Chapter 6

Development of Microcontroller based ECG system
6.1 Hardware requirements

Since the main aim is to develop a microcontroller based ECG system, hardware should be able to fulfill certain requirements:

i. The system should use a low cost, easily available general purpose controller with peripheral and external interrupt handling capabilities.

ii. The system should be able to transmit and receive ECG messages to and from any point to any point and hence has to use a GSM MODEM connected to the controller through a RS232 standard connector.

iii. The system should be able to handle data storage in EEPROM

iv. The system should be able to display relevant messages like ‘BUSY’, ‘CONNECTED TO ECG’, ‘DISCONNECTED’ etc., to give the status of the data being processed

6.2 System block diagram

The Microcontroller based ECG system is based on signal transmission and reception through GSM. The entire system can be visualized in the form of a block as shown in Figure 6.1.
The block mainly consists of a Microcontroller, 89C51, which is the heart of the system, a GSM MODEM to send and receive data through the SIM, a 16x2 LCD display to display the relevant messages, a 24C04 EEPROM memory to store the ECG data and RS232 and I2C connectors to connect different blocks.

6.2.1 The Microcontroller\textsuperscript{[1, 2]}:

The 8051 family of micro controllers is based on an architecture which is highly optimized for embedded control systems. It is used in a wide variety of applications from military equipment to automobiles to the keyboard on the PC. The 8051 family of microcontrollers is available in a wide array of variations from manufacturers such as Intel, Philips, and Siemens, with numerous features and peripherals.
such as I2C interfaces, analog to digital converters, watchdog timers, and pulse width modulated outputs.

The basic architecture consists of the following features:

- One 8051 processor cycle consists of twelve oscillator periods. Each of the twelve oscillator periods is used for a special function by the 8051 core.
- An 8-bit ALU
- 32 discrete I/O pins (4 groups of 8) which can be individually accessed two 16 bit timer/counters
- Full duplex UART
- 6 interrupt sources with 2 priority levels
- 128 bytes of on board RAM
- Separate 64K byte address spaces for DATA and CODE memory
- separate 64K byte address spaces for DATA and CODE memory
The Architecture of 8051:

Figure 6.2: The Architecture of the 8051 Microcontroller
Pin out diagram of 8051:

Figure 6.3: Pin out diagram of the 8051 Microcontroller
Memory Organization

The 8051 architecture provides the user with three physically distinct memory spaces. Each memory space consists of contiguous addresses from 0 to the maximum size, in bytes, of the memory space. Address overlaps are resolved by utilizing instructions which refer specifically to a given address space. The three memory spaces as shown in Figure 6.4, function as described below.

The CODE Space:

The first memory space is the CODE segment in which the executable program resides. This segment can be up to 64K (since it is addressed by 16 address lines). The processor treats this segment as

Figure 6.4: Memory organization of the 8051 Microcontroller
read only and will generate signals appropriate to access a memory device such as an EPROM. Many embedded systems use EEPROM which allows the memory to be overwritten either by the 8051 itself or by an external device. This makes upgrades to the product easy to do since new software can be downloaded into the EEPROM rather than having to disassemble it and install a new EPROM.

The DATA Space:

The second memory space is the 128 bytes of internal RAM on the 8051. This segment is typically referred to as the DATA segment. The RAM locations in this segment are accessed in one or two cycles depending on the instruction. This access time is much quicker than access to the XDATA segment because memory is addressed directly rather than via a memory pointer such as DPTR which must first be initialized. Variables stored in the DATA segment can also be accessed indirectly via R0 or R1. The register being used as the memory pointer must contain the address of the byte to be retrieved or altered. The DATA segment contains two smaller segments of interest. The first sub-segment consists of the four sets of register banks which compose the first 32 bytes of RAM. The 8051 can use any of these four groups of eight bytes as its default register bank. The selection of register banks is changeable at any time via the RS1 and the RS0 bits in the Processor Status Word (PSW). These two bits combine into a number from 0 to 3 (with RS1 being the most significant bit) which indicates the register bank to be used. The second sub-segment in the DATA space is a bit addressable segment
in which each bit can be individually accessed. This segment is referred to as the BDATA segment. The bit addressable segment consists of 16 bytes (128 bits) above the four register banks in memory. The 8051 contains several single bit instructions which are often very useful in control applications and aid in replacing external combinatorial logic with software in the 8051 thus reducing parts count on the target system. It should be noted that these 16 bytes can also be accessed on a "byte-wide" basis in the DATA space.

