LAB NO : 7 DATE : 19/02/2025

Title : PROGRAMS ON MULTIPLEXED SEVEN SEGMENT DISPLAY

# Lab Exercise 1: Write C program to display the number “1234” serially in the seven segment display.

## Code :

#include <LPC17xx.h>

// Array for converting digits 0-9 to their 7-segment display encoding

unsigned char tohex[10] = {0x3F, 0x06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};

// Array to represent the digits we want to display ("1234")

unsigned char digits[4] = {1, 2, 3, 4};

unsigned int i = 0, j = 0;

int main() {

SystemInit();

SystemCoreClockUpdate();

// Configure GPIO pins for the 7-segment displays

// GPIO0 pins 0 to 7 will be used for the 7-segment segment control (8 bits)

LPC\_GPIO0->FIODIR |= 0xFF0; // Set bits 4-11 as output for the segments of 7-segment

// GPIO1 pins 23 to 26 will be used for digit selection (4 bits for 4 digits)

LPC\_GPIO1->FIODIR |= 0xF << 23; // Set bits 23-26 as output for selecting digits

while(1) {

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

// Select the current digit (activate the appropriate digit by shifting 1 to the correct bit)

LPC\_GPIO1->FIOPIN = (1 << (i + 23)); // Activate one of the four digits (0-3)

// Send the corresponding 7-segment pattern for the current digit

LPC\_GPIO0->FIOPIN = tohex[digits[i]] << 4; // Shift the 7-segment pattern for the current digit

for(j = 0; j < 100000; j++);

// Clear the 7-segment segments before displaying the next digit

// LPC\_GPIO0->FIOCLR |= 0xFF0; // Turn off all segments

}

}

return 0;

}

# Lab Exercise 2: Write C program to simulate a 4 digit BCD down counter. Use a timer for delay.

## Code :

#include <LPC17xx.h>

// Array for mapping digits 0-9 to their 7-segment display encoding

unsigned char tohex[10] = {0x3F, 0X06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};

// Variables used for controlling the countdown and loop indices

unsigned int i = 0, j = 0;

// Array to store the digits for each of the four 7-segment display positions

long int arr[4] = {9, 9, 9, 9}; // Starts at 9999, representing a four-digit countdown

void initTimer0(void){

LPC\_SC->PCONP |= (1<<1); //Power up TIM0

LPC\_SC->PCLKSEL0 &= (0x3 << 3); // Set clock for Timer 0 to CCLK

LPC\_TIM0->CTCR = 0x0;

LPC\_TIM0->PR = 999;

//25000 clck @25Mhz = 1ms

LPC\_TIM0->TCR = 0x02; //Reset timer

}

void delay(unsigned int millisec){

LPC\_TIM0->TCR = 0x02; //Reset Timer

LPC\_TIM0->TCR = 0x01; //Enable Timer

while (LPC\_TIM0->TC < millisec);

LPC\_TIM0->TCR = 0x00; //Disable timer

}

int main(){

// Initialize the system and update the system core clock (usually required for LPC17xx setup)

SystemInit();

SystemCoreClockUpdate();

initTimer0();

// Configure GPIO pins for the 7-segment displays

// LPC\_GPIO0 pins 0 to 7 are set as outputs to control the 7-segment display (8 bits)

LPC\_GPIO0->FIODIR |= 0xFF0; // Set bits 4 to 11 as output (8 bits for 7-segment display)

// LPC\_GPIO1 pins 23 to 26 are set as outputs to select which 7-segment digit to control

LPC\_GPIO1->FIODIR |= 0xF << 23; // Set bits 23-26 as output (4 bits for digit select)

while (1){

LPC\_TIM0->TCR = 0x02; //Reset Timer

LPC\_TIM0->TCR = 0x01; //Enable Timer

while (LPC\_TIM0->TC < 1000){

temp = counter;

LPC\_GPIO1->FIOPIN = 0 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 1 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 2 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 3 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

}

LPC\_TIM0->TCR = 0x00; //Disable timer

counter--;

if (counter == 0)

counter = 9999;

}

}

# Lab Exercise 3: Write C program for 4 digit BCD up/down counters on seven segment display using a switch and timer delay of 1-second between each count.

