DEVELOPING A SMART CONTROL SYSTEM OF ELECTRICAL APPLIANCES BY COMPUTER INTERFACING

N.K. Das[(1)], Capt. M. Ziauddin Alamgir, BN[(2)], C.K.Das[(3)], M. M. Hasan[(4)], M. M. Rahman[(5)] and M. F. Alam[(5)]

1, 3.   Assistant Professor, Department of Electrical Electronic and Engineering, Chittagong University of Engineering & Technology, E-mail: nipu16@gmail.com

2.   Senior Instructors, Department of Electrical Electronic and Communication Engineering, Military Institute of Science and Technology, E-mail: naziazafir@yahoo.com

4, 5.   Lecturer, Chittagong University of Engineering & Technology, E-mail: mahbub.rahman336@yahoo.com

ABSTRACT

The use of computer interfacing systems for controlling the device is spreading at an increasingly fast pace, with analogous wired system being substituted by computer interfaced systems alternatives in growing number of industries. As the technology progress many control system have been developed ranging from high end stuff to our common daily life. The main objective of this project is to develop a roadmap on communication and control technologies for distributed electrical equipments. Equipments located at any remote place can be monitored and controlled. The process is mainly to communicate between the two remote controlling device and setup connection between devices which are to be controlled. The main hallmark is to get access to the controlling section from any distance location in a convenient way and supervising the system accordingly. The speciality of this project is that the operator will able to control different devices at home/industry by using a single PC. An added advantage of this control system is that the operator will able to know the status of the device to be controlled from a remote place. Here, the algorithm of controlling any kind of electrical equipments is developed. The speeds of a DC motor are controlled as a test case and also monitor the voltage and current during different speed level of the motor.

KEY WORDS— Distance control, Advance Monitoring, IP addresses based control, Client server interface, Visual basic.

1.0   INTRODUCTION

A Control System is a very important aspect in modern technology. It is an interconnection of components connected or related in such a manner as to command, direct, or regulate itself or another system. A controller is a control system that manages the behavior of another device or System [11] as shown in figure 1. It is a device or group of devices that serves to govern in some predetermined manner the performance of electric devices. A smarter control makes this transformation possible by bringing the philosophies, concepts and technologies that enabled the internet to the utility and the electric component. More importantly, it enables the industry's best ideas for control modernization to achieve the full potential. The electric industry is poised to make the transformation from a centralized, producer-controlled network to one that is less centralized and more consumer-interactive. The move to a smarter control promises to change the industry's entire business model and its relationship with all stakeholders, involving and affecting utilities, regulators, energy service providers, technology and automation vendors and all consumers of electric power. Examples of control systems can be found all around, and in fact there are very few mechanical or electro-mechanical systems that do not include some kind of a feedback control device. In robotics, control design algorithms are responsible for the motion of the manipulators. In flight applications, control algorithms are designed for stabilization, altitude regulation and disturbance rejection. Cruise control is an interesting application in which the automobile's speed is set at a fixed value. In electronic amplifiers feedback is used to reduce the damaging influence of external noise. In addition, these days control systems can be found in diverse fields ranging from semiconductor manufacturing to environmental regulation.

2.0   METHODOLOGY

This project can be effectively and conveniently utilized for the control of different appliances. As this project could be extended to control about 255 devices, this could be used for computerization of an office, home, or a firm. An added advantage of this project is that the status of the device can be ad monitored & controlled from distance station [7].
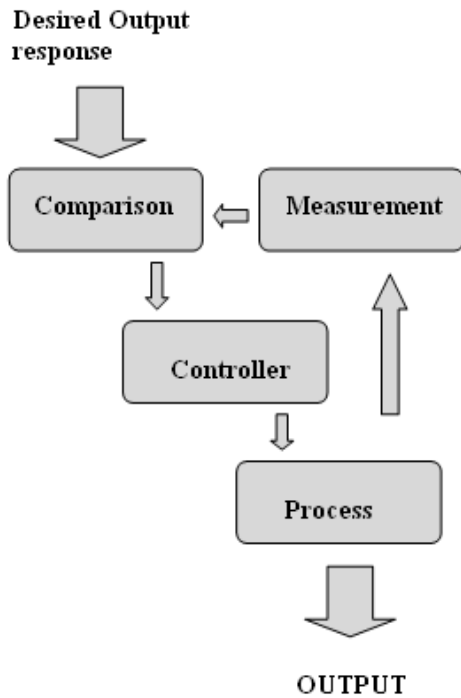
Fig 1: Block diagram of a controller

