****

**INTERFACING LPC2148 WITH LCD FOR WORLD CLOCK**

**REVANTH ROY .T – CCE23034**

**INTERFACING LPC2148 WITH LCD FOR WORLD CLOCK**

**SOURCE CODE:**

**#include <lpc214x.h>**

**#include <stdint.h>**

**#include <stdlib.h>**

**#include <stdio.h>**

**void delay\_ms(uint16\_t j) /\* Function for delay in milliseconds \*/**

**{**

**uint16\_t x,i;**

**for(i=0;i<j;i++)**

**{**

**for(x=0; x<6000; x++);    /\* loop to generate 1 millisecond delay with Cclk = 60MHz \*/**

**}**

**}**

**void LCD\_CMD(char command)**

**{**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((command & 0xF0)<<8) ); /\* Upper nibble of command \*/**

**IO0SET = 0x00000040; /\* EN = 1 \*/**

**IO0CLR = 0x00000030; /\* RS = 0, RW = 0 \*/**

**delay\_ms(5);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = RW = 0)    \*/**

**delay\_ms(5);**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((command & 0x0F)<<12) ); /\* Lower nibble of command \*/**

**IO0SET = 0x00000040; /\* EN = 1 \*/**

**IO0CLR = 0x00000030; /\* RS = 0, RW = 0 \*/**

**delay\_ms(5);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = RW = 0)    \*/**

**delay\_ms(5);**

**}**

**void LCD\_INIT(void)**

**{**

**IO0DIR = 0x0000FFF0; /\* P0.12 to P0.15 LCD Data. P0.4,5,6 as RS RW and EN \*/**

**delay\_ms(20);**

**LCD\_CMD(0x02);  /\* Initialize lcd in 4-bit mode \*/**

**LCD\_CMD(0x28);  /\* 2 lines \*/**

**LCD\_CMD(0x0C);   /\* Display on cursor off \*/**

**LCD\_CMD(0x06);  /\* Auto increment cursor \*/**

**LCD\_CMD(0x01);   /\* Display clear \*/**

**LCD\_CMD(0x80);  /\* First line first position \*/**

**}**

**void LCD\_STRING (char\* msg)**

**{**

**uint8\_t i=0;**

**while(msg[i]!=0)**

**{**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((msg[i] & 0xF0)<<8) );**

**IO0SET = 0x00000050; /\* RS = 1, EN = 1 \*/**

**IO0CLR = 0x00000020; /\* RW = 0 \*/**

**delay\_ms(2);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = 1, RW = 0) \*/**

**delay\_ms(5);**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((msg[i] & 0x0F)<<12) );**

**IO0SET = 0x00000050; /\* RS = 1, EN = 1 \*/**

**IO0CLR = 0x00000020; /\* RW = 0 \*/**

**delay\_ms(2);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = 1, RW = 0) \*/**

**delay\_ms(5);**

**i++;**

**}**

**}**

**void LCD\_CHAR (char msg)**

**{**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((msg & 0xF0)<<8) );**

**IO0SET = 0x00000050; /\* RS = 1, EN = 1 \*/**

**IO0CLR = 0x00000020; /\* RW = 0 \*/**

**delay\_ms(2);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = 1, RW = 0) \*/**

**delay\_ms(5);**

**IO0PIN = ( (IO0PIN & 0xFFFF00FF) | ((msg & 0x0F)<<12) );**

**IO0SET = 0x00000050; /\* RS = 1, EN = 1 \*/**

**IO0CLR = 0x00000020; /\* RW = 0 \*/**

**delay\_ms(2);**

**IO0CLR = 0x00000040; /\* EN = 0, RS and RW unchanged(i.e. RS = 1, RW = 0) \*/**

**delay\_ms(5);**

**}**

**unsigned char flag=0;**

**void rtc\_int(void)\_\_irq{**

**ILR = 0X01;**

**flag = 1;**

**VICVectAddr = 0X00000000;**

**}**

**void RTC\_INIT(){**

**ILR = 0X01;**

**CCR  = 0X13;**

**CCR =  0X11;**

**CIIR = 0X01;**

**SEC = 0X05;**

**MIN = 0X12;**

**HOUR = 0X06;**

**VICIntEnable = 0x00002000;**

**VICVectCntl0 = 0x0000002D;**

**VICVectAddr0 = (unsigned)rtc\_int;**

**}**

**int tzOffset = 0;**

**int tzOffsetHr = 0;**

**int tzOffsetMin = 0;**

**char timestring[8];**

**void showTime() {**

**int ofsettedMin = (MIN + tzOffsetMin + 60) % 60;**

**int ofsettedHr = (HOUR + tzOffsetHr + 12) % 12;**

**sprintf(timestring,"%02d:%02d:%02d", ofsettedHr, ofsettedMin, SEC);**

**LCD\_CMD(0x86);**

**LCD\_STRING(timestring);**

**}**

**char gmtString[6];**

**void showTimeZone() {**

**LCD\_CMD(0xC3);**

**sprintf(gmtString, "%c%02d:%02d", tzOffset >= 0 ? '+' : '-', (tzOffset >= 0 ? tzOffset : -tzOffset ) / 2, tzOffset % 2 ? 30 : 0);**

