Digital Controller of a Diesel Generator using an Embedded System

Kwang Seon Ahn*

Abstract: We have designed an embedded controller for the control of a diesel generator using an embedded system. The generator is monitored and controlled remotely via the internet in real-time. The proposed digital controller is designed to handle precisely the distortions and noises of the signals emanating from the diesel generator, and enables abnormal operation of the diesel generator to be notified to the remote manager using the Short Message Service (SMS) of the Internet, which enables the appropriate personnel to take action by remote control according to the incoming messages.

Keywords: *Diesel Generator, Embedded Controller, Remote Control*

1. Introduction

The embedded system is a specially designed computer system that is completely encapsulated by the device it controls. The embedded system has specific requirements and performs pre-defined tasks.

The diesel generator is used when electricity is not readily available, or when power failures occur due to natural disasters such as typhoons or floods, or during other unexpected crises.

Generally, the diesel generator operates in analog. The analog type controller cannot be processed precisely due to the distortions and noises coming from the data. In order to increase data accuracy, the controller needs to be digitalized.

Most digital remote controllers currently in use include microprocessors, but they are inferior to embedded systems which offer functionality and quality in diesel controllers [1,2].

An embedded system is a computer that is equipped with an operating system, and as a result, applications are easily developed using utilities like a network socket program and a Graphic User Interface (GUI). Also, the modification and updating of implementations can be accomplished easily by reprogramming the code instead of constructing new hardware.

The proposed digital controller uses embedded hardware and software. The embedded controller connects to a remote server via the internet; therefore the generator can be monitored and controlled remotely from the web server. Urgent messages from the generator will also be reported to the engineer using SMS.

2. Overall System Configuration

The proposed system consists of an embedded controller and a web server. The embedded CPU controls the diesel generator and uses the internet network to send data to the web server. The web server stores and analyzes the data that comes from the local embedded controller. Figure 1 shows how the embedded controller transfers the data to the web server, and how the web server stores the received data in MySQL DB. The administrator PC connects to the web server and monitors the data stored in the MySQL DB. If the embedded controller sends an urgent message to the web server, the TCP/IP Server daemon will send the received urgent message to the engineer using the SMS service.

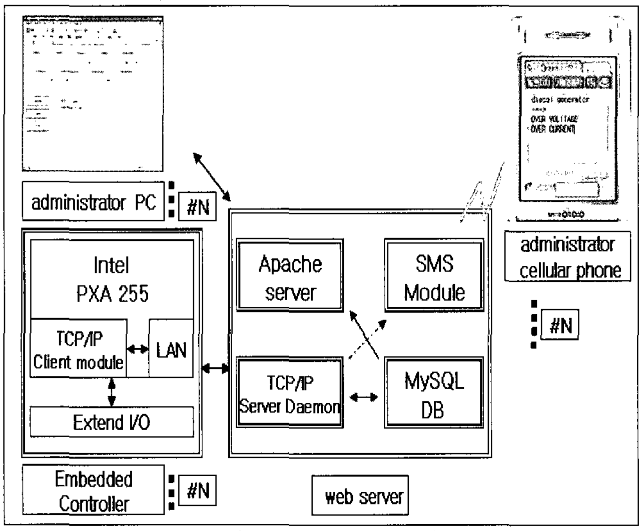


Fig. 1. Entire system block diagram

3. System Operation

3.1 Embedded Controller Hardware Design

The embedded system we developed in this paper uses the HBE-EMPOS Tiny, which has the Intel Xscale PXA255 (400Mhz) microprocessor [3] as a test bed, and

Manuscript received October 13, 2006; accepted November 19, 2006.
Corresponding Author: Kwang Seon Ahn
* Dept. of Computer Engineering, Kyungpook National University, Daegu, Korea (gsahn@knu.ac.kr)

the Linux 2.4.19 is ported in this system. As can be sees in Figure 2, the Extend I/O board includes the voltage detection circuit, current detection circuit, analog I/O circuit, and level I/O. Tool chain for the ARM processor is used for compiling C program. PSpice of Orcad [4] is used for the simulation of the circuits. Verification was performed by taking an actual measurement with the instruments during run time. Apache-2.0.52, PHP-5.0.4, and MySQL-4.1.12 are used for the web server.

The analog signal from the generator is received by the 8 channel ADC 0809 (A /D converter) and is transformed into digital data. The transformed data is processed in the application programs in the HBE-EMPOS Tiny. Table 1 shows the memory maps for the I/O data from the generator.

