Biometric Pulse Counting System using Microcontroller 8051

[1] Sayantan Dutta,[2] Kaushik Samanta,[3] Anusree Sarkar,[4] Sudeep Mondal,[5] Sumanta Mandal,[6] Tarun Karak

[1] - [6] Final Year Student, Department of Applied Electronics & Instrumentation Engineering,

University Institute of Technology, The University Of Burdwan, Burdwan, West Bengal, India

ABSTRACT: This paper gives a novel idea about the design and implementation of a microcontroller based pulse counter. Intel 8051 is used as the main microcontroller unit and the system has been fabricated to count the total number of pulse in a minute. IR sensors are connected as the input sensors which read the blood flow through the fingers. The microcontroller counts the number of input signals and thus generates an output on a display window build using three 7 segment IC. This system is highly efficient, accurate and economical and thus can be effectively used as an instrument to measure pulse on medical grounds.

Keywords: Microcontroller, pulse, sensors, IR, IC, medical

I. INTRODUCTION

In medical science, Pulse is defined as the regular beating of the heart, especially when it is felt at the wrist or side of the neck. Heart rate is the number of heartbeats per unit time. An IR pair is used to measure this pulse through the veins of a finger. The use of IR to sense heart rate was studied from earlier biomedical monitoring systems [1] − [5]. The microcontroller based embedded system has been created to be efficient and accurate. Thus, Intel 8051 used in this system carried out the decision making processes by using the input analog signals to count and process the value of heart beats per minute in the 7-segment display. Three 7- segment displays have been included in the output module of the system.
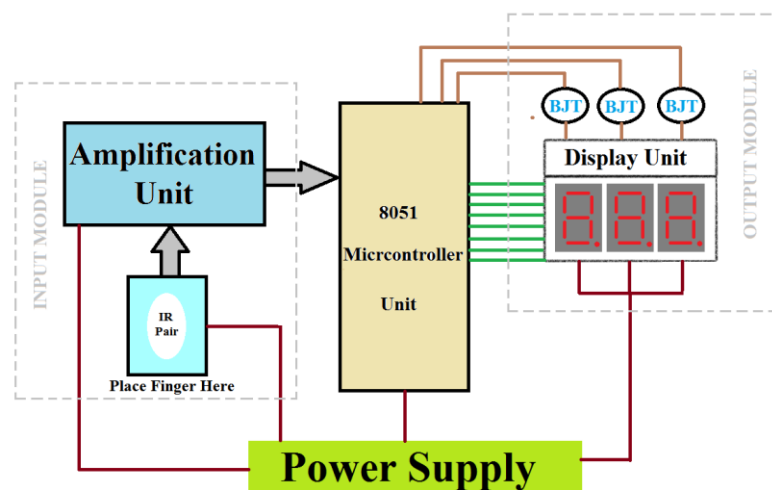

Fig 1. Block diagram of the system

Figure 1 shows the complete block diagram of the system. A beating heart increases the concentration of red blood cells each time it pumps. This change in concentration is detected and an average count is maintained. A low cost microcontroller makes the system simple and yet accurate to count the heart beat.

II. RELATED WORK

This system takes ideas from others which have been systemised earlier. It is an extension to a sensor system that has been given earlier [5]. But here the system is making use of a low cost microcontroller 8051 which makes the programming easy and cuts down the cost per device. The accuracy of the system is also high as external crystal provides the basis of timer count. This count is crucial for the increment counter that has been made to be used in the program.

