ARMv7 PROJECT

SIBA PRASAD CHOUDHURY

Digital Clock With Alarm and Medicine Reminder over Bluetooth using ARMv7-TDMI (LPC2148)

Contents

1 ABSTRACT

This projects aims to design a basic digital clock using ARMv7 based micro-controller (LPC2148/LPC2138) and interface it with display and Bluetooth module for alarm and medicine reminder features.

2 INTRODUCTION

The designed digital clock has a LCD for displaying the time and date. The clock has alarm feature. For people who forget to take their medication it is a very useful thing to have a reminder. The medicine reminder feature will display the message to take medicine on LCD and also it will send the same message over Bluetooth.

3 MOTIVATION

Motivation is the reason for people's actions, willingness and goals. My motivation to work on this project came from the fact that we see a a lot interesting gadgets around us all the time. But being an embedded systems enthusiast I always wanted to design such a gadget myself. So I made a digital clock with all the basic features. This device is no way near industrial standard, but it surely have helped me to apply the concepts I have learned to use in real life.

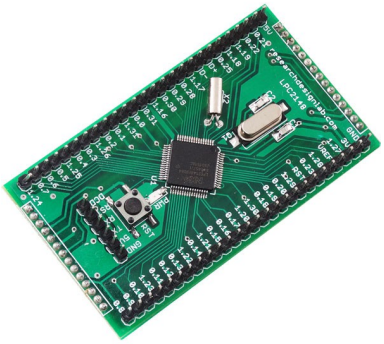
4 FEATURES

1.Can display Time.
2.Can set Alarm.
3.Can edit the Time or Alarm time using keyboard.
4.Can send alarm message over Bluetooth.

5 TOOL'S DESCRIPTION

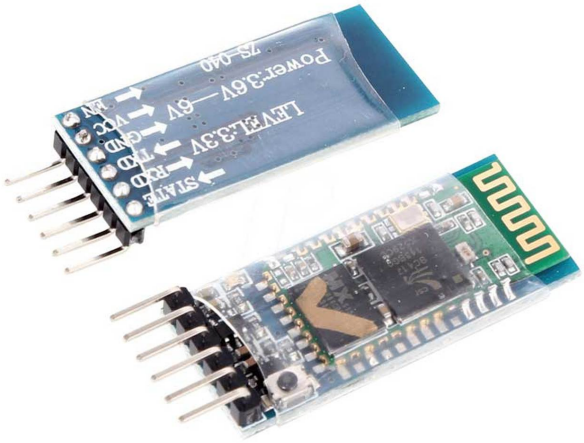
i.LPC2148/LPC2138

The LPC2148 is an ARM7TDMI-S based 32-bit RISC Microcontroller with Thumb extensions 512KB on-chip Flash ROM with In-System Programming (ISP) and In-Application Programming (IAP), 32KB RAM, Vectored Interrupt Controller, Two 10bit ADCs with 14 channels, USB 2.0 Full Speed Device Controller, Two UARTs, one with full modem interface. Two I2C serial interfaces, Two SPI serial interfaces Two 32-bit timers, Watchdog Timer, PWM unit, Real Time Clock with optional battery backup, Brown out detect circuit General purpose I/O pins. CPU clock up to 60 MHz, On-chip crystal oscillator and On-chip PLL.



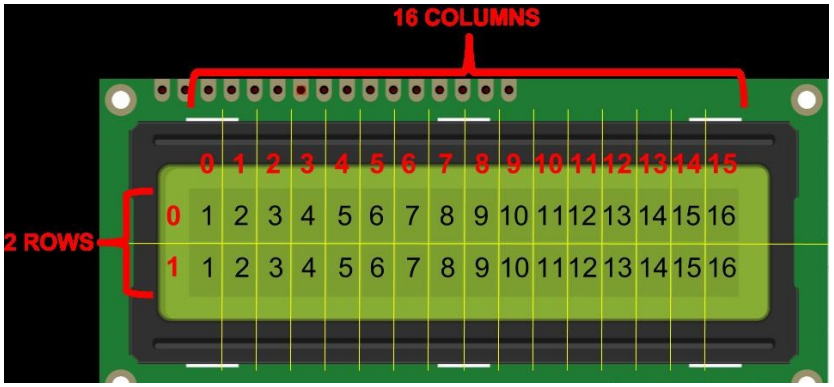
ii.Bluetooth Transmitter/Receiver Module(HC-05/HC-06

