Design and Construction of a Computer Based Power Billing System

Abubakar Sadiq Mohammed

Electrical/Computer Engineering Department, Federal University of Technology Minna, Nigeria
E-mail: <abussadiq@yahoo.com>

Abstract

Meter reading is an important part of billing system for residential and commercial buildings. This often does not reflect the exact or true electricity consumption of such buildings, resulting in an inefficient bill mainly due to human errors. Thus, it is essential to have n efficient and effective system for this measurement purposes. This project is about the design and construction of a personal computer (PC) based electronic billing system. It is made up of integrated circuits, active and passive hardware devices. And a Visual Basic software program used for programming the PC parallel port. The device is used to measure accurately the electric power consumed by a residential or commercial buildings which is more economical compared to the electromechanical devices. Utility companies can directly monitor and control electrical power supply billing of its spatially distributed consumers without engaging the services of meter readers. The meter displays the sale rate of power and the consumed power per minute, respectively.

***Keywords**: Power supply, power converter, analog-to-digital converter, parallel port, switch control, Visual Basic.*

Introduction

The importance of an effective and efficient billing system cannot be over emphasized. This is very critical as small errors can lead to substantial loss of revenue. The main theme of this project is design and construction of a PC based billing system to display the consumed power per minute, the sale rate of power per minute and other parameters (IEEE 2008).

Fig. 1 gives the overall block diagram of the computerized power billing system. This is an electronic meter system used to measure accurately the electric power consumed by a company or an individual. A scratch card can be incorporated such that when a user wants to recharge, he buys and scratches the credit card. He keys in the pin number onto a user interface and click “load” in the computer system. The computer will then confirm the validity of the card and displays the credit wattage bought. As the user plugs in the load, the system will display the consumed power, the balance credit and the rate of sale of power. When the credit is exhausted, the computer switches off power to the user’s load until he loads in credit again.

The display area is the monitor of the PC with a prompt to place a load. A Visual Basic program (see Appendix) has been written to manipulate the incoming digital signal from the Analog-to-Digital Converter (ADC) to display the correct load. The software program can be written in various languages (C, C++, Microsoft Visual Basic, FORTRAN, assembly Language etc) but Microsoft Visual Basic was used for this project (Byron 2002).

Computer Parallel Ports

The parallel port is the most commonly used port for interfacing home projects. It allows the input of up to 9-bits or the output of 12-bits at any one given time, thus requiring minimal external circuit to implement many simpler tasks. The port is composed of 4 control links, 5 status lines and 8 data lines. It is found commonly at the back of PC a D-type 25 pin female connector. (Hall 1992; Intersil 2008).

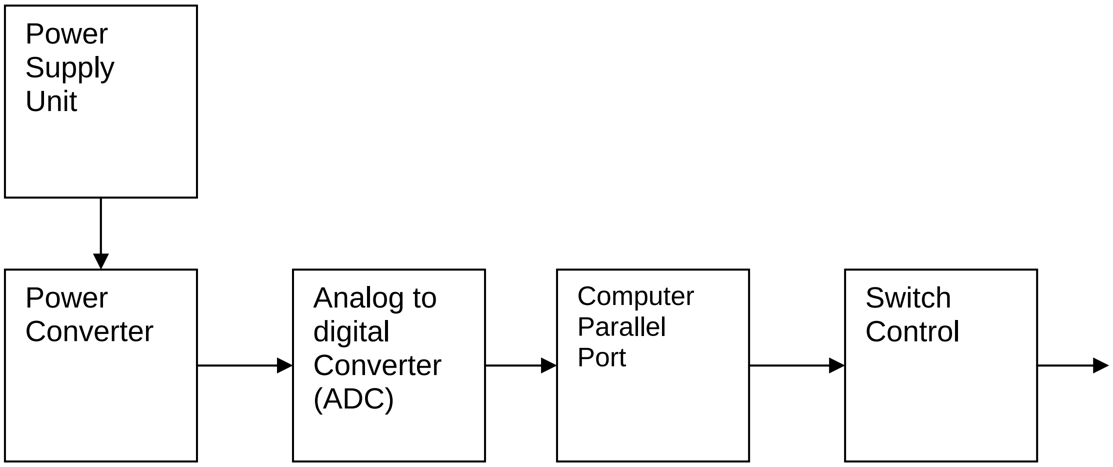

Fig. 1. Block diagram of computerized power billing machine.

