USB Interface

E/14/262 Punchihewa PS
E/14/302 Samarakoon SMHM
E/14/317 Senanayake SMAJ

1. Proposal
    1.1. Overview of the project explaining rationale and outcome
    1.2. Distribution of the work among the group members
    1.3. Associated figures
2. Project Work
    2.1. Should have the separate sections according to the work distribution and should be written by the member who is responsible for each section. I will give marks separately for each member.
    2.2. Should include additional supporting materials such as figures, pseudo codes etc.
    2.3. Annotated Source Code should be included in Appendix

2.4. Extended Communication

3. Results and discussion
    3.1. Discuss what you have achieved and could not achieved according to the project plan giving reasons.
    3.2. Discuss the advantages and disadvantages of your approach.
    3.3. Critically analyse the contribution of each member in the project and how to improve the contributions if possible.
4. Future work
    4.1. Discuss how this project can be improved and also extended towards an industrial project.
5. References

1. Proposal

1.1 Overview

Communication via USB is the most popular mode of interfacing nowadays. Almost all the peripherals that have other kind of interfacing (serial port, parallel port..) are converting the ports to be compatible with USB. Building a general USB interface that can later be extended to industrial components handling is of major concern. Because of this reason leaders in microcontroller designing have explicitly designed microchips that can support USB protocol. This attempt is to program one such microchip to build a general I/O board. The board will be hereafter referred as the GIOB. GIOB has the following features.

- can read 16 input lines
- can write to 8 output lines
- no external power supply required, power is supplied via the PC's USB bus
- compatible with USB 2.0, 3.0

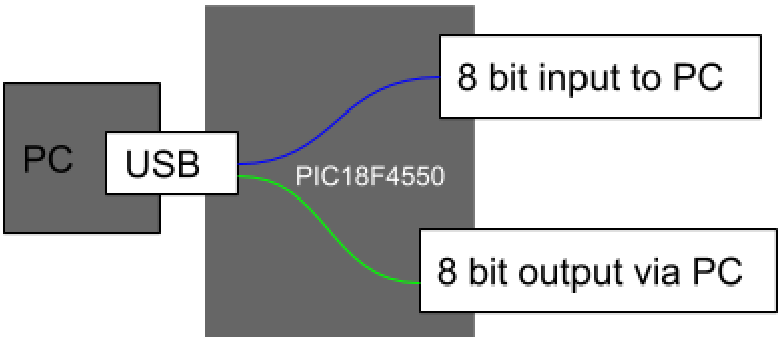

Figure 1.1

To design GIOB we use the PIC18F4550 USB module which is a 8 bit PIC microcontroller. We chose this module as it has a stable clock for full-speed USB operation and is widely used for USB communication. Also the designer of this microcontroller, MICROCHIP has provided the tools to program the IC. Since this IC is the widely used component in this field there is an active supportive community.

To start with the project, we are going to refer [USB interface to the PIC Microcontroller]
Then, extend the design and the concept to more generalized plug and play I/O interface with maximum data lines possible.

1.2 Workload Distribution

workload can be categorized as,

1. Proteus simulation and PCB design E14317
2. PIC18F4550 programming E14302
3. Interface Development E14262
4. Updating the Report in respective fields

1.3 Associated Figures

2. Project Work

2.1 Proteus Simulation

2.1.1.1 About Proteus

It is a software suite from **Labcentric Electronics** containing **schematic**, **simulation** as well as **PCB designing**.

- **ISIS** is the software used to draw schematics and simulate the circuits in real time.The simulation allows human access during run time,thus providing real time simulation.
- **ARES** is used for PCB designing.It has the feature of viewing output in 3D view of the designed PCB along with components.
- The designer can also develop 2D drawings for the product.

Features

ISIS has wide range of components in its library. It has sources, signal generators, measurement and analysis tools like oscilloscope, voltmeter, ammeter etc., probes for real time monitoring of the parameters of the circuit, switches, displays, loads like motors and lamps, discrete components like resistors, capacitors, inductors, transformers, digital and analog Integrated circuits, semiconductor switches, relays, microcontrollers, processors, sensors etc.

