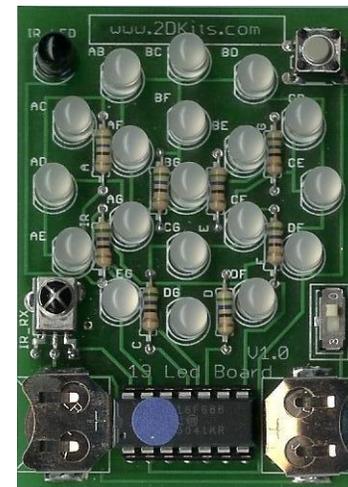


# The Blinkie



Circuit #6c

# **Table of Contents**

Introduction  
Parts List  
Assembly Procedure  
Troubleshooting  
Soldering Hints  
How It Works  
Display Mode  
IR Test mode  
Program mode  
EEProm Pattern Format  
Sample Programming



## The Blinkie

The heart of this circuit is a 16F688 PIC produced by a company called Microchip. A PIC is a tiny, yet surprisingly powerful little computer. By itself, it can't do much – it needs somehow to interact with the world – we are going to do this by giving it senses:

- Sight – an IR receiver
- Touch – a push button

and ways to communicate:

- To us – 19 bi-color light emitting diodes (LEDs)
- To other PICs – an infrared (IR) LED

By building this blinkie, we hope you have a lot of fun, as well as learn how easy it is to assemble and solder a circuit, as well as gain a desire to learn more!

### Parts List

1 – 16F688 PIC chip	8 – 56 $\Omega$ resistors
19 – Bi-color LEDs	1 – push button
1 – Infrared (IR) detector	4 – 1.5V button batteries
1 – IR LED	1 – sliding power switch
1 – 14 pin socket	1 – board w/battery clips

## Assembly Procedure

Refer to *Figures 1 – 2* for the following steps.

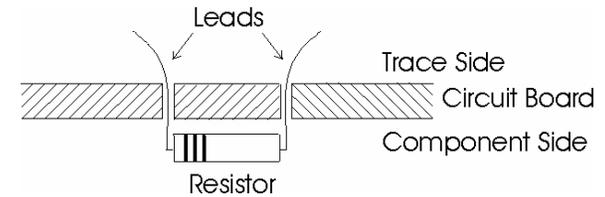
- ( ) 1. Orient the circuit board with the foil (trace) side down and the holes for the switch along the bottom.



*Figure 1*

- ( ) 2. Mount the IC socket in the 14 holes near the top of the board, as shown in *Figure 1*, with the notch pointing toward the left of the board.
- ( ) 3. Mount the IR detector up and to the left of the IC socket, as shown in *Figure 1*.
- ( ) 4. Mount the push button on the upper right of the board, just left of center, as shown in *Figure 1*.
- ( ) 5. Mount the switch up and to the right of the IC socket, as shown in *Figure 1*.

- ( ) 6. Flip the board over and solder all the leads to the circuit board pads. Clip off the excess leads after soldering.
- ( ) 7. Flip the board back to the component side.



*Figure 2*

- ( ) 8. Mount the  $56\Omega$  resistors in the 2 holes as shown by the silk screening on the board. Bend the leads slightly to hold the resistor in place, as shown in *Figure 2*.
- ( ) 9. Flip the board over and solder all the leads to the circuit board pads. Clip off the excess leads after soldering.
- ( ) 10. Flip the board back to the component side.
- ( ) 11. Insert the bi-bolor LEDs and the IR LED, as shown in *Figure 1*. As you insert them, the longer lead will go to the **right** side of the board. Bend the leads slightly to hold the LED's in place, as shown in *Figure 3*. You may need to place half the LEDs, solder and clip the leads, before you can place and solder the rest.
- ( ) 12. Flip the board over and solder all the leads to the circuit board pads. Clip off the excess leads after soldering.

- ( ) 13. Place the pin-back clip on the large circuit board pad at the top of the board and solder in place.
- ( ) 14. Flip the board back to the component side and install the PIC chip. Pin one goes to the left side of the socket.
- ( ) 15. Install the four button cell batteries in the battery holders. The positive side (+) of the batteries face up.
- ( ) 16. Turn on the switch and watch your blinkie go!

### **Troubleshooting**

If your blinkie doesn't flash, then you'll need to do a little troubleshooting to finish your project. The following steps should isolate most problems.

