

Texas Instruments Microcontroller HOW-TO GUIDE

Interfacing Keypad with MSP430F5529



Contents at a Glance

PS PRIMER MSP430 kit.....	3
Keypad.....	4
Interfacing keypad	4
Interfacing keypad with MSP430F5529	5
Pin Assignment with MSP430F5529	6
Circuit Diagram to Interface Keypad with MSP430F5529 ..	7
C Program to 4 X 4 matrix keypad using MSP430F5529	7

PS PRIMER MSP430 kit

The MSP430 Primer kit is specifically designed to help students to master the required skills in the area of embedded systems. The kit is designed in such way that all the possible features of the microcontroller will be easily used by the students. The kit supports with Code Composer Studio v4 and later through USB port.

Texas Instrument Microcontroller, MSP430 Primer Kit is proposed to smooth the progress of developing and debugging of various designs encompassing of speed 16-bit Microcontrollers. It integrates on board PS2, UART, ADC, PWM, Temperature Sensor, Relay, Buzzer, I2c Seven Segment, I²C Serial EEPROM, Temperature Sensor LM35, Matrix Keypad, Switch, LED, Stepper Motor Driver, Traffic Light Controller, I²C RTC, LCD & GLCD Display to create a stand-alone versatile test platform. User can easily engage in Primer in this platform, or use it as reference to application Primer.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

Keypad

A **keypad** is a set of buttons arranged in a block or "pad" which usually bear digits, symbols and usually a complete set of alphabetical letters. If it mostly contains numbers then it can also be called a **numeric keypad**. Here we are using 4 X 4 matrix keypad.

Interfacing keypad

The rows are connected to an output port and the columns are connected to an input port.

To detect a pressed key, the microcontroller grounds all rows by providing 0 to the output latch, and then it reads the columns. If the data read from the columns is D3-D0=1111, no key has been pressed and the process continues until a key press is detected. However, if one of the column bits has a zero, this means that a key press has occurred. For example, if D3-D0=1101, this means that a key in the D1 column has been pressed.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

After a key press is detected, the microcontroller will go through the process of identifying the key. Starting with the top row, the microcontroller grounds it by providing a low to row D0 only; then it reads the columns.

If the data read is all 1s, no key in that row is activated and the process is moved to the next row. It grounds the next row, reads the columns, and checks for any zero. This process continues until the row is identified. After identification of the row in which the key has been pressed, the next task is to find out which column the pressed key belongs to.

Interfacing keypad with MSP430F5529

We now want to scan a keypad in MSP430F5529 Primer Board. In case of 4X4 matrix Keypad both the ends of switches are connected to the port pin i.e. four rows and four columns. So in all sixteen switches have been interfaced using just eight lines.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

1Keypads arranged by matrix format, each row and column section pulled by high or low by selection J5, all row lines(P2.4 – P2.7) and column lines(P2.0 to P2.3) connected directly by the port pins.

Note: While using Keypad ensure all slide switches (SW20-SW27) to no connection position because the same lines used for both slide switches and matrix keypads.

