**PROBLEM:**

Write a code to make OLED I2c display (SH1106 or SSD1306 based display) work with any TI controller (preferably MSP430) without the use of a display booster pack. Interface it with a keypad so as to display Ferrari logo when '\*' is pressed and Lamborghini logo when '#' is pressed on the keypad.

**COMPONENTS REQUIRED:**

1. **MSP430G2231 MICROCONTROLLER**

* 16KB Flash
* 512B RAM
* 2 Timers
* P1.6 – SCL
* P1.7 – SDA
* USCI\_B0 provides support for SPI (3 or 4 pin) and I2C.
* USCI B0 I2C Interrupt enable - UCB0CIE
* USCI\_B0 I2C slave address UCB0SA
* USCI\_B0 I2C own address UCB0OA

1. **SH1106 OLED DISPLAY**

* 132X64 dot matrix OLED
* Logic voltage supply-1.65V-3.5V
* Dc-Dc supply voltage-3V-4.2V
* Common cathode type OLED panel
* D0-SCL
* D1-SDA
* I2C configuration

|  |  |  |
| --- | --- | --- |
| IM0 | IM1 | IM2 |
| 0 | 1 | 0 |

* Slave address – 0X3C(write)/0x3D(read)

1. **4X4 KEYPAD**

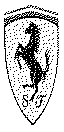
* Circuit Rating: 35V (DC), 100mA, 1W

1. **PCF8574**

* I2C to Parallel-Port Expander
* Slave address – 0x20

1. **TWO BITMAP IMAGES**

1) Ferrari\_logo



Corresponding values are in the file ferrari\_logo.h file

2) Lamborghini\_logo



Corresponding values are in the file Lamborghini\_logo.h file

**NOTE**: These two images are made in a way to fit 128X64 screen of the OLED display and the corresponding table for displaying both the images is made with the help of LCDAssistant application.

**FLOW CHART:**

START

* Include the required header files
* Define the variables and constants
* Stop watchdog timer
* Define the required ports and pins as INPUT and OUTPUT
* Make the required setting for I2C interfacing such as mode, pins, clock, slave address, interrupt etc.

If Value==’\*’

**TRUE FALSE**

1. Display initialisation
2. Display Lamborghini\_logo

* Display initialisation
* Display Ferrari\_logo

**TRUE FALSE TRUE**

While Value==’#’

While Value==’\*’

**ALGORITHM:**

**STEP 1:** Start

**STEP 2:** Include the additional header files needed and declare and define the required

Variables and Constants.

**STEP 3:** Stop the watchdog timer and define the required ports and pins as INPUT and

OUTPUT. Make the required setting for I2C interfacing such as mode, pins, clock,

Slave address, interrupt etc.

**STEP 4:** If the value is not equal to ‘\*’ go to **STEP 7** else go to next step.

**STEP 5:** Initialise the display and display the Ferrari\_logo.

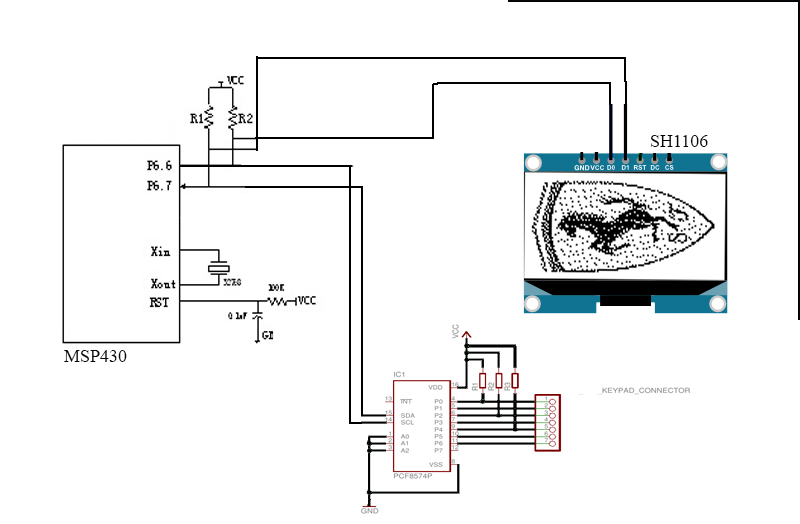
**STEP 6:** wait until the value changes to ‘#’ due to interrupt and then go to **STEP 4.**

**STEP 7:** Initialise the display and display the Lamborghini\_logo.

**STEP 8:** wait until the value changes to ‘\*’ due to interrupt and then go to **STEP 4.**

**COST ESTIMATION:**

|  |  |  |
| --- | --- | --- |
| **SL.NO** | **COMPONENTS** | **COST(INR)** |
| 1 | MSP430G2553 | 180 |
| 2 | SH1106 | 445 |
| 3 | 4x4 Keypad | 125 |
| 4 | PCF8574 | 110 |
| **TOTAL** | | 860(approx.) |

**CONNECTION DIAGRAM:**

**PROGRAM:**

#include <msp430g2231.h>

#include <i2c.h> //i2c with msp430g2231

#include <i2ckeypad.h> //i2c with keypad through pcf8574

#include <SH1106.h> //i2c with display

#include <ferrari\_logo.h> //image of ferrari logo

#include <Lamborghini\_logo.h> //image of lamborghini logo

#define ROWS 4 //4x3 keypad

#define COLS 3

#define PCF8574\_ADDR 0x20 //slave address of pcf8574

i2ckeypad kpd = i2ckeypad(PCF8574\_ADDR, ROWS, COLS);

#define OLED\_ADDR 0x3C //Slave address of SH1106

char key; //holds the value of key pressed

SH1106 display(OLED\_ADDR);

void main(void) {

WDTCTL = WDTPW + WDTHOLD; //stop watchdogtimer

i2c\_init(); //i2c initialisation

kpd.init(); //keypad initialisation

display.init(); //display initialisation

while(1)

{

if(key=='\*')

{

display.resetDisplay(); //reset display

//draw ferrari logo

display->drawBitmap(1,1,64,128,ferrari\_logo);

}

while(key=='\*')

{

key = kpd.get\_key(); //loop until value change;

}

if(key=='#')

{

display.resetDisplay(); //reset display

//draw lamborghini logo

display->drawBitmap(1,1,64,128,Lamborghini\_logo);

}

while(key=='#')

{

key = kpd.get\_key(); //loop until value change;

}

}

}