GSM BASED FLOOD WARNING SYSTEM USING FPGA

Wan Mohd Azim Bin Wan Mohd Zubir
Centre of Electronic Engineering Study
Faculty of Electrical Electronics Engineering
University Technology Mara
Shah Alam Selangor
wazimwzubir@gmail.com

***Abstract*— Flood is one of the frequent natural disasters occur each year. It can occur due to heavy rain, dam outbreak or overflow water in the river. In other words, rivers can be one of the biggest causes of flood. By monitoring the condition of water level in river or drain, flood can be anticipated early and warnings can be delivered to public in order to prevent loss of property and life. GSM based Flood Monitoring and Warning System (FMWS) is proposed here to convey the condition of river or give warning to people at affected area. Any data received from water level and water flow rate meter is processed and delivered to main system before SMS notifications are sent to public. Currently most of GSM based applications are developed using microcontroller such as PIC or 8051 microcontroller. With the flexibility in designing the hardware, FPGA is chosen to be the heart of the system. An interface unit, UART, between GSM modem and FMWS controller is modeled using Quartus II software and implemented on the Altera DE2 Board.**

Keywords- flood ,GSM, water level,water flow rate sensor, SMS autorespond

I. INTRODUCTION

Flood disaster can be predicted and countermeasure can be taken if monitoring and warning system are implemented at any rivers or drain that cause the flood. A flood monitoring system in real-time in [1] using 300 KHz H-ADCP (Horizontal Acoustic Doppler Current Profiler) is implemented at China. The system measures two horizontal dimension of speed across 200m Yangtze river in China, and calculates the level of water using index-velocity method. The data is received by hydrological information centre in Huangling Temple and it is transmitted from the regional river Yangtze. It uses LAN, satellite and VHF for transmitting and receiving data. Study of the flood warning system in [2] also applied to the Ping river, Chaing Mai province, Thailand, where the floods hit every so often. The system provides five channels data logger: two automatic raingage, two TDR and an air temperature. These data are transmitted to the main server Upper North Water Hydrological And Administration Center (UNHWAC) using Digital Radio signal Packages (DRP) with the interval time set by SCADA system (supervisory control and data acquisition) [10]. Besides based on geographical and hydrological data, wireless sensors and video surveillance system [4-5] are also used to forecast flooding. Complex algorithms to incorporate the effects of noise and winds are also provided in [6] and the system requires significant complex models and excessive processing power. Thus a less complex, more robust and less expensive monitoring system is demanded.

Flood Monitoring and Warning System with GSM capabilities is proposed which the system is composed of water flow rate and level sensors. They can be placed in the river or drain and above the river depending on type of sensors used. The sensors are used to detect the changes in water level and speed. A system with GSM capabilities with give a reminder via SMS to field staff (client), so immediate action can be taken to avoid the worst case occurred. This will also be the fastest way in providing information or alarming public of any possible flash floods disaster. Hence, a flood detection system that offers low cost, low power consumption, ease of installation and reliability has to be developed in order to prevent loss of property and life.

Since FPGA is known for its flexibility and robust, the proposed system is developed using FPGA and in this paper, the interfacing part between GSM modem and FPGA is studied and modeled using Verilog code. When GSM SMS system is designed with PIC or specific processor, the interfacing part or its configuration is not similar with other controllers. Thus, a controller developed using FPGA which enables the interfacing between GSM module and FPGA board has to be developed.

This paper proposes a FPGA-based system that can send SMS (short message service) to field staff (client) or residents if water level rises rapidly. In order to develop such system, a few goals have to be set:

- Design an interface module i.e. RS232 which will send AT commands to GSM.
- Transmit data via SMS such as status of river : NORMAL, ALERT and DANGER

