```
/* ========================================
 *
 * Copyright YOUR COMPANY, THE YEAR
 * All Rights Reserved
 * UNPUBLISHED, LICENSED SOFTWARE.
 *
 * CONFIDENTIAL AND PROPRIETARY INFORMATION
 * WHICH IS THE PROPERTY OF your company.
 *
 * ========================================
*/
#include <project.h>
#include <stdio.h>
#include <math.h>
uint16 xadcResult = 0;
uint16 yadcResult = 0;
char xsend = 4;
char ysend = 4;
char bstate =0;

void R_XADC();
void R_YADC();
void C_XADC();
void C_YADC();
void C_B();

int main()
{
    CyGlobalIntEnable; /* Enable global interrupts. */

    /* Place your initialization/startup code here (e.g. MyInst_Start()) */

    UART_1_Init();
    UART_1_Start();
    UART_2_Init();
    UART_2_Start();
    unsigned char j = 10;
    //int Result[2];
    //int bytesWritten;
    //char S1[8];

    ADC_SAR_1_Start();
    ADC_SAR_2_Start();
    ADC_SAR_1_StartConvert();
    ADC_SAR_2_StartConvert();
    //UART_1_PutString("Hello World!");
    for(;;)
    {
        R_XADC();
        R_YADC();
        C_XADC();
        C_YADC();
```

```
        C_B();
        UART_1_WriteTxData(xsend);
        UART_1_WriteTxData(ysend);
        UART_1_WriteTxData(bstate);
        UART_2_PutChar(xsend);
        UART_2_PutChar(ysend);
        UART_2_PutChar(bstate);
        UART_2_PutChar(' ');
        CyDelay(j);
        /* Place your application code here.
        if (UART_1_GetRxBufferSize() > 0) {
            uint8 c = UART_1_GetChar();
            UART_1_PutChar(c);
            Control_Reg_1_Write(c);
        }*/
    }
}


void R_XADC()
{
  if( ADC_SAR_1_IsEndConversion(ADC_SAR_1_WAIT_FOR_RESULT) )
        {
            xadcResult = ADC_SAR_1_GetResult16();       // read the adc and ↵
assign the value adcResult
            xadcResult = (int)xadcResult;
            xadcResult = (xadcResult)/5;              // adcResult is the ↵
integer scaling between 0-10
        if (xadcResult & 0x8000)
        {
            xadcResult = 0;       // ignore negative ADC results
        }
        else if (xadcResult >= 0xfff)
        {
            xadcResult = 0xfff;      // ignore high ADC results
        }
        }
}
void R_YADC(){
  if( ADC_SAR_2_IsEndConversion(ADC_SAR_2_WAIT_FOR_RESULT) )
        {
            yadcResult = ADC_SAR_2_GetResult16();       // read the adc and ↵
assign the value adcResult
            yadcResult = (int)yadcResult;
            yadcResult = (yadcResult)/5;              // adcResult is the ↵
integer scaling between 0-10
        if (yadcResult & 0x8000)
        {
            yadcResult = 0;       // ignore negative ADC results
        }
        else if (yadcResult >= 0xfff)
        {
```

```
            yadcResult = 0xfff;      // ignore high ADC results
        }                                           // delay in milliseconds
        }
}

void C_XADC()
{
    if (0 <= xadcResult && xadcResult < 100){
            xsend = 'A';
        }
        if (100 <= xadcResult && xadcResult < 200){
            xsend = 'B';
        }
        if (200 <= xadcResult && xadcResult < 300){
            xsend = 'C';
        }
        if (300 <= xadcResult && xadcResult < 400){
            xsend = 'D';
        }
        if (400 <= xadcResult && xadcResult < 500){
            xsend = 'E';
        }
        if (500 <= xadcResult && xadcResult < 600){
            xsend = 'F';
        }
        if (600 <= xadcResult && xadcResult < 700){
            xsend = 'G';
        }
        if (700 <= xadcResult && xadcResult < 800){
            xsend = 'H';
        }
        if (800 <= xadcResult && xadcResult < 900){
            xsend = 'I';
        }
}

void C_YADC()
{
    if (0 <= yadcResult && yadcResult < 100){
            ysend = 'J';
        }
        if (100 <= yadcResult && yadcResult < 200){
            ysend = 'K';
        }
        if (200 <= yadcResult && yadcResult < 300){
            ysend = 'L';
        }
        if (300 <= yadcResult && yadcResult < 400){
            ysend = 'M';
        }
        if (400 <= yadcResult && yadcResult < 500){
            ysend = 'N';
```

```
        }
        if (500 <= yadcResult && yadcResult < 600){
            ysend = 'O';
        }
        if (600 <= yadcResult && yadcResult < 700){
            ysend = 'P';
        }
        if (700 <= yadcResult && yadcResult < 800){
            ysend = 'Q';
        }
        if (800 <= yadcResult && yadcResult < 900){
            ysend = 'R';
        }
}

void C_B(){
    if (B1_Read()==0){
        bstate = 'S';
    }
    else if (B2_Read()==0){
        bstate = 'T';
    }
    else if (B3_Read()==0){
        bstate = 'U';
    }
    else if (B4_Read()==0){
        bstate = 'V';
    }
    else{
        bstate='W';
    }
}

/* [] END OF FILE */
```