Digital Clock

Based on PIC16F877A

Embedded System Design

By Methsara Dissanayeka

Introduction

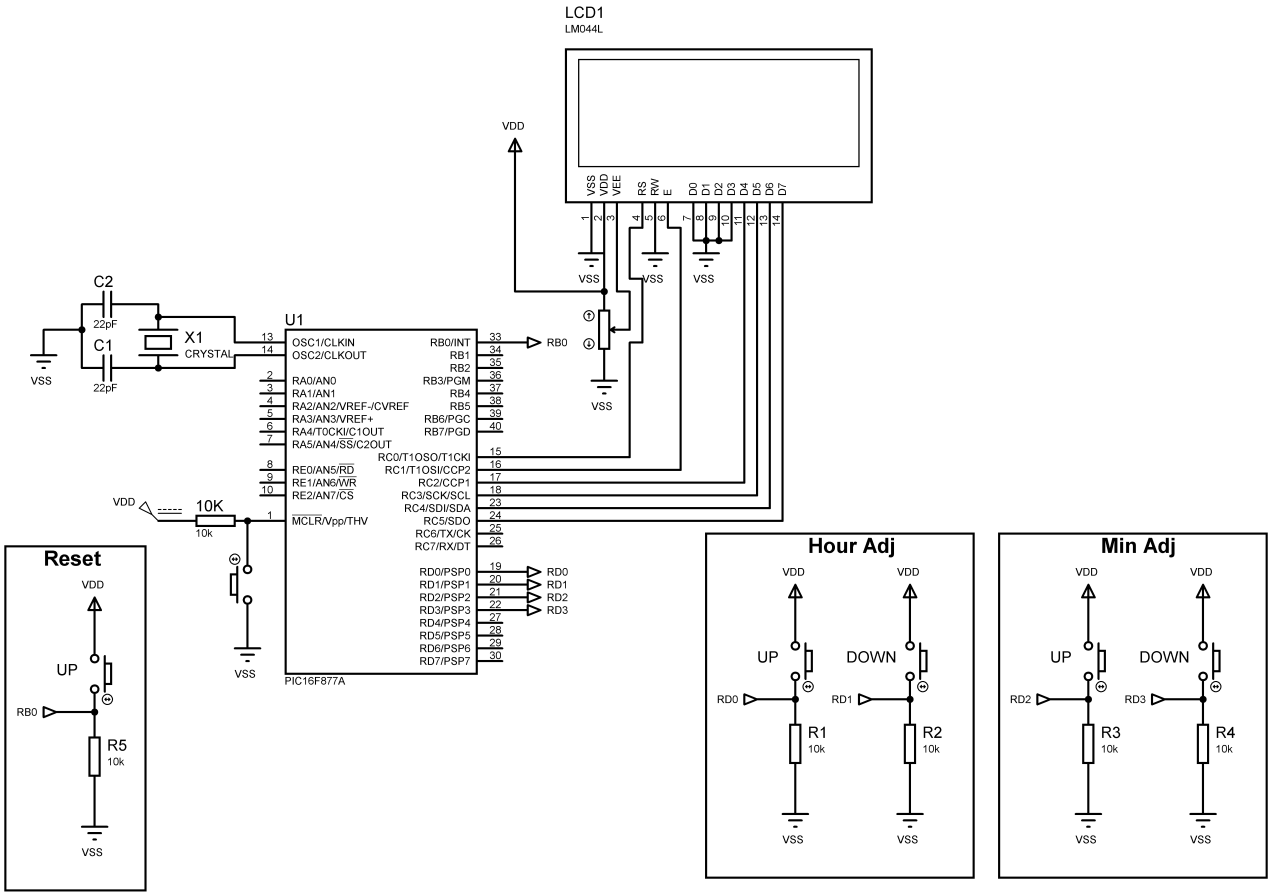
This is a project based on designing a digital clock using the PIC16F877A microcontroller. PIC16F877A is a widely used microcontroller in the embedded system industry due to its rich features and reliability. MicroC is a popular high-level programming language used to program PIC microcontrollers, including the PIC16F877A. It is a user-friendly language that simplifies the development process by providing a range of pre-defined libraries and functions, making it easy to code complex applications. The digital clock system operates using the PIC16F877A microcontroller, which is a powerful and versatile device that is commonly used in a wide range of applications. To connect the system to the display, I utilised PORT C, which is a flexible and configurable port that is commonly used for LCD interfacing. The LCD display provides a clear and easy-to-read interface for the user, displaying the current time and allowing for easy time adjustments.To take inputs and set the time, I used PORT D and RB0 (external interrupt pin). This setup is incredibly user-friendly, allowing the user to easily make changes to the hours and minutes using up and down push buttons. Additionally, the microcontroller runs in execution mode, which ensures that the system runs smoothly and efficiently. In terms of design, this project required careful planning and attention to detail. I had to consider a range of factors, including power consumption, circuit stability, and overall functionality. Through careful testing and iteration, I was able to create a reliable and efficient clock system that meets my design goals.MicroC is a powerful programming language that can be used to implement accurate clocks using the TIMER0 module in PIC microcontrollers. The TIMER0 module is a timer/counter that can be used to count events, generate interrupts at specific time intervals, and provide accurate timing in embedded applications. To implement an accurate clock using TIMER0, programmers can use MicroC to write code that configures the TIMER0 module to generate interrupts at a specific frequency, which can then be used to update the clock display. By using TIMER0 interrupts, the clock can maintain accuracy over long periods of time.

