ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT

EE447 Introduction to Microprocessors

Term Project Final Report

Water Level Controller

Name : Yasincan Bozkurt

ID : 2304202

Name : Afer Olkay

ID : 2305126

Contents

Introduction

The objective of this laboratory project is to design and implement a water level controller system using a microcontroller. The system will continuously monitor the water level using a sensor and attempt to maintain it within a predetermined range using two water pumps. The current configuration and measurements will be displayed on a Nokia 5110 LCD screen using the SPI module, and the system will also use the GPIO for the on-board RGB LED and pushbuttons. The system has three main functions: water level sensing, controlling the water level, and providing a user interface.

Team Organization

Motor driving, water level sensor reading, potentiometer for setting limits parts are done by Yasincan Bozkurt. LCD driving, interrupt and periodic operation parts and LED parts are done by Afer Olkay

Overall Project

In the project, all operations are handled by interrupts and subroutines, with the main program only used for calling functions that are functional subparts in different files. The components and their detailed port connections will be outlined in this section of the report.

Components & Connections

Components of the Project
TM4C123GH6PM Board
NOKIA 5110 LCD Screen
4x4 Keypad
Potentiometer
2 Water Pumps
Water Level Sensor
2 BD139 BJT
LM358 Opamp

Table 1. Componts of the project

Components of the project is listed in Table1.

Peripherals	**Pins on Peripheral / Pin Name**	**TM4C123G**
LCD Screen	SSI0CLK	PA2
	SSI0FSS	PA3
	SSI0RX	PA4
	SSI0TX	PA5
	RESET	PA6
	D/C	PA7
	LED	VBUS
	GND	GND
Keypad	S1	PB0
Pot	Signal	PE2
Water Sensor	Signal	PE3
	Red Led	PF1
	Blue Led	PF2
	Green Led	PF3

Table 2. Connections of Peripherals

In order to avoid confisions and multiply the same port breadboard is used. The ground, 3.3V, and VBUS pins of the Tiva board are multiplied on the breadboard.