3.0 EXPERIMENTAL DESIGN

The program to control the appliances is written in C [1]. It is compiled using Turbo C compiler. The flow chart of this program is shown in figure 2 and the program sequence as follows

- On running the program SRC_CODE.C the menu appears asking for the operation to be done.
- The very first option shows the list of the loads that could be controlled using this software along with their codes
- The outstanding part of this software is activating of required PROFILES at once so that the burden of controlling of individual devices is eliminated.
- User defined profile can be set.
- As a basic requirement the control over individual devices is also provided. If the user by mistake tries to switch ON the device which is already ON, the software detects it & the corresponding message is displayed.

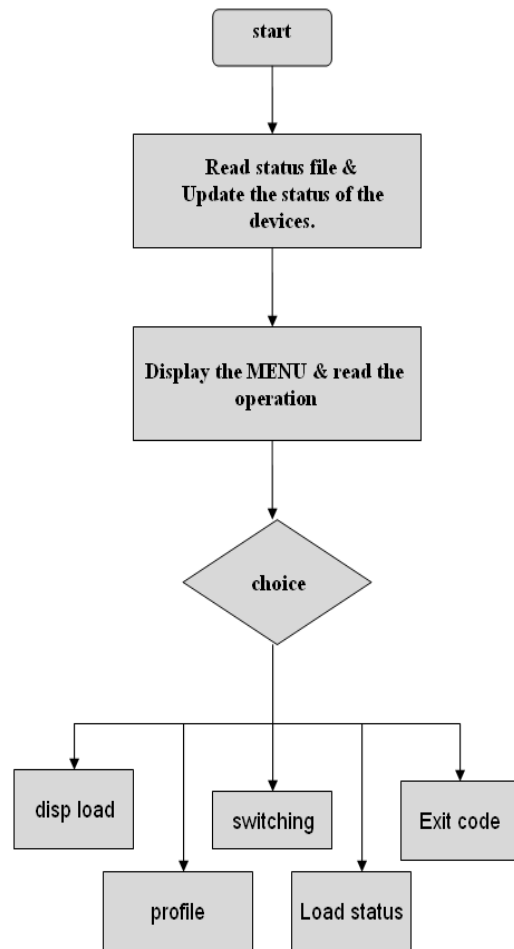The software also allows the user to know the current status of the loads.

Fig 2: Flow chart of the programme

4.0 SYSTEM OVERVIEW

Electrical equipments are connected with the remote server [10]. In figure 3 shows the interconnection of electrical equipments with remote server which is to be controlled. There are two ways to log in to server PC, one is Microsoft remote assistance system and another is creating a webpage [2-9]. In this project virtual IP used when the system was driven by Microsoft remote assistant mode and In the case of webpage, real IP was used. When the client log in to the server and can easily control the equipments as well as will get status update of them. To view the current status of the electrical component an application has been developed by using this application an end user wail is able to view the live status. In this project it was implemented by visual basic and C language [1, 3]. A TCP/IP listener is responsible for accepting data which is sent by the device via internet to a specific port of the desired address. This data consist of id number of the client pc, which shows the status of the electrical components within an instant.