II. METHODOLOGY

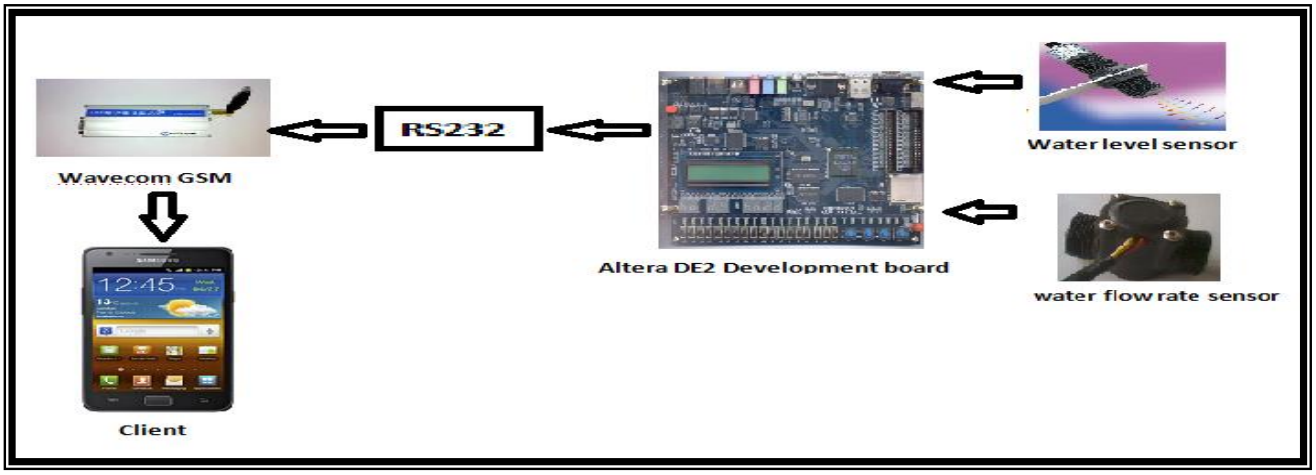

Figure 1: System hardware architecture

The architecture of the system design consists of three units as shown in Figure 1, the controller FPGA ALTERA DE2 development board , the Wavecom GSM modem, and the interface circuit that include the water level and water flow rate sensors. The main function of the Wavecom GSM modem is as the medium of the communication between the controller and the user through RS232 which is serial communication standard protocol. Meanwhile, the main function of the controller FPGA ALTERA DE2 development board is as the main processing unit which continuously process the measurement taken from the water level and the water flow rate sensor and compares them with threshold level before sending message through the GSM network in case of high water level or water flow rate. The main module in the system is the GSM controller which is modeled using Verilog code and synthesized by Altera Quartus II 9.1 software. The GSM controller consists of clock divider, synchronous FIFO, ROM memory and UART transmitter.

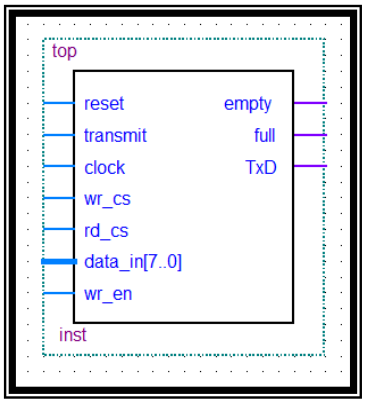

Figure 2: Top Module of GSM controller

The top module of GSM controller which controls the message to be fetched from the memory and sent to Wavecom GSM modem before being processed as SMS is shown in Figure 2. There are seven inputs being inserted to this top module. One of the most important input data is the 8 bit *data_in*. This input data consists of the AT command that are set in ROM. The second most important input data is *transmit*. The transmit signal is always high because to activate the transmission process. Data in transmit is coming from the sensors. The third important input would be the *clock* to drive the function of this module design. Output should be sent to Wavecom GSM is coming from the data output *Tx*. *TxD* has to be connected to Rx of Wavecom GSM through RS232 converter, which converts TTL voltage level to RS232 voltage level in case of sending data to GSM MODEM.

Figure 3 shows the Register Transfer Level (RTL) schematic generated from the Verilog code. The Verilog design program has many components and function. The most important components are the clock divider(A) , synchronous FIFO (B) and UART transmitter (C). The function of clock divider is to slow down the timing period. This is because synchronous FIFO needs to read data in ROM memory and the system clock provided by Altera DE2 board is 50MHz and the reading requires several cycles to perform the function. The ROM is used store the AT commands which is the most important component in the system. The function of synchronous FIFO is to generate the AT commands and send to UART transmitter module. The *data_out* coming from synchronous FIFO converts the processed data from parallel to serial. Then after the data become serial, it'll send the data to Wavecom GSM modem. The transmitted data is an AT command and send SMS to the user.