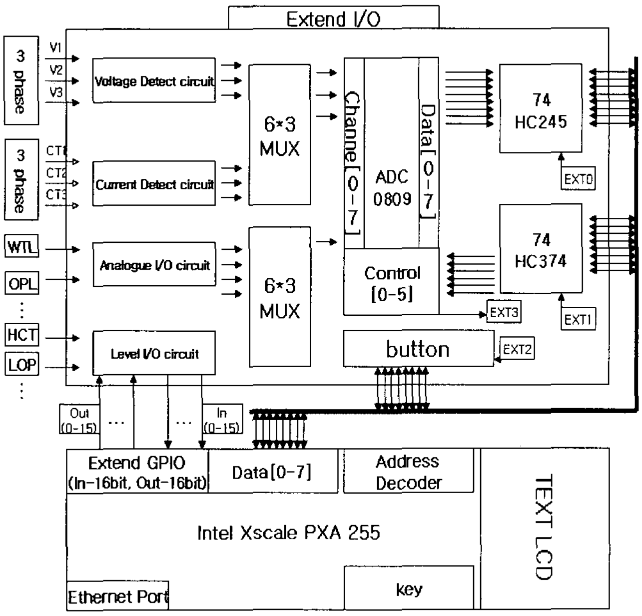


Fig. 2. Diesel generator embedded controller

Table 1. Extend I/O Memory Map

GPIO(9)	Frequency
0xf17000000(Virtual)	Text LCD
0xf16000000(Virtual)	Internal LED
0xf15000000(Virtual)	Primary key
0xf14000000(Virtual)	Extend key
0xf13000000(Virtual)	Extend GPIO Out
0xf12000000(Virtual)	Extend GPIO In
0xf11000000(Virtual)	Adc Control Out
0xf10000000(Virtual)	Adc Data In

3.2 Software Design

The application program for the embedded controller is created by using the C language. Figure 3 shows the flow chart of the main program. The process includes the following steps: Function initialize, Device driver open, Signal register, I/O port initialize, and Getgauge.

Getgauge in the program transforms the input data into the visible digital data. It performs key processing, displaying output, error processing, and button processing.

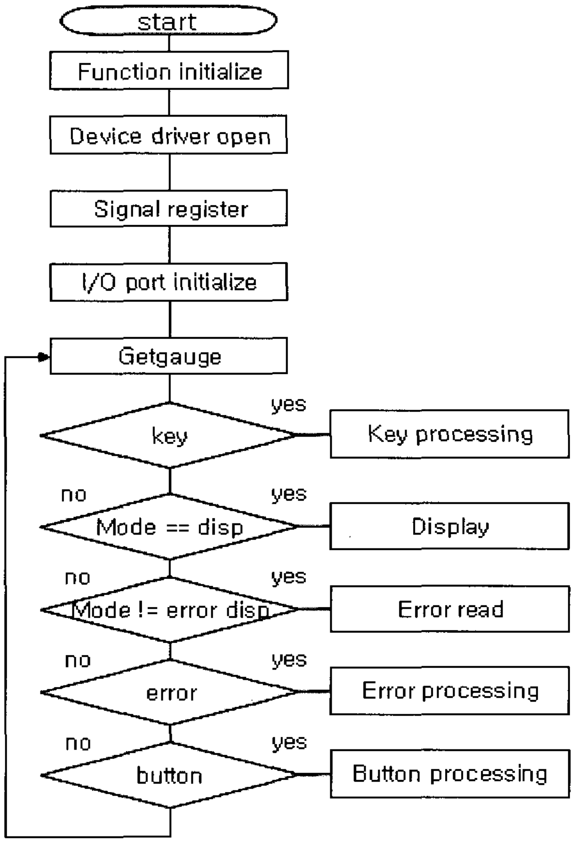


Fig. 3. Main Code Flow Chart

3.3 Communication Module

The TCP/IP protocol is widely used in such internet applications as email transfer, file transfer using FTP, and web page creations using the HTTP [5]. Therefore, TCP/IP protocol can be used to communicate messages between the embedded controller and the web server.

Various data such as the input voltages collected from the diesel generator are saved into files. At this time, the TCP/IP client module creates sockets, and immediately tries to connect to the web server. It reads the log files, and sends them to the TCP/IP server daemon of the web server through the sockets.

The web server should run its TCP/IP server daemon to receive the data. The TCP/IP server daemon should be running in the background mode so that it operates only when there is a request from the TCP/IP client module.