Flowchart

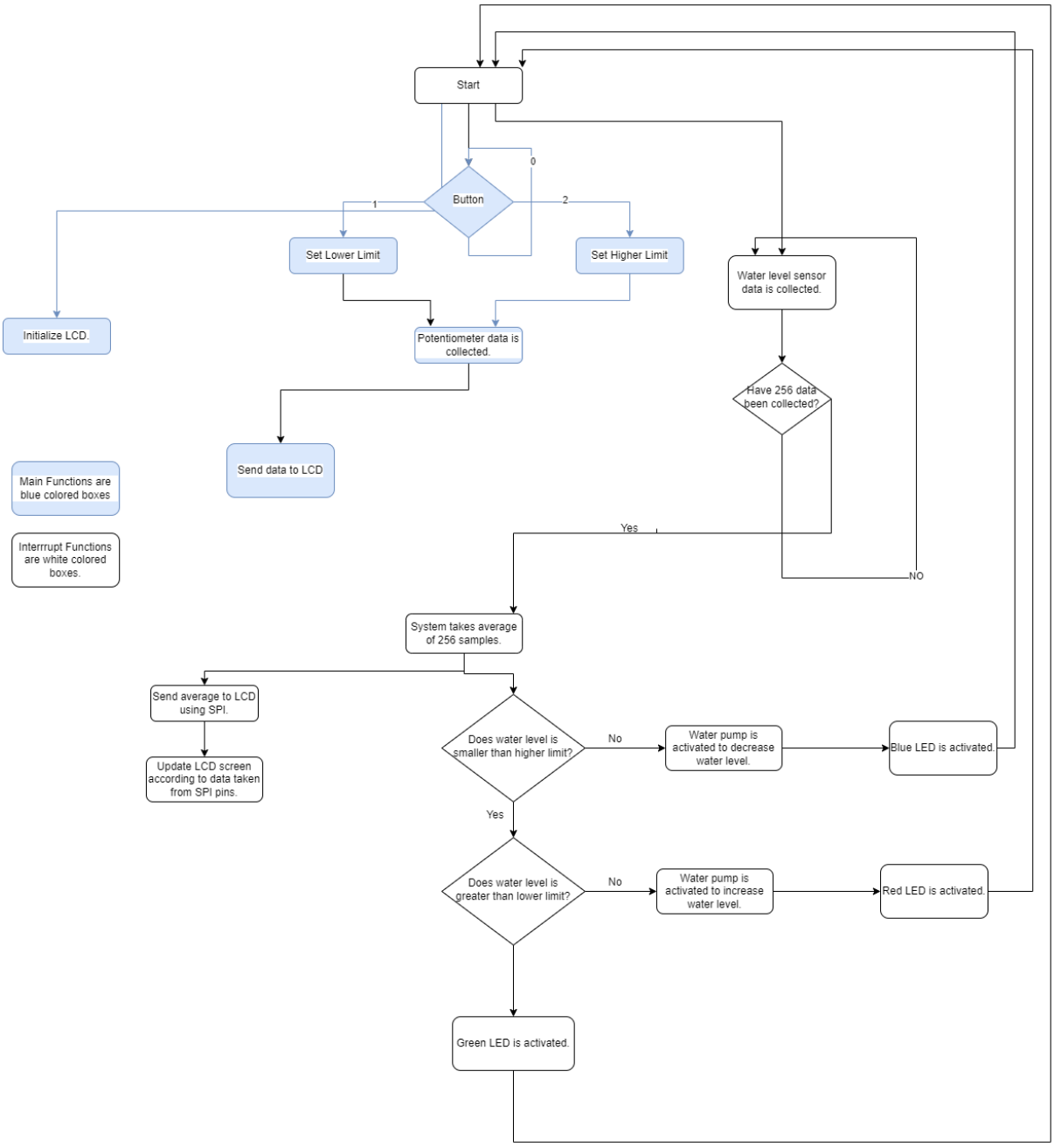

Figure 1. Flowchart of the Project

Submodules

Water Level Sensing Unit

This module uses water level sensor to read water level. Water level sensor is a varying resistance that change its resistance according to water level. The code uses a timer interrupt to periodically read the water level from the sensor and store the value in an array. Once the array is filled with 256 samples, the code calculates the average water level by adding up all the values in the array and dividing the total by the number of samples. This average water level is then used to update the output elements such as the LCD screen and the RGB LED. The output elements will display the current water level and may change their state (e.g. turn on or off) depending on whether the water level is within a predefined range.

The code includes various functions and variables related to the control of a water level system. The buffer array is used to store samples of the water level, and the currentBufferSize variable keeps track of the number of samples in the buffer. The currLevel variable stores the current water level. The str array is used as a string buffer for displaying text on the screen. The state variable is used to keep track of the current state of the system, and the lowLimit and highLimit variables store the lower and upper bounds of the predetermined range for the water level. The num2str function is used to convert a number to a string and store it in the msg array. The Delay function is used to introduce a delay in the program. The periodicReadInitialization function is used to enable and set up a timer for periodic readings of the water level. The Systic_Handler function is called periodically as specified by the timer and is used to read the water level and update the system accordingly.

Water Level Control Unit

Water level control unit uses two water pumps to control the water level. It checks upper limit and lower limit of the water level which are controlled by user using potentiometer. User can use 4x4 keypads button to switch between upper limit setting, lower limit setting and no operation mode. If water level is higher than upper limit, one water pump starts working to decrease the water level. If water level is lower than lower limit, one water pump works to increase the water level. Water pumps are controlled with two BJT’s.