III. SYSTEM DESIGN DESCRIPTION

The whole system has been designed to be used with a 9 V DC supply. The Complete system has been divided into three different parts. These are:

    A   The Input Module
    B   The Microcontroller Unit
    C   The Output Display Unit

A. The Input Module:

This consists of amplification circuits and IR pair. The two major components of this module are :
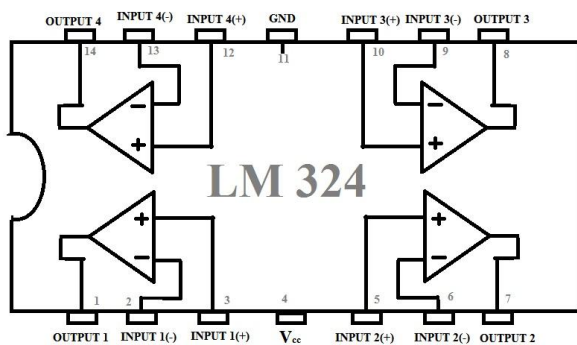
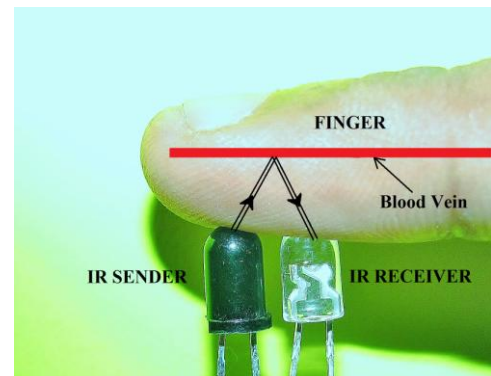

Fig 2a. LM 324 Quad Processor OP-AMP

Fig 2b. IR Pair with the Finger placed over it

- LM 324 : It is a Dual-in-Line Packaged Quad processor which consists of four independent, high gain, internally frequency compensated operational amplifiers which are specifically designed to operate on single power supply over a wide voltage range[6]. Three of them have been used to amplify the signals from the IR. Figure 2a is the internal circuit diagram of the IC.
- IR PAIR : An IR transmitter and receiver pair is used. Here the transmitter transmits the signal through the finger which is placed over it. Whenever there is an increase in concentration of red blood cells in the vein, the signals detect by bouncing back to the receiver IR. This way each time the blood is pumped throughout the body, a localised area of the human body will help to detect it. This signal is then send to the microcontroller unit. Figure 2b describes the working of the IR pair.

B. The Micrcontroller Unit:

An Intel 8051 microcontroller is used as the main decision making processor in the system. It is a low cost, Harvard architecture, CISC instruction set single chip Microcontroller [7]. Its architecture includes CPU, RAM, ROM, I/O, Interrupt logic, timer, etc in a single processor. It has a 8 bit accumulator and 8 bit ALU thus making it an 8 bit processor.

This unit also uses a crystal to generate clock frequency. This crystal has been specifically chosen to have 11.0593MHz of frequency and the timer and delay programs have been generated taking this into consideration.

C. The Output Display Unit:

This module consists of three different 7-segment display units. They are all common anode type which has been connected serially.A 7 segment display or a 7-segment indicator is a form of electrical device consisting of seven light emitting diodes arranged in a manner to display Arabic numerical. The glowing of each of these diodes can be controlled by the input which is generally provided to it by a microcontroller [8]. The switching mechanism of these components have been connected and controlled by the microcontroller. The source is used using a BJT and is then independently connected to each of the display units.

The 2N2222 is a NPN type transistor used along with the 7 segment display IC. It works at very high speed with low to medium amounts of power. This is basically used as a switching device that does not allow direct flow of voltage between microcontroller port and the display units.

