

Checking Battery Condition and Multiplexing I/O Lines

Two Mini Applications, by Scott Edwards

THIS month's first application was contributed by Guy Marsden of ART TEC, Oakland, California. Guy, a former visual effects specialist for the movies (Star Trek the Motion Picture, Ghostbusters, 2010) makes his living helping artists incorporate electronics into their work. One such work was powered by 12-volt lead-acid battery, and Guy devised a simple, effective way for the Stamp to monitor the battery and sound an alarm at charging time.

Figure 1 and listing 1 show Guy's method. He's using the brightness of the LED as a relative indication of battery voltage. The Stamp reads this brightness as a variable resistance across the photocell. When the photocell resistance exceeds a preset limit, the Stamp sounds an alarm.

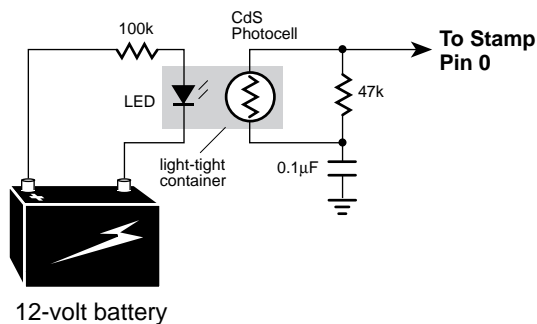


Figure 1. Battery-monitor setup.

Although Guy used a commercially made optoisolator, you should get similar results with a roll-your-own version. Just mount an LED

facing a photocell and cover the assembly to block outside light.

This approach can be used more generally to convert a variable voltage into a form that can be read with the Pot command. Just keep a couple of LED characteristics in mind:

- LEDs have a relatively high forward voltage of approximately 1.5 to 2.1 volts. They don't light at all if the input voltage is below their forward voltage. They require a series resistor to make sure that the current through them is reasonable (usually up to 25 mA continuous). To calculate the value of this resistor, use the following formula:

Series Resistor = (Input Voltage - LED Forward Voltage) / LED current

For example, suppose your input is 9 volts. Just guessing, you figure the LED forward voltage at 1.9 volts. And you decide that 10 milliamperes is a safe bet for current. (Remember that most electronic formulas involving current want the value in amperes; 10 mA = 0.01 amperes.) The series resistor should be $(9 - 1.9) / 0.01 = 710$ ohms. Pick the closest standard value (or the closest value that you have on hand) and you're done.

- An LED's forward voltage decreases with temperature, increasing the current that will pass through it with a given series resistance. More current usually means more light output, but brightness decreases with temperature. To make matters worse, data isn't available for most common LEDs. Even when data is

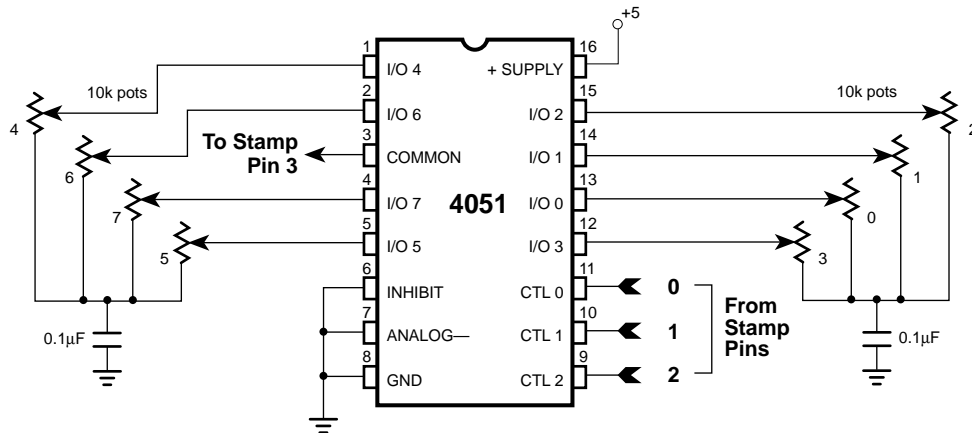


Figure 2. Four Stamp pins read eight pots through a 4051 analog multiplexer.

available, it doesn't usually provide much detail on these effects.

So I'm wondering: Is Guy's circuit usable across a wide range of temperatures, or is it restricted to the great indoors? I started to set up an experiment to find out, then decided that a contest would be more fun.

Here's the deal: Design and construct a Stamp project to test the output of Guy's circuit over a range of at least 32 to 90 degrees F with a steady input voltage. (Temperature control can be as simple as an ice bucket in which the ice is allowed to melt.) Send me a drawing of the circuit, the BASIC program listing, a brief description of your procedure, and a copy of the data generated by it. I'll accept entries until October 31, 1995.

The reader who submits the best entry (my call) will receive the newest version of the LCD Serial Backpack display with 16x1 LCD. This latest model features switchable 2400- or 9600-baud operation, low-voltage reset circuit, improved contrast control, and an easy-to-read 16-character display. All runners-up will receive a coupon good for 10 percent off anything I sell; see the Sources box.

