GSM

Global System for Mobile Communication

GSM is a digital mobile telephony system GSM digitizes and compresses data, then sends it down a channel with two other streams of user data, each in its own time slot. It operates at either the 900 MHz or 1800 MHz frequency band. Interfacing GSM Fig.1 shows how to interface the GSM with microcontroller. The GSM is communicate the microcontroller with mobile phones through **UART**. To communicate over **UART** or **USART**, we just need three basic signals which are namely, RXD (receive), TXD (transmit), GND (common ground).

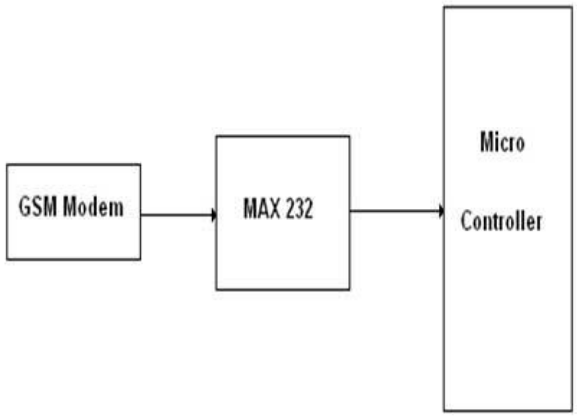

Fig. 1- Interfacing UART to Microcontroller

GSM modem interfacing with microcontroller for SMS control of industrial equipment. The sending SMS through GSM modem when interfaced with **microcontroller** or PC is much simpler as compared with sending SMS through **UART**.

For Connection to GSM module you should use **"AT Commands"** that we will discuss about it here:
At first some general rules about AT Commands:

- Every AT command needs to be terminated by \r\n (CR + LF) combination.
- Our Connection must be with UART or console command line of PC's.
- If we use the UART connection it is notable that don't forget to use the "\r\n" at the end of the commands.
- please note, send the commands in uppercase only.

Get Start to Connect to Webserver via GPRS by GSM Sim900

Now comes the interesting part. Given below is a list of AT commands that you must fire in the exact given sequence to connect to a webserver using TCP over GPRS. The symbol => indicates the command you need to enter, <= indicates the response received from the SIM900. Text within /** **/ are my comments for further explanation. The response may contain additional information than what is shown here. I have listed down only the important part of the response that you need to be aware of.

Here is the sequence:

/** First test if everything is okay **/

=> AT

<= AT

/** This should come back. SIM900 default is to echo back commands you enter **/

<= OK

/** This string should tell you all is well**/

=>AT+CPIN?

/**This is to check if SIM is unlocked. This sample assumes unlocked SIMs**/

<= +CPIN: READY

/** If your response contains this, then it means SIM is unlocked and ready**/

=>AT+CREG?

/**This checks if SIM is registered or not**/

<=+CREG: 0,1

/**This string in the response indicates SIM is registered**/

=>AT+CGATT?

/**Check if GPRS is attached or not**/

<=+CGATT: 1

/**A response containing this string indicates GPRS is attached**/

=>AT+CIPSHUT

/**Reset the IP session if any**/

<=SHUT OK

/**This string in the response represents all IP sessions shutdown. **/

=>AT+CIPSTATUS

/**Check if the IP stack is initialized**/

<=STATE: IP INITIAL

/**This string in the response indicates IP stack is initialized**/

=>AT+CIPMUX=0

/**To keep things simple, I'm setting up a single connection mode**/

<=OK

/**This string indicates single connection mode set successfully at SIM 900**/

=>AT+CSTT= "APN", "UNAME", "PWD"
/**Start the task, based on the SIM card you are using, you need to know the APN, username and password for your service provider**/

<= OK

/**This response indicates task started successfully**/

=> AT+CIICR

/**Now bring up the wireless. Please note, the response to this might take some time**/

<=OK

/**This text in response string indicates wireless is up**/

=>AT+CIFSR

/**Get the local IP address. Some people say that this step is not required, but if I do not issue this, it was not working for my case. So I made this mandatory, no harm.**/

<= xxx.xxx.xxx.xxx /**If previous command is successful, you should see an IP address in the response**/

=>AT+CIPSTART= "TCP" , "www.google.com", "80"

/**Start the connection, TCP, domain name, port**/

<= CONNECT OK

/**This string in the response indicates TCP connection established**/

=>AT+CIPSEND

/**Request initiation of data sending (the request)**/

<= >

/**The response should be the string “>” to indicate, type your data to send**/

=> xxxxxx

/**Just type anything for now**/

=>#026

/**Now type the sequence #026. This tells the terminal.exe to send the hex code 0x1a (which is Ctrl+Z) to indicate end of data sending**/

<= xxxxxxxxxx

/**You should get some response back from the server…it would generally be a complain that the request string was not valid…but that is a different subject…you have established the connection**/

/**To close the connection**/

=>AT+CIPSHUT

/**Request shutting down of the current connections**/

<=SHUT OK

/**Indicates shutdown successful**/

Get Start to Send and Receive SMS by GSM Sim900

----------Sending SMS----------
/** First test if everything is okay **/

=> AT

<= AT

/** This should come back. SIM900 default is to echo back commands you enter **/

<= OK

=> AT+CMGF=1

/**AT command to set SIM900 to SMS mode**/

=> AT+CMGS=\"+ZZxxxxxxxxxx\"

/** change ZZ with country code and xxxxxxxxxxx with phone number to sms **/

=>Hello , this is a sample text

/** at the end you should write ctrl+z OR ASCII number of that “26” **/

----------Receiving SMS----------

----------Sending SMS----------
/** First test if everything is okay **/

=> AT

<= AT

/** This should come back. SIM900 default is to echo back commands you enter **/

<= OK

=> AT+CMGF=1

/**AT command to set SIM900 to SMS mode**/

=> AT+CNMI=1,2,0,0,0

/** Decides how newly arrived SMS messages should be handled **/

<= +CMT : “+ 00000000”,”TEXT”

Source:

www.lastminuteengineers.com
www.randomnerdtutorials.com
www.os.mbed.com
www.pantechsolutions.net

Shahed University summer-2019
By: Mohammad Hossein Movasaghinia