CHAPTER - 7

Software Development,
Results,
Discussions & Conclusion
7.1 INTRODUCTION

Remote monitoring and control of data [1] usually large in number present in a plant is conventionally done using structured cables running between the field devices and the control room. The control room makes use of custom made programs to perform the necessary man-machine-interface. Now-a-days new devices appear (Mobile phones supporting wireless application protocols) offering the possibility to be instantly and more efficiently informed the instantaneous changes occurring in the critical parameters of the plant. The present development of GSM based RMACS for industrial process parameters presents how cellular telephony technologies like the short message service (SMS) may be implemented effectively to assist the control and monitoring of a process plant. The techniques that can be used to provide visualization and control signals along with messages and alarms from the plant is explored in this work. This is particularly useful in large and distributed process such as chemical industry, oil industry etc.

The objective of the present work is to equip the plant decision makers with all the possible plant information as quick as possible in pocket-sized devices and also eliminate the need for wires. The work enables the technical and maintenance personnel supervision and control of machinery [2] and process from a single mobile phone. The process work can be implemented in real time process plant. Online data was successfully transferred to the mobile devices of the expert people at regular time intervals. As alarm management program is written to provide an alert signal as and when any received data exceeds as a set threshold value. The operational range of the plant is expected to improve significantly with these new emerging data transfer techniques.

RMACS can perform process parameters monitoring by means of LPC 2148 device control and device monitor by means of the SMS service of GSM technology [3]
user mobile number alteration by password authentication. To achieve these entire functionality Embedded C language program is written in Keil μ vision 4 IDE.

RMACS is also provided with a graphic screen along with the usual text display on personal computers. The user is able to monitor the plant in a very user-friendly manner where details are represented in dynamic graphical interfaces to achieve these functionality a visual basic 6 software package is used and also password verification is incorporated.

7.2 SOFTWARE DEVELOPMENT

The software developed [4] for design and implementation of GSM based remote monitory and control system for industrial process parameters are

- Embedded C
- Visual basic

As mentioned earlier the main expectation of the overall system is providing remote monitoring and controlling capabilities for the user in the industries. Focusing on the user requirements, following capabilities have been provided for the user in the software.

**Monitoring:** This is the main feature of the system where extracted information is presented for the operator in near real-time. Monitoring has been divided into two sections.

1. Monitoring the data through mobile phone using Kiel software
2. Monitoring the data in PC using visual basic 6 software.

**Controlling capabilities:** Controlling capabilities are required to initialize the on/off conditions of four devices in the implemented system. The software for this is developed in embedded C.

**Communication with remote terminals:** The users should be able to communicate with the GSM MODEM using Kiel software installed in at the PC.

**User administration:** The users are divided into three user levels. Each user is given a user name and password where users have to pass the authentication procedure to get the access for the system. The user actions are tracking down while he/she is in the system. When the users want to monitor the status of the devices in the mobile phone
he/she has to enter the password along with the status command. For this the software is developed in Kiel. When users want to monitor the log data current data values and set point values of sensors and Graphical representation of parameters the software is developed in VB6.

After gathering all the user requirements the system is designed with the following features [5].

**High flexibility** – Users are allowed to do the changes of the system without changing the code.

**High user friendliness** – graphical interface and other features are made simple and easily understandable as much as possible.

**High reliability** – trust worthiness of the system is highly considered where all the information is recorded to prove the system reliability.

### 7.3 EMBEDDED C

In the present work the embedded C language [6] in Keil is used for the development of GSM based RMACS for industrial process parameters. The development tool used is Keil development tool [7]. This tool offers numerous features and advantages that help us to develop embedded applications quickly and successfully. They are easy to use and are guaranteed to help us to achieve the design goals in a timely manner. Keil development tools for the Microcontroller architecture support every level of software developer from the professional applications engineer to the student just learning about embedded software development [8]. The Keil development tools are designed to solve the complex problems facing embedded software developers. The integrated development environment used here is Keil μVision4 IDE.