Double Your I/O

The next application is for those of you who can never have enough input/output (I/O) lines. Although I'm demonstrating it with the Pot command, the same method will work with almost any Stamp input or output statement.

Take a look at figure 2 and listing 2. It's based around a 4051 multiplexer/demultiplexer chip. This device works like a digitally controlled rotary switch, shown conceptually in figure 3. Depending on the binary number on its control bits, the chip connects one of eight I/Os to a single common pin. This connection is analog and bidirectional, so you can use it for input, output, or both. The Pot command is a good example of both, since it switches back and forth between input and output as it measures the time required to discharge a capacitor through an unknown resistance.

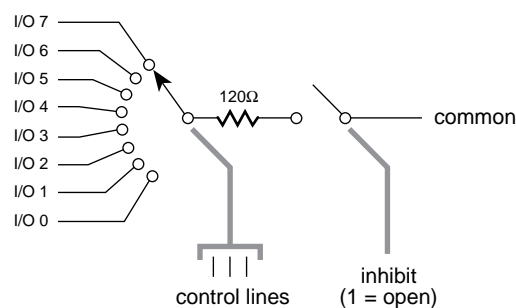


Figure 3. The 4051 multiplexer works like a digitally controlled rotary switch. The three control bits select I/O 0 (000 binary) through 7 (111 binary).

For an investment of four pins--three control bits and one common—you receive a dividend of eight I/Os. Better than Wall Street most days.

There are a few limitations to this trick. The first is evident if you try the pot demonstration. You can never quite get a reading of zero; the

lowest the readings will go is about 8. This is because the 4051's electronic rotary switch isn't perfect--it has a resistance of approximately 120 ohms.

A second limitation is that inputs to the 4051 must never exceed 0.5 volts above the supply voltage or below ground. This means that the 4051 can't be used to directly receive RS-232 serial signals through a series resistor as the Stamp's I/O pins can.

The last restriction is common to both the 4051 and the Stamp, so it really isn't much of a limitation: Current through the 4051 should never exceed 25 mA.

If you like the 4051, but still need more I/O, try the 4067. It provides 16 I/Os for five Stamp pins. Check your favorite data book (such as the

CMOS Cookbook by Don Lancaster) for details.

Sources

Enter the LCD Serial Backpack contest and win a new LCD display for your Stamp projects (a \$40 value) or 10 percent off Stamp accessories and microcontroller projects seen here in Nuts & Volts. Entries (or questions, suggestions, requests) to: Scott Edwards Electronics, 964 Cactus Wren Lane, Sierra Vista, AZ 85635; phone 520-459-4802; fax 520-459-0623; e-mail (Compuserve) at 72037,2612; or via Internet 72037.2612@compuserve.com.

For more information on the BASIC Stamp, contact Parallax Inc., 3805 Atherton Road no. 102, Rocklin, CA 95765; phone 916-624-8333; fax 916-624-8003; BBS 916-624-7101.

' Listing 1. BASIC Stamp Battery Monitor

' CHEAP AND SIMPLE LOW BATTERY WARNING
' Guy Marsden, March 1995. tekart@well.com
,

' Uses a Cds opto coupler with LED input such as: CLM6000.
' Put a 33 or 47K resistor across cell and a .1uF cap to gnd.
' Use a high value resistor in series with the LED wired directly
' to the battery. I used 100k for a 12volt system. This value is
' enough to produce a resistance of 20k at nominal battery voltage.
,

' When the voltage drops, the Cds resistance increases. By setting
' the SCALE of the POT function under a low battery condition, you
' will then have a range to work with. Determine your setpoint
' using a variable bench supply and a DVM.

symbol Batt = b2
symbol LoBatt = 220

CheckBatt:

 pot 0,76,Batt ' check battery voltage
 if Batt > LoBatt Alarm ' if less than established value
goto CheckBatt

Alarm:

 sound 1,100,100 ' beep piezo alarm
 pause 100
goto Alarm

' Listing 2. Multiplexing Stamp I/O Lines

' Program: MULTIPOT.BAS (Multiple pots using a 4051 multiplexer)
' This program demonstrates how to connect and measure multiple
' pots using a 4051 multiplexer chip. The 4051's control inputs
' (11,10,9) connect to Stamp pins 0, 1, and 2 respectively.
' The common I/O pin (3) of the 4051 goes to Stamp pin 3.
' By writing a value between 0 and 7 to its pins, the Stamp can
' select one of eight variable-resistance inputs through the
' 4051. See the schematic for details.

```
SYMBOL pot_sel = b2          ' Pot number 0-7 selected through 4051.
SYMBOL pot_val = b3          ' Result of the pot measurement.

Let dirs = %0111             ' Make the lower 3 pins outputs to drive 4051.
Again:
  for pot_sel = 0 to 7       ' For each of the eight pots:
    let pins = pot_sel       ' Write pot number to the 4051.
    pot 3,150,pot_val        ' Perform pot measurement on selected pot.
    debug "pot #",#pot_sel," ",#pot_val,cr ' Display result.
  next pot_sel               ' Read the next pot.
  pause 2000                 ' Wait two seconds.
  debug cr                   ' Insert a carriage return on screen.
goto Again                   ' Do it again (endless loop).
```