

Basic Express Application Note

Using RTime to Measure Resistance

Introduction

One common use for I/O pins is to measure the analog value of a variable resistance. Although a built-in ADC (Analog to Digital Converter) is perhaps the easiest way to do this, it is also possible to use a digital I/O pin. This can be useful in cases where you don't have enough ADC channels, or if a particular processor doesn't have ADC capability.

Procedure RTime

BasicX provides a special procedure called RTime for this purpose. RTime measures the time it takes for a pin to change state to a defined value. By connecting a fixed capacitor and a variable resistor in an RC circuit, you can use an I/O pin to measure the value of the variable resistor, which might be a device such as a potentiometer or thermistor.

There are two common ways to wire an RTime system. The first is to tie the variable resistor to ground. Figure 1 shows this configuration. The advantage here is less chance of damage from static electricity:

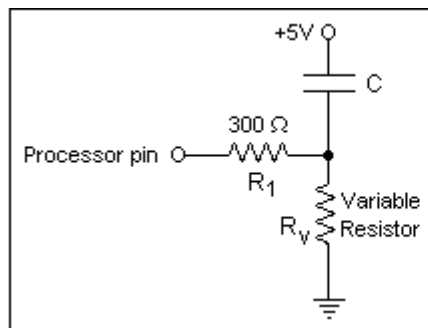


Figure 1

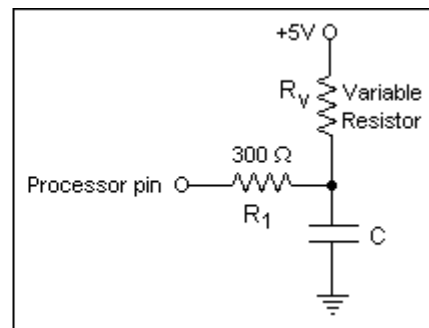


Figure 2

The second configuration is shown in Figure 2. Here we use the opposite connection, where the capacitor C is tied to ground and the variable resistor R_V is tied to 5 volts:

In both circuits resistor R_I is there to protect the BasicX chip's output driver from driving too much current when charging the capacitor.

To take a sample, the capacitor is first discharged by taking the pin to the correct state. In the case of Figure 1, the pin needs to be taken high (+5 V) to produce essentially 0 volts across the capacitor, which causes it to discharge. In figure 2 the desired state is low or ground. In either case, the capacitor is discharged for at least 4 time constants, which is $4 R_I C$. Once this time has elapsed, the capacitor

should be sufficiently discharged.

At this point, the RTime procedure is called. RTime releases the capacitor by making the pin a tristate input. The procedure then measures how long it takes to charge the capacitor to the trip point of the input pin. This time interval is a function of R_V . If R_I is negligible compared to R_V , the time interval is proportional to R_V .

You may need to experiment with different values of capacitors to optimize the process.

Example 1, using the circuit from Figure 1:

```
Dim TimeInterval As Single

' Discharge the capacitor on pin 15.
Call PutPin(15, bxOutputHigh)
Call Delay(20.0E-3) ' Wait about 20 ms.

' Wait for a logic low on pin 15.
Call RTime(15, 1, TimeInterval)
```

Example 2, using the circuit from Figure 2:

```
Dim TimeInterval As Single

' Discharge the capacitor on pin 15.
Call PutPin(15, bxOutputLow)
Call Delay(20.0E-3) ' Wait about 20 ms.

' Wait for a logic high on pin 15.
Call RTime(15, 0, TimeInterval)
```

Potentiometer example

Symbols in this section

τ	Time constant
C	Capacitance
I	Current through pot and capacitor
I_0	Initial current through component
R_I	Resistance of fixed resistor
R_V	Resistance of pot
t	Time
V_0	Initial voltage across component
V_C	Voltage across capacitor
V_{RI}	Voltage across fixed resistor R_I
V_{RV}	Voltage across pot
V_{trip}	Voltage trip point for the input pin

Circuit behavior

In this example, we'll use the same configuration as Figure 1. A potentiometer is used as the variable resistor R_V and has a maximum resistance of 50 k Ω (see Figure 3 below):

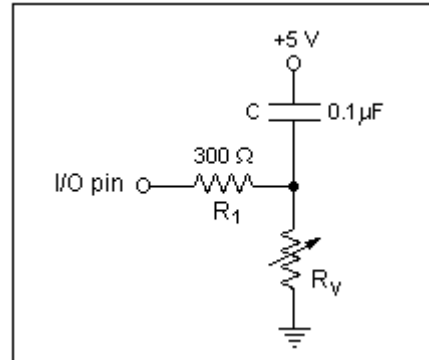


Figure 3

The following derivation assumes R_V is large compared to R_I . That is, $R_I / R_V \ll 1$.

Discharging capacitor -- the first step is to discharge the capacitor by raising the I/O pin. The pin needs to be held high long enough to allow V_C to decay to a negligible value.