#### 7.3.1 Keil μVision4 IDE

Keil μVision4 IDE is the framework with all necessary tools integrated and is available for a large number of ARM processors [9]. It ensures easy and consistent Project Management. A single project file stores source file names and saves configuration information for Compiler, Assembler, Linker, Debugger, Flash Loader, and other utilities. It has powerful simulation capabilities that give you serious benefits for rapid, reliable embedded software development. When we start a new μVision4 project, we select the target device from the Device Database and μVision4 automatically sets required tool options and customizes dialogs. μVision4 displays only
those options that are relevant to the selected device and prevents selection of incompatible directives.

The μVision4 IDE/Simulator/Debugger accelerates learning efforts by providing a single environment for editing, simulating, and testing target hardware. Most debugger and editor functions may be quickly accessed from the toolbar. The μVision4 editor offers many standard and advanced software editing features. Color syntax highlighting and text indentation are optimized for editing C source code. Most editor functions may be quickly accessed from the toolbar or the context menu. Program debugging is directly controlled in the editor. This natural environment allows you to quickly integrate and test source code changes. μVision4 for ARM provides an execution profiler that records execution statistics for each CPU instruction. For experienced users, μVision4 adds new features such as Source Outlining, Function Navigation, Editor Templates, Incremental Search, Configuration Wizard, Logic Analyzer, CAN and I²C Simulation, Flash Programming, and JTAG Debugging. μVision4 contains powerful features and interfaces to external tools which help to complete the projects on time.

Kiel’s μVision4 IDE toolsets provide a powerful, easy to use and easy to learn environment for developing embedded applications.

- They include the components you need to create, debug, and assemble your C/C++ source files, and incorporate simulation for microcontrollers and related peripherals.
- The RTX RTOS Kernel helps you to implement complex and time-critical software.

### 7.3.2 Software Development Cycle

The project development cycle of keil μVision4 is

1. Create a project, select the target device from the Device Data base, and configure the tool settings
2. Create your source files in C/C++ or Assembly
3. Build your application with the Project Manager
4. Debug and correct errors in source files, verify and optimize your application
5. Download your code to Flash ROM or SRAM and test the linked application
Each component shown in the Figure 7.1 is described in detail in the following section

![Figure 7.1: Keil μVision4 project development cycle](image)

- **μVision4 IDE**: The μVision4 IDE [10] is a window-based software development platform combining a robust editor, Project Manager, and Make Utility tool. μVision4 supports all the Keil tools including C/C++ Compiler, Macro Assembler, Linker, Library Manager, and Object-HEX Converter.

- **μVision4 Device Database**: The μVision Device Database offers a convenient way to select and configure the device and project parameters. It includes preconfigured settings, so that we can fully concentrate on our application requirements. In addition, we can add our own devices, or change existing settings. The features of the Device Database are:
  a. Initialize the start up code and device settings
  b. Load the configuration options for the assembler, compiler, and linker
  c. can add and change microcontroller configuration settings

- **μVision4 Debugger**: The μVision Debugger is completely integrated into the μVision4 IDE. It provides the following features:
  a. Disassembly of the code on C/C++ source- or assembly-level with program execution in various stepping modes and various view modes, like assembler, text, or mixed mode
  b. Multiple breakpoint options including access and complex breakpoints
  c. Bookmarks to quickly find and define your critical spots
  d. Review and modify memory, variable, and register values
  e. List the program call tree including stack variables
  f. Review the status of on-chip microcontroller peripherals
g. Debugging commands or C-like scripting functions
h. Execution Profiling to record and display the time consumed, as well as the cycles needed for each instruction
i. Code Coverage statistics for safety-critical application testing
j. Various analyzing tools to view statistics, record values of variables and peripheral I/O signals, and to display them on a time axis
k. Instruction Trace capabilities to view the history of executed instructions