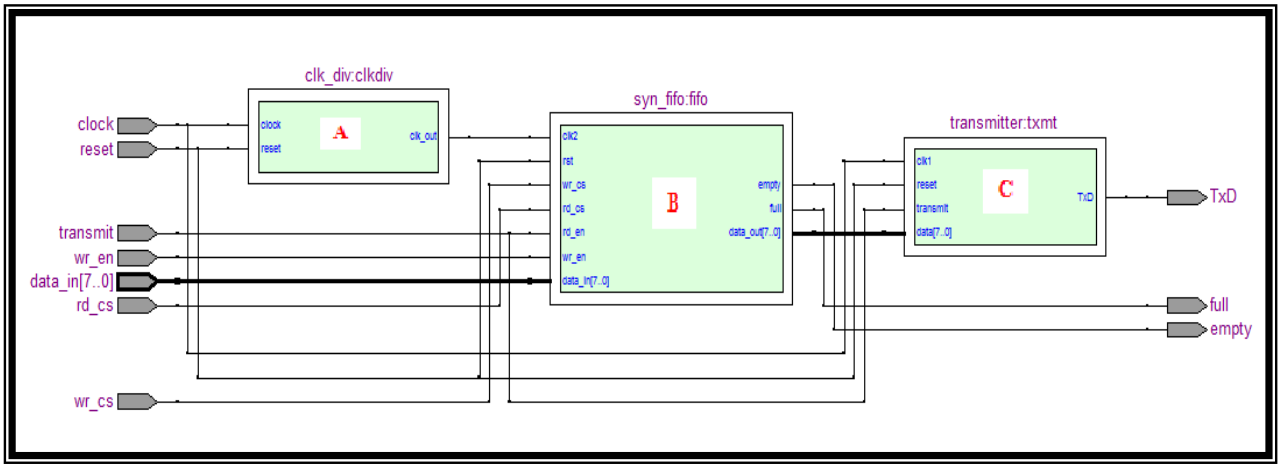

Figure 3: Block diagram of GSM controller

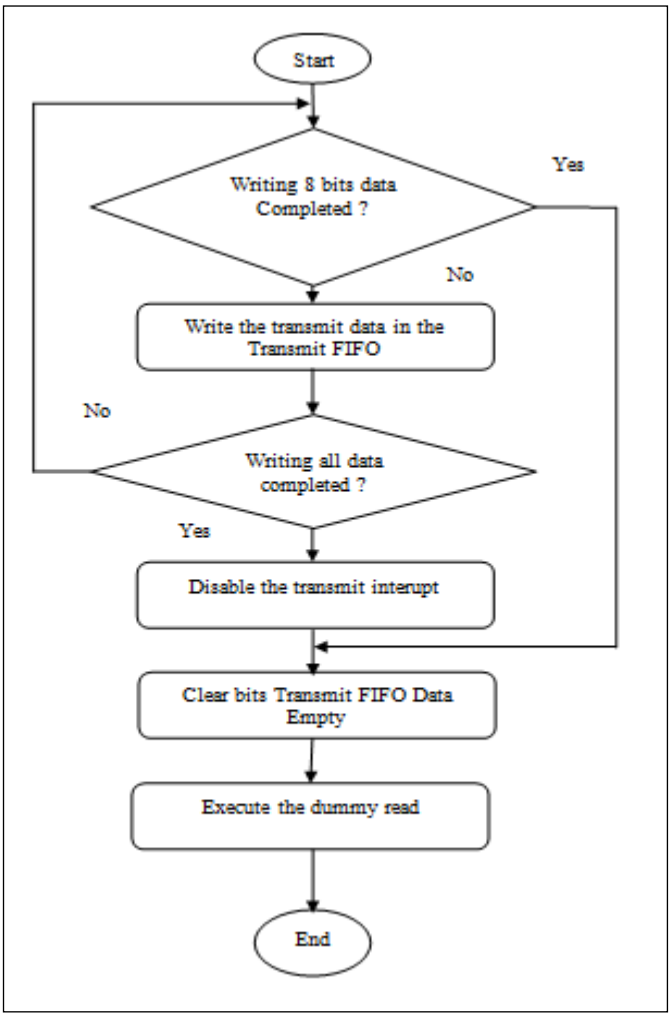

Figure 4: Flow chart of synchronous FIFO process

Figure 4 shows the flow chart of the synchronous FIFO process. The function of this synchronous FIFO is to read from the stored data memory. The stored data in memory have 8 bits data which contains of AT commands. If all the 8 bits data completed, it'll clear the bit transmit FIFO Data Empty. If no, it'll will write the transmit data in the transmit FIFO. After all data completed, the program will disable the transmit interrupt which is *Transmit-FIFO-data-empty* and *Receive-FIFO-data-full.* The program will execute the dummy read to make sure to clear the interrupt source. Then, program will end the process and transmit the data to UART transmitter.

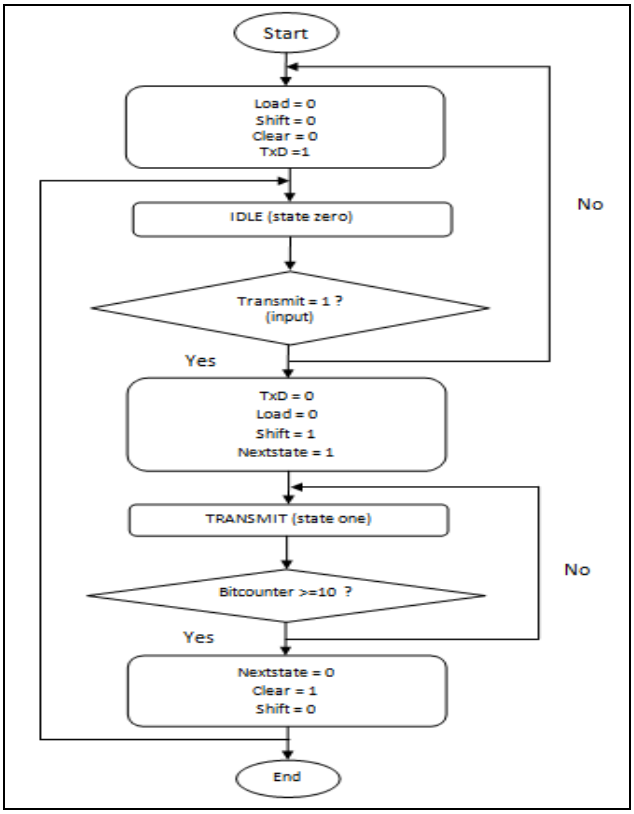

Figure 5: Flowchart of UART Transmitter operation

Figure 5 shows the flow chart of UART transmitter operation. The UART transmitter operation is described here. The *Load, shift* and *clear* signals are set to zero (LOW) while output *TxD* is set to one (HIGH) because the program is idling (not transmiting). If *transmit* input is one (HIGH), the program knows that the transmission is about to begin. In this condition, output *TxD* and *load* are remained zero (LOW) while *shift* and *nextstate* signals are remained one (HIGH). Meanwhile, *start* bit is always equal to zero while *end* bit is always equal to one (HIGH) because when transmission is ready to end , the output *TxD* goes to HIGH again and the transmission will stop. From that moment, the data is in serial mode which it is converted from parallel to serial. In serial protocol, the 8 bits transmission will be added by 3 other bits like *start, parity* and *stop* bit.

In TRANSMIT, when the *bit-counter* is equal or greater to 10 or reaches 11 bits of transmission, it means that it has completed its task of transmitting. Shift will be set to zero (LOW) until then. Clear must go to one (HIGH) to clear the *bit-counter* back to zero and get ready for the next transmission. Next-state is set to zero (LOW) because when transmission is over it needs to go back to IDLE state so that it can wait and transmit data again. After all the data has been transmitted, it will end the program. If the *bit-counter* is not greater or equal to 10 (11 bits) , the program are still in TRANSMIT mode.