Figure 4 is the TCP/IP server daemon pseudo code. *select()* allows communication between the TCP/IP clients [6]. The TCP/IP client module connection is conducted only when requested, and *read()* reads the data and stores them in the MySQL DB when the TCP/IP client sends the data.

```
if(-1==select(fd_max+1,&temps,0,0,&timeout))
   exit(1);
for(fd=0;fd<fd_max+1;fd++){
   if(FD_ISSET(fd,&temps)){
      if(fd==serv_sock){    /* in case of connection request*/
         clnt_len=sizeof(clnt_addr);
         clnt_sock=accept(serv_sock,
            (struct sockaddr*)&clnt_addr,
            &clnt_len);
         FD_SET(clnt_sock,&reads);
         if(fd_max<clnt_sock)
         fd_max=clnt_sock;
      }else{
         str_len=read(fd,message,sizeof(message));
         message[str_len]='₩0';
         if(str_len==0){    /* in case of disconnection request*/
            FD_CLR(fd,&reads);
            close(fd);
         }else{              /* in case of data received */
            write(fd,message,str_len);
            strcpy(strque,"insert into woorig values (0,0,'");
            if(!isalnum(message[str_len-1]))
               message[str_len-1]='₩0';
            strcpy(strque+strlen(strque),message);
            strcpy(strque+strlen(strque),"',0)");
            mysql_query(&mysql,strque);
         }
      }
   }
}
```

Fig. 4. TCP/IP Server Daemon

3.4 Construction of remote web servers

The operating system of the web server is LINUX, and the APM (Apache+PHP+MySQL) package has been installed in the construction of the web server. Tables 2 and 3 are the MySQL DB tables for storing the data that has been sent from the TCP /IP client. These data are parsed according to their data types as mentioned above, and then stored in the corresponding table. In the tables, SQ is the counter number that automatically increases whenever data has been transferred from each of the client modules.

Table 2. MySQL DB input table

Field	Type	Null	Key	Remarks
SQ	int		PRI	Counter number increases automatically
ID	int	YES		Separate ID of each Controller
VLT	int	YES		Input Voltage
CUR	int	YES		Input Current
FRQ	int	YES		Input Frequency
VAT	int	YES		Battery Remaining
PF	text	YES		Power Factor
0IL	text	YES		Oil Pressure
TMP	int	YES		Coolant Temperature
KW	int	YES		Power
HOU	int	YES		Operation Hours

Table 3. MySQL DB error table

Field	Type	Null	Key	Remarks
SQ	int		PRI	Counter number increases automatically
ID	int	YES		Separate ID of each Controller
ER	text	YES		Error Description
ET	int	YES		Error Level

Figure 5 shows the PHP Code used for transferring information from the generator to the server through the web page. It checks the information that is being transferred from the TCP / IP client module, and shows the error contents whenever there is any updated information on the web.

From the code shown below, it uses the $sequence for the new input data. It receives count number $sq in the process of searching MySQL DB, and places $sq to $sequence. Then the $sq sends queries for the values that are greater than $squence, puts the data into sessions, and reloads the pages. Both efficiency and real-time detection are implemented in this process.

```
try{
   $mysql = new safe_mysqli('localhost', 'root', '', 'test');
   $stmt = $mysql->prepare(
      "SELECT * from woorig where sq > ?");
   $sequence = 0;
   while(true){
      $stmt->bind_param('i',$sequence);
      $stmt->execute();
      $stmt->bind_result($sq,$id,$er,$et);
      while($stmt->fetch()){
         $sequence = (int)$sq;
         echo "$sq. $er<br>";
      }
      sleep(1);
   }
   $stmt->free_result();
}catch(SQLException $e){
   die("SQL Error : ". $e->getSQLError().
      " in <hr><PRE>".$e->getSQL()."</PRE><hr>");
}catch(DBException $e){
   die("Database Error: ". $e->GetMessage());
}catch(Exception $e){
   die(exception_dump($e));
}
```

Fig. 5. Web Page PHP Code

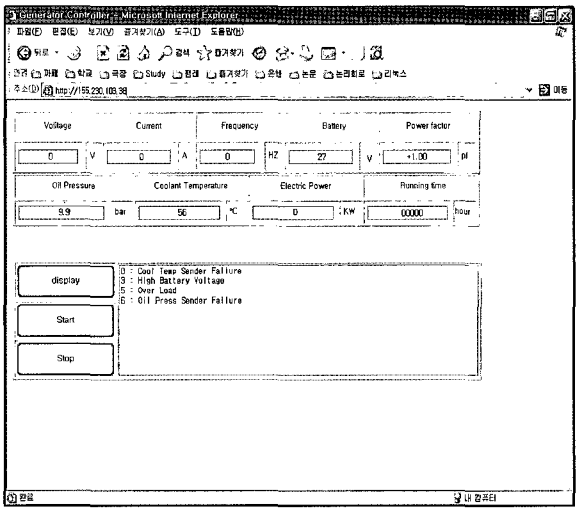


Fig. 6. Remote control or monitoring of the web page

3.5 Short Message Service (SMS)

The functions to detect malfunction and status of the system are implemented, and that information is reported to the server through the web. The web server installs the

