) via extended leads using twin-core

Table 4: Arm output (LK4)		
LK4	+	–
	Arm output low on arm, open on disarm	Arm output open on arm, low on disarm

Table 5: Receiver lockout selections				
Lockout Identity	LK1	LK2	LK3	LK4
1	+	+	+	+
2	+	+	+	–
3	+	+	–	+
4	+	+	–	–
5	+	–	+	+
6	+	–	+	-
7	+	–	–	+
8	+	–	–	–
9	–	+	+	+
10	–	+	+	–
11	–	+	–	+
12	–	+	–	–
13	–	–	+	+
14	–	–	+	–
15	–	–	–	+
16	–	–	–	–

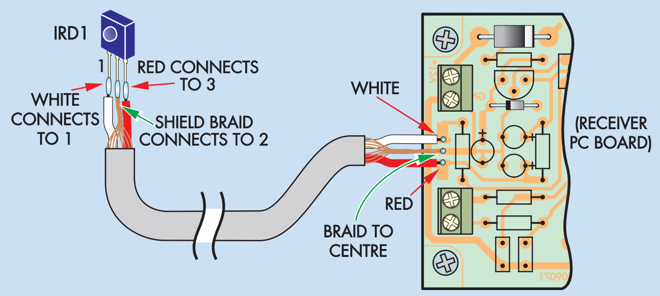
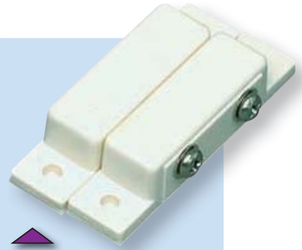


Fig.8: the IR receiver (IRD1) can be connected via twin-core shielded cable as shown here.



Above: you can buy both NO and NC reed switch assemblies.

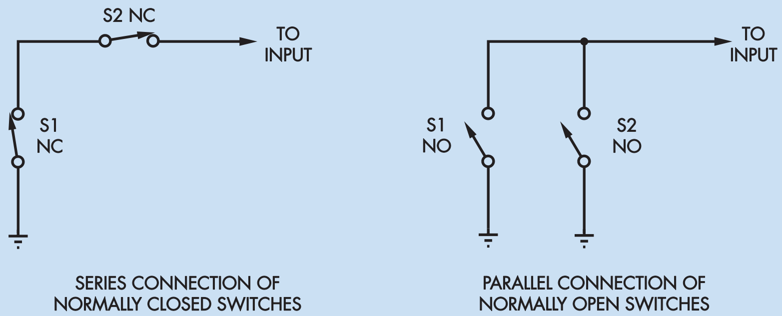


Fig.9: here's how to wire the two different sensor types (NO and NC) to the alarm inputs on the receiver board.

shielded cable (eg, if the receiver is mounted on one side of a wall, but infrared reception is needed on the other side). Fig.8 shows how this is done.

The two alarm inputs (Input1 and Input2) can be used in conjunction with reed switch magnet assemblies that change state when a door or window is opened or closed. You can use either normally closed (NC) or normally open (NO) types. (See last month's weather station project for the lowdown on reed switches.)

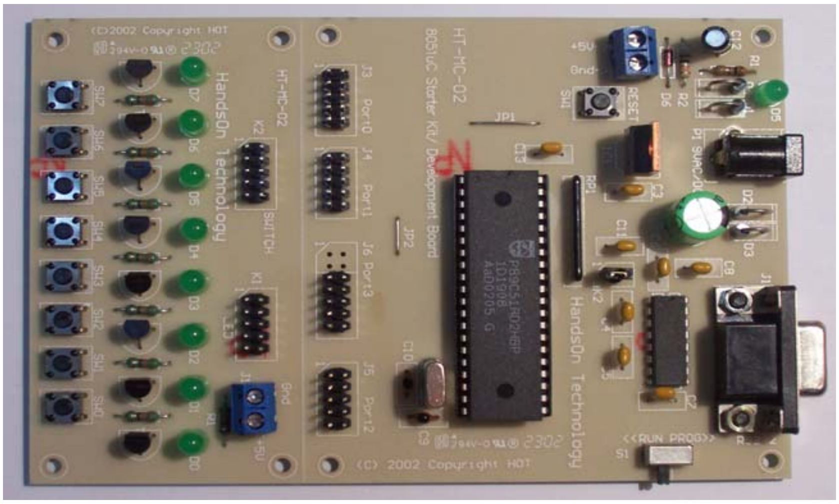
As shown in Fig.9, NC types are connected in series, while NO types are connected in parallel. However, for best security, use only one sensor per input.