IV. IMPLEMENTATION

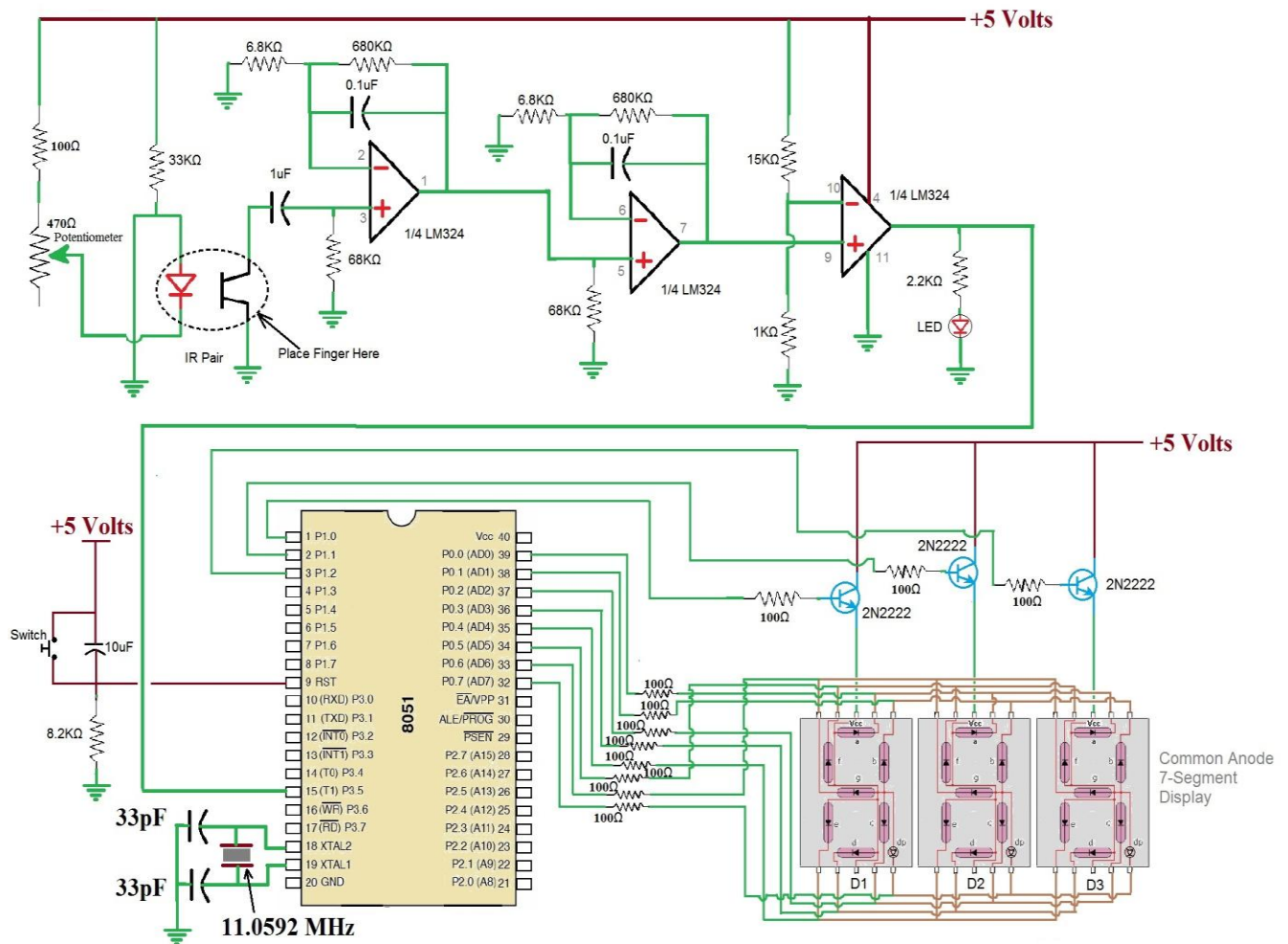The complete circuit diagram is shown in figure 3.

Fig 3. The complete system circuit diagram

For a normal person, the blood flows through the veins 72 times per minute. The input IR pair senses this and each time this is sensed a signal is given to the microcontroller which counts the signals received. A potentiometer has been used to calibrate the intensity and the efficiency of the IR in natural conditions. The microcontroller then sends this logical count to the display units which give then count of heart rate.

The Microcontroller programming has been firstly implemented in Kiel μvision 4. After the required debugging the hex codes have been transferred to a chip burner which assigns the assembly language program into the Intel 8051 used in the physical device. A window showing the program simulation and implementation is given in figure 4.

The microcontroller program has been developed to check the analog input that is received from the amplifier circuits. This signal is taken into the processor through port P3.5. Ports P1.0, P1.1 and P1.2 are used to turn ON and OFF each of the &-segment display independently. The output is defined by the ports P0.0 to P0.7 which are connected from a to h of the 7-segment display respectively. Binary codes of three digits are used by each of the ports to define each of the diodes of the display units. For example if port 0.0 has the value 010, then the second display unit D2 will have its 'a' diode switched ON [9]. This serially switches ON or OFF the respective diodes in the 7-segment display so as to show the correct value the output that has been calculated.