■ Assembler: An assembler allows you to write programs using microcontroller instructions. It is used where utmost speed, small code size, and exact hardware control is essential. The Keil Assemblers translate symbolic assembler language mnemonics into executable machine code while supporting source-level symbolic debugging. In addition, they offer powerful capabilities like macro processing. The assembler translates assembly source files into re-locatable object modules and can optionally create listing files with symbol table and cross-reference details. Complete line number, symbol, and type information is written to the generated object files. This information enables the debugger to display the program variables exactly. Line numbers are used for source-level debugging with the μVision Debugger or other third-party debugging tools.

■ C/C++ Compiler: The ARM C/C++ compiler is designed to generate fast and compact code for the ARM7, ARM9 and Cortex-Mx processor architectures; while the Keil ANSI C compilers target the 8051, C166, XE166, and XC2000 architectures. They can generate object code that matches the efficiency and speed of assembly programming. Using a high-level language like C/C++ offers many advantages over assembly language programming:

a. Knowledge of the processor instruction set is not required. Rudimentary knowledge of the microcontroller architecture is desirable, but not necessary.

b. Details, like register allocation, addressing of the various memory types, and addressing data types, are managed by the compiler

c. Programs receive a formal structure and can be split into distinct functions. This contributes to source code reusability as well as a better application structure.

d. Keywords and operational functions that resemble the human thought process may be used
e. Software development time and debugging time are significantly reduced

f. Can use the standard routines from the run-time library such as formatted output, numeric conversions, and floating-point arithmetic

g. Through modular programming techniques, existing program components can be integrated easily into new programs

h. The C/C++ language is portable, enjoys wide and popular support, and is easily obtained for most systems. Existing program code can be adapted quickly and as needed to other processors.

- **Object-HEX Converter:** The object-hex converter creates Intel HEX files from absolute object modules that have been created by the linker. Intel HEX files are ASCII files containing a hexadecimal representation of your application program. They are loaded easily into a device program for writing to ROM, EPROM, FLASH, or other programmable memory. Intel HEX files can be manipulated easily to include checksum or CRC data.

- **Linker/Locator:** The linker/locator combines object modules into a single, executable program. It resolves external and public references and assigns absolute addresses to relocatable program segments. The linker includes the appropriate run-time library modules automatically and processes the object modules created by the Compiler and Assembler. We can invoke the linker from the command line or from within the μVision4 IDE. To accommodate most applications, the default linker directives have been chosen carefully and need no additional options. However, it is easy to specify additional custom settings for any application.

- **Library Manager:** The library manager creates and maintains libraries of object modules. Library files provide a convenient way to combine and reference a large number of modules that may be used by the linker. The linker includes libraries to resolve external variables and functions used in applications. Modules from libraries are extracted and added to programs only if required. Modules, containing routines that are not invoked by your program specifically, are not included in the final output. Object modules extracted by the linker from a library are processed exactly like other object modules. There are a number of advantages to using libraries: security, speed, and minimized
disk space are only a few. Libraries provide a vehicle for distributing large numbers of functions and routines without distributing the original source code. For example, the ANSI C library is supplied as a set of library files.

μVision4 is a Windows application that encapsulates the Keil microcontroller development tools as well as several third-party utilities. μVision4 provides everything that need to start creating embedded programs quickly. μVision4 includes an advanced editor, project manager, and make utility, which work together to ease your development efforts, decreases the learning curve, and helps to get started with creating embedded applications quickly. The μVision4 IDE main screen is shown in Figure 7.2.

Figure 7.2: μVision4 main screen
❖ Creating a Project File

Creating a new µVision4 project requires just three steps:

1. Select the Project Folder and Project Filename
2. Select the Target Microcontroller
3. Copy the Startup Code to the Project Folder

❖ Selecting the Folder and Project Name

To create a new project file, select the Project – New Project - Menu. This opens a standard dialog that prompts you for the new project file name. It is good practice to use a separate folder for each project. We may use the Create New Folder button in this dialog to create a new empty folder. Select the preferred folder and enter the file name for the new project as shown in Figure 7.3. µVision4 creates a new, empty project file with the specified name. The project contains a default target and file group name, which you can view on the Project Window.

