```c
/*
 *  Copyright 2010 by Spectrum Digital Incorporated.
 *  All rights reserved. Property of Spectrum Digital Incorporated.
 */

/*
 *  SAR functions Test
 *
 */
#include "usbstk5515.h"
#include "sar.h"

Uint16 preKey  =0;
Uint16 keyCnt1 =0;
Uint16 keyCnt2 =0;

/* ------------------------------------------------------------------------ *
 *                                                                          *
 *   Init_SAR(void)                                                         *
 *       Initialize SAR ADC                                                 *
 *                                                                          *
 * ------------------------------------------------------------------------ */
void Init_SAR(void)
{
    *SARCTRL    = 0x3C00;       // Select AIN3, which is GPAIN1
    *SARCLKCTRL = 0x0031;       // 100/50 = 2MHz
    *SARPINCTRL = 0x7104;       // interrupção???
    *SARGPOCTRL = 0;
    return;
}

void Read_GPAIN1(void)
{
    Uint16 val, i;

    for(i=0;i<30; i++)
        asm("   nop");

    while(1)
    {

        for(i=0;i<100; i++)
            asm("   nop");

        *SARCTRL = 0xB400;

        while(1)
        {
            for(i=0;i<50; i++)
                asm("   nop");
            val = *SARDATA;
            if((val&0x8000) == 0)
                break;
        }
    }
}
/* ------------------------------------------------------------------------ *
 *                                                                          *
 *   Get_Sar_Key(void)                                                      *
 *       Find key pressed depending on value returned by SAR ADC            *
 *                                                                          *
 * ------------------------------------------------------------------------ */
Uint16 Get_Sar_Key(void)
{
    Uint16 val, i;

    *SARCTRL = 0xB800;
    while(1)
```

```
    {
        for(i=0;i<5; i++)
            asm("   nop");
        val = *SARDATA;
        if((val&0x8000) == 0)
            break;
    }
    // Account for percentage of error
    if((val < SW1 + 12) && (val > SW1 - 12))
        val = SW1;
    if((val < SW2 + 12) && (val > SW2 - 12))
        val = SW2;
    if((val < SW12 + 12) && (val > SW12 - 12))
        val = SW12;
    if((val < NoKey + 12) && (val > NoKey - 12))
        val = NoKey;

    if(val == NoKey)  // No button pressed
    {
        preKey = NoKey;
        return NoKey;
    }
    else if((val==SW1)||(val==SW2)||(val==SW12)) // Button pressed
    {
        if((val != preKey) && (preKey != SW12))// New button
        {
            preKey = val;
            return val;
        }
        else if((val != preKey) && (preKey == SW12))// New button, Account for 2 button
situation
        {
            preKey = NoKey;
            return val;
        }
        else              // Same button
        {
            return NoKey;
        }

    }
    else
    {
        preKey = val;
        return NoKey;
    }

}
```