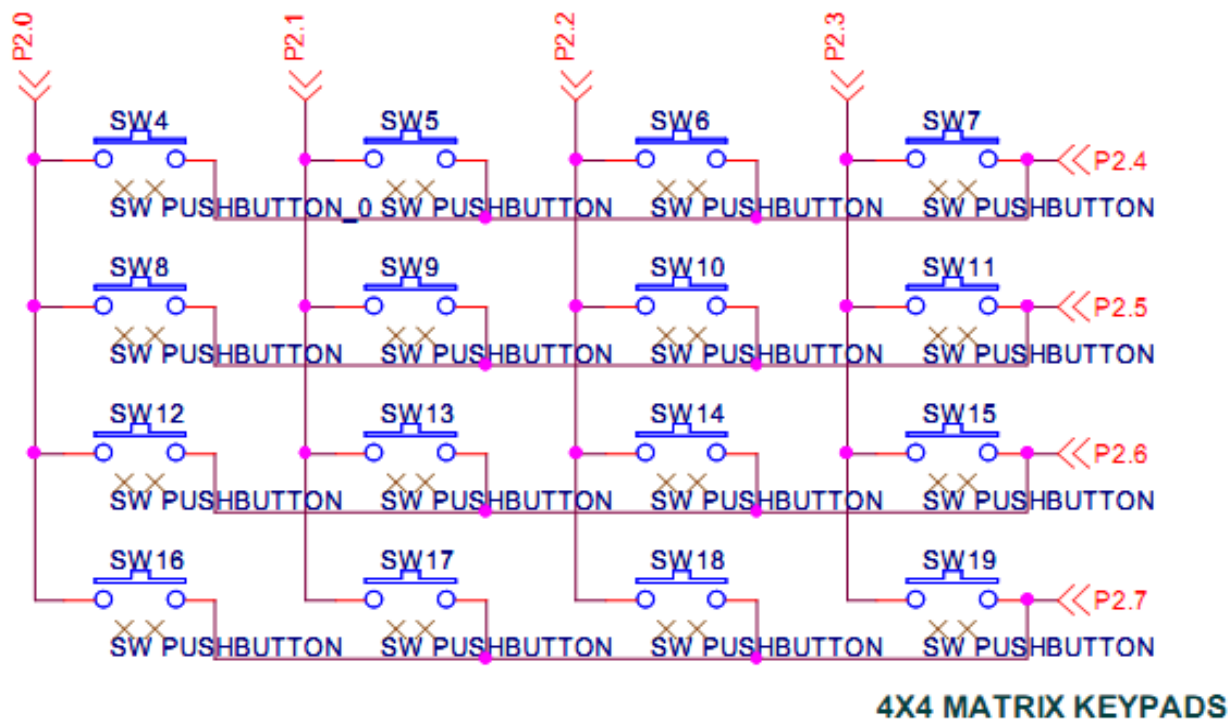


Pin Assignment with MSP430F5529

	4x4 Matrix Lines	MSP430F5529 Lines	4x4 Matrix Keypad
ROW	ROW-0	P2.4	
	ROW-1	P2.5	
	ROW-2	P2.6	
	ROW-4	P2.7	
COLUMN	COLUMN-0	P2.0	
	COLUMN-1	P2.1	
	COLUMN-2	P2.2	
	COLUMN-3	P2.3	

Join the Technical Community Today!
<http://www.pantechsolutions.net>

Circuit Diagram to Interface Keypad with MSP430F5529



C Program to 4 X 4 matrix keypad using MSP430F5529

```
*****
Title : Program to keypad interfacing
*****
```

```
#include <msp430x552x.h>
#include <stdio.h>
#define keyport P2OUT //Keypad Port
#define COL1 (0x10 & P2IN)
#define COL2 (0x20 & P2IN)
#define COL3 (0x40 & P2IN)
#define COL4 (0x80 & P2IN)
```

Join the Technical Community Today!
<http://www.pantechsolutions.net>


```

const unsigned char Msg1[] = "MATRIX KEYPAD ";
unsigned char lsb,msb;
unsigned int a=1,j=0;
unsigned char key_press;
unsigned char i,k,key=0;
unsigned char Key_Val[] = {'C','8','4','0','D','9','5','1','E','A','6','2','F','B','7','3'};
unsigned char get_key(void);

```

```

void DelayMs(int Ms);
void LCD_init(void);
void LCD_cmd4(unsigned char cmd);
void LCD_dat4(unsigned char byte);

```

```

void main(void)
{

```

```

    unsigned int count=0;
    unsigned char Value=0;

```

```

    WDTCTL = WDTPW + WDTHOLD; // Stop WDT
    P6DIR |= 0xF0; // Set P6.4-P6.7 to Output direction
    P7DIR |= 0xF0; // Set P7.4-P7.7 to Output direction
    P2DIR = 0x0F; // Set P2.0 to 2.3 Output ,Set P2.4 to 2.7 Input
    P2REN = 0xFF; // Set P2.0 to 2.7 Pull up Register enable
    P2OUT = 0xF0; // Set P2.0 to 2.7 Out Register.