Figure 7.3: Create New Project dialog box
Selecting the Target Microcontroller

After selecting the folder and decided upon a file name for the project, μVision4 asks you to choose a target microcontroller. This step is very important, since μVision customizes the tool settings, peripherals, and dialogs for that particular device. The Select Device 1, 2 dialog box lists all the devices as shown in Figure 7.4 from the μVision Device Database. We may invoke this screen through the Project – Select Device for Target… Menu in order to change target later.

Figure 7.4: Selecting the target microcontroller dialog box

Copying the Startup Code

All embedded programs require some kind of microcontroller initialization or startup code1, 2 that is dependent of the tool chain and hardware you will use. It is required to specify the starting configuration of your hardware. All Keil tools include chip-specific startup code for most of the devices listed in the Device Database. Copy
the startup code to project folder and modify it there only. μVision4 automatically displays a dialog to copy the startup code into your project folder. Answer this question with YES. μVision will copy the startup code to the project folder and adds the startup file to the project as shown in Figure 7.5. The startup code files are delivered with embedded comments used by the configuration wizard to provide you with a GUI interface for startup configuration.

![Copy startup code dialog box](image)

**Figure 7.5: Copy the startup code dialog box**

- **Using the Project Windows**

Once we have created a project file successfully, the Project Window shows the targets, groups, and files of project. By default, the target name is set to Target 1, while the group’s name is Source Group 1. The file containing the startup code is added to the source group as shown in Figure 7.6. Any file, the startup file included, may be moved to any other group you may define in future.
Creating Source Files

Select the File – New… Menu to create a new source file. This action opens an empty Editor Window to enter your source code as shown in Figure 7.7. Save the new source file using the File – Save Menu.
❖ Adding Source Files to the Project
After you have created and saved source file, add it to the project. Right-click a file group in the Project Window and select Add Files to Group from the Context Menu. Then, select the source file or source files to be added as shown in Figure 7.8.

Figure 7.8: Adding Source Files to the Project

❖ Using Targets, Groups, and Files
The µVision’s very flexible project management structure allows you to create more than one Target for the same project. A Target is a defined set of build options that assemble, compile, and link the included files in a specific way for a specific platform. Multiple file groups may be added to a target and multiple files may be attached to the same file group as shown in Figure 7.9.
Setting Target Options

Open the Options for Target dialog from the Build Toolbar or from the Project Menu as shown in Figure 7.10.
Setting Group and File Options

In μVision4, properties of objects and options can be set at the group level and on individual files. Use this powerful feature to set options for files and groups that need a configuration different from the default settings as shown in Figure 7.11. To do so, open the Project Window:

• Invoke the Context Menu of a file group and select Options for Group to specify the properties, compiler options, and assembler options for that file group
• Invoke the Context Menu of a source file and select Options for File to specify the properties, compiler, or assembler options for that file

Figure 7.11: Setting Group and File options dialog

Configuring the Startup Code

Keil tools include files with chip-specific startup code for most of the supported devices. Keil startup files contain assembler code with options you can adjust to your particular target system. Most startup files have embedded commands for the μVision4 Configuration Wizard, which provides an intuitive, graphical, and convenient interface to edit the startup code as shown in Figure 7.12.
Figure 7.12: Configuring the Startup Code dialog box