For initial conditions, assume the voltage across the capacitor has reached a steady-state 5 V. Also assume the pin acts like an ideal switch. At time $t = 0$, the pin is taken high, and the initial current through R_I is

$$I_0 = (V_0 / R_I) = (5 \text{ V}) / (300 \Omega) = 16.7 \text{ mA}$$

The voltage V_C across the capacitor at time $t > 0$ is

$$V_C = \frac{I_0 R_I}{e^{(t/\tau)}}$$

Here, the time constant $\tau = (300 \Omega) (0.1 \mu\text{F}) = 30.0 \mu\text{s}$. A rule of thumb is to allow the capacitor to discharge for at least 4 time constants (120.0 μs , in this case), which means the voltage will decay as follows:

$$V_{RI} = \frac{V_0}{e^{(4\tau/\tau)}}$$

$$= (5 \text{ V}) / e^4 = 91.6 \text{ mV} \cong 1.8 \% \text{ of maximum}$$

Here we are neglecting the effect of R_V on the capacitor discharge. This is a reasonable approximation only if R_I is negligible compared to R_V .

Measuring time delay -- now that the capacitor is discharged, we set the processor I/O pin to input-tristate, which essentially disconnects R_I from the circuit. At new time $t = 0$, the initial current I_0 through capacitor C and potentiometer R_V is

$$I_0 = (V_0 / R_V) = (5 \text{ V}) / (50 \text{ 000 } \Omega) = 100 \text{ } \mu\text{A}.$$

At time $t > 0$, the current I is

$$I = \frac{I_0}{e^{(t/\tau)}}$$

The time constant $\tau = R_V C = (50 \text{ 000 } \Omega) (0.1 \text{ } \mu\text{F}) = 5.00 \text{ ms}$. This represents is the time it takes for the current through the circuit to drop by $(1 - 1/e)$ or about 63 % below its maximum value.

The voltage V_{RV} across the pot at time $t > 0$ is

$$V_{RV} = \frac{I_0 R_V}{e^{(t/\tau)}}$$

Now we need to know the trip point V_{trip} for the input pin, which is the voltage at which the pin transitions from logic high to logic low. This point can vary somewhat from pin to pin and needs to be determined empirically.

If we know V_{trip} , the resistance of the pot becomes

$$R_V = \frac{t}{C \ln (V_0 / V_{trip})}$$

For the Atmel chip used in a BasicX system, V_{trip} is in the neighborhood of 2.5 V and does not appear to be sensitive to temperature changes. Assuming $V_0 = 5 \text{ V}$ and $V_{trip} = 2.5 \text{ V}$, the resistance is therefore

$$R_V = 1.4427 (t / C)$$

In other words, for a constant capacitance, the resistance is linearly proportional to time.

In this example, the time required to charge the capacitor to $V_{trip} = 2.5 \text{ V}$ is

$$t = (0.1 \text{ } \mu\text{F}) (50 \text{ k}\Omega) / 1.4427 = 3.47 \text{ ms}.$$

Optimization -- In this particular application, there is a tradeoff between accuracy and the time required for measurements. Higher accuracy requires a longer measurement time. RCtime has a full-scale value of about 71.1 ms and resolution of about 1.085 μs ($1 / 65 \text{ 535}$ of full scale), which needs to be taken into account.

For a given resistance, once you've determined the optimum trip time t , this is how to size the capacitor:

$$C = \frac{t}{R_V \ln (V_0 / V_{trip})}$$

Low values of R_V -- the derivations in this section assume R_I is negligible compared to R_V . If this is not the case, the equations are somewhat more involved. In particular, as the ratio R_I / R_V increases, it becomes less valid to assume the capacitor is fully discharged at the beginning of each measurement cycle. In the limiting case where $R_V = 0$, the capacitor never discharges at all.

Code example

```
Sub GetResistance( _
    ByRef Resistance As Single)

    ' Measures the resistance of a pot attached to an I/O pin.

    Const PotPin As Byte = 15
    Const DischargeTime As Single = 120.0E-6 ' Seconds
    Const Capacitance As Single = 0.1E-6     ' Farads
    Const TripVoltage As Single = 2.5       ' Volts
    Const InitialVoltage As Single = 5.0    ' Volts

    Dim TimeInterval As Single, K As Single

    ' Raise the pin and discharge the capacitor.
    Call PutPin(PotPin, bxOutputHigh)
    Call Delay(DischargeTime)

    ' Set the pin to input-tristate, then measure how long the pin stays
    ' at logic-high.
    Call Rctime(PotPin, 1, TimeInterval)

    K = 1.0 / (Capacitance * Log(InitialVoltage / TripVoltage))

    Resistance = TimeInterval * K

End Sub
```

This example is used with the circuit in Figure 3. Source code can be found in file Resistance.bas.

© 1998-2001 by NetMedia, Inc. All rights reserved.

Basic Express, BasicX, BX-01, BX-24 and BX-35 are trademarks of NetMedia, Inc.

All other trademarks are the property of their respective owners.

2.00.A