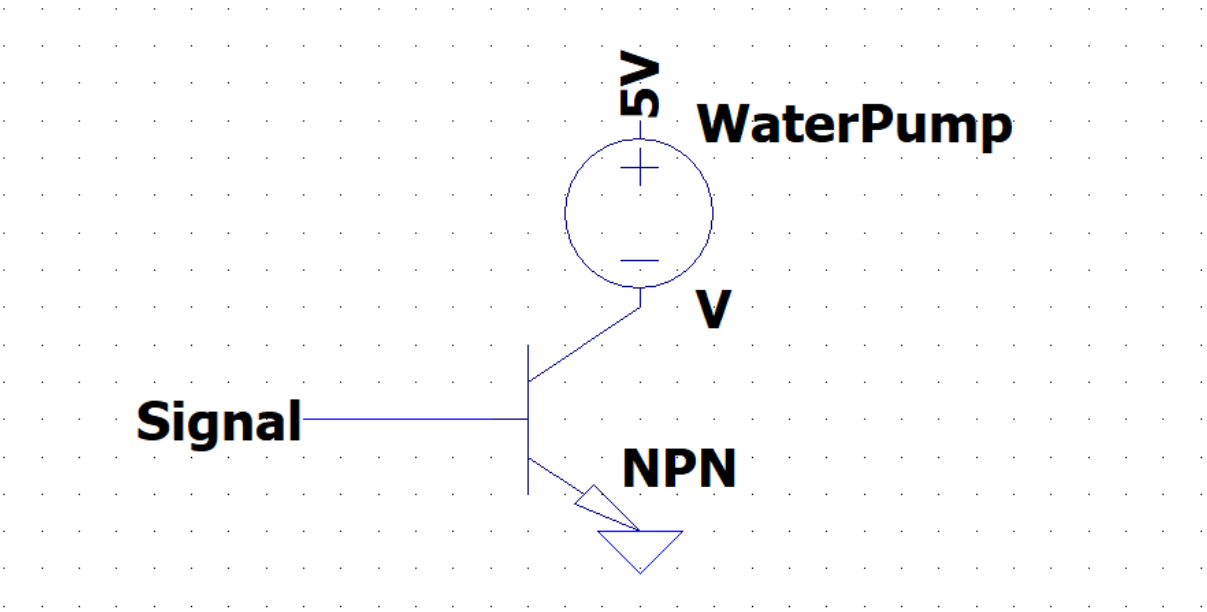

Figure 2. BJT Configuration

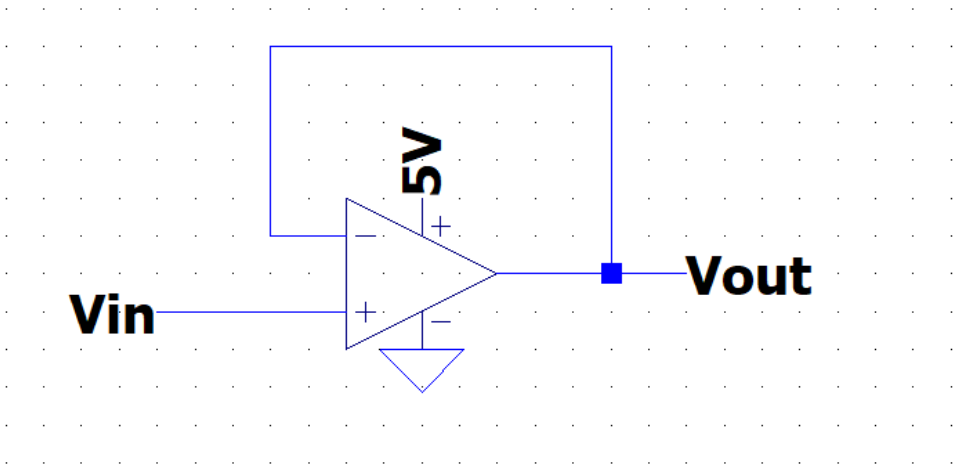

Figure 3. Opamp Configuration

As can be seen in Figure 2, BJT's are used in common emitter configuration. Signal pin is connected to base of the BJT. Signal pins are LED ouput pins of the TM4C123H6PM board. These are namely PF1 for the water pump which increase water level and PF2 for the water pump which decrease the waer level. We used 5V to supply the water pumps. This supply is separate from board. The reason for this is water pumps are drawing too much current that can cause problems for board. Also, we seperated board and motor driving circuits by using buffer circuits. We use LM358 opamp with configuration given in Figure 3 in order to avoid any possible damages on board.

User Interface

LCD Screen

The aim of this module is show users the level of the water and its upper and lower limits. The LCD driver code control the display on a Nokia 5110 LCD screen. The screen is connected

to the microcontroller through the SSI0 module using the pins PA2, PA3, PA4, PA5, PA6, and PA7. The driver uses the GPIO and SSI registers to control the communication between the microcontroller and the screen. The ASCII table is also provided in the code, which contains the 6x8 pixel font data for each ASCII character that can be displayed on the screen. The driver includes routines for initializing the screen, sending commands and data to the screen, and displaying text on the screen. It also includes a routine for drawing a horizontal dot plot graph on the screen to visualize the water level data.

LEDs

Leds are used to give information about the comparison of water level with limits. We used a funtion called updateLED. This function updates the state of the RGB LED according to the current water level. If the water level is above the high limit, the blue LED is turned on. If the water level is below the low limit, the red LED is turned on. Otherwise, the green LED is turned on. The function first turns off all the LEDs by using a bitmask to clear the appropriate bits in the DATA register of the GPIOF port. It then sets the appropriate bits in the DATA register to turn on the desired LED.

Overall System

Pictures taken from overall system can be seen in figure 4,5 and 6.