Hardware

The main hardware components of the system are discussed in this section.

Hardware Properties

The pin out of the DB-25 connector is shown in Fig. 2. The photo on the left in Fig. 2 shows the back panel of an expansion card, with a parallel port's 25-pin female D-sub connector on the left side of the panel. (The other connector is for a video monitor). The photo on the right shows the 36-pin female Centronics connector used on most printers. The lines in the DB-25 connector of the parallel port are divided into 17 signal lines and 8 earth lines. The 17 signal lines are divided into; 4-control lines, 5-status lines, and 8-data lines (data bus). The new implementation of the port allows bi-directional communication by means of these lines. Each one of these lines (control, status data) can be indexed in an independent way by means of registers.

Parallel Port Registers

The data, control and status lines are connected to their corresponding registers inside the computer. By manipulating these registers in a program, one can easily read or write to the parallel port with programming languages such as C and Visual basic. The registers found in the standard parallel port are; the data registers, the status registers, and the control registers respectively. And as the names imply, the data register is connected to the data lines, the control register is connected to the control lines and the status register to the status lines. The parallel port has three commonly used base addresses. These are listed in the Table 1 below. These addresses may change from machine to machine.

When the computer is first turned on, BIOS (Basic Input / Output System) will determine the number of ports you have and assign device labels LPT1, LPT2 and LPT3 to them. BIOS first look at address 3BCh. If a parallel port is found there, it is assigned as LPT1, it then searches location 378h, and assign the next free device label if a parallel card is found there. If a parallel card is found there, it is assigned the next free device level. It will however be either LPT1 if not found at 3BCh or LTP2 if found at 3BCh. The last port is 278h and follows the same procedure like the other two ports. Therefore, it is possible to have a LPT2 which is at 378h and not at the expected address 278h (Axelson 1996).

Software

Parallel Port Programming

Almost all programming languages allow programmers to access the parallel port using some library functions. QBasic is perhaps the easiest language to program the parallel port with. Microsoft Visual Basic has been a favorite programming language for real world projects. Its ease of use and flexibility made it popular and it was used for programming the parallel port in line with the expected results. (See Appendix for the Visual Basic program).

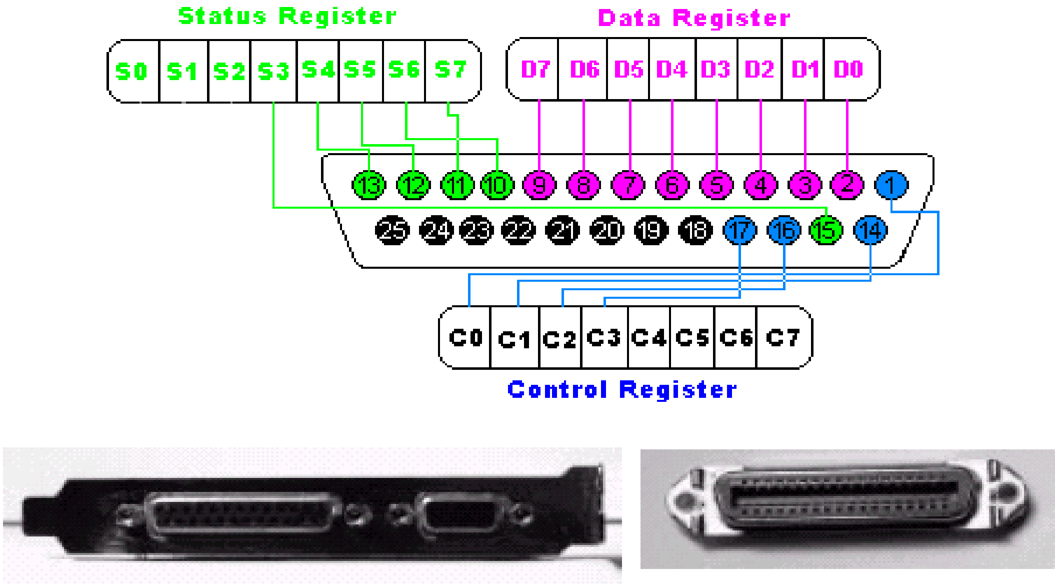

Fig. 2. Pin-out components of the parallel port.

Table 1. Parallel port addresses.