1. Recheck your solder connections. Cold solder joints and broken joints will cause erratic performance or failure. Reheat any questionable solder connections until they flow and look shiny and secure.
2. Check the values and orientation of your parts. A reversed LED won't light, and a reversed PIC will not work at all. (Resistors do not have polarity.)
3. Check for bits of solder, wire ends, or other foreign matter which may be lodged in the wiring.

## Soldering Hints

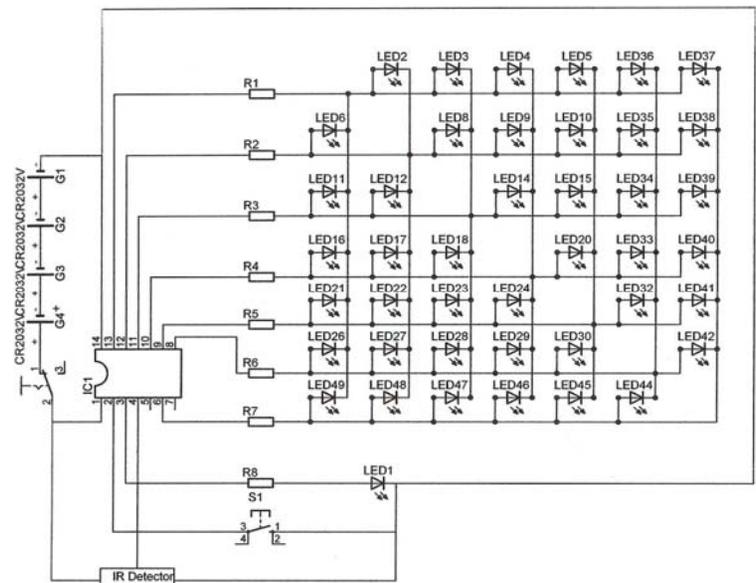
Soldering is not like gluing; the solder forms an alloy with the metals to be connected that creates a stable electrical path and a certain amount of mechanical attachment. For the small connections we will be doing on this project, we will be using a 25 or 30 watt soldering iron works well. We will be using rosin core solder – the acid core solder sold for plumbing would eat your components in a short time.

Here's how to make a good joint:

- Prepare the joint. Bend the component lead slightly after it passes through the printed circuit board (this helps hold it in place while soldering).
- Prepare the tool. The soldering iron should be up to temperature. Clean the tip by quickly brushing it against a damp sponge or cloth. Melt a little solder (a 2mm length) onto the tip so it's shiny. This is called "tinning". The solder coating helps conduct heat from the tip to the joint.
- Place the tip in contact with the component lead and the printed circuit board pad.
- Place the solder against the joint directly opposite the tool. It should melt within 2 seconds, and flow around the joint. If it takes longer than that, you're not getting enough heat into the joint.
- Keep the soldering iron in place until the solder flows freely and completely covers the joint. If the heat is removed too soon, the solder will tend to "ball up" and not stick well to the conductors. The solder joint should look "wetted", with concave shapes.

- Let the joint cool without movement at room temperature. This usually takes only a few seconds.
- If a joint is moved before it cools, it will take on a dull, satin look that is characteristic of a cold solder joint. A cold solder joint is fragile and conducts poorly – reheat the joint until the solder flows freely, and hold it still until it cools.
- Keep the tip of the soldering iron clean. Wipe off flux and excess solder regularly in the damp sponge or cloth, and re-tin if needed.

### How It Works



Blinkie Circuit Diagram

The PIC chip runs a program that outputs a bit pattern to the I/O pins. There is a short delay after each pattern is sent, before the next one is sent. These patterns give the illusion that the light is chasing itself around a circle. The PIC chip includes a built-in RC (resistor/capacitor) oscillator circuit that provides clock timing for the microprocessor.

The PIC chip is programmed using PIC assembler (an assembler type programming language). The program is burned into the FLASH memory on the chip. The IR receiver picks up IR (infra-red) signals from other blinkies, which causes the PIC to change the blink pattern/rate.

### **Display Mode**

When turned on, Display mode is the default state.

Button press – shows what pattern is running (in binary on leds)

Hold button down – shows what pattern, then increments pattern counter

Remote control keys

“channel +” moves to next pattern

“channel – “ moves back one pattern

“0” moves to 1<sup>st</sup> pattern

“on/off” enters program mode

If the demo mode bit is set in the eeprom, the blinkie will display 255 entries of the pattern, then advance to the next pattern.