2.1.1.2 PIC18F4550

The microcontroller is very famous due its functionalities and features such as ADC and USB Integration. A typical PIC18F4550 comes in various packages like DIP, QPF and QPN. It has 16 bit Instruction Set Architecture, (ISA) and also has an Extended Instruction Set as a special feature. PIC18F4550 has 256 bytes of EEPROM, 2KB of SRAM and 32KB of flash memory. The modern protocols for communications like USB, SPI, EUSART are supported. PIC18F4550 is compatible to work with different internal and external clock sources. It comes with four built-in timers or an external oscillator can be interfaced for clocking. The frequency limit for a PIC18F4550 is from 31 KHz to 48 MHz respectively. Microcontroller is compatible with the USB speeds,Full speed (12 Mbps) and Low speed (1.2 Mbps). 40 pins of PIC18F4550 are divided into 5 ports. 35 pins are Input-Output pins which can be configured for general Input or Output by setting registers associated with them. Three registers are associated with each port

TRIX - This register assigns the direction of the pins
LATX - The latch registers reds and modifies the write operation on the value of I/O pin and stored the output data that is to be passed on to the external hardware
PORTX - Reads the device level, stores the Input level of the pins and reads and registers the input signal from the external device if the pin is configured as Input

2.1.2 Design & Functionality

2.1.2.1 Direct 8 IO

Circuit Functionality

Functionality is to perform a basic IO operations to the PIC18F4550 via USB. Therefore we planned to read 2 inputs from SPST Switches and write 3 outputs to light 3 LEDs. This demonstration is only for the purpose of the Proteus simulation and the final circuit will be almost same to this except for the further developments.

To mimic the IO functionalities in the virtual simulation, we developed a simple GUI program. The GUI program has the ability to perform IO operations to the microcontroller.

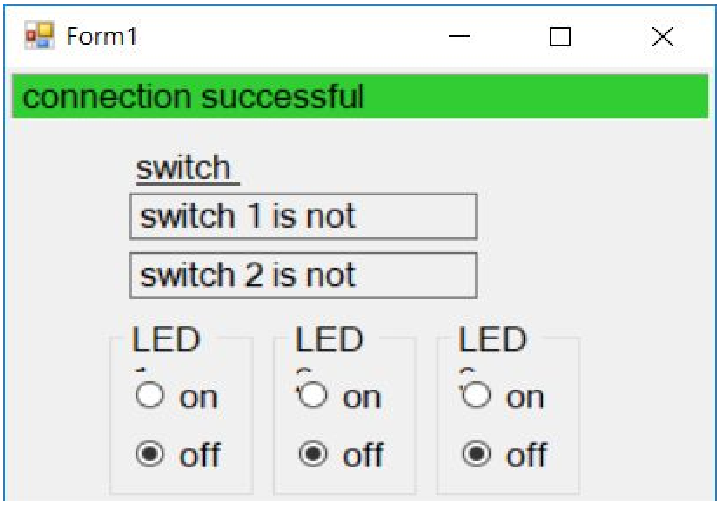

Figure 01: GUI for the IO operations

Circuit Design

The design of the circuit, that we implemented in Proteus is below.

