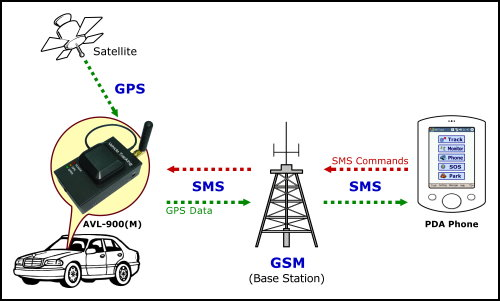
SMS BASED GENERATOR START/STOP SYSTEM



INTRODUCTION-:

It offers an integrated SMS alert and remote control system. This unit can be integrated into any diesel generator in the Bundu Power range and allows for the following items to be monitored and controlled.  
  
**1.** Up to 9 different alarm signals to be sent directly to your cell phone.  
e.g: "Diesel Low" or "Oil Pressure Low"  
  
**2.** The Generator can be Started or Stopped by sending an SMS to the generator from anywhere in the world. Used in conjunction with the alert you can decide when to turn on and off your generator even if you’re not at home or the office when the lights go out.  
  
The SMS unit is mounted inside the generator housing (space permitting) and runs off the generators battery. When the generator is in automatic mode it will start-up when the power fails and in doing so sends an SMS to UP TO 5 predetermined phone numbers. An SMS can then be sent back to the generator using a predetermined "code" and command. The code allows you to safe guard the generator from malicious SMS have to turn it on and off. If the generator has a problem when running it will again send an SMS to up to 5 users notifying them of the problem encountered. When the generator is shuts down the user is also notified, allowing them to confirm that the generator has successfully shutdown.  
  
The unique SMS Alert and command system allows users to quickly and easily control their Generator worldwide, with the knowledge that problems can be sorted out quickly and efficiently. In addition to that, the security system allows for safe and secure use without interference

# INTRODUCTION-:

The project is aimed at developing and testing the use of mobile phones to remotely control an appliance control system. The microcontroller would then control an device based on the information given to it. The proposed solution will need to be easy to use, simple, secure, robust and be useful on most mobile phones. To achieve this testing will need to be carried out to create a useful system.

The report consists of a background into the area of 8051 microcontroller and mobile communication, how they are interfaced to each other and AT (Attention) commands set used in communication.

## HARDWARE USED

1. AT command supporting GSM mobile phone.
2. 89S52 Microcontroller
3. Max 232 IC.
4. Relays
5. Relay driver IC ULN 2803
6. Voltage regulator 7805.
7. Diode IN4007
8. GSM Phone

# SOFTWARE USED

1. Keil u-Vision 3.0

Keil Software is used provide you with software development tools for 8051 based microcontrollers. With the Keil tools, you can generate embedded applications for virtually every 8051 derivative. The supported microcontrollers are listed in the µ-vision

1. **PRO51 Programmer Software**

## THEORY OF OPERATION

In this project we interfaced 8051 microcontroller with Motorola’s C168 GSM mobile phone to decode the received message and do the required action. The protocol used for the communication between the two is AT command.

The microcontroller pulls the SMS received by phone, decode it, recognizes the

Mobile no. and then switches on the relays attached to its port to control the appliances. After successful operation, controller sends back the acknowledgement to the user’s mobile through SMS.

**AT-Command set**

The following section describes the AT-Command set. The commands can be tried out by connecting a GSM modem to one of the PC’s COM ports. Type in the test-command, adding CR + LF (Carriage return + Line feed = \r\n) before executing. Table gives an overview of the implemented AT-Commands in this application. The use of the commands is described in the later sections.

AT-Command set overview

|  |  |
| --- | --- |
| **Command** | **Description** |
| AT | Check if serial interface and GSM modem is working. |
| ATE0 | Turn echo off, less traffic on serial line. |
| AT+CNMI | Display of new incoming SMS. |
| AT+CPMS | Selection of SMS memory. |
| AT+CMGF | SMS string format, how they are compressed. |
| AT+CMGR | Read new message from a given memory location. |
| AT+CMGS | Send message to a given recipient. |
| AT+CMGD | Delete message. |

A BRIEF INTRODUCTION TO 8051 MICROCONTROLLER:

When we have to learn about a new computer we have to familiarize about the machine capability we are using, and we can do it by studying the internal hardware design (devices architecture), and also to know about the size, number and the size of the registers.

         A microcontroller is a single chip that contains the processor (the CPU), non-volatile memory for the program (ROM or flash), volatile memory for input and output (RAM), a clock and an I/O control unit. Also called a "computer on a chip," billions of microcontroller units (MCUs) are embedded each year in a myriad of products from toys to appliances to automobiles. For example, a single vehicle can use 70 or more microcontrollers. The following picture describes a general block diagram of microcontroller.

**89s52:** The AT89S52 is a low-power, high-performance CMOS 8-bit microcontroller with 8K bytes of in-system programmable Flash memory. The device is manufactured using Atmel’s high-density nonvolatile memory technology and is compatible with the industry-standard 80C51 instruction set and pinout. The on-chip Flash allows the program memory to be reprogrammed in-system or by a conventional nonvolatile memory pro-grammer. By combining a versatile 8-bit CPU with in-system programmable Flash on a monolithic chip, the Atmel AT89S52 is a powerful microcontroller, which provides a highly flexible and cost-effective solution to many, embedded control applications. The AT89S52 provides the following standard features: 8K bytes of Flash, 256 bytes of RAM, 32 I/O lines, Watchdog timer, two data pointers, three 16-bit timer/counters, a six-vector two-level interrupt architecture, a full duplex serial port, on-chip oscillator, and clock circuitry. In addition, the AT89S52 is designed with static logic for operation down to zero frequency and supports two software selectable power saving modes. The Idle Mode stops the CPU while allowing the RAM, timer/counters, serial port, and interrupt system to continue functioning. The Power-down mode saves the RAM con-tents but freezes the oscillator, disabling all other chip functions until the next interrupt



The hardware is driven by a set of program instructions, or software. Once familiar with hardware and software, the user can then apply the microcontroller to the problems easily.

The pin diagram of the 8051 shows all of the input/output pins unique to microcontrollers:



The following are some of the capabilities of 8051 microcontroller.

* Internal ROM and RAM
* I/O ports with programmable pins
* Timers and counters
* Serial data communication

The 8051 architecture consists of these specific features:

* + - 16 bit PC &data pointer (DPTR)
    - 8 bit program status word (PSW)
    - 8 bit stack pointer (SP)
    - Internal ROM 4k
    - Internal RAM of 128 bytes.
    - 4 register banks, each containing 8 registers
    - 80 bits of general purpose data memory
    - 32 input/output pins arranged as four 8 bit ports: P0-P3
    - Two 16 bit timer/counters: T0-T1
    - Two external and three internal interrupt sources Oscillator and clock circuits.

THEORY-:

it basically works like this:

- System is controlled by SMS commands: start, stop, status, timer on, timer off.

- Requesting "status" it returns: Load in Amps (which tells me that the pump is working fine), running time, and whether timer is currently activated or not.  
- The generator starts up connected to a small load, about 0.4A for a ceiling fan, but is disconnected from the pump.  The startup sequence is:

1.  Give start signal to generator

2.  Check current sensor to see if output is more than 0.4A, if not shut down generator and send error message that generator failed to start.

3. Wait for 2 minutes

4. Close pump relay

5.  Check current sensor to make sure the pump came on, if not shut down generator and send error message.  
  
The shutdown sequence is in reverse and also uses the current sensor to make sure everything is off as it should be.