Control registers for the interrupt system and the peripherals on the 8051 are contained in internal RAM at locations 80 hex and above. These registers are referred to as special function registers (or SFRs). Many of them are bit addressable. The bits in the bit addressable SFRs can either be accessed by name, index or bit address. Thus, the EA bit of the Interrupt Enable SFR can be referred to as EA, IE.7, or 0AFH. The SFRs control actions such as the function of the timer/counters, the UART, and the interrupt sources as well as their priorities. These registers are accessed by the same set of instructions as the bytes and bits in the DATA segment. A memory map of the SFRs indicating the registers which are bit addressable is shown in Figure 6.5.
Figure 6.5: Special Function Registers of the 8051 Microcontroller

**The IDATA Space:**

Certain 8051 family members such as the 8052 contain an additional 128 bytes of internal RAM which reside at RAM locations 80 Hex and above. This segment of RAM is typically referred to as the IDATA segment. Because the IDATA addresses and the SFR addresses overlap, address conflicts between IDATA RAM and the SFRs are resolved by the type of memory access being performed, since the IDATA segment can only be accessed via indirect addressing modes.
**The XDATA Space:**

The final 8051 memory space is 64K in length and is addressed by the same 16 address lines as the CODE segment. This space is typically referred to as the external data memory space (or the XDATA segment). This segment usually consists of some sort of RAM (usually an SRAM) and the I/O devices or external peripherals to which the 8051 must interface via its bus. Read or write operations to this segment take a minimum of two processor cycles and are performed using DPTR, R0, or R1.

**Processor Status:**

Processor status is kept in a bit addressable SFR called PSW (Processor Status Word). This register contains the carry bit, an auxiliary carry bit which is used with BCD operations, the Accumulator parity flag and overflow flag, two general purpose flags, and two bits which select the register bank to use as the default. As mentioned before, the register bank selection bits make a two bit number from 0 to 3 which indicate the bank to be used. Bank 0 begins at the base of the DATA segment (address 00H); bank 1 begins at address 08H, bank 2 at address 10H and bank 3 at address 18H.
Processor Status Word (PSW) - Bit Addressable

<table>
<thead>
<tr>
<th>CY</th>
<th>AC</th>
<th>F0</th>
<th>RS1</th>
<th>RS0</th>
<th>OV</th>
<th>USR</th>
<th>P</th>
</tr>
</thead>
</table>

CY: Carry Flag  
AC: Auxiliary Carry Flag  
F0: General Purpose Flag  
RS1: Register Bank Selector 1. MSB of selector  
RS0: Register Bank Selector 0. LSB of selector  
OV: Overflow Flag  
USR: User Definable Flag  
P: Accumulator Parity Flag

Interrupts on the 8051:

The basic 8051 supports six interrupt sources: two external interrupts, two timer/counter interrupts, and a serial byte in/out interrupt. These interrupt sources force the processor to vector to one of five locations in the lowest part of the CODE address space (serial input and serial output interrupts share the same vector). The interrupt service routine must either reside there or be branched to from there.

The 8015 supports two interrupt priority levels: low and high. The nature of the interrupt mechanism is very standard and thus, a low level interrupt service routine can only be interrupted by a high level interrupt and a high level interrupt service routine cannot be interrupted. Each interrupt source can be individually set to one of two priority levels by altering the value of the IP (Interrupt Priority) SFR. If an interrupt source's corresponding bit in this register is set, it will
have high priority. Similarly, if the corresponding bit is cleared the interrupt will be of low priority and subject to being interrupted by any high priority interrupts.

<table>
<thead>
<tr>
<th>Interrupt Source</th>
<th>Vector Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power On Reset</td>
<td>0000H</td>
</tr>
<tr>
<td>External Interrupt 0</td>
<td>0003H</td>
</tr>
<tr>
<td>Timer 0 Overflow</td>
<td>000BH</td>
</tr>
<tr>
<td>External Interrupt 1</td>
<td>0013H</td>
</tr>
<tr>
<td>Timer 1 Overflow</td>
<td>001BH</td>
</tr>
<tr>
<td>Serial Receive / Transmit</td>
<td>0023H</td>
</tr>
<tr>
<td>Timer 2 Overflow</td>
<td>002BH</td>
</tr>
</tbody>
</table>

*Table 6.1: Interrupt sources and vector address*

**On-Board UART**

The 8051 features an on board, full duplex UART which is under software control. The UART is configured via the SCON (Serial CONtrol) SFR. The SCON register allows the user to select the UART mode, enable reception, and check UART status.