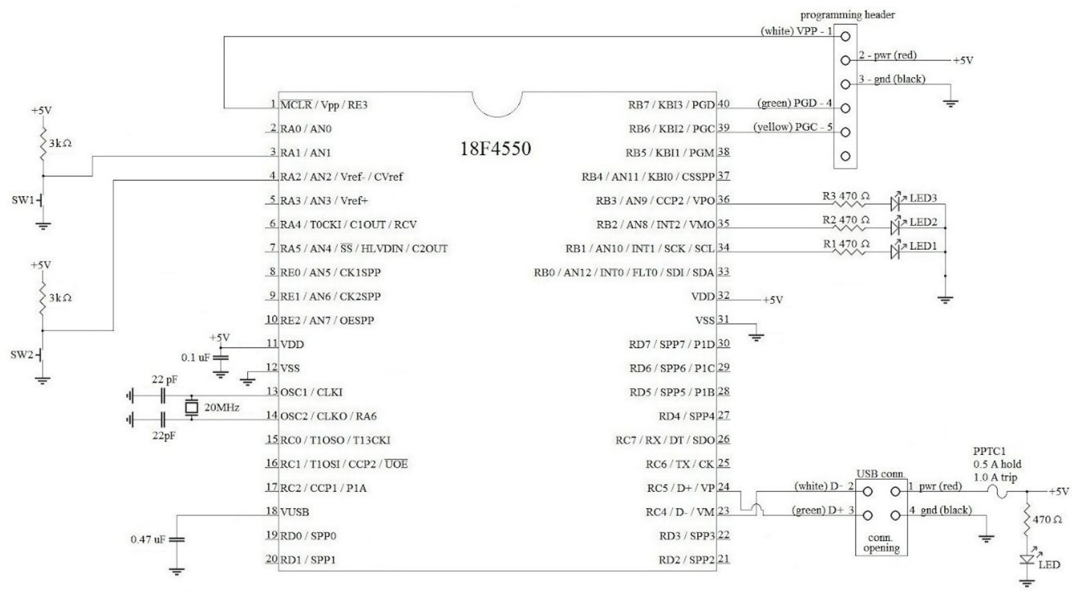

Figure 02: Circuit Design

Design Notes

- In the above design ***+5v*** is attached to the power from bus, not from an external power supply.
- But in the means of a virtual simulation, these ***+5v*** are supplied externally.

Components

- PIC18F4550
- 6 Pin Programming Header
- Crystal Oscillator(20MHz)
- A USB Connector
- Resistors(3kΩx2 & 470Ωx3)
- Capacitors(0.1μFx1 & 22pFx2 & 0.47μF)
- 3 LEDs
- Interactive SPST Switch(x2)

The Proteus schematic of the above circuit is below.