Fig 4. The Programming window showing the assembly language program implemented and simulated

The Assembly Language program code for this system is given in Appendix 1. The circuit has been fabricated on a single PCB board and a common 5 volt supply has been used. Figure 5 gives the real time image of the circuit that has been implemented.

Fig 5. The final fabricated circuit of the complete system

V. RESULTS

The result of this system is the output count which the microcontroller displays on the 7-segment display. An ideal graph showing the voltage vs time curve of a beating heart is shown in figure 6. In this a threshold has been defined which verifies the amount of voltage over which the IR senses obstacle and the microcontroller increments its count by one.

Fig 6. The Voltage vs Time graph obtained for 5 seconds

The count is incremented for a time period of 60 seconds. This count is then displayed by the three 7-segment displays which is nothing but the heart beat of the person.

VI. CONCLUSION

The system developed here is user friendly and efficient. The result and output is a simple count that depends on the IR sensors. However there are a few disadvantages of the system and its specifications. The system considers the environmental factors and other parameters while checking the sensor value. This error may be generated. This error can only be avoided if the calibration of the system has been done correctly in the environment where the system is being used. The IR sensors correctly can detect changes in the blood flow 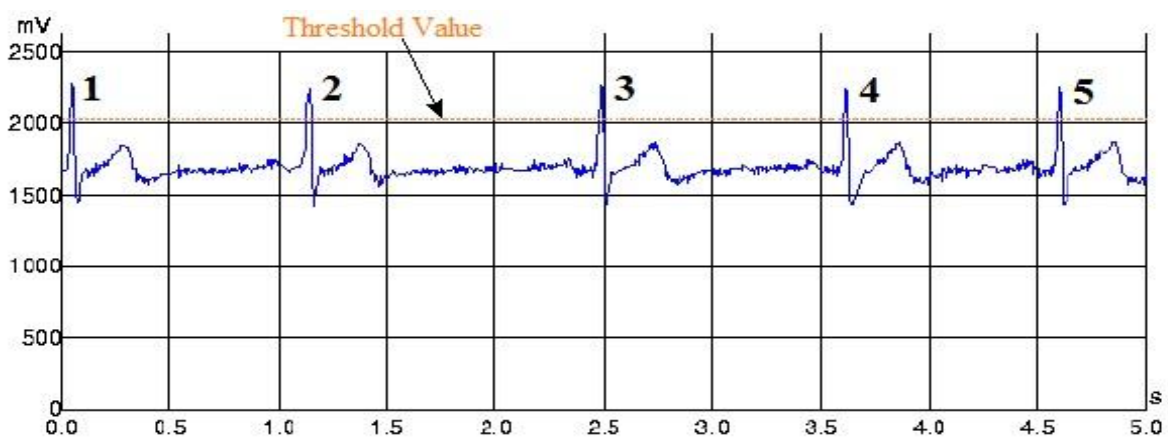thus making sure that the increment operator in the microcontroller program code work efficiently to produce the final result. It is also cost effective and easy to build. There is very little requirement of understanding the system before using it, thus making it easy, helpful and less time consuming.