HC-05 is a Bluetooth module which is designed for wireless communication. It has range up to ¡100m which depends upon transmitter and receiver, atmosphere, geographic urban conditions. It is IEEE 802.15.1 standardized protocol. It uses serial communication to communicate with devices. It communicates with microcontroller using serial port (USART).



iii.16*2 LCD Display

LCD is an electronic display module which uses liquid crystal to produce a visible image. The 16×2 LCD display is a very basic module commonly used for basic display. The 16×2 translates to a display of 16 characters per line in 2 such lines. In this LCD each character is displayed in a 5×7 pixel matrix.



iV.4*4 Keypad Matrix

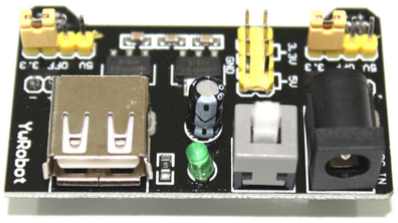
The 4*4 matrix keypad usually is used as input in a project. It has 16 keys in total, which means the same input values. The 4*4 Matrix Keypad Module is a matrix non- encoded keypad consisting of 16 keys in parallel. The keys of each row and column are connected through the pins outside – pin Y1-Y4 as labeled beside control the rows, when X1-X4, the columns.

V. Push button Switch and Resistors

We need one push button switch and two 10 kilo ohm resisters in our project.

Vi.3.3V Power Supply

This DC power supply can be easily plugged in on any breadboard. The output of this module can be switched from 5V DC to 3.3V DC using a 2 pin shunt jumper. One may provide power to this module from External DC adapter or USB A-B cable. Red LED indicates that power is supplied from external Power adapter. Green LED indicates that USB power is present on board.



Vii.Mini Breadboard

Anatomy of a Mini Breadboard. ... These tie points take the form of holes within the breadboard, into which wires and components can be pushed. They're useful for basic prototyping, but breadboards don't accommodate anything with two closely spaced rows of pins, such as the header on the Raspberry Pi.

Viii.Jumper Wires

A jumper wire is an electrical wire, or group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering.

ix.Proteus Simulator and Library for HC-05

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards.

Proteus Link - https://www.labcenter.com/downloads/

HC-05 Library Link - https://www.theengineeringprojects.com/2016/03/bluetooth-library-for-proteus.html

x.Keil uVision 4/5

The Keil μVision IDE combines project management, run-time environment, build facilities, source code editing, and program debugging in a single powerful environment. μVision is easy-to-use and accelerates your embedded software development. μVision supports multiple screens and allows you to create individual window layouts anywhere on the visual surface. Here we will use MDK-Arm IDE.