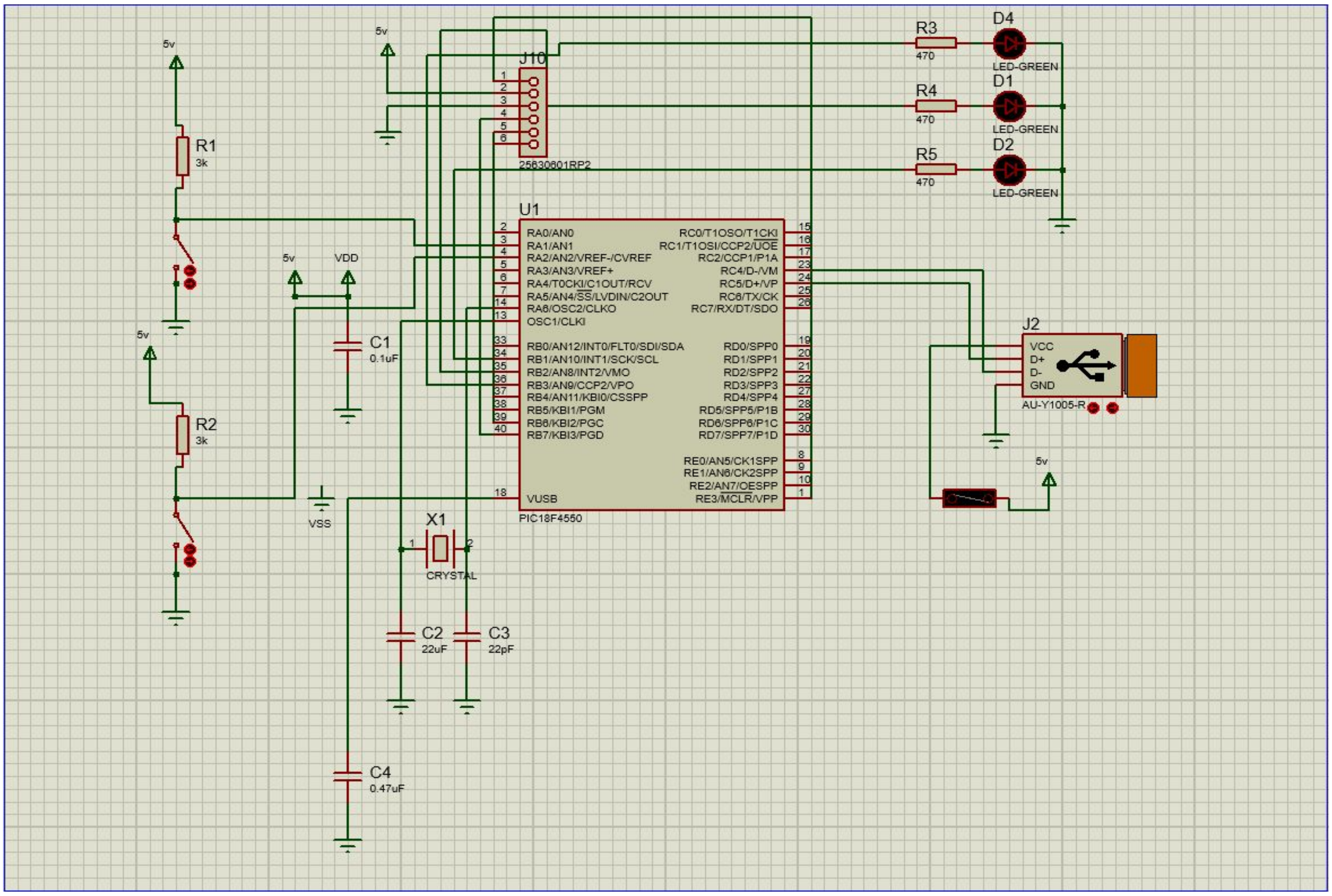

Figure 03: Proteus Schematic

Procedure

1. First of all we installed the 'Proteus virtual USB driver', in order to identify the virtual USB as an device by the OS, while in the simulation.The above mentioned driver can be found in the following site. (https://proteus-virtual-usb-drivers.software.informer.com/7.8/)
2. Then, we programmed the PIC to function according our task. The C program written for the task is compiled to an hex file using MPLab IDE (v8.85)
3. The compiled hex file is then uploaded to the virtual PIC18F4550 in the Proteus simulation.