```

Join the Technical Community Today!
<http://www.pantechsolutions.net>


```
P7DIR = 0xF0;    // Set data, P7.4-P7.7 to Output direction
P5DIR = 0xC0;    // Set r/w & rs,P5.6 & 5.7 to O/P direction
P4DIR = 0x80;    // Set en, P4.7 to Output direction
DelayMs(100);
```

```
LCD_init();
DelayMs(50);
```

```
LCD_cmd4(0x80);
for(j=0;j<16;j++)
{
    LCD_dat4(Msg1[j]);
    DelayMs(5);
}
```

```
while(1)
{
    while((count = get_key())==0); //Wait untill a key Pressed
    Value = Key_Val[count];
    LCD_cmd4(0xC0);
    LCD_dat4(Value);
    P6OUT = ( Value << 4 );
}
}
```



```
unsigned char get_key(void)
```

```
{
```

```
    k=1;
```

```
    for(i=0;i<4;i++)
```

```
    {
```

```
        keyport = ((0x01<<i) ^ 0xff);    //Scan for a Key by  
sending '0' on ROWS
```

```
        if(!COL1){    //when a key pressed numbers 1--16  
will be returned
```

```
            key = k+0;
```

```
            while(!COL1);
```

```
            return key;
```

```
        }
```

```
        if(!COL2){
```

```
            key = k+1;
```

```
            while(!COL2);
```

```
            return key;
```

```
        }
```

```
        if(!COL3){
```

```
            key = k+2;
```

```
            while(!COL3);
```

```
            return key;
```

```
        }
```

Join the Technical Community Today!

<http://www.pantechsolutions.net>


```

        if(!COL4){
            key = k+3;
            while(!COL4);
            return key;
        }
        k+=4;
        keyport |= (0x01<<i);
    }
    return 0;
}

```

```

void DelayMs(int Ms)
{
    int k;
    while(Ms>0)
    {
        for(k=0;k<104;k++);
        Ms--;
    }
}

```

```

void LCD_init(void)
{
    LCD_cmd4(0x33);
    LCD_cmd4(0x22);
}

```

Join the Technical Community Today!
<http://www.pantechsolutions.net>


```
LCD_cmd4(0x22);
```

```
LCD_cmd4(0x22);
```

```
LCD_cmd4(0x28);           // 28 for four bit mode
```

```
LCD_cmd4(0x0c);
```

```
LCD_cmd4(0x06);
```

```
LCD_cmd4(0x01);
```

```
}
```

```
void LCD_cmd4(unsigned char cmd)
```

```
{
```

```
    P5OUT = 0x00;    // RW=0,RS=0
```

```
    P7OUT = cmd;
```

```
    P4OUT = 0x80;    // En = 1;
```

```
    P4OUT = 0x00;    // En = 0;
```

```
    cmd = (cmd<<4) & 0xF0;
```

```
    P7OUT = cmd;
```

```
    P4OUT = 0x80;    // En = 1;
```

```
    P4OUT = 0x00;    // En = 0;
```

```
    DelayMs(3);
```

```
}
```

Join the Technical Community Today!

<http://www.pantechsolutions.net>


```
void LCD_dat4(unsigned char byte)
{
    P5OUT = 0x80;    // RW=0,RS=0
    P7OUT = byte;
    P4OUT = 0x80;    // En = 1;
    P4OUT = 0x00;    // En = 0;

    byte = (byte<<4) & 0xF0;
    P7OUT = byte;
    P4OUT = 0x80;    // En = 1;
    P4OUT = 0x00;    // En = 0;
    DelayMs(3);
}
```


Did you enjoy the read?

Pantech solutions creates information packed technical documents like this one every month. And our website is a rich and trusted resource used by a vibrant online community of more than 1,00,000 members from organization of all shapes and sizes.

Join the Technical Community Today!
<http://www.pantechsolutions.net>

What do we sell?

Our products range from Various Microcontroller development boards, DSP Boards, FPGA/CPLD boards, Communication Kits, Power electronics, Basic electronics, Robotics, Sensors, Electronic components and much more . Our goal is to make finding the parts and information you need easier and affordable so you can create awesome projects and training from Basic to Cutting edge technology.

Join the Technical Community Today!
<http://www.pantechsolutions.net>