Before the system is implemented on the board, the system has been experimentally tested using Hyper terminal with RS232 interfacing serial port of the PC through Wavecom GSM modem. With the arrangement as follows:

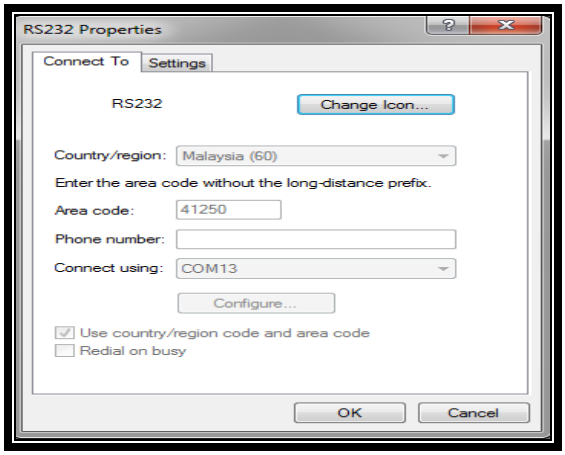

Figure 6: Order COM to communicate

Figure 6 shows the configuration in connecting the Wavecom GSM Modem in Hyperterminal using port COM13.

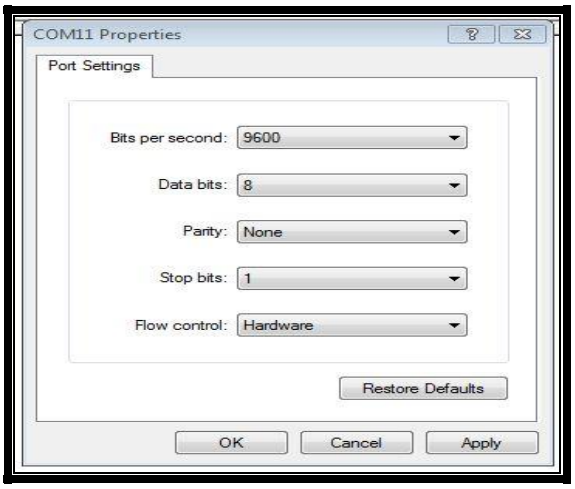

Figure 7: Setting Baud rate, Data Bits, Parity, Stop Bits, Flow Control

Figure 7 shows the setting needed in RS232 serial communication link with COM13 with baud rate 9600 bps, data bits 8 bits, parity None, Stop Bits 1 bit, Flow Control None. Figure 8 shows the received SMS from PC.

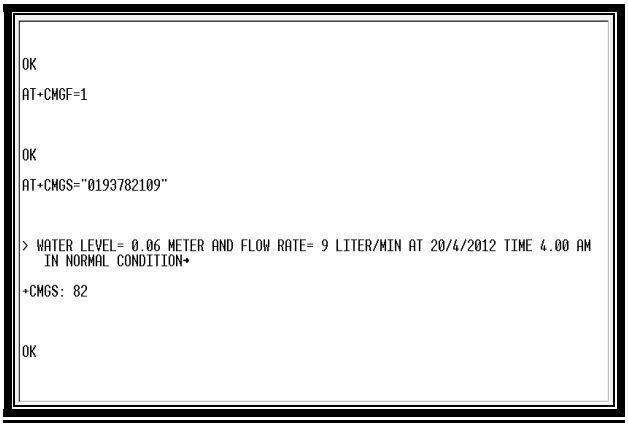

Figure 8 : Set of AT command to be sent as SMS

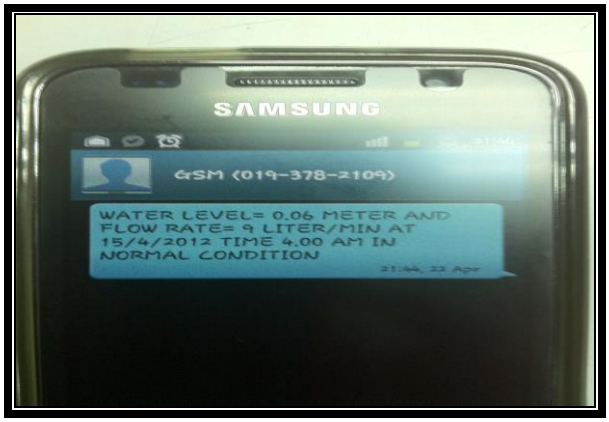

Figure 9 : SMS received from the system

III. RESULT AND DISCUSSION

Figure 10 and Figure 11 show the simulation results for the GSM control unit and the UART transmitter. Figure 10 shows the simulation for the control unit. From control unit, it sends parallel data which represents the ASCII code for the characters consist either a message or an AT commands to UART transmitter. The example given in this figure 10 is sending the AT commands to initialize the GSM and setting message as a text mode "AT+CMGF=1". In the simulation, it shows the transmitted data in the form of characters but when doing implementation in hardware FPGA Altera DE2 development board, it is a stream of bits. Example 8'b01000001 for ASCII code is character A.