**Building the Project**

Several commands are available from the Build Toolbar or Project Menu to assemble, compile, and link the files of your project. Before any of these actions are executed, files are saved. Translate File – compiles or assembles the currently active source file Build Target – compiles and assembles those files that have changed, then links the project Rebuild – compiles and assembles all files, regardless whether they have changed or not, then links the project While assembling, compiling, and linking, μVision4 displays errors and warnings in the Build Output Window as shown in Figure 7.13. μVision4 displays the message 0 Error(s), 0 Warning(s) on successful completion of the build process. Though existing warnings do not prevent the program from running correctly, you should consider solving them to eliminate unwanted effects, such as time consumption, undesirable side effects, or any other actions not necessary for program.
Figure 7.13: Build output window dialog box

Creating a HEX File

Check the Create HEX File box under Options for Target — Output, and µVision4 will automatically create a HEX file during the build process as shown in Figure 7.14. Select the desired HEX format through the drop-down control to generate formatted HEX files, which are required on some Flash programming utilities.
7.4 VISUAL BASIC 6

VISUAL BASIC [11] is a high level programming language which evolved from the earlier DOS version called BASIC. BASIC means Beginners' All-purpose Symbolic Instruction Code. Visual Basic (VB) was developed from the BASIC programming language. In the 1970s, Microsoft started developing ROM-based interpreted BASIC for the early microprocessor-based computers. In 1982, Microsoft QuickBasic revolutionized Basic and was legitimized as a serious development language for MS-DOS environment. Later on, Microsoft Corporation created the enhanced version of BASIC called Visual Basic for Windows. It is a very easy programming language to learn. The code looks a lot like English Language. Different software companies produced different versions of BASIC, such as Microsoft QBASIC, QUICKBASIC, GWBASIC, IBM BASICA and so on. However, people prefer to use Microsoft Visual Basic today, as it is a well developed programming language and supporting resources are available everywhere. Now, there are many versions of VB exist in the market, the most popular one and still widely used by many VB programmers is none other than Visual Basic 6.

Visual Basic 6 is an ideal programming language for developing sophisticated professional applications [12] for Microsoft Windows. It makes use of Graphical User Interface for creating robust and powerful applications. The Graphical User Interface as the name suggests, uses illustrations for text, which enable users to interact with an application. This feature makes it easier to comprehend things in a quicker and easier way. Coding in GUI environment is quite a transition to traditional, linear programming methods where the user is guided through a linear path of execution and is limited to small set of operations. In GUI environment, the number of options open to the user is much greater, allowing more freedom to the user and developer. Features such as easier comprehension, user-friendliness, faster application development and many other aspects such as introduction to ActiveX technology and Internet features make Visual Basic an interesting tool to work with.

Visual Basic (VB) is an event-driven programming language. This is called because programming is done in a graphical environment unlike the previous version BASIC where programming is done in a text only environment and executed
sequentially in order to control the user interface. Visual Basic [13] enables the user to
design the user interface quickly by drawing and arranging the user elements. Due to
this spent time is saved for the repetitive task.

The important features of Visual Basic (VB) 6 are;

- Full set of objects to 'draw' the application
- Lots of icons and pictures for use
- Response to mouse and keyboard actions
- Clipboard and printer access
- Full array of mathematical, string handling, and graphics functions
- Can handle fixed and dynamic variable and control arrays
- Sequential and random access file support
- Useful debugger and error-handling facilities
- Powerful database access tools
- ActiveX support
- Package & Deployment Wizard makes distributing the applications simple

On start up of the project, Visual Basic 6.0 will display the following dialog box
as shown in Figure 7.15. We can choose to start a new project, open an existing project
or select a list of recently opened programs. A project is a collection of files that make
up your application. There are various types of applications [14] that we could create;
however, we shall concentrate on creating Standard EXE programs (EXE means
executable program). Now, click on the Standard EXE icon to go into the actual Visual
Basic 6 programming environment.

Figure 7.15: Visual Basic 6 Main Screen
In the present work the Visual Basic 6 is used for the development of log data, current data values and set point values of sensors and graphical representation of parameters in GSM based RMACS for industrial process parameters.