**LCD\_STRING(gmtString);**

**}**

**int main(void){**

**LCD\_INIT();**

**RTC\_INIT();**

**LCD\_STRING("Time: ");**

**showTime();**

**LCD\_CMD(0xC0);**

**LCD\_STRING("GMT");**

**showTimeZone();**

**IO1DIR &= ~(1 << 24);**

**IO1DIR &= ~(1 << 26);**

**while(1){**

**if(flag){**

**showTime();**

**flag=0;**

**}**

**if ((IO1PIN & (1 << 24)) == 0) {**

**tzOffset -= 1;**

**if (tzOffsetMin == 0) {**

**tzOffsetMin = 30;**

**tzOffsetHr -= 1;**

**} else {**

**tzOffsetMin = 0;**

**}**

**showTime();**

**showTimeZone();**

**delay\_ms(250);**

**}**

**if ((IO1PIN & (1 << 26)) == 0) {**

**tzOffset += 1;**

**if (tzOffsetMin == 0) {**

**tzOffsetMin = 30;**

**} else {**

**tzOffsetMin = 0;**

**tzOffsetHr += 1;**

**}**

**showTime();**

**showTimeZone();**

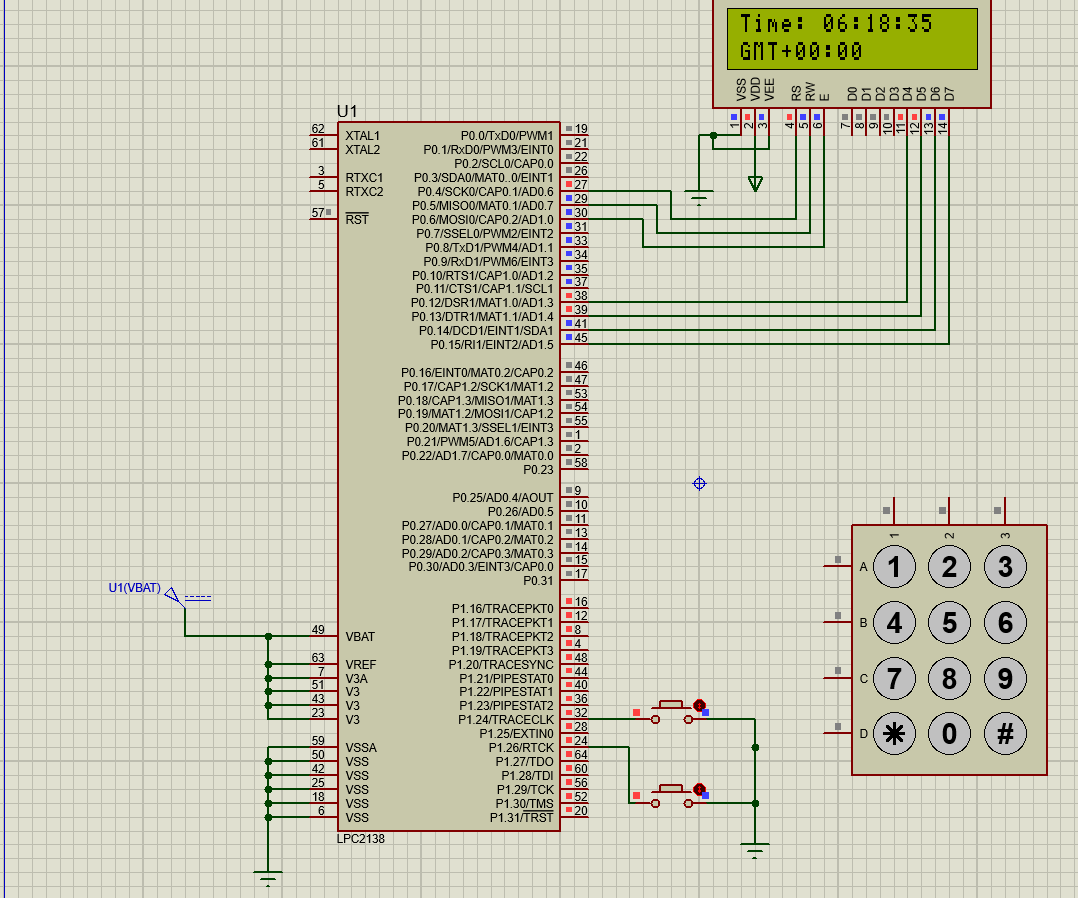
**delay\_ms(250);**

**}**

**}**

**}**

**IMAGE OF CIRCUIT :**



**Applications of Interfacing LPC2148 with LCD for a World Clock**

1. **Global Time Display:**

* Displaying the current time for multiple time zones simultaneously, useful in airports, international offices, and meeting rooms.

1. **Digital Dashboard:**

* Integrated into IoT systems or dashboards for smart homes or offices to provide a global time
* view alongside other metrics.

1. **Educational Tool:**

* Used in academic setups to teach the concepts of embedded systems, interfacing, and real time clock programming.

1. **Travel Accessories:**

* Portable world clock devices for travelers to view time zones of different regions easily.

1. **Smart Devices:**

* Implementation in smart gadgets, such as wristwatches or home automation displays.

**Advantages of Interfacing LPC2148 with LCD for a World Clock:**

1. **Low Power Consumption:**
   * LPC2148 is based on the ARM7 core, which is highly efficient, making it ideal for battery-powered devices.
2. **Compact Design:**
   * Combines the microcontroller and LCD for a minimalistic and portable design.
3. **Real-Time Processing:**
   * With an inbuilt RTC (Real-Time Clock) module, the LPC2148 can maintain accurate timekeeping.
4. **Flexible Interfacing:**
   * The LPC2148 supports multiple communication protocols (I2C, SPI, UART), allowing easy integration with other devices like sensors or wireless modules.

**Disadvantages of Interfacing LPC2148 with LCD for a World Clock:**

**Limited Resources**:

1. **RAM and Flash Memory**: The LPC2148 has limited RAM (up to 32KB) and flash memory (up to 512KB)

**Power Consumption**:

1. Continuous running of the microcontroller and LCD screen can consume significant power