Figure 11, the UART Transmitter results show that the data in frame of 11-bits including the start and stop bits, also sending the frame to the serial output TxD , where bit-counter is a counter for the number of bits to be sent.

In UART transmitter, when sending message it can only send a 8-bit binary number in each clock cycle, so it is necessary to make the data join the queue and wait to be sent. Every time when it finishes sending data, there will be a sign signal idle. After sign signal idle, the transmitter begins to transmit at this state. In UART communication protocol, at the output TxD, the line is going to high to show that there is no data to be transmitted. When the line goes to low, the transmitter knows that transmission is about to begin. That is why the start bit is always equal to zero or low. The transmission is going to stop when the line goes to high again. At this time, the communication is ready to end. This is why the end bit is always equal to one or high. When the idle is low, the next data is allowed to be sent.

In this project, the packet data setup with 11 bits which are start bit,8 data bits, 1 parity bit and a stop bit. Transmit will make sure that the bit-counter is not over than 11 bits. The transmitting process has to be completed its task when bit-counter reaches 11 bits. To make for another new transmission, the clear must go to high condition, because it wants to clear the bit-counter back to zero and waiting for another new data bit transmission.

Beside that, shift also must put to high because that will shift the shift-register. So, the next bit can be shifted to TxD output pin. The transmitter will transmit the data again when the next state go back to IDLE condition. The state which is bit-counter (0-10) stays in TRANSMIT until it is equal to 10 and goes back to IDLE. When IDLE, it will waiting for new transmission.

In UART protocol, transmission begins from the least significant bit (LSB) as bits are shifted; one by one is sent until the most significant bit (MSB). Beside that, transmitting process needs to consider the baud rate transmitting. The speed is specified in baud rate for example how many bits-per-seconds can be sent. For this project, the baud rate is 9600 which means 9600 bits-per-seconds can be sent in one frame consisting of 11 bits. The packet data (11 bits) will go to the next state if the counter reaches 5208. 5208 is the calculated result in order to get a 9600 baud rate.

Counter = FPGA clock speed/baud rate
= 50MHz/9600
= 5208

50MHz FPGA clock needs to be reduced in order to have the baud rate, so the counter is used. Therefore the transmission from IDLE to TRANSMIT and vise versa only happens at each 5208 count of counter. Counter is also reset each time it reaches 5208 so that it can start to count up to 5208 again and do the same thing over and over again.