Additional Features

- Reset Button
- Updated formatted month visibility in display
- Updated theme
- Interrupt based Rest Button to use Digital clock as timer.

The blinking of the colons can't see properly in simulation due to rewriting on the date array. To See that properly. I used another point to blink “.” On Second line. Date is Starting from 1st of Jan in 2000. Assume that each month has 30 days.To reduce codeines initialisation of bits, I called whole registry at one.

Schematic Diagram

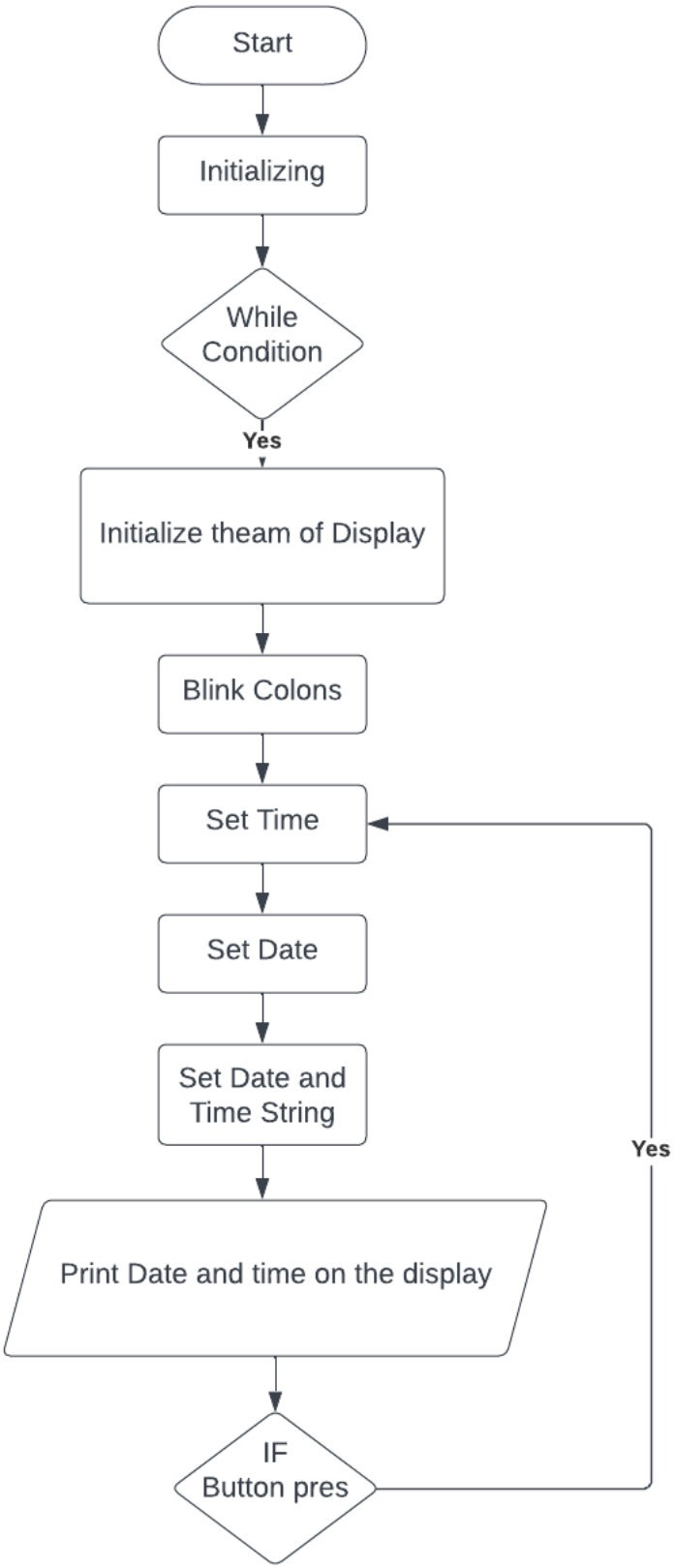


Schematic Digram

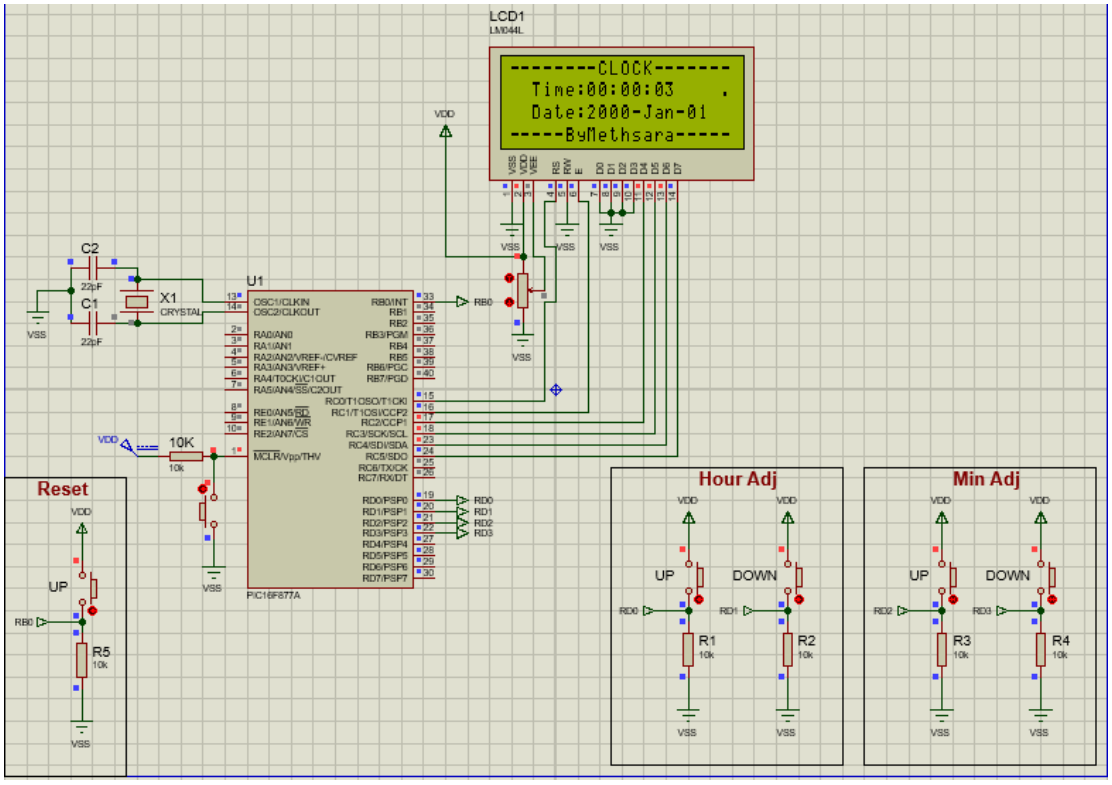
Component list

Name	Quantity
Microcontroller (PIC16F877A)	1
LCD Display (LM044L)	1
Push Button	6
10K ohm Resistor	6
Crystal (4 MHz)	1
22 pF Disk Capacitor	2
Variable Resistor	1
5 V Voltage Source	1

