KeyPad Input

Adapted from Prof. Martin Fox’s ECE266 slides
Agenda

• Overview

• Objectives: Display of Data input from a keypad.

• LCD: Use to display toggled in messages

• Enter: First time ‘Press Enter’; Subsequently toggle between Name, Character, Temperature, and KeySwitch subroutines. Just add to previous state variable.
KEYSTATE: new state var.

- Mechanical switching: many LoopTimes
- 10 ms LoopTime chosen because max. bounce < 10 ms.
- KEYSTATE can go from 0-4. [5 states]
- Released? Require release during 2 successive passes around main loop. Take action AFTER release, i.e. update display. Reset to look for new press.
KeyPad

- Keypad to be scanned is 4x3; 3 Columns, 4 Rows
- Columns are hard wired to 3 ports:
  - Col1=A5, Col2=A4, Col3=A2
- Rows are hard wired to PortB 4,3,2,1 i.e.
  - B4=Row1, B3=Row2, B2=Row3, B1=Row4
- PortB has the feature of being configured for a weak pullup if open; therefore will read “1” if open[need OPTION_REG,7 clear].
Keypad wiring diagram

Col 1 [A5 new]

Col 2 [A4 new]

Row1

To display
PORTB pins
Multiple function B4:1

- PortB 4,3,2,1 BOTH send data to LCD and receive data from keypad in this implementation.
- MUST convert to inputs to read keypad, but set back to outputs to operate display. To make input need TRISB,x=1,output x=0.
KeySwitch: States

KEYSTATE=0
Call AnyKey; has any key been pressed?
Y=goto KEYPSTATE1, n=Return

KEYSTATE=1
Call GetKey; which key was pressed?
;Put in KEYCODE

KEYSTATE=2
Call HandleKeycode; update display variable string

KEYSTATE=3
Call AnyKey; debounce, wait for 0

KEYSTATE=4
Call AnyKey; debounce then DisplayC string
char string[] = { '+', '0', '.', '0', '0', '0', 0, 0 };  
char keystate;  
char keycode;  

void KeySwitch()  
{  
    switch( keystate )  
    {  
        case 0:  
            if ( AnyKey() )  
                keystate = 1;  
            break;  
        case 1:  
            keycode = GetKey();  
            if ( keycode == 0 )  
                keystate = 0;  
            else  
                keystate = 2;  
            break;  
        case 2:  
            HandleKeycode(keycode);  
            keystate = 3;  
            break;  
        case 3:  
            if ( !AnyKey() )  
                keystate = 4;  
            break;  
        case 4:  
            if ( !AnyKey() )  
            {  
                DisplayC( 0x80, string );  
                keystate = 0;  
            }  
            break;  
    }  
}
Summarize

This lab shows how to decode general matrix key arrays i.e. keyboards. Allows input of data.

The following steps are needed:

• Debounce initial keyswitch
• Determined which key was pressed
• Take action ONCE for key press
• Wait for release, then reset.