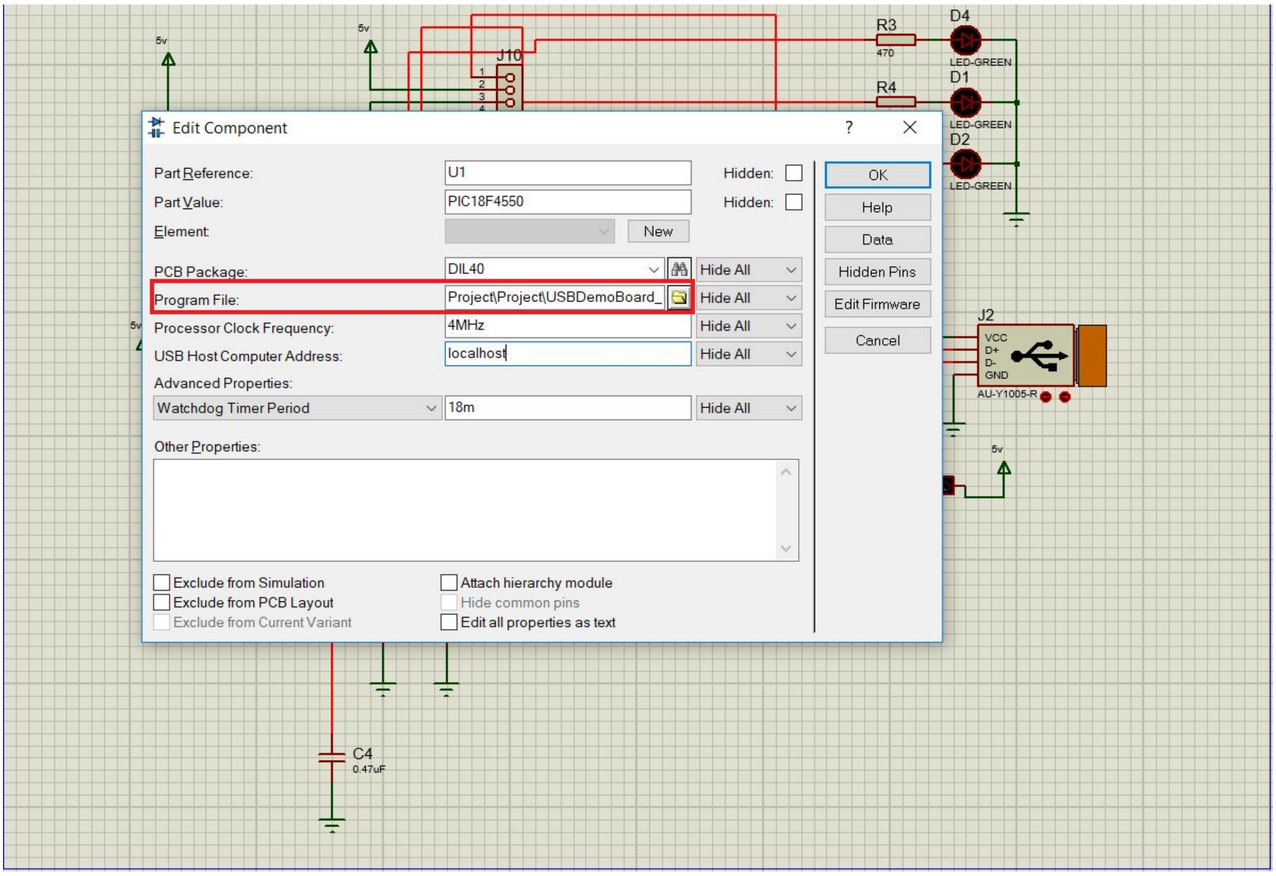

Figure 04: Uploading the hex file

4. After uploading the hex file, the simulation is ran. if successful, now we can perform IO operations to the PIC microcontroller.