Fig 3: Interconnection of electrical equipments with remote server

Fig 4: Circuit arrangements for the project

5.0   PROJECT WORK

In this system internet is used to control & collect information of an electrical component at every instant & passes this information to a remote PC. As data being passed to the remote PC containing a private IP address will now listen to a specific port assigned for it. A database created at the server end will store these data. The database now contains all the necessary information it needs to update the information of any particular equipment. This page Contains detailed information of present status of the electrical component. As new data being inserted into the database, it will update the status containing speed, voltage, current, and temperature etc of the component at the client end. In this project we controlled a DC motor through interfacing with a computer. Here microcontroller was used to control and check the status of the DC motor. The programming code for the microcontroller is given in Appendix-A. Equipment arrangement of the project is shown in figure 4 and Stimulated View of Status Checking Unit in figure 5.
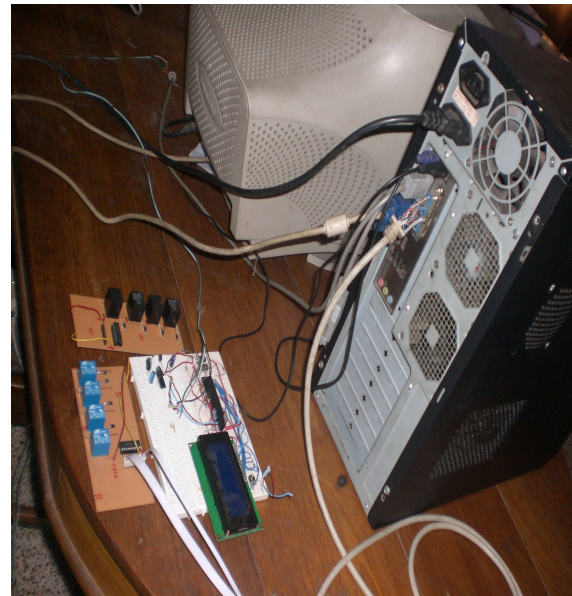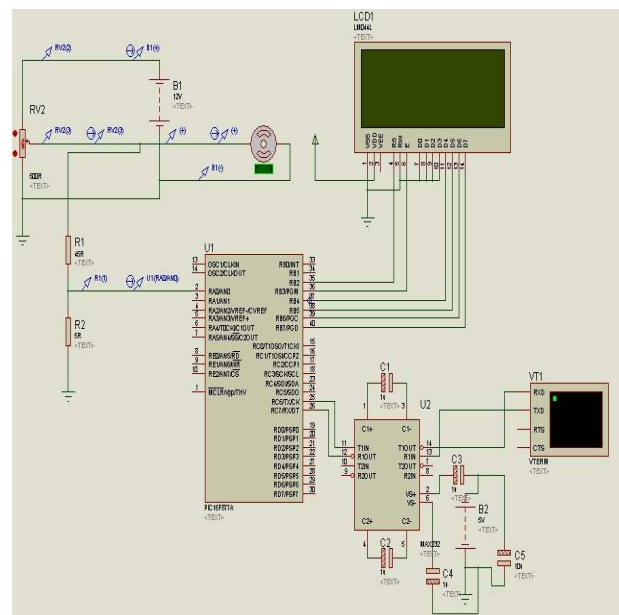

Fig 5: Stimulated View of Status Checking Unit

6.0   APPLICATIONS

Office equipment, medical equipment

There is a wide variety of motorized office equipment such as personal computers, computer peripherals, copy machines and fax machines as well as smaller items such as electric pencil sharpeners. Controllers for these types of equipment are built into the equipment. Some quite sophisticated controllers are used to control the motors in computer disc drives. Medical equipment may include very sophisticated controllers.

Commercial applications

Commercial buildings have larger heating ventilation and air conditioning equipment than

that found in individual residences. In addition, motors are used for elevators, escalators and other applications.

Industrial applications

Many industrial applications are dependent upon motors (or machines), which range from the size of human thumb to the size of a railroad locomotive. The controllers can be built into the driven equipment, installed separately, installed in an enclosure along with other machine control equipment or installed in motor control centres. Control centres are multi-compartment steel enclosures designed to enclose many motor controllers. In this case the controllers communicate with each other so they can work the electrical equipment together as a team.

7.0   CONCLUSION

This project implemented a smart control system by this system the client can control and monitor the electrical equipment from any place effectively and conveniently. As this project could be extended to control about 255 devices, this could be used for computerization of an office, home, or a firm. Though it is quiet costlier, the circuit is simple and the working mechanism could be easily understood. An added advantage of this project is that the will be able to know the status of the device which is to be controlled. The program to control the appliances is written in C language which is more users friendly and easy to understand than other programming languages.