Address	Notes
3BCh-3BFh	Used for parallel ports which were incorporated into video cards and now, commonly an option for ports controlled by BIOS. Does not support ECP address.
378h-37Fh	Usual address for LPT1
278h-27Fh	Usual address for LPT2

Design and Implementation

The design and construction of a computerized power billing machine is divided into six modules namely; Power supply unit, Power converter, Analog to Digital Converter (ADC), Computer parallel port, Switch control and the Output point (Fig. 1).

Power Supply Unit

The 240/12V transformer steps down the 240Vac to 12Vac as shown in Fig. 3. The full-wave rectifier turns the ac voltage into dc voltage. A 3300uf/25V capacitor filters off the ripple in the converted dc volt. And LM7805 is a voltage regulator that regulates the voltage to 5V, for both ADC and computer parallel port respectively. A 1K resistor serves as a precaution by limiting the current into the indicator LED.

Power Converter

The second transformer is connected as a current to voltage converter as shown in Fig. 4. As the load is applied to the system, the current drawn by the load is converted into dc voltage. The second full-wave rectifier turns the ac voltage into dc voltage. The 3300uf/25V capacitor filters off the ripple in the converted dc voltage and the 100 Ohms resistor serves as a load resistor. If a load of say 170W is plugged into the system and the converted current to voltage is 9V. The filtering capacitor will charge to 9V. And if the load is reduced to 100W and the converted voltage is 6V. The capacitor will slowly discharge to 6V and within this period, the measuring system must have displayed several erroneous readings on the load due to the slow discharge of the capacitor. In order to ensure accurate reading and measurement, a load resistor of 100 Ohms is connected to discharge the residual voltage when the voltage drops. The 20K pot resistor is used to divide the input voltage to the ADC. (Millman 1979).

Analogue To Digital Converter (ADC)

An analog to digital Converter accepts an analog voltage as input and provides a digital output as shown in Fig. 5. This digital information is in binary format consisting of 0s and 1s only. The finds a fractional binary number that gives the closest approximation to the fraction formed by the input and the reference voltage (Seshanna 2006). The ADC accepts 5V as a maximum input voltage. Pin6 of the ADC0804 is the input pin as shown in Fig. 5. Pins 3 and 5 are connected together and brought to ground voltage momentarily in order to set the ADC in continuous conversion mode. Pin1 of the parallel port is used to initialize the ADC for this conversion by putting a 0.5s pulse width to bias the transistor C945, thereby allowing a ground voltage to move from the emitter to Pins3 and 5 via the collector. The 0.5s pulse width is generated from the system program. The digital outputs of the ADC are Pins 11 to 18 and are connected to Pins 2 to 9 of the parallel port for data transfer. The program data for the ADC can be collected and manipulated to get a desired result through the system data (Intersil 2008).

Computer Parallel Port

The parallel port is the commonly used for interfacing applications involving data transfer to and from the PC (Hall 1992). This is where data is transferred from a device to the system using the parallel port see Fig. 6. It is also the interface between the PC and the device whereby once a credit is loaded into the system it sends a message to the device for the load to work. The parallel port used is the PS/2(Bi-directional) port as shown in Fig. 6 (Axelson 1996; Craig 1998).

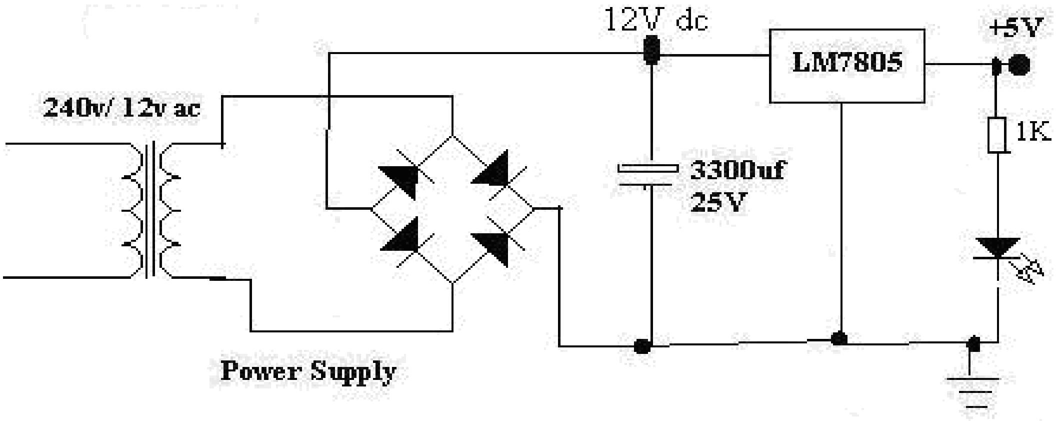

Fig. 3. Power Supply Unit.