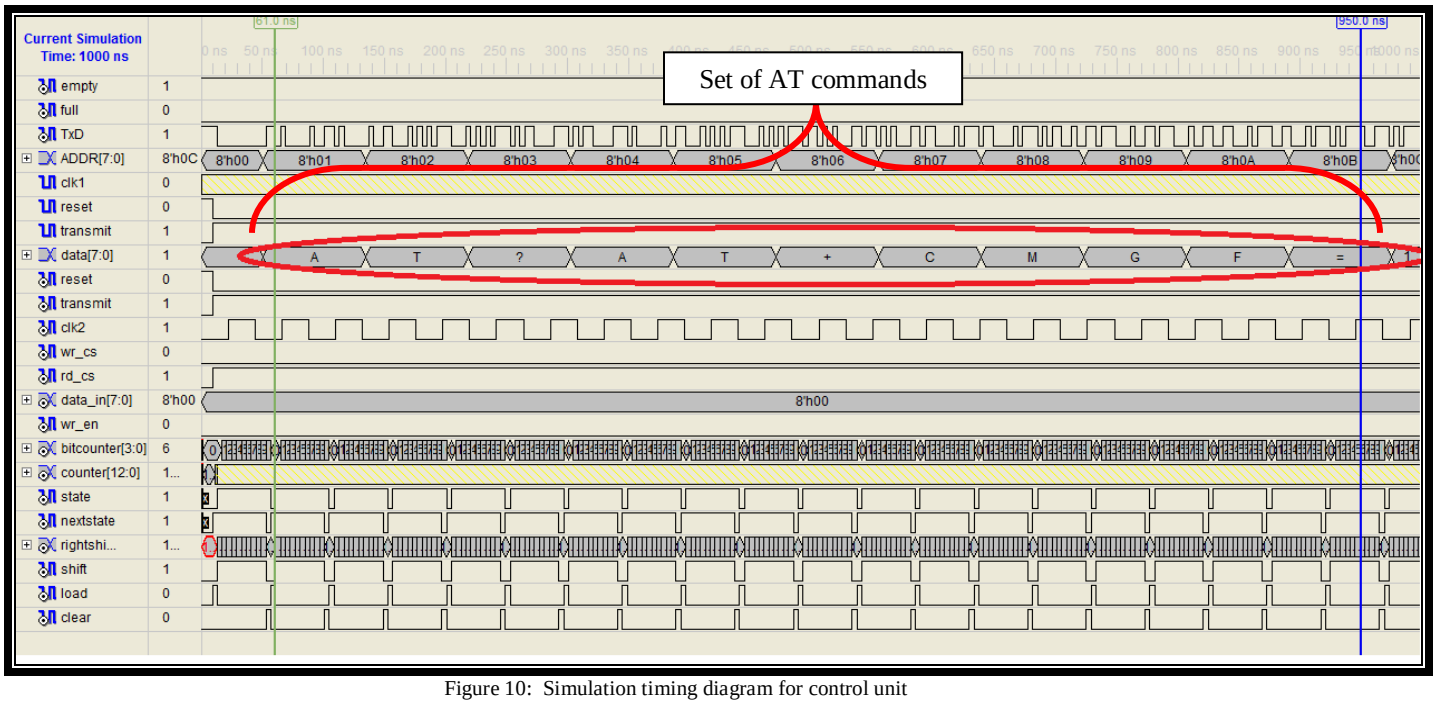

Figure 10: Simulation timing diagram for control unit

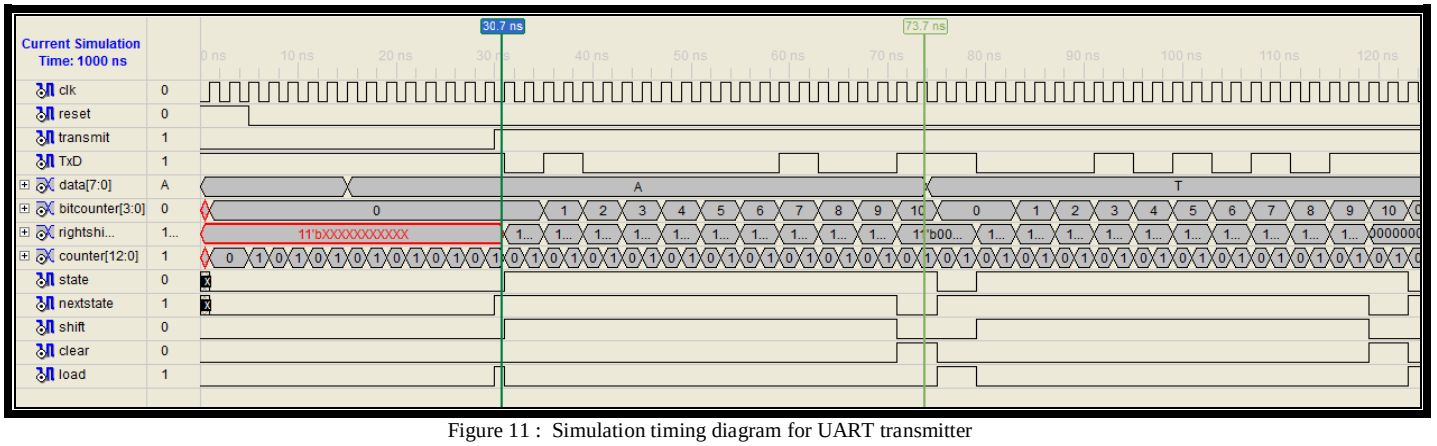

Figure 11 : Simulation timing diagram for UART transmitter

IV. CONCLUSION

This paper present the implementation of GSM in FPGA for a FPGA-based monitoring system for water level and water flow rate. The system is suitable for a real time monitoring in flood situation. All parts of the design have been simulated using ISIM simulator and implemented using Altera Quatus II 9.1 software tools. The design has targeted Altera FPGA "Cyclone II EP2C035F672C" chip. Based on the simulation results, the design is considered successful, however, the design still need some trouble shooting to make the system well functioning after implementing it on the Altera board.