When the user starts the program, the window to identify the user is shown in Figure 7.16, where he/she enters his/her login and password.

![Figure 7.16: Log in and password form](image)

By input a password it prevents the access of an unauthorized person to the program, thus protecting the data of RMACS. The login for admin and user are shown in Figure 7.17 (a) & (b).

![Figure 7.17 (a): Log in for admin](image)  ![Figure 7.17 (b): Log in for user](image)

After validating the access, a main window with buttons related to different functions is shown in Figure 7.18.
The Remote Monitoring and Control System form consists of the following menus:

- **File Menu:** This menu is used to open, save and exit the file.
- **View Menu:** This menu is used to view the current data values, high limit values of sensors and device status as shown in Figure 7.19.

![Remote Monitoring and Control System Main Form](image)

**Figure 7.18: Remote Monitoring and Control System Main Form**

![Device status, Current data and High limit values of sensors](image)

**Figure 7.19: Device status, Current data and High limit values of sensors**
- **Statistics Menu:** This menu is used to show the statistics of measurement in graphical representation. It is shown in Figure 7.20.

![Figure 7.20: Graphical Representation of Temperature Measurement, Pressure Measurement, Humidity Measurement and Level Measurement](image)

- **LogData Menu:** This menu is used to show the data of temperature, pressure, humidity and liquid level sensors. This form also consists of current data and time as shown in Figure 7.21. Once we select the required date and click retrieve, the data will be present in the form of a graph in Microsoft Excel as shown in Figure 7.22. If the user wishes to print the graph select the file and click print option in MS-Excel.
Figure 7.21L: Log data for Industrial Process Parameters

![Log data for Industrial Process Parameters](image)

Figure 7.22 Graphical representation of data in MS-Excel

The back end used in the present work for Visual Basic 6 is Microsoft Access. The log data form in Microsoft Access is shown in Figure 7.23.
7.5 ALGORITHM FOR THE ENTIRE SYSTEM

The algorithm of the present implementation of Remote Monitoring and Control System of GSM based industrial process parameters are given below.

1. Make setup ready by connecting RMACS to sensors, PC and devices from respective slots/connectors.
2. Switch RMACS ON
3. Read the input data from sensors
4. Store the data in different variables
5. Compare all the sensor data with threshold limit
6. If there is any deviation alert the user(s) by SMS to his/her mobile phone through GSM Modem or System Switch ON the controlled devices
7. If the user wishes to change the status of the input values or set points or device control he/she has to add password for authentication along with the AT command in his/her mobile phone and send as SMS to GSM modem in the required format
8. Display the measured values on RMACS LCD, Personal Computer and Remote User(s) mobile phone
9. User can view the log data, status of the parameters, and graphs in Personal Computer for further analysis

7.6 FLOWCHART FOR THE ENTIRE SYSTEM

The flowchart of the present implementation of Remote Monitoring and Control System of GSM based industrial process parameters are shown in Figure 7.24.
Figure 7.24: Flowchart for the entire system
7.7 RESULTS & DISCUSSIONS

As discussed in chapter 2 in the development of RMACS all the process parameter like temperature pressure humidity and level are monitored and controlled remotely using mobile phone through GSM modem.

The main objective of the present research work is to develop an integrated prototype wireless remote monitoring and control system to replace manual control and supervision in the plant. The implementation of data of RMAS is communicated via SMS to concerned user(s) through GSM modem. The quality and improvement of RMACS depends main on the maintenance of security and authentication of information data with the help of hardware as well as software implementation which provides better information about the process parameter.

The author has developed an indigenous RMACS integrated wireless Remote Monitoring and Control System. This work is unique nature and has been worked out through instrumentation technology. In the previous chapters the complete details about design and development of instrumentation and interface with GSM network for the Remote Monitoring and control system are presented. This is a prototype module for remote monitoring and control of process parameters remotely through GSM using mobile phone. Considering the feature of a remote monitoring and control of the process parameters and the cost for communication GSM [15] short message service is suitable to use for transmitting data in large scale field measurement system.

