

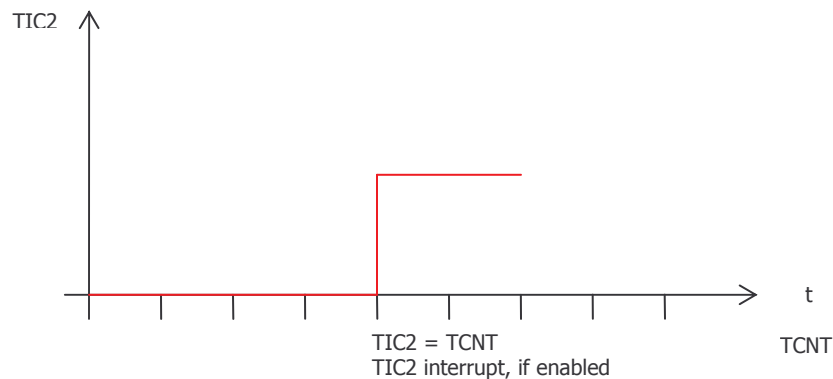
Timer Registers

1. Control Registers
2. Mask registers
3. Flag registers
4. Parameters
5. Results

Timer Input Capture

Capture the value of TCNT when a voltage occurs on the TIC pin and move TCNT to the TIC register. Fire the TIC interrupt, if enabled.

It's a very simple function and does not do much.



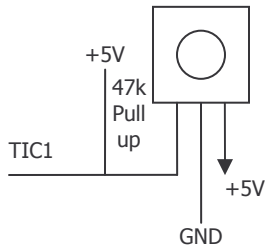
Timer Function Init

1. Initialize the TCLK register for that function.
2. Initialize any function parameters for that function, if they have any.
3. Do an initial clear to erase the first interrupt. This is done in the appropriate TFLG register.
 - a. This takes care of past events that have occurred so that they don't immediately interrupt the CPU when I enable the interrupt.
4. Enable the interrupt, if using them. This is in one of the TMSK registers.

The very last thing boot should do is a CLI to enable interrupts.

There are two levels of masking – the physical, on device masking, controlled by TMSK, and the software mask that is toggled using SEI and CLI.

Infrared Receiver



Traditionally modulate a 40 KHz sine wave. Its frequency is modulated when you push a button on the remote control. The receiver is a demodulator which produces a digital pulse of ones and zeros.

Open drain output – only drives to 0. It does not drive to +5V. When it disconnects, the pull up resistor pulls the voltage up to +5V.

The demodulator goes low and high based on the frequency of the sine wave it receives.

When an additional frequency is there, it is 0. When just the 40KHz carrier frequency is there, it goes high.

The question is: What are the frequencies that will show up? It depends on the remote control being used.

The concept of remote control has been around since the 1800s. Granted, they didn't use infrared.

The sensor is performing A/D conversions in a certain sense. It receives an analog input and then turns either on or off. Digital demodulation behavior.

The protocol used varies based on the remote control. The Sony remote control protocol is considered one of the simplest standard protocols. This is the preferred modus operandi.

Sony Remote Control Protocol

Start pulse + 7 command bits + 5 address bits.

The 5 address bits are unique. If it is your TV remote, they will be one value. If it is your VCR, they will be another value. The address bits specify the device.

The actual key pushed is given by the 7 command bits. We get a change every 600 microseconds. That is the basic unit of modulation.

Because our component pulls low, it is the width of the low time that determines whether the value given is a 1 or a 0.

We only need to capture the first 7 bits because they

How does this apply to TIC?

What if on every edge, we program TIC to capture time. The ISR simply moves the TIC value into an array of numbers. When the ISR has captured the 8th one, set a global flag so that the remote control process knows that there is a command. It starts looking at the value sin the array, subtracts them, looks at the width, and then gives a 0 or a 1.

```
TIMEVAL[8]
```

The ISR:

```
RTIMEVAL[i] = TIC1  
i++
```