If the demo mode bit is not set in the eeprom, the blinkie will display 255 entries of the pattern, then send broadcast an infection IR command setting any blinkie nearby to its current pattern, or the infection

pattern number it last received, which ever has been updated last.

### IR Test mode

Mode entered by holding button down when turning on power.

Unit will show current IR state machine index value in red. (binary counter)

The max state for the IR state machine index will be in green. (binary counter)

Button press clears resets the “max state” and resets the machine index to 0. It also sends a “TV” remote “on/off” IR sequence. Will turn on/off any nearby TV if it uses the RC5 IR command format.

Getting a valid IR RC-5 command will cause the device to:

- 1) flash all the leds twice
- 2) display the RC-5 command
- 3) return to reporting the state machine index info

Green will be the device code, a TV remote used code “0”

Red will be the command code.

## Program mode

Dwayne created the programming mode. After trying it both Dale and Dwayne agree that its not user friendly. We question how many people will ever use it but we will document it anyway!

Mode entered by pressing “on/off” on remote while in Display mode.

### Remote control keys

“channel +” moves to next memory location

“channel – “ moves back one memory location

“0” – “9” used to enter numeric value for location

“on/off” exits program mode

“mute” or “Volume –“ sets value to zero

The green leds show the current address, the red leds show the value stored.

### Example

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1

Shows Adress of 2, value of 170 (0xAA)

## EEProm Pattern Format

For 19 bi-color led board

Ad	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit1	Bit0
0	Pattern Type	dem o	Pattern type (zero disables eeprom pattern)						
1	Speed	"0" is the fastest, "255" is the slowest							
2	Entry 0	G01	R01	G02	R02	G03	R03	G04	R04
3	Entry 0	G05	R05	G06	R06	G07	R07	G08	R08
4	Entry 0	G09	R09	G10	R10	G11	R11	G12	R12
5	Entry 0	G13	R13	G14	R14	G15	R15	G16	R16
6	Entry 0	G17	R17	G18	R18	G19	R19	---	EoP
7	Entry 1	G01	R01	G02	R02	G03	R03	G04	R04
8	Entry 1	G05	R05	G06	R06	G07	R07	G08	R08
9	Entry 1	G09	R09	G10	R10	G11	R11	G12	R12
10	Entry 1	G13	R13	G14	R14	G15	R15	G16	R16
11	Entry 1	G17	R17	G18	R18	G19	R19	---	EoP
...	...								
...	...								
...	...								
247	Entry 50	G01	R01	G02	R02	G03	R03	G04	R04
248	Entry 50	G05	R05	G06	R06	G07	R07	G08	R08
249	Entry 50	G09	R09	G10	R10	G11	R11	G12	R12
250	Entry 50	G13	R13	G14	R14	G15	R15	G16	R16
251	Entry 50	G17	R17	G18	R18	G19	R19	---	EoP
252	---	---	---	---	---	---	---	---	---
253	---	---	---	---	---	---	---	---	---
254	---	---	---	---	---	---	---	---	---
255	---	---	---	---	---	---	---	---	---

## Sample Programming

### Quick speed hack.

- 1) in display mode
- 2) remote "on/off" button (program mode)

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 1								Address = 0							

- 3) remote "channel +" (next address)

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 48 (0x30)								Address = 1							

- 4) remote "mute" or "volume -" (clear value)

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 0								Address = 1							

- 5) remote "5"

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 5								Address = 1							

- 6) remote "on/off" (exit programming)

- 7) The pattern should be running a little faster now.

**Turn demo mode on.**

- 1) in display mode
- 2) remote "on/off" button (program mode)

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 1								Address = 0							

- 3) remote "volume -" (clear value)

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 0								Address = 0							

- 4) remote "1"

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 1								Address = 0							

- 5) remote "2"

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 12 (0x0C)								Address = 0							

- 6) remote "9"

Green (Value)								Red (Address)							
8	7	6	5	4	3	2	1	8	7	6	5	4	3	2	1
Value = 129 (0x71)															

- 7) remote "on/off" (exit programming)

8) Blinkie will now cycle through the patterns, and will not infection other blinkies.