In proposed work a low-cost high efficient RMACS was presented. The proposed device consists of a LPC2148 ARM TDMI-S processor SIM 300 GSM MODEM, Temperature Pressure Humidity and level sensors and a mobile phone. The monitoring information is sent to a remote mobile phone via SMS through GSM modem. In the present system we have used SIM 300 GSM module it is possible to send collected sensory information through this protocol resulting in future reduction in the maintenance cost. On the low power side the overall power of the device can be future reduced by using a low power microcontroller in addition to a low power GSM module.
The second part of this work is devoted to development of RMACS with a graphic screen along with the usual text display on Personal Computer. The details of the software of this system are discussed in previous sections. The author has developed exclusive software in visual basic to monitor the plant in a very user friendly manner where details are represented in dynamic graphical interface and reports.

The proposed system has been tested for remote monitoring and control of temperature, pressure, humidity level parameters and the results of the system are shown in Table 7.1. The individual parameter results are discussed in detail in previous chapters.

**Table 7.1: Results of RMACS**

<table>
<thead>
<tr>
<th>Process Parameters</th>
<th>RMACS action</th>
<th>Alert SMS received by User</th>
<th>SMS sent by User</th>
<th>RMACS Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>Temperature</td>
<td>Crossed set point, SMS generated to user</td>
<td>SMS received “Temperature crossed limit”</td>
<td>Switch ON Fan</td>
<td>Fan will be switched ON</td>
</tr>
<tr>
<td>Pressure</td>
<td>Measured value is less than set point, SMS not generated to user</td>
<td>No SMS received by user</td>
<td>No SMS sent by user</td>
<td>No action taken by system</td>
</tr>
<tr>
<td>Humidity</td>
<td>Measured value is near to set point, SMS generated to user</td>
<td>SMS received “Humidity crossing limit”</td>
<td>Switch ON Bulb</td>
<td>Bulb will be switched ON</td>
</tr>
<tr>
<td>Level</td>
<td>Measured value is equal to set point, SMS generated to user</td>
<td>SMS received “Liquid Level crossing limit”</td>
<td>Switch ON Buzzer</td>
<td>Buzzer will be switched ON</td>
</tr>
</tbody>
</table>

Remote monitoring and control of devices and retrieval of information relating present status of inputs using GSM have been successfully demonstrated through SMS based message transfer between user mobile and GSM modem. The results that are displayed on the mobile phone are shown in Figure 7.25.
If the remote user wants to set the system in auto mode i.e. RMACS will not wait for the user response via SMS and the action will be taken by the system. The command typed in the mobile phone is “SET SYS AUTO”. If the user wants to set the system in manual mode i.e RMACS will wait for user response by SMS further to take any action. The command typed in the mobile phone is “SET SYS MANUAL”. The message typed in the mobile phone is as shown in Figure 7.26.

![Figure 7.25: Status of Process Parameters in Remote user Mobile Phone](image)

Figure 7.25: Status of Process Parameters in Remote user Mobile Phone

If the remote user wants to monitor the status of the individual parameter for example temperature, he has to send the command STAT TEMP to GSM Modem as shown in Figure 7.27 (a) and the GSM modem will sends signal to the LPC2148

![Figure 7.26: AT Commands display in Mobile phone](image)

Figure 7.26: AT Commands display in Mobile phone

If the remote user wants to monitor the status of the individual parameter for example temperature, he has to send the command STAT TEMP to GSM Modem as shown in Figure 7.27 (a) and the GSM modem will sends signal to the LPC2148
processor through UART and is decoded in the processor and that message is sent to GSM modem via SMS to user mobile phone containing the status in the requested format as shown in Figure 7.27 (b).