**Serial Control Register (SCON) - Bit Addressable**

<table>
<thead>
<tr>
<th>SM0</th>
<th>SM1</th>
<th>SM2</th>
<th>REN</th>
<th>TB8</th>
<th>RB8</th>
<th>TI</th>
<th>RI</th>
</tr>
</thead>
</table>

- SM0  Serial Port Mode Specifier 0, MSB
- SM1  Serial Port Mode Specifier 1, LSB.
- SM2  Multiprocessor Mode enable, in mode 0, this bit should be 0. In mode 1, if SM2=1, RI will not be set unless a valid stop
bit was received. In modes 2 and 3 if SM2=1, RI will not be set unless the ninth data bit is 1.

- **REN** Receive Enable Flag. Must be 1 to allow UART to receive data
- **TB8** The ninth data bit that will be sent in mode 2 and 3.
- **RB8** In mode 0 this bit is unused. In mode 1 if SM2=0, RB8 is the stop bit that was received. In modes 2 and 3 RB8 is the ninth data bit that was received.
- **TI** Transmit interrupt flag. Must be cleared by software
- **RI** Receive interrupt flag. Must be cleared by software

### 6.2.2 I2C

I2C is a form of inter-device communication (inter-integrated circuit) interface created and popularized by Phillips. I2C is a serial format data link which uses two wires (one for data and one for clock). Each device has its own ID on the link to which it will respond, data transfers are bi-directional, and the bus can have more than one master. Two I/O pins are taken from port 1 for the I2C interface and a set of SFRs are added to control the I2C and aid in implementing the protocol of this interface.

I2C [3] bus is used mainly with single-chip micro controller based systems that require general-purpose circuits like EEPROM, RAM, real time clock, LCD & audio/video tuning circuits. A key advantage of this is that only two lines can connect multiple devices. All devices have built-in addresses.
In circuit with multiple devices, one device (usually the micro controller) takes the role of master. Another device (only one of the multiple devices at any one time) acts as a slave device. Master device takes control of SCL; i.e. SCL is set low or high under the control of master (usually micro controller). Slave device accepts the data from micro controller (e.g. writing into memory) or sends the data to micro controller (reading from memory) under the control of master device.

Four different conditions exist in I2C bus transfers. These are Start, Stop, Bit transfer (read/write) and Acknowledge.

6.2.3 EEPROM Memory 24C04 - Two-wire Serial EEPROM

**Features**

- Low-voltage and Standard-voltage Operation
  - 2.7 (VCC = 2.7V to 5.5V)
  - 1.8 (VCC = 1.8V to 5.5V)
- Internally Organized 128 x 8 (1K), 256 x 8 (2K), 512 x 8 (4K), 1024 x 8 (8K) or 2048 x 8 (16K)
- Two-wire Serial Interface
- Schmitt Trigger, Filtered Inputs for Noise Suppression
- Bidirectional Data Transfer Protocol
- 100 kHz (1.8V) and 400 kHz (2.7V, 5V) Compatibility
- Write Protect Pin for Hardware Data Protection
- 8-byte Page (1K, 2K), 16-byte Page (4K, 8K, 16K) Write Modes
- Partial Page Writes Allowed
- Self-timed Write Cycle (5 ms max)
• High-reliability
  – Endurance: 1 Million Write Cycles
  – Data Retention: 100 Years
• Automotive Devices Available
• 8-lead JEDEC PDIP, 8-lead JEDEC SOIC, 8-lead Ultra-Thin Mini-MAP (MLP 2x3), 5-lead SOT23, 8-lead TSSOP and 8-ball dBGA2 Packages
• Die Sales: Wafer Form, Waffle Pack and Bumped Wafers

Description

The AT24C04 provides serial electrically erasable and programmable read-only memory (EEPROM) organized as 512 words of 8 bits each. The device is optimized for use in many industrial and commercial applications where low-power and low-voltage operation are essential.