APPENDIX-A

The programming code for the microcontroller which was used to control the DC Motor is given below.

```
unsigned char datav,datai;
unsigned float j,v,k,i;
unsigned char x[6], y[9];
// float to string conversion function (user defined)
   void conv (unsigned short row, unsigned short col, float b)
   {
   float a;
   if ((b>=0)&&(b<10))        // 0 <= x < 10
   {
   a=b;
   x[0]=  (int)a;
   x[1]=  (int)(a*10) - x[0]*10;
   x[2]=  (int)(a*100) -x[0]*100- x[1]*10;

   y[0]= x[0] + 48;
   y[1]= 46;
   y[2]= x[1] + 48;
   y[3]= x[2] + 48;
   y[4]= 0;
   y[5]= 0;
   y[6]= 0;
   y[7]= 0;
   y[8]= 0;

   }
   if ((b>=10)&&(b<100))       // 10 <= x < 100
   {
   a=b;
   x[0]=  (int)a;
   x[1]=  (int)(a*10) - 10*x[0];
   x[2]=  (int)(a*100) -100*x[0]- 10*x[1];
   x[3]=(int)(a*1000)-1000*x[0]-10*x[1]- 0*x[2];

   y[0]= x[0] + 48;
   y[1]= x[1] + 48;
   y[2]= 46;
   y[3]= x[2] + 48;
   y[4]= x[3] + 48;
   y[5]= 0;
   y[6]= 0;
   y[7]= 0;
   y[8]= 0;

   }
   if ((b>-10)&&(b<0))         // -10 < x < 0
   {
   a=-b;
   x[0]=  (int)a;
   x[1]=  (int)(a*10) - x[0]*10;
   x[2]=  (int)(a*100) -x[0]*100- x[1]*10;

   y[0]= 45;
   y[1]= x[0] + 48;
   y[2]= 46;
   y[3]= x[1] + 48;
   y[4]= x[2] + 48;
   y[5]= 0;
   y[6]= 0;
   y[7]= 0;
   y[8]= 0;
   }
   lcd_out (row, col, y);
   }
   void main (void)
   {
   // ADC initiation
   adcon0=0b00000001;
   adcon1=0b10000000;
   //

   trisb=0;        // LCD output
   trisa=255;       // Voltage input
```

```c
// LCD initiation
lcd_init(&portb);
lcd_cmd(lcd_cursor_off);
lcd_cmd(lcd_clear);
//
lcd_out(1,1, "V[V]=");
lcd_out(2,1, "I[A]=");

while (1)
{
j=adc_read(0);
v=(50*j/1024);
k=(v/9.677419355);
i=(50*k/62);
conv (1,6,v);
conv (2,6,i);

Lcd_Out(3,1, "f[Hz]=50");

delay_ms(100);

Usart_Init(8900);
datav=v;
datai=i;
Usart_Write(datav);         // Send voltage via
USART
delay_ms(500);
Usart_write(datai);         // Send Amp via
USART
}
}
```

Source code for the write operation

```vb
Private Declare Function Inp Lib "inpout32.dll"
Alias "Inp32" (ByVal PortAddress As Integer) As
Integer. Private Declare Sub Out Lib
"inpout32.dll" Alias "Out32" (ByVal PortAddress
As Integer, ByVal Value As Integer)

  Dim Value As Integer
  Dim PortAddress As Integer
  Private Sub cmdWriteToPort_Click()
  'Write a byte to a port, read it back,
  '& display the result.
  Out PortAddress, Value
  Text1.Text = Inp(PortAddress)
  Value = Value + 1
  If Value = 255 Then Value = 0
  End Sub
  Private Sub Form_Load()
  'Test program for inpout32.dll
  Value = 0
  'Change this value to match the port to write to:
  '(Usual parallel-port addresses are &h378,
&h278, &h3BC)
  PortAddress = &H378
  End Sub
```

Source code for the read operation:
```vb
  Dim a
  Private Sub Form_Load()
  MSComm1.PortOpen = True
  End Sub

  Private Sub MSComm1_OnComm ()
  On Error Resume Next
  'mscomm1.RThreshold
  a = Asc(MSComm1.Input)
  'b = MSComm1.Output
  Label1.Caption = a
  Label3.Caption = a / 9.677419355
  Label2.Caption = 50 * Label3 / 62
  Debug.Print a
  'MsgBox b
  End Sub
```

REFERENCES

C programming for microcontroller by Joe Pardue 2nd Edition, published by smily micro, page-137,273,275,287,289.

Serial port complete by Jan Axelson, 2nd Edition. Lakeview Research LLC, page 43-76.

Visual basic for electronic engineering applications by Vincent Himpe, 2nd edition, page-37-47&99-108.

www. wikipidia.org/remote desktop protocol.

www.electrofriends.com

www.picgallery.com

Device switching using PC's parallel port by Ranjith, page-1-3&6-8.

Muhammad H.R Rashid "Power Electronics" 3rd edition, Page-641.

Wayne Thomasi, "Advanced Electronics Communications Systems", 6th Edition, page-629

Wireless device server-www.lantronic.com

Control System Engineering by NISE, 4th edition.