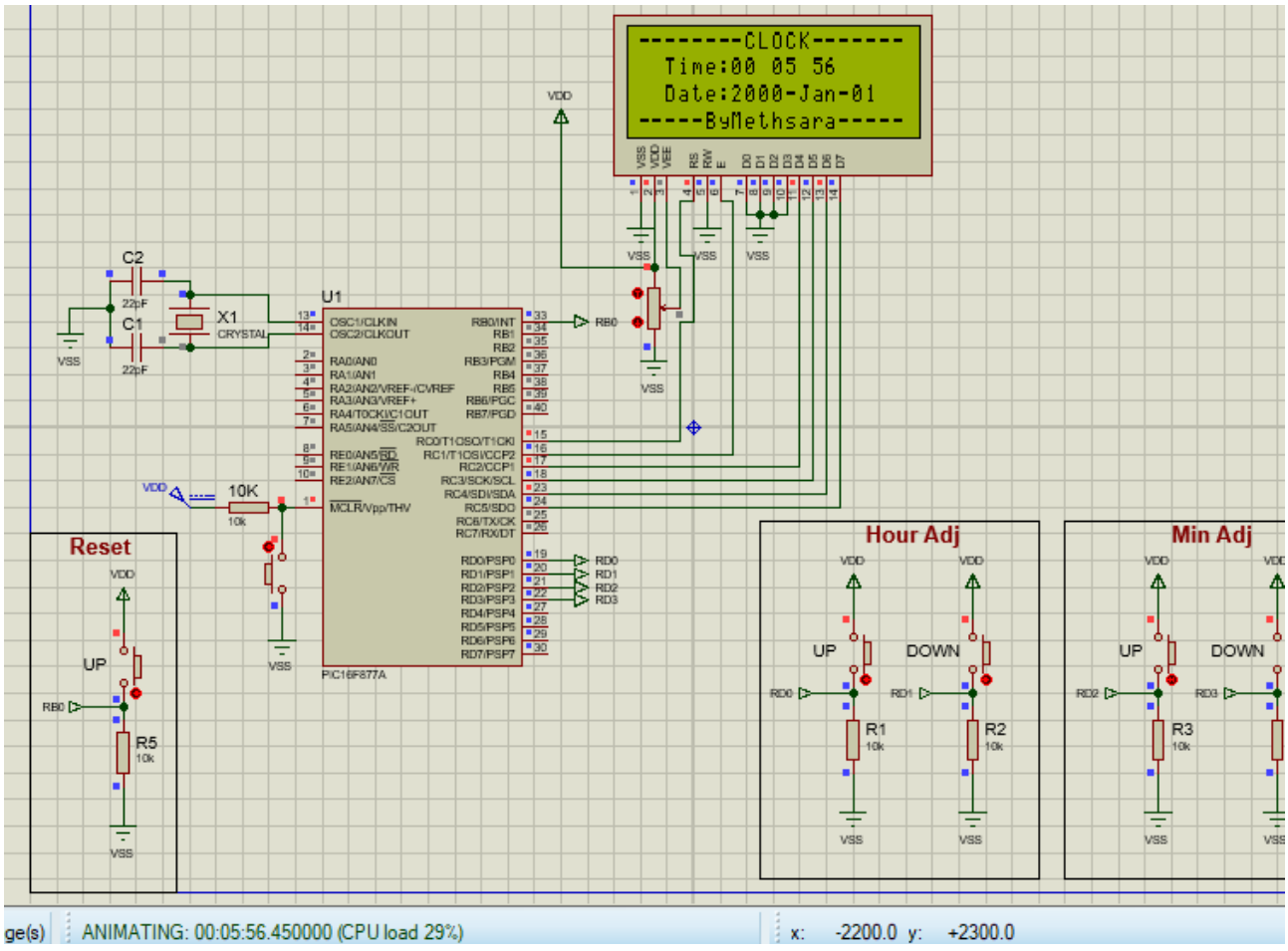
Flowchart of the program



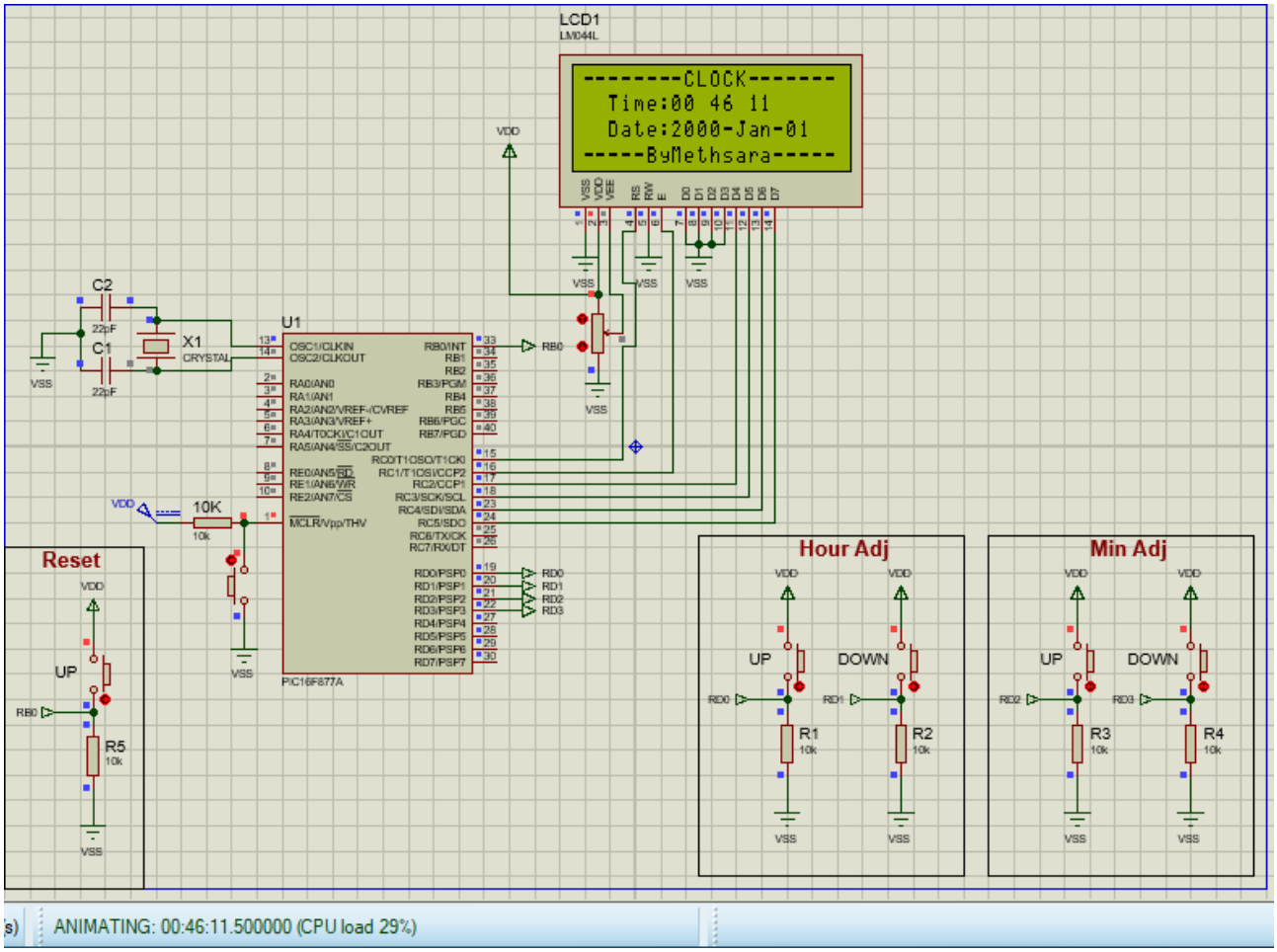
Test Results



Simulations snap



Simulation after 5 min and 56 sec



Simulation after 46 min and 11 sec

Conclusion

The PIC16F877A microcontroller, which is a dependable and flexible technology extensively used in the embedded system sector, will be utilised to construct a digital clock for this project. The clock system makes use of MicroC, a straightforward high-level programming language that streamlines the creation process by offering pre-defined libraries and functions. The clock can be readily modified via PORT D and RB0, which accept input from up and down push buttons, and the clock display is attached to PORT C, which is frequently used for LCD interface. The microcontroller operates in execution mode to guarantee a seamless and effective process.

References

- https://www.microchip.com/en-us/product/PIC16F877A
- Micro C Documentation