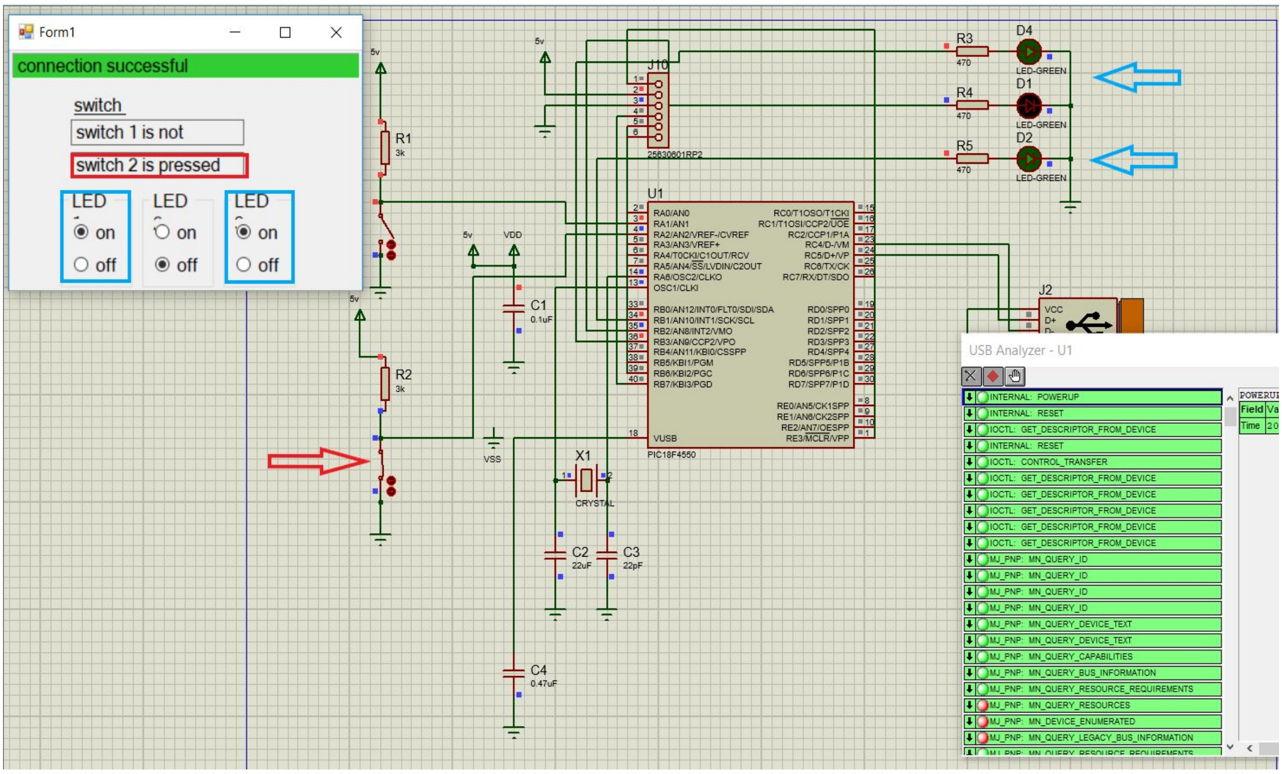

Figure 05: Successful simulation

2.1.2.2 Extended 8 O/16I

The previous implementation of the circuit is extended with 8 outputs and 16 inputs. The schematic is below.

