

Assignment 4: Photocell Bargraph

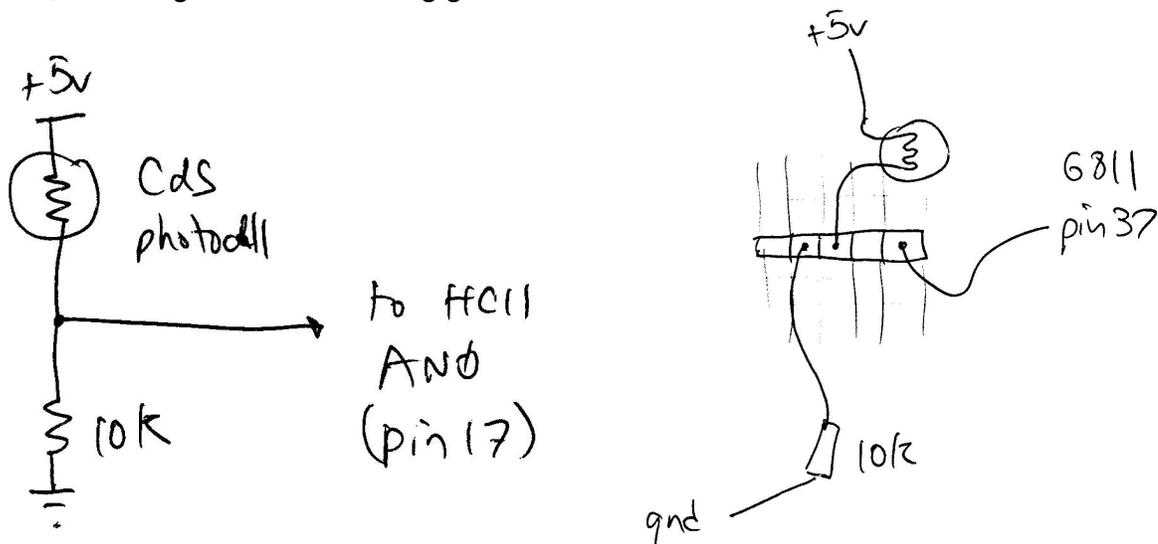
due Wednesday, Oct 16

Overview

Using provided sample code, write a program that repeatedly displays a converted analog reading as a bargraph of 0 to 15 asterisks.

How To

The photocell is wired in a “voltage divider” circuit as shown. The diagram on the left is the electrical schematic; on the right is a visual wiring guide.



The voltage divider generates an output voltage that is a function of the ratio of the two resistances in the legs of the converter. In this instance, the photocell is in the upper leg. Its resistance decreases with increasing light, causing a higher voltage result in this case.

Sample Code

The file `analog2hex.s` demonstrates how to initialize the HC11's analog-to-digital converter and make a conversion. The core functionality in the following code snippet:

```
        bset <OPTION,X,#0x80    ; set high bit of option -> turn on A/Ds

loop:   ldaa #0                 ; select channel 0
        staa ADCTL              ; start conversion
done1p: ldaa ADCTL
        bpl done1p              ; if high bit set, it's done
        ldaa ADR1               ; grab result!
```

The full sample program repeatedly makes analog-to-digital conversions and prints to the serial line an ASCII representation of their hex value.

How to Go About It

To verify that your hardware is functioning properly, it is suggested that you first build the circuit and run the supplied demo program.

Features

Your solution must:

- demonstrate the use of subroutines. Make sure to initialize the stack pointer at the beginning of your code.
- repeatedly print the bar graph of 0 – 15 asterisks based on the high nybble of the converted value, with the linefeed and carriage return characters between each bar. You should see a sideways scrolling bar graph on your terminal emulator when it is working.

For extra credit, you can generate a graph of 0 – 31 asterisks using the upper five bits of the converted value.

What to Turn in

Hand in your assembled .lst file and a screen capture of your program in action.