## Code :

#include <LPC17xx.h>

// Array for mapping digits 0-9 to their 7-segment display encoding

unsigned char tohex[10] = {0x3F, 0X06, 0x5B, 0x4F, 0x66, 0x6D, 0x7D, 0x07, 0x7F, 0x6F};

// Variables used for controlling the countdown and loop indices

unsigned int i = 0, j = 0, temp, counter=9999, switchState=1, flag = 1;

// Array to store the digits for each of the four 7-segment display positions

long int arr[4] = {9, 9, 9, 9}; // Starts at 9999, representing a four-digit countdown

void initTimer0(void){

LPC\_SC->PCONP |= (1<<1); //Power up TIM0

LPC\_SC->PCLKSEL0 &= (0x3 << 3); // Set clock for Timer 0 to CCLK

LPC\_TIM0->CTCR = 0x0;

LPC\_TIM0->PR = 999;

//25000 clck @25Mhz = 1ms

LPC\_TIM0->TCR = 0x02; //Reset timer

}

void delay(unsigned int millisec){

LPC\_TIM0->TCR = 0x02; //Reset Timer

LPC\_TIM0->TCR = 0x01; //Enable Timer

while (LPC\_TIM0->TC < millisec);

LPC\_TIM0->TCR = 0x00; //Disable timer

}

int main(){

// Initialize the system and update the system core clock (usually required for LPC17xx setup)

SystemInit();

SystemCoreClockUpdate();

initTimer0();

LPC\_PINCON->PINSEL4 &= 0xFCFFFFFF; //For Key

LPC\_GPIO2->FIODIR &= ~(1<<12);//0x00001000;

// Configure GPIO pins for the 7-segment displays

// LPC\_GPIO0 pins 0 to 7 are set as outputs to control the 7-segment display (8 bits)

LPC\_GPIO0->FIODIR |= 0xFF0; // Set bits 4 to 11 as output (8 bits for 7-segment display)

// LPC\_GPIO1 pins 23 to 26 are set as outputs to select which 7-segment digit to control

LPC\_GPIO1->FIODIR |= 0xF << 23; // Set bits 23-26 as output (4 bits for digit select)

while (1){

LPC\_TIM0->TCR = 0x02; //Reset Timer

LPC\_TIM0->TCR = 0x01; //Enable Timer

while (LPC\_TIM0->TC < 1000){

temp = counter;

LPC\_GPIO1->FIOPIN = 0 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 1 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 2 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

temp /= 10;

LPC\_GPIO1->FIOPIN = 3 << 23;

LPC\_GPIO0->FIOPIN = tohex[temp%10] << 4;

for(i=0; i<1000; i++);

}

LPC\_TIM0->TCR = 0x00; //Disable timer

switchState = (LPC\_GPIO2->FIOPIN >> 12) & 1;

if (switchState == 0){

flag = !flag;

if (flag) counter = 9999;

else counter = 0;

}

if (flag)

counter--;

else counter++;

if (counter == 0)

counter = 9999;

}

}

# Lab Exercise 4: Write a program for 4 digit Hexadecimal up/down counters on seven segment using a switch and timer with a delay of 1-second between each count.

## Code :

#include <LPC17xx.h>

unsigned int i, value = 65535;

unsigned seven\_seg[16]={0x3F,0x06,0x5B,0x4F,0x66,0x6D,0x7D,0x07,0x7F,0x6F,0x77,0x7C,0x39,0x5E,0x79,0x71};

unsigned int seg\_select[4] = {0, 1<<23, 2<<23, 3<<23};

void display() {

int digits[4] = {0,0,0,0};

int ind = 0,x, val = value;

while(val > 0) {

x = val % 16;

val = val / 16;

digits[ind++] = x;

}

for(ind = 0; ind < 4; ind++) {

LPC\_GPIO1->FIOPIN = seg\_select[ind];

LPC\_GPIO0->FIOPIN = seven\_seg[digits[ind]] << 4;

for(i = 0; i < 1000; i++);

}

}

int main(void) {

SystemInit(); //Add these two function for its internal operation

SystemCoreClockUpdate();

LPC\_PINCON->PINSEL0 &= 0xFF0000FF; //Configure Port0 PINS P0.4-P0.11 as GPIO function

LPC\_PINCON->PINSEL3 &= ~(0xF<<23); //Configure P1.23 to P1.26 as GPIO function

LPC\_PINCON->PINSEL4 &= 0xFFFFFFF0; //Configure Port2 PIN P2.0 and P2.1 as GPIO function

LPC\_GPIO0->FIODIR |= 0xFF0; //Configure P0.4-P0.11 as output port

LPC\_GPIO2->FIODIR &= ~(0); //Configure pin 0 for input

LPC\_GPIO1->FIODIR |= 0xF << 23;

LPC\_GPIO1->FIOPIN = 0;

while(1) {

for(i = 0; i < 1000; i++);

if (LPC\_GPIO2->FIOPIN & 1)

value += 1;

else

value -=1;

if (value == 0)

value = 65535;

if (value == 65536)

value = 0;

display();

}

}