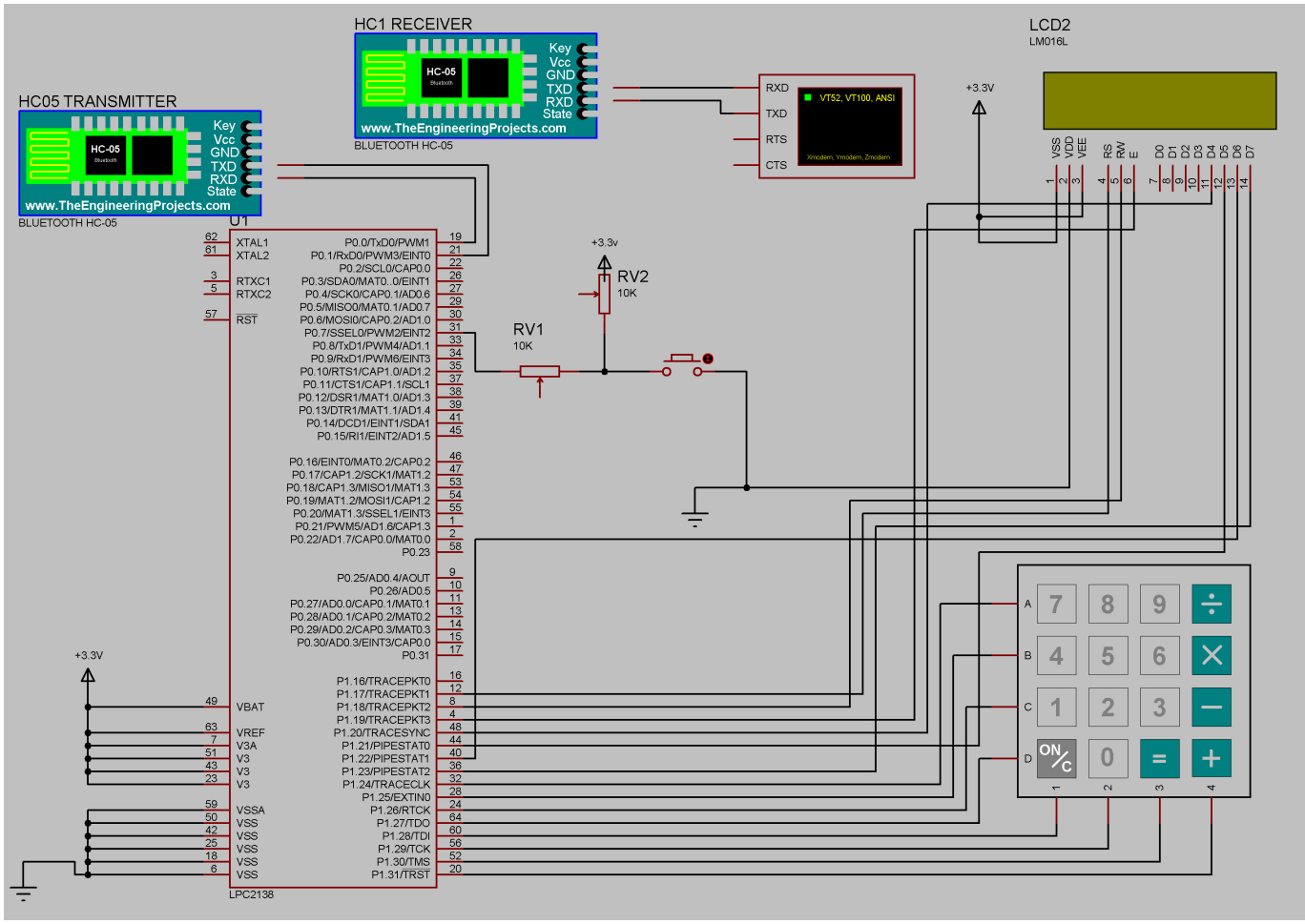
Link - https://www.keil.com/

xi.Virtual Serial Port Driver by Eltima Software

Virtual Serial Port Driver creates virtual COM ports and a virtual null modem cable for each pair of them (which connects two virtual serial ports to each other). Virtual ports fully emulate real ones, so that an application communicating with a virtual port never suspects the difference. Moreover, using virtual ports is often more convenient than using real ones since virtual COM port connection offers unmatched stability.

Link - https://www.virtual-serial-port.org/products/

6 CIRCUIT CONNECTION DIAGRAM



7 WORKING PROCESS

In this project, a digital clock is designed using LPC2148 microcontroller. LPC2148 has an inbuilt RTC. Real Time Clock (RTC) is a time counter that counts real time continuously. LPC2148 RTC can be clocked by a separate 32.768 KHz oscillator or by a programmable prescaler divider based on the APB clock. It maintains a calendar and clock and provides seconds, minutes, hours, month, year, day of week, day of month and day of year. It has power supply pin that can be connected to a battery or to the main 3.3V. It uses little power in power down mode. And most important, it has Alarm functionality. HC-05 is a Bluetooth device used for wireless communication. It works on serial communication (UART). When Alarm interrupt is triggered we send a message through UART to the HC-05 module, which then transmits the message in wireless medium, which is picked up by another Bluetooth receiver such as our smart phone or smart watch.

There is a button present in the designed clock. For editing the time or for editing the alarm time we can use this button. Pressing this button will create an interrupt and then we can edit the time or alarm.

8 CODE EXPLANATION

The coding has been done as a keil project. Open the file named which has extension ".uvproj" using keil microvision 4 or 5 IDE.

The main source file is "medicine.c". This contains the necessary macro definitions and header files.