![Figure 7.27 (a) & (b): Status Request and Display on Remote User Mobile Phone](image)

The RMACS is tested with different set points. If the measured value is near or more than the set limit then the processor will sends an alert SMS to remote user(s) concerned to the plant through GSM modem as shown in Figure 7.28.

![Figure 7.28: Alert Message Display on Remote User Mobile Phone](image)

The authority(s) concerned to the plant can control the set point by changing the input value or set point value by sending command as SET TEMP XXXX to GSM modem through his/her mobile phone where XXXX indicates the value as shown if
Figure 7.29 or can enable/disable the controlled devices in the plant by sending command as shown in Figure 7.30.

![Figure 7.29: Changing the parameter value on System by Remote Mobile Phone](image1)

![Figure 7.30: Enable/Disable the Controlled Devices by Remote Mobile Phone](image2)

The concerned authority can also monitor the measurement values in Personal computer in the form of text and graphical representation is shown in Figure 7.31.
The achieved analytical results from RMACS are

- System allowed the provision of security such that system took no action against the instructions received from un authorized number the required task was performed only when the pre-configured number instructed the system.

- The system makes use of the ubiquitous GSM technology by which it empowers the user to exhibit control over the device from the remote destinations. The user has the freedom to move about and still control over the device and can also get information about the device status by making a call to the GSM modem.

- System sent breach alert when the input value crosses the threshold limit was detected.
Remote controlling capability of the system allowed user to switch ON/OFF through simulating the controlled devices as directed by the incoming SMS.

The system automatically performed tests and checked support for available features and SMS sending and receiving capability and configured system accordingly.

Finally, the main aim and objective of the author is to develop and implement a more sophisticated Integrated Wireless Remote Monitoring and Control System using GSM to contribute to industrial automation.

7.8 CONCLUSION

Modern control systems for industrial applications require automated data reading on-line monitoring and control processing, so remote access is an important feature. One possible solution for remote access and control is GSM technology available almost everywhere and requires no cabling. Considering the feature of a remote monitoring and control of the industrial process parameters and the cost for communication GSM network is suitable to use for transmitting data in large-scale field measurement system.

The developed system accomplishes the objective of good performance, low-cost, secure, ubiquitously accessible, auto-configurable, remotely controlled solution for automation of process industries. The approach discussed in the proposed system is novel and has achieved the target to monitor and control process parameters remotely using the SMS-based system satisfying user needs and requirements.

The RMACS has provided a low cost custom built monitoring and controlling system specifically for process industries. Thus this system can be customized to suit any other industrial requirement related to monitoring and controlling provided industrial sensors are in use. Similarly the RMACS can be extended to a variety of other possible application areas so long as there is the need for him/her monitoring of a measurable parameter and possibly the necessity of extending some form of control that is achievable via electronic actuators [16]. Health care is such a sector that may benefit from the deployment of the RMACS system in the monitoring of patients. The system is cost effective as compared to the previously existing systems. Hence we can conclude that the require goals and objectives of RMACS have been achieved.
The RMACS can be further enhanced by

- As a small data logger which is compatible with a huge range of sensors and will power any sensor requiring up to 24 VDC supply.
- The hardware of the system will be self contained and can not be prone to electric failure. This system will have its own encapsulated UPS and charging system.
- The system has to be programmed remotely using any windows terminal emulator.
- The integration of a GSM modem connected to the RMACS, with the purpose to send a message in case of power failure or an equipment malfunction.
- This system can be empowered with computer vision where with the help of image processing pictures and videos of remote areas can be sent to the user via MMS.
- This system can also be carried out for the communication industry, automotive industries, home automation system, health care systems, defense etc as hardware requirement is same, the difference being extending some form of control that is achievable via electronic actuators.
- To develop an RMACS system as Universal Remote Monitoring and Control for industrial applications by incorporating different wireless communication medium, like Zigbee, Ethernet, GPS, RS232 etc, according to the user requirement to interact, communicate and control many devices at a single stretch.
REFERENCES