Appendix 1

```
ORG 000H                          // origin
MOV DPTR,#LUT                     // moves starting address of LUT to DPTR
MOV P1,#00000000B                 // sets P1 as output port
MOV P0,#00000000B                 // sets P0 as output port
MAIN: MOV R6,#230D                // loads register R6 with 230D
  SETB P3.5                       // sets P3.5 as input port
  MOV TMOD,#01100001B             // Sets Timer1 as Mode2 counter & Timer0 as
                                  Mode1 timer

  MOV TL1,#00000000B              // loads TL1 with initial value
  MOV TH1,#00000000B              // loads TH1 with initial value
  SETB TR1                        // starts timer(counter) 1
BACK: MOV TH0,#00000000B          // loads initial value to TH0
  MOV TL0,#00000000B              // loads initial value to TL0
  SETB TR0                        // starts timer 0
HERE: JNB TF0,HERE                // checks for Timer 0 roll over
  CLR TR0                         // stops Timer0
  CLR TF0                         // clears Timer Flag 0
  DJNZ R6,BACK
  CLR TR1                         // stops Timer(counter)1
  CLR TF0                         // clears Timer Flag 0
  CLR TF1                         // clears Timer Flag 1
  ACALL DLOOP                     // Calls subroutine DLOOP for displaying the count
  SJMP MAIN                       // jumps back to the main loop
DLOOP: MOV R5,#252D
BACK1: MOV A,TL1                  // loads the current count to the accumulator
  MOV B,#4D                       // loads register B with 4D
  MUL AB                          // Multiplies the TL1 count with 4
  MOV B,#100D                     // loads register B with 100D
  DIV AB                          // isolates first digit of the count
  SETB P1.0                       // display driver transistor Q1 ON
  ACALL DISPLAY                   // converts 1st digit to 7seg pattern
  MOV P0,A                        // puts the pattern to port 0
  ACALL DELAY
  ACALL DELAY
  MOV A,B
  MOV B,#10D
  DIV AB                          // isolates the second digit of the count
  CLR P1.0                        // display driver transistor Q1 OFF
```

```
        SETB P1.1                               // display driver transistor Q2 ON
        ACALL DISPLAY                           // converts the 2nd digit to 7seg pattern
        MOV P0,A
        ACALL DELAY
        ACALL DELAY
        MOV A,B                                 // moves the last digit of the count to accumulator
        CLR P1.1                                // display driver transistor Q2 OFF
        SETB P1.2                               // display driver transistor Q3 ON
        ACALL DISPLAY                           // converts 3rd digit to 7seg pattern
        MOV P0,A                                // puts the pattern to port 0
        ACALL DELAY                             // calls 1ms delay
        ACALL DELAY
        CLR P1.2
        DJNZ R5,BACK1                           // repeats the subroutine DLOOP 100 times
        MOV P0,#11111111B
        RET
DELAY: MOV R7,#250D                             // 1ms delay
 DEL1: DJNZ R7,DEL1
        RET
DISPLAY: MOVC A,@A+DPTR                         // gets 7seg digit drive pattern for current value in A
        CPL A
        RET
LUT: DB 3FH                                     // LUT starts here
     DB 06H
     DB 5BH
     DB 4FH
     DB 66H
     DB 6DH
     DB 7DH
     DB 07H
     DB 7FH
     DB 6FH
END                                            // End of the program
```

REFERENCES

[1] R.S.Khandpur, "*Handbook of Biomedical Instrumentation*", 2nd edition, Tata-Magrow-Hill Publication,2003, New Delhi
[2] John Webster, "*Medical Instrumentation: Application and Design*", 3rd Edition, Wiley India Pvt. Ltd, July 2009
[3] Richard Aston, "*Principles of Biomedical Instrumentation and Measurement*", Merrill Pub Co. , 1990
[4] Ayang-ang C. and Sison L., "*Electrocardiograph Pre-Filtering, QRS Detection, and Palm Display Programming for Biomedical Applications,*"*in Proceedings of the ECE Conference*, University of St. Tomas, Manila, 2001.
[5] M Fezzali, M.B Sellah, M. Bedda, "*Heart rate monitoring", The International Arab Journal of Information Technology,* Vol. 5, No. 4, October 2008
[6] Datasheet LM324 Quad processor, Texas Instruments
[7] M.A Mazidi, J.G mazidi, R.D Mckinlay, "*The 8051 Microcontroller and its embedded systems*", 13th Edition, Pearson Publications, 2008 Dorlings Kindersley (India) Pvt. Ltd.
[8] 7-segment display, Wikipedia, http://en.m.wikipedia.org/wiki/Seven-segment_display
[9] S. Salivahnan, S. Aliverignan, "*Digital Circuits And Design*", 3rd Edition, Vikas Publications Pvt. Ltd., October 2011