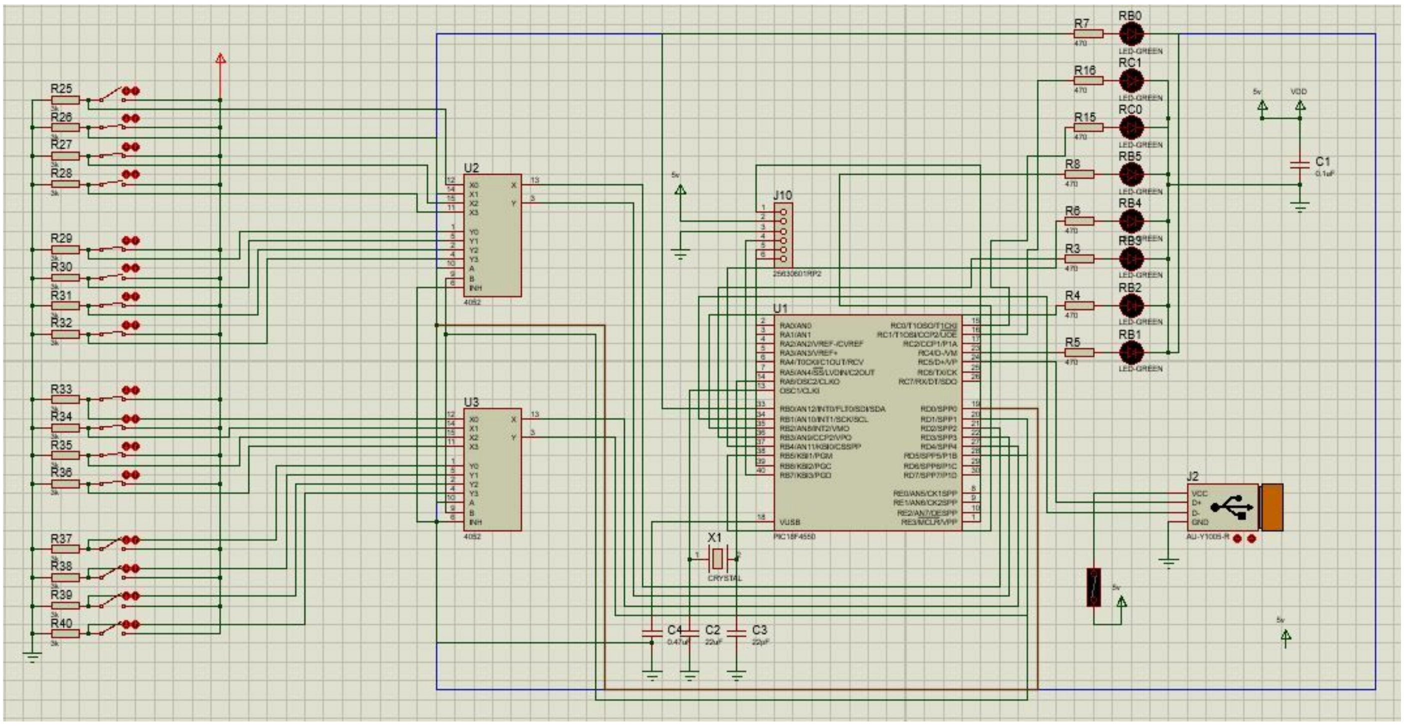

Figure 06: Extended I/O Proteus Schematic

Components

- PIC18F4550
- 6 Pin Programming Header
- Crystal Oscillator(20MHz)
- A USB Connector
- Resistors(3kΩx16 & 470Ωx8)
- Capacitors(0.1μFx1 & 22pFx2 & 0.47μF)
- 8 LEDs
- Interactive SPST Switch(x16)
- Dual channel 4:1 Multiplexers(x2)

