

## RCin

### Syntax

*Variable* = **RCin** *Pin*, *State*

### Overview

Count time while pin remains in *state*, usually used to measure the charge/ discharge time of resistor/capacitor (RC) circuit.

### Operators

**Pin** is a Port.Pin constant that specifies the I/O pin to use. This pin will be placed into input mode and left in that state when the instruction finishes.

**State** is a variable or constant (1 or 0) that will end the Rcin period. Text, High or Low may also be used instead of 1 or 0.

**Variable** is a variable in which the time measurement will be stored.

### Example

```
Dim Result as Word           ' Word variable to hold result.
High PORTB.0                 ' Discharge the cap
DelayMs 1                    ' Wait for 1 ms.
Result = RCin PORTB.0, High  ' Measure RC charge time.
Print Dec Result, " "        ' Display the value on an LCD.
```

### Notes

The resolution of **RCin** is dependent upon the oscillator frequency. If a 4MHz oscillator is used, the time in state is returned in 10us increments. If a 20MHz oscillator is used, the time in state will have a 2us resolution. Declaring an Xtal value has no effect on **RCin**. The resolution always changes with the actual oscillator speed. If the pin never changes state 0 is returned.

When **RCin** executes, it starts a counter. The counter stops as soon as the specified pin is no longer in *State* (0 or 1). If *pin* is not in *State* when the instruction executes, **RCin** will return 1 in *Variable*, since the instruction requires one timing cycle to discover this fact. If pin remains in *State* longer than 65535 timing cycles **RCin** returns 0.

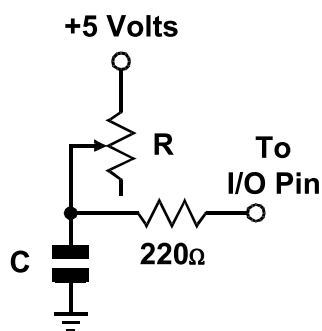


Figure A

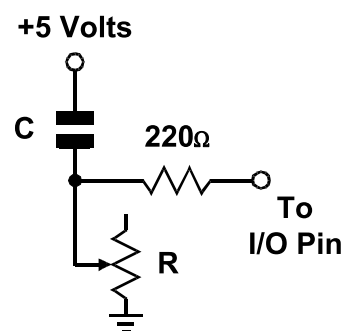


Figure B

The diagrams above show two suitable RC circuits for use with **RCin**. The circuit in figure B is preferred, because the PICmicro's logic threshold is approximately 1.5 volts. This means that the voltage seen by the pin will start at 5V then fall to 1.5V (a span of 3.5V) before **RCin** stops. With the circuit in figure A, the voltage will start at 0V and rise to 1.5V (spanning only 1.5V) before **RCin** stops.