Date: October 19th, 2009

Project: CardPOV v1

Description: Project Log 001

Hardware Selection:

Microcontroller: A microcontroller with onboard USB and sufficient memory space for future code features is desired. For this, the **PIC18f2550**, featuring 32k of flash and 2k of data memory, as well as onboard USB, will be used.

Power Source: A slim power source with sufficient capacity to run the onboard circuitry is desired. For this, a **CR2032** coin cell battery, providing 3V power and a 230mAh capacity, will be used.

LEDs: Generic surface mount LEDs

Switches: Generic tactile switches

Firmware:

Version 1 will make use primarily of the timer0 peripheral to coordinate LED pulses. For static timing purposes, it will be assumed that the POV will be moved repetitively through a linear path 30in in length. POV characters will be stored in the onboard data EEPROM memory. Target frequency operation is 8MHz.

Pin Usage:

Of particular importance for the initial layout of the CardPOV, pins should be selected in such a way so as to not conflict with future pin needs. For this reason, pins that are needed for the USB, comparator, and ADC should not be used. Initial pin declarations are therefore as follows:

Pushbutton 1: RC0 (pin 11)

Pushbutton 2 (available): RC1 (pin 12)

Pushbutton 3 (available): RC2 (pin 13)

POV LEDs: RB5-RB0, RC7, RC6 (pins 26-21, 18, 17)

Date: January 8th, 2010

Project: CardPOV

Version: 1

Revision: 0

Description: Log 002

Work continued on CardPOV v1 yesterday, but was quickly brought to a halt when the following problems arose:

1. An ordering error resulted in the accidental purchase of PIC18LF25J50 microcontrollers instead of PIC18F25J50 microcontrollers
2. A second ordering error resulted in the accidental procurement of 0.75 ohm resistors instead of 75 ohm resistors
3. In the PCB layout, ground was referred to by 2 different names (GND and VSS), and as such these 2 nets were never connected, resulting in circuit connections that were incomplete.

To help mitigate these problems in the future, I plan to check orders and PCB layouts more carefully.

Date: January 11th, 2010

Project: CardPOV

Version: 1

Revision: 0

Description: Project Log 003

After having an opportunity to lay out a prototype of the CardPOV concept, it has become apparent that a few revisions will need to be made to the future plans for CardPOV. Accelerometers I’ve seen so far appear to be too pricy to be commercially viable, and as such the closed-loop POV device is (at present time) therefore not going to be visited. Secondly, due to the height of the CR2032 battery and holder, it is not presently beneficial to remove the tact switches in favor of capacitive touch sensing. Thus, unless the battery holder can be directly incorporated into the board, the capacitive touch sensing POV version will probably not be visited.

Those unrealities considered, there are some additions that will be made to the POV project:

1. Added a hole to the PCB to attach a string to allow the card to be swung via string (theoretically increasing sign readability)
2. While the closed-loop and touch POV versions of the project have been cancelled, a new version of the project, known as SuperPOV, has been envisioned. SuperPOV will increase the number of onboard LEDS (thus increasing resolution), and the PC-side application will then be modified to include UTF-16 support. In this way, a POV device will be created that will allow for the display of non-Latin character sets, such as Chinese and Arabic.

Date: January 17th, 2010

Project: CardPOV

Version: 1

Revision: 0

Description: Log004

I corrected the error to version 1 of the CardPOV board, and the resulting board operated properly. As such, the hardware for CardPOV v1 is now stable. Firmware will be stable for the CardPOV v1 device once I incorporate the multiple string support feature.

On a sidenote, while working on CardPOV v1 today, Karen came up with 2 positively brilliant ideas I would’ve never thought to think about myself. First, when this device goes up for sale, she suggested that I could allow users to pick the POV color that they wanted (brilliant, good business idea). Secondly, she suggested moving away from red as a POV color and instead using blue, green, or teal. This makes a lot of sense, really, as the human eye has a much better response to these colors than red (optimal response for teal), which means I could considerably increase the POV display’s readability without any increase in led intensity or power (ingenious!).

Included below is a table of POV colors, voltages, and resistor values for the different POV colors:



Date: January 30th, 2010

Project: CardPOV

Version: 1

Revision: 0

Description: Log 005

I spent a considerable amount of time yesterday working on multiple string support code for the CardPOV project. Currently, POV string data is written starting at address 0800h. The first value is the number of POV signs in memory, followed by a series of entries consisting of the length of a given sign and its starting location in memory.

The code has thus far presented me with a few difficulties, some of which remain unsolved. The first major mystery was that when I programmed the circuit with the Pickit2 in debug mode, it wouldn’t work as a standalone device, the pickit was required for operation. In an attempt to remedy this, I used the config directive supported by the pic18f devices and added the configuration bits to the code. *A complete listing of all of the configuration bits and their symbolic values for the config directive can be found in the .inc file of the PIC microcontroller in use.* That alone did not fix the problem, however. I later discovered that *when programming a PIC microcontroller with the debugger, the debugger inserts bits of code that require the circuit to be used with the debugger to work properly.*

As of the time of this writing, one other error still eludes me, however. For reasons thus far unidentified, bit 0 of portA does not register a value of 1 when 3.3V is applied, in spite of the fact that it is set up as an input in the trisA register.

Date: February 8th, 2010

Project: CardPOV

Version: 1

Revision: 0

Description: Log 006

Having tried using a different pin on port A and checking the silicon errata, the button debouncing problem blocking the path to multiple string support remained a mystery. Fortunately, with the help of a nice fellow from the Microchip online forums, it was revealed that the analog control register ANCON0 is banked and not accessible from the default bank select register upon a reset. As such, the values that I was writing to the ANCON0 register were not actually being written, and as a result the portA pins were being used as analog inputs (instead of digital inputs). By changing the bank select register to bank F, I gained write access to the ANCON0 register, which allowed me to initialize portA as digital I/O and fixed the error. With that error fixed, multiple string support is working by and large.

Date: February 14th, 2010

Project: CardPOV

Version: 1

Revision: 0

With multiple string support supported, CardPOV v1 was nearly finished. Still, a few lingering errors seem to be latent in the hardware, and these needed to be investigated. In particular, the display was still a bit unreadable, debouncing of the string select pushbutton appeared to be incomplete, and there’s an unusual artifact of some kind present in the first string (“NBitWonder”) POV output. After uping the timer prescaler (moving from 1:32 -> 1:64), the POV display appeared to be more readable. The button debouncing problem that appeared to be there seemed to go away after the device was reprogrammed. The artifact still remains, however, but that will hopefully be addressed soon.