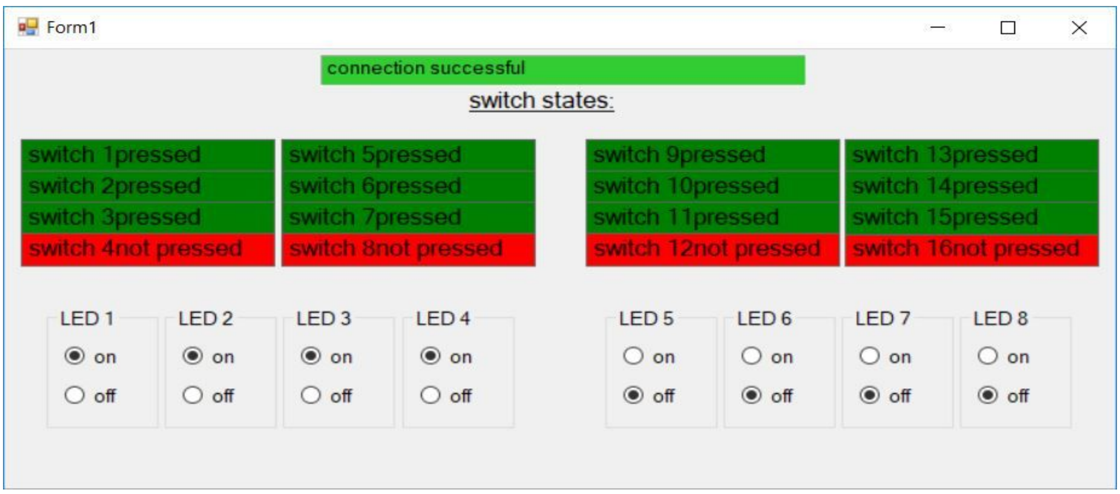

Figure 07: Extended I/O GUI

2.2 Programming

Figure 08: Programming Steps

2.3 PCB Design

2.3.1 Layout of the GPIOB

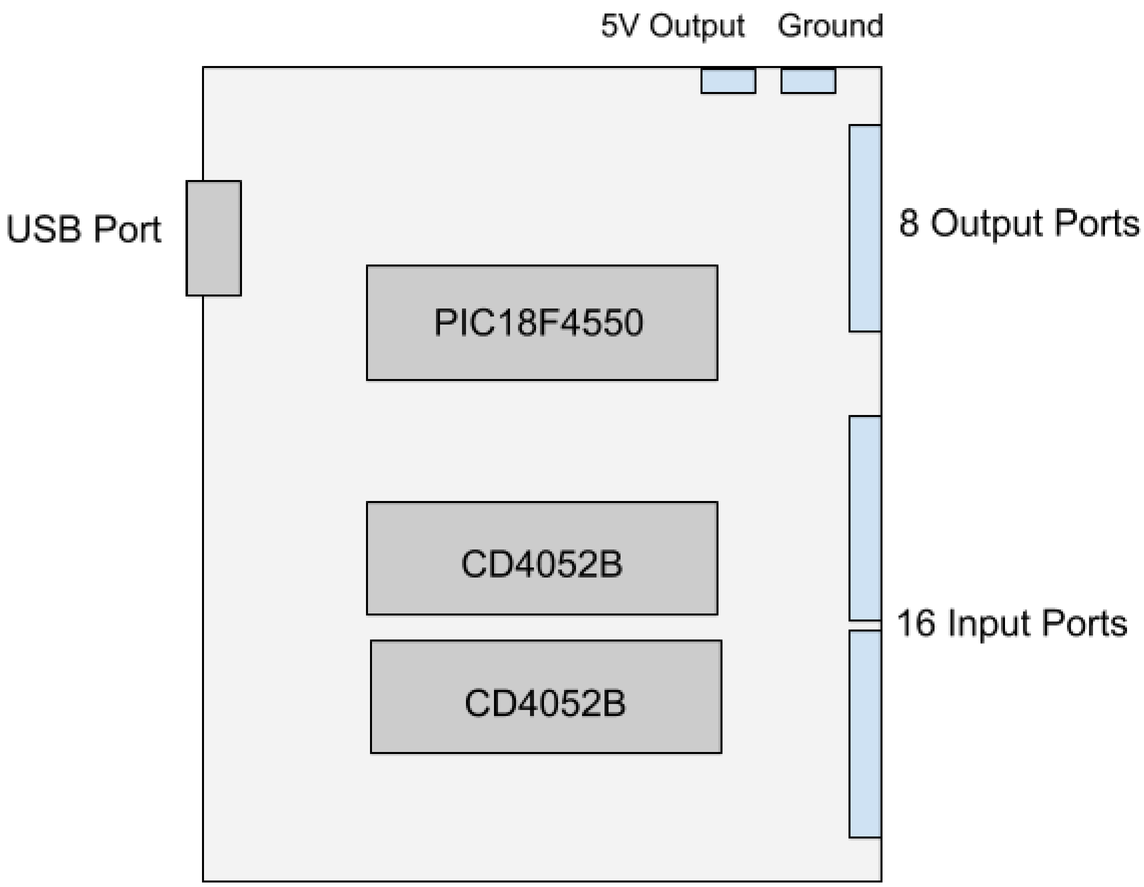

Figure 09: GPIOB Layout

3 Results and discussion

3.1 Limitations of GIOB

As mentioned in the overview GIOB has only 8 output lines. That is because GIOB is driven only via PC USB power bus. The

References

http://www.ti.com/lit/ds/scds307c/scds307c.pdf
https://www.st.com/resource/en/datasheet/CD00256013.pdf
http://www.ti.com/lit/ds/symlink/cd4097b.pdf
16:1 mux
MC14067
MPC506
74HC4067
74HCT4067
http://www.kevinmfodor.com/home/My-Blog/microcontrollerinputprotectiontechniques
https://arduino.stackexchange.com/questions/13126/best-way-to-protect-a-digital-or-analog-input-from-12volts

AC switch as an input to PIC

Difference between optoisolator(optocoupler) and solid state relay

Protecting Output Pin
best-way-to-protect-a-digital-output