SMS to send instant error messages to the TCP/IP server by SUREM [8]. The operation is similar to that of the general socket program. The PHP module connects error messages to the SMS server and transfers them by using the *sendto()* in Figure 7. An urgent message is sent to the engineer from the server.

```
/* Connect the socket to the specified server */
if(connect(sd, (struct sockaddr *)&sad, sizeof(sad)) < 0){
  fprintf(stderr, "connect failed\n");
  exit(1);
}
if(sendto(sd, (char*)&AcsPkt, PACKET_SIZE, 0,
  (struct sockaddr *)&sad, sizeof(sad)) < 0)
  perror("Sending datagram message");
if(read(sd, (char*)&AcsPkt,(PACKET_SIZE)) < 0){
  perror("Error receving data from server");
  exit(1);
}
if(AcsPkt.szStatus == 'O') {
/* sending success */
}else {
/* sending fail */
}
```

Fig. 7. SUREM PHP module

3.6 System Test

The final system test is composed of the following operations: Extended I/O Circuit test, embedded controller test, TCP/IP communication test, web server test, and remote control test.

Extended I/O board converts the AC voltage from the generator to DC voltage by using the A/D converter. Figure 8 shows the results of the testing conducted using the output voltage of the generator. The comparison of the generated voltage with the measured voltage between AC 320V and AC 450 shows that as the AC voltage increases, the measured value also increases. This difference could be adjusted by tuning the resister value of the OP amp in the extended I/O circuit. Testing of current, frequency, and the engine temperatures is conducted using a similar method.

As a result of our experiments, it may be concluded that the remote web servers and SMS function properly.

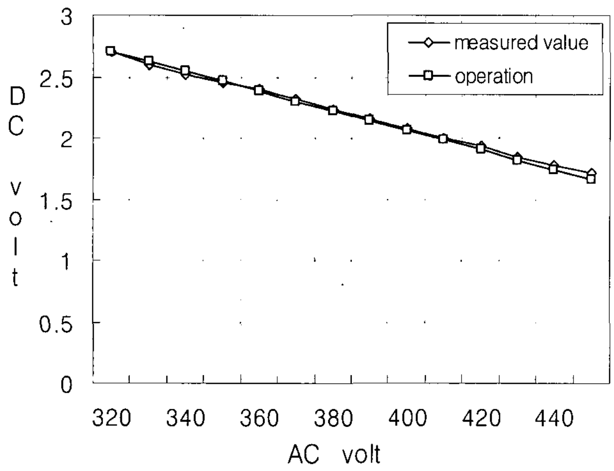


Fig. 8. Transformed DC voltage from generated AC voltage

4. Conclusion

This research concerns remote controllers which effectively control diesel generators. The developed system is capable of real-time monitoring and remote controlling, and is very effective in responding to malfunctions.

The developed remote embedded controller has the following characteristics. First of all, it makes the generator work more efficiently by providing accurate data. Secondly, the administrator can check the status of the generator at any remote location as long as there is access to the web. Finally, in cases of emergency, urgent messages can be sent immediately to the engineer by using the SMS service.

We are sure that our embedded controller offers a wide scope of application in the field of remote digital controllers in the diesel generator industry.

Reference

[1] J. Teng, C. Tseng, and Y. Chen, "Integration of networked embedded systems into power equipment remote control and monitoring," in *2004 IEEE Region 10 Conference Volume C*, vol 3, pp. 566-569, Nov. 2004.

[2] M. Moallem, "A laboratory test-bed for embedded computer control," *IEEE Transactions on Education*, Volume 47, Issue 3, pp. 340-347, Aug. 2004.

[3] *HBE-EMPOS Tiny User's Manual ver1.0,* Hanback Electronics Co.

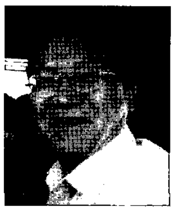
[4] *Simulation with PSpice Workbook*, Cadence Design Systems Inc., pp. 1-45.

[5] M. T. Jones, "Embedded with the Mailman," in *Embedded Systems Programming*, pp. 45-52, Oct. 2001.

[6] S. W. Yoon, "TCP/IP socket programming," in *FREELEC*, pp. 284-292, 2004.

[7] H. M. Shin, *Advanced PHP for Windows*, Daelim Co., pp. 166-171, 2003.

[8] *surem* [Online]. Available: http://www.surem.co.kr



Kwang Seon Ahn
Ahn received a Ph.D. degree in Electronics Engineering from Yonsei Univ. in 1980. He has been a professor at Kyungpook National Univ. since 1977. His current research interests are in the areas of Embedded System Design, SoC, and LINUX.