Alternatively, you can use a PIR detector or a glass breakage detector on one or both of the inputs. ***EPE***

Reproduced by arrangement with SILICON CHIP magazine 2009.
www.siliconchip.com.au

Low Cost 8051μC Starter Kit/ Development Board *HT-MC-02*

HT-MC-02 *is an ideal platform for small to medium scale embedded systems development and quick 8051 embedded design prototyping.* ***HT-MC-02*** *can be used as stand-alone 8051μC Flash programmer or as a development, prototyping and educational platform*



Main Features:

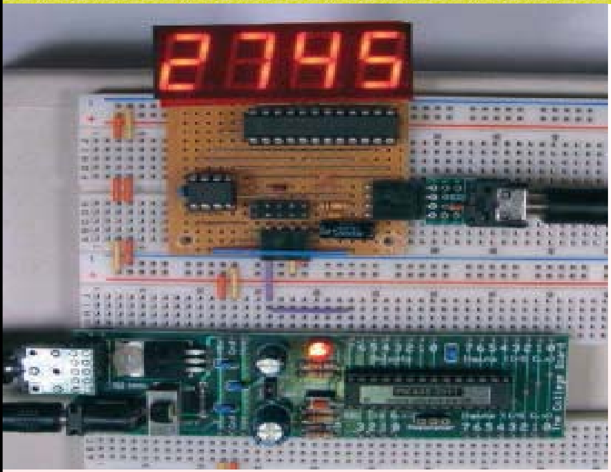
- 8051 Central Processing Unit.
- On-chip Flash Program Memory with In-System Programming (ISP) and In Application Programming (IAP) capability.
- Boot ROM contains low level Flash programming routines for downloading code via the RS232.
- Flash memory reliably stores program code even after 10,000 erase and program cycles.
- 10-year minimum data retention.
- Programmable security for the code in the Flash. The security feature protects against software piracy and prevents the contents of the Flash from being read.
- 4 level priority interrupt & 7 interrupt sources.
- 32 general purpose I/O pins connected to 10pins header connectors for easy I/O pins access.
- Full-duplex enhanced UART **–** Framing error detection Automatic address recognition.
- Programmable Counter Array (PCA) & Pulse Width Modulation (PWM).
- Three 16-bits timer/event counters.
- AC/DC (9~12V) power supply – easily available from wall socket power adapter.
- On board stabilized +5Vdc for other external interface circuit power supply.
- Included 8x LEDs and pushbuttons test board (free with ***HT-MC-02*** while stock last) for fast simple code testing.
- Industrial popular window ***Keil*** C compiler and assembler included (Eval. version).
- Free ***Flash Magic*** Windows software for easy program code down loading.

PLEASE READ ***HT-MC-02 GETTING STARTED MANUAL*** BEFORE OPERATE THIS BOARD
INSTALL ACROBAT READER (AcrobatReader705 Application) TO OPEN AND PRINT ALL DOCUMENTS

HandsOn Technology

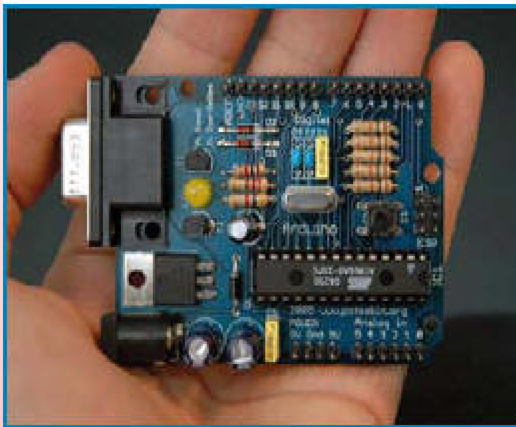
http://www.handsontec.com

creativity for tomorrow's better living...



HandsOn Technology is a manufacturer of high quality educational and professional electronics kits and modules, uController development/evaluation boards. Inside you will find Electronic Kits and fully assembled and tested Modules for all skill levels. Please check back with us regularly as we will be adding many new kits and products to the site in the near future.
Do you want to stay up to date with electronics and computer technology? Always looking for useful hints, tips and interesting offers?

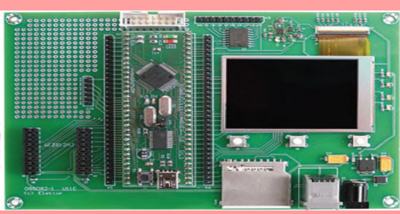
Inspiration and goals...



HandsOn Technology **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...**
HandsOn Technology provides Designs, ideas and solutions for today's engineers and electronics hobbyists.

Creativity for tomorrow's better living...

HandsOn Technology believes everyone should have the tools, hardware, and resources to play with cool electronic gadgetry. HandsOn Technology's goal is to get our "hands On" current technology and information and pass it on to you! We set out to make finding the parts and information you need easier, more intuitive, and affordable so you can create your awesome projects. By getting technology in your hands, we think everyone is better off
We here at HandsOn like to think that we exist in the same group as our customers >> curious students, engineers, prototypers, and hobbyists who love to create and share. We are snowboarders and rock-climbers, painters and musicians, engineers and writers - but we all have one thing in common...we love electronics! We want to use electronics to make art projects, gadgets, and robots. We live, eat, and breathe this stuff!!
If you have more questions, go ahead and poke around the website, or send an email to sales@handsontec.com. And as always, feel free to let your geek shine - around here, we encourage it...



http://www.handsontec.com