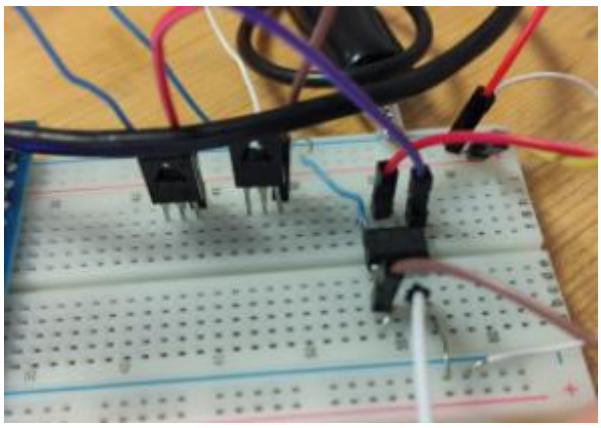

Figure 4. Driver Circuit of Water Pumps

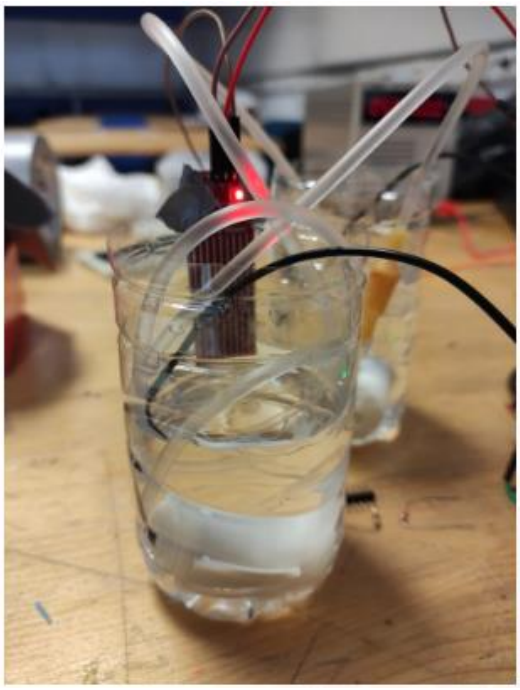

Figure 5. Water Level Sensing Unit

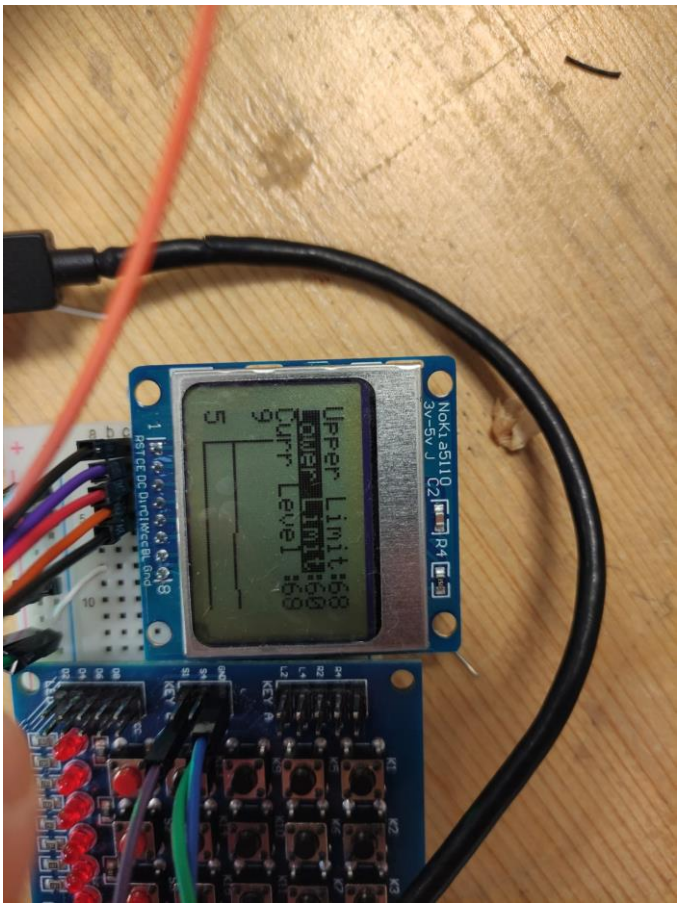

Figure 6. User Interface

Conclusion

In this project, we designed and implemented a water level controller system using a microcontroller and various hardware components. The system was able to continuously monitor the water level using a sensor and maintain it within a predetermined range using two water pumps. We also displayed the current configuration and measurements on a Nokia 5110 LCD screen and controlled an RGB LED and pushbuttons using the microcontroller.

We fulfilled the requirements of several functions including water level sensing, water level control, and a user interface. We also added additional features such as the ability to set the range using a potentiometer or keypad, and the option to display the water level as a graph on the LCD screen.