Pin Configuration

<table>
<thead>
<tr>
<th>Pin Name</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td>A0 - A2</td>
<td>Address Inputs</td>
</tr>
<tr>
<td>SDA</td>
<td>Serial Data</td>
</tr>
<tr>
<td>SCL</td>
<td>Serial Clock Input</td>
</tr>
<tr>
<td>WP</td>
<td>Write Protect</td>
</tr>
<tr>
<td>NC</td>
<td>No Connect</td>
</tr>
<tr>
<td>GND</td>
<td>Ground</td>
</tr>
<tr>
<td>VCC</td>
<td>Power Supply</td>
</tr>
</tbody>
</table>

Table 6.2: Pin name and their functions of EEPROM Memory 24C04
NORMAL DATA BIT WRITE/READ:

During transfer of data from master (micro controller) to slave device (e.g. EEPROM), SDA is set to logic’0’ or logic’1’ only when SCL is low. After a small delay, SCL is pulsed high to clock in data. During read operation, SDA is an input line and its logic state is clocked in with SCL going high. Master (microcontroller) can then read the level of SDA.

START CONDITION:

Start is a special condition where SDA changes its state from high to low when SCL is high. Both SCL and SDA are controlled by master (microcontroller).

STOP CONDITION:

Stop is also a special condition where SDA goes from low to high when SCL is high. Both SCL & SDA are controlled by master (microcontroller).

ACKNOWLEDGE:

After transmitting eight data bits from micro controller to the device (e.g. EEPROM), direction of SDA line is reversed. One more clock pulse is given by micro controller. During this period, the slave device drives SDA to low. This indicates the acceptance of data by receiving device (e.g. EEPROM). When data is read from slave device (e.g. EEPROM), after reading eight bits, direction of SDA is
reversed. SDA is driven low (to send acknowledge) or high (to send no-acknowledge) and then SCL is pulsed

How to write a byte in EEPROM:

- Action is started with start condition generated by master (microcontroller).
- 1010 0000 (A0h) is transmitted as address for EEPROM.
- EEPROM responds with acknowledge during next clock pulse
- 0101 1000 (58h) is transmitted as byte address with EEPROM
- Again EEPROM responds with acknowledge during next clock pulse.
- Finally data byte 0011 0000 (30h) is transmitted.
- EEPROM acknowledges during next clock pulse.
- At the end, master (microcontroller) generates stop condition.

6.2.4 LCD Display Unit

LCD display is a single-line, 16-character unit. This is a standard unit available in market. Interface with micro controller is accomplished via four data lines D7-D4 and two control lines RS & E. Using these six lines, microcontroller displays all messages. Some LCD modules come with additional one or two pins. These extra pins are used for backlighting. There is no fixed standard for the additional pins. LCD controller is a flexible controller and can be used with 8-bit or 4-bit micro controller. To avoid problems, extra delays in software
are provided after every write command so that before writing another command/data, LCD module should be ready (not busy). Further, only four data lines (D4-D7) have been used while the other four data lines (D0-D3) are left disconnected. Thus even though we are using an 8-bit micro controller, the LCD module has been interfaced for 4-bit mode. Again, to save pin count, RS line is shared with SDA (serial data) line for memory (IC3) since at any given moment microcontroller will either interface with the LCD module or the memory, and this does not affect the system operation

**LIQUID CRYSTAL DISPLAY (LCD):**

Liquid Crystal Displays have materials, which combine the properties of both liquids and crystals. LCD consists of two glass panels, with the Liquid Crystal material sandwiched in between them. The inner surfaces of the glass plates are coated with transparent electrodes, which define the character, symbols or patterns to be displayed.

The LCDs are lightweight with only a few millimeters thickness. Since the LCDs consume less power, they are compatible with low power electronic circuits and can be powered for long durations.

The LCDs don’t generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. The LCDs have long life and wide operating temperature range. A brighter display can be obtained by providing backlighting.

LCD has single line display, Two-line display, four line display
Single line display

Two-line display

The Controller has two 8-bit registers. Instruction Registers (IR) and Data Registers (DR). The IR stores the instruction codes and address instruction for Display Data RAM (DD RAM) and Character Generator RAM (CG RAM). The DR temporarily stores data to be written to/read from DD RAM or CG RAM. When an address code is written to IR, the data (of the specified address) is automatically transferred from the DD RAM or CG RAM to the DR

Display Data RAM (DD RAM)

The characters to be displayed are written into the Display Data RAM (DDRAM) in the form of 8-bit character codes.

Character Generator ROM (CG ROM)

The character generator ROM generates 5x8 dot or 5x10 dot character patterns from 8-bit character codes.

Character Generator RAM (CG RAM)
In the character generator RAM, the user can rewrite character patterns by program.