Then we have Interrupt Service Routine for RTC interrupt and Keypad Interrupt. The RTC ISR simply handles the interrupt produced by both the time and alarm. The Keypad ISR handles the interrupt produced by the edit switch and make it possible to take input from keypad.

medicine.c | LCD Header.h | Keypad Header.h | Startup.s

```
21  // RTC Interrupt Service Routine
22  void ISR_RTC(void) __irq
23  {
24   char str[20];
25   if(ILR & 0x01)
26   {
27       sec = CTIME0 & MASKSEC;
28     min = (CTIME0 & MASKMIN) >> 8;
29     hour = (CTIME0 & MASKHOUR) >> 16;
30     ILR = 0x01;
31     LCD_CMD(0x85);
32     sprintf(str,"%d: %d: %d",hour,min,sec);
33     LCD_INIT();
34     LCD_STRING(str);
35
36   }
37   else if(ILR & 0x02)
38   {
39       LCD_INIT();
40     LCD_STRING("Medicine Alarm");
41     UART0_Init();
42     UART0_Send_String("Medicine Alarm");
43     ILR = 0x02;
44   }
45   VICVectAddr = 0;
46  }
```

```
13  // KEYPAD Interrupt Service Routine
14  void ISR_KEYPAD(void) __irq
15  {
16   edit_function();
17   EXTINT = 0x04;
18   VICVectAddr = 0x00;
19  }
```

Then we have initialization functions for both RTC and keypad.

```
// RTC INITIALIZATION
void RTC_Init()
{
   ILR = 0x01;
   CCR = 0x02;
   CIIR = 0x07;
   PREINT = (int)(fpclk/32768)-1;
   PREFRAC = (int)(fpclk-((PREINT+1)*32768));
   SEC = 5;
   MIN = 9;
   HOUR = 11;
   ALSEC = 11;
   ALMIN = 9;
   ALHOUR =  11;
   CCR = 0x01;
   VICIntSelect &= (~(1<<13));
   VICVectCntl1 = 0x20 | 13;
   VICVectAddr1 = (long)ISR_RTC;
   VICIntEnable |= (1<<13);
}
```

medicine.c | LCD Header.h | Keypad Header.h | Startup.s

```
69  // KEYPAD INTERRUPT INITIALIZATION
70  void keypad_interrupt_init()
71  {
72      PINSEL0 &=(~((1<<15)|(1<<14))); // P0.7
73      PINSEL0 |=(0x3)<<14;
74      EXTINT = 0x04;
75      EXTMODE = 0x04;
76      EXTPOLAR = 0x00;
77      VICIntSelect &= (~(1<<16));
78      VICVectCntl0 = 0x20 | 16;
79      VICVectAddr0 = (long)ISR_KEYPAD;
80      VICIntEnable |= (1<<16);
81  }
```

Then in the main function we initialize the RTC, LCD, Keypad and then enter into a infinite loop.

```
int main()
{
     PINSEL2 = 0x00000000;
   LCD_INIT();
   LCD_CMD(0x80);
   LCD_STRING("Time:");
   LCD_CMD(0xC0);
   LCD_STRING("Press EDIT");
   RTC_Init();
   keypad_interrupt_init();
   while(1)
   {

   }
   return 0;
}
```

LCD and UART are configured in their respective header file. Time and alarm editing is done by "edit function()" and that is defined in a separate header file named "edit header.h". Editing requires keypad input and that is done using "key detect()" function which is defined in a header file named "keypad header.h".

9 SIMULATION EXPLANATION

Complete simulation file can be downloaded from the github link provided in download section of the project.
Step 1.
Open the "Medicine Reminder.pdsprj" file with Proteus software.
Step 2.
Double click on the LPC2148 microcontroller. Select the program file as "medicineReminderToday.hex" which is present in the codes folder that you have downloaded from github. Click okay.
Step 3.
Then double click on the HC-05 transmitter and make the physical port as COM1. Click okay. Repeat the same process for HC-05 receiver ad make the physical port as COM2.
Step 4.
Open the Virtual Serial Port Driver software by eltima software and in the manage port section select COM1 and COM2 and click on add pair button.
Step 5.
Now click on the "run simulation" button present in the bottom left corner of Proteus software and the simulation will run as intended.

10 LINKS TO PROJECT RESOURCES

Github Repository Link -

https://github.com/sibaprasadchoudhury/digitalClock.git

11 FUTURE SCOPE

Future work will build upon the improvement of the current design. Time and Date display will be added and we can have multiple alarm with different message for each alarm. Also we can have a buzzer for alarm indication or even use a Bluetooth speaker for alarm sound. The project will be further updated in github if any improvements are made.

12 CONCLUSION

Embedded systems and Internet of things is booming in today's world. The world is more connected than it was ever before. This project serves as a gateway to the world of embedded systems and connected devices. The road ahead is certainly interesting and challenging.