V. FUTURE RECOMMENDATION

The system still needs some development for the future work. In this project, the system is used to send SMS but for the future development, the system could also control the system using GSM through mobile's user to controller. In other words, the system could receive the instruction from user. Another improvement is to use GPRS (General Packet Radio Service) for remote monitoring instead of using GSM, to make the design a web-based design that will reduce the cost of usage by avoiding the use of SMS messages in case of GSM.

ACKNOWLEDGMENT

Alhamdulillah, praise to Allah who gave mercy, His blessings and guidance to solve this paper in a timely manner. Special appreciation goes to my supervisor, Dr. Azilah Saparon for her supervision and knowledge that was presented to the completion of this paper. To those who indirectly contributed in this research, your kindness means a lot to me. Thank you very much.

REFERENCES

[1]. Yun-Zhong, Li; Yan-Peng, Hu; De-Xun, Ye; Real-Time Discharge Monitoring During Flood Season Using Broadband H-ADCP at Huangling Temple Hydrology Station Downstream Three Gorges Dam in Yangtze River, Yangtze River Three Gorges Hydrology Bureau, Yichang, China. Teledyne RD Instruments, San Diego, California, USA. Hening, Huang, 2004.

[2]. Junkhiaw, Sutep; Sivongs, Wit; Sukhapunnaphan, Thada; Developing Flood Warning System for Upland Watersheds of the Chao Phraya Basin, 6th International Study Conferences on GEWAX in Asia and GAME Kyoto International Community House, Japan. Tangtham, Nipon, 2004.

[3] Eddie M.C. Wong, "A Phone-Based Remote Controller for Home and Office Automation". IEEE Transactions on Consumer Electronics, Vo1.40, No.1, February 1994, pp. 28- 34. 2006

[4] Feng Xu; Wentao Wang; , "Design of flood control and video surveillance system of water resources," Advanced Information Management and Service (IMS), 2010 6th International Conference on , vol., no., pp.432-435, Nov. 30 2010-Dec. 2 2010

[5] Basha, E.; Rus, D.; , "Design of early warning flood detection systems for developing countries," Information and Communication Technologies and Development, 2007. ICTD 2007. International Conference on , vol., no., pp.1-10, 15-16 Dec. 2007

[6] Qian Wang; Yun Du; Huaiping Xue; Shengjun Wu; , "Research on flood monitoring system for middle reaches of Yangtze China using RS and GIS," Geoscience and Remote Sensing Symposium, 2005. IGARSS '05. Proceedings. 2005 IEEE International , vol.2, no., pp. 1409- 1412, 25-29 July 2005

[7] Young L C.Yoshiro Imai,Makoto Ooga,et.al,"Mobile Phone-enhance User Interface of Remote Monitoring System",IEEE Proceedings of the International Conference on Mobile Business,2005,pp.63-68, 2005

[8] Xuemei LI, Qiuchen YUAN,Wuchen WU, Xiaohong PENG, Ligang HOU, "Implementation of GSM SMS Remote Control System Based on FPGA", VLSI & System Laboratory Beijing University of Technology Beijing, China, 2007

[9] W.M.El-Medany, M.R. El-Sabry, "GSm-Based Remote Sensing and Control System Using FPGA", Proceedings of the international Conference on Computer and Communication Engineering, ICCCE'08,IEEE catalog number CFP0839D,V II,pp. 1093-1097,2008

[10] Iyemori, Toshihiko; Kamei, Toyohisa; Takeda, Masahiko; Nose, Masahito, 2006. Importance of Real-Time Data Acquisition and Dissemination in Geophysical Research — Lessons Learnt from Quasi Real-Time Data Service for 10 Years. Kyoto University. Asia Oceania Geosciences Society, Singapore. 2006