6.2.5 GSM MODEM SIMCOM [5]

GSM modem is a specialized type of modem which accepts a SIM card, and operates over a subscription to a mobile operator, just like a mobile phone. When a GSM modem is connected to a computer, this allows the computer to use the GSM modem to communicate over the mobile network. While these GSM modems are most frequently used to provide mobile internet connectivity, many of them can also be used for sending and receiving SMS and MMS messages.

A GSM modem can be a dedicated modem device with a serial, USB or Bluetooth connection, or it can be a mobile phone that provides GSM modem capabilities.

A GSM modem exposes an interface that allows applications to send and receive messages over the modem interface. The mobile operator charges for this message sending and receiving as if it was performed directly on a mobile phone. To perform these tasks, a GSM modem must support an “extended AT command set” for sending / receiving SMS messages, as defined in the ETSI GSM 07.05 and 3GPP TS 27.005 specifications.

GSM modems can be a quick and efficient way to get started with SMS, because a special subscription to an SMS service provider is not required. In most parts of the world, GSM modems are a cost effective solution for receiving SMS messages, because the sender is paying for the message delivery.
The SIM300 is a tri band GSM / GPRS solution in a compact plug-in module. Featuring an industry – standard interface, the SIM300 delivers GSM / GPRS 900/1800/1900MHz performance for voice, SMS, Data and Fax with low power consumption.

The modem support interfaces like:

- Interface to external SIM 3V/1.8V
- 60 pins board – to – board connector
- Two analog audio interfaces
- Keypad interface
- LCD interface
- RTC interface
- Serial interface and debug interface
- Antenna connector and antenna pad

6.2.6 RS 232

One of the advantages of a serial system is that it lends itself to transmission over telephone lines. The serial digital data can be converted by modem, placed onto a standard voice-grade telephone line, and converted back to serial digital data at the receiving end of the line by another modem. Officially, RS-232 is defined as the “Interface between data terminal equipment and data communications equipment using serial binary data exchange.” This definition defines data terminal equipment (DTE) as the computer, while data communications equipment (DCE) is the modem. A modem cable has pin-to-pin connections, and is designed to connect a DTE device to a DCE device.
6.3 Hardware development

Figure 6.6 (a): Circuit diagram of the GSM ECG system doctor side
Figure 6.6 (b): Circuit diagram of the GSM ECG system patient side
ECG system Doctor Side:

The GSM SIMCOM 300 modem is connected through a female connector to the Receive data pin (RXD), and transmit data pin (TXD) of the 8051 microcontroller via RS232 driver. The received data or the data to be transmitted is stored in the EEPROM memory chip, IC24C04, connected through I2C interface. The messages generated from the microcontroller are displayed on a 16x2 LCD display. A buzzer is driven from the P1.1 pin of the microcontroller when data is received. An 11.09MHz crystal oscillator is connected to the X1 and X2 pins of the microcontroller to generate external clock pulses. The entire circuitry is fed by an IC 7805 fixed positive 5V voltage regulator [6-9].

ECG system and amplifier Patient Side:

At the patient side, a preamplifier to amplify signals from the developed sensors, filters to eliminate noise, a low power audio ECG amplifier to provide displayable ECG signals is designed and developed. The design and development of the amplifier system is discussed in chapter 5.

Another GSM ECG system is developed for the patient side as described above. The output of the designed amplifier system is fed to the GSM ECG system to be able to transmit the achieved ECG data.
Description of the GSM ECG System:

The microcontroller is the heart of the system which controls all the devices on the board. The microcontroller initializes the power up delays for the peripherals, initializes all the peripherals. It clears the message array, clears receive interrupt. The microcontroller checks if the Receive Interrupt (RI) bit in Serial Control Register (SCON) is high. The high status of the RI bit of SCON indicates the availability of data in SBUF SFR. After receiving the data, the microcontroller resets the RI bit of SCON, displays the message ‘Message Received’, after a small delay displays ‘Reading Message’ and then sends command to the modem to receive the message. The number of received message is checked with the database and if found invalid, a message ‘Invalid’ is displayed and then the message in the SIM is deleted after displaying ‘Message Deleting’. On finding a valid number, the controller generates a beep and displays the message ‘Calling Doctor’ and makes the data available at the output, to be connected to the CRO. If the data is not available, the microcontroller displays the message ‘No Signal’ on the LCD display.
References:


[3] I2C data sheets

[4] 24c04 datasheets

[5] GSM MODEM SIMCOM300 datasheets