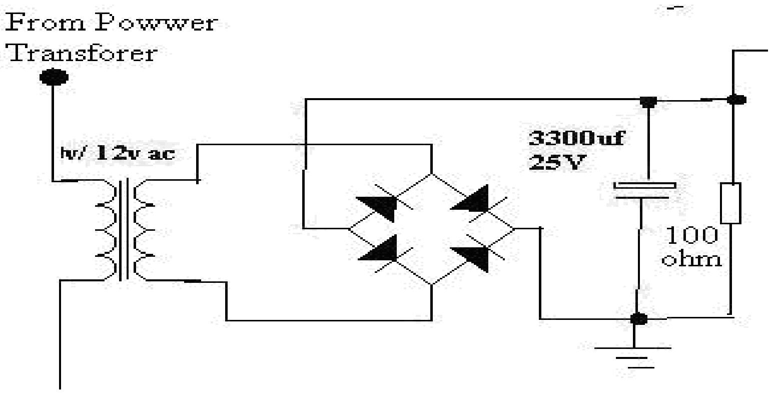

Fig. 4. Power Converter.

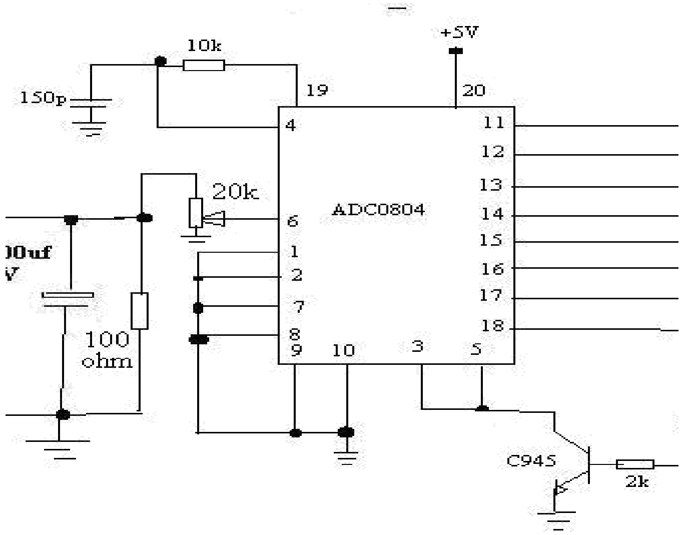

Fig. 5. Analogue to digital converter (ADC).

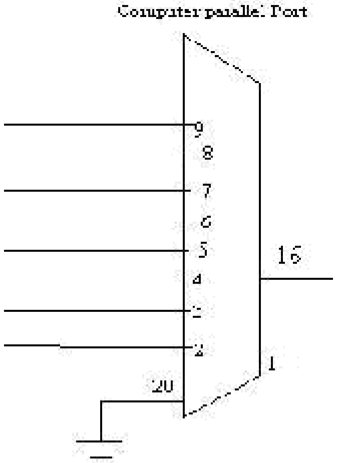

Fig. 6. Computer parallel port.

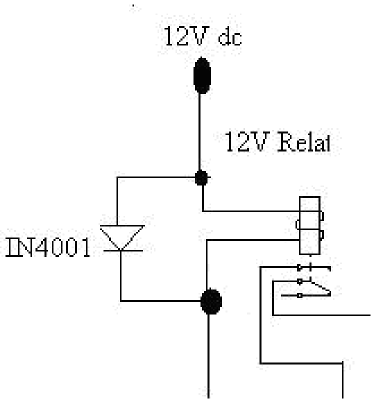

Fig.7. Switch control (relay).

Fig. 8. Ouput point.

Switch Control

The relay operates based on the principle of electromagnetic induction. It is basically a mechanical switch operated by a magnetic coil see Fig. 7. These relays are available with coils that can operate or close the contacts from millivolts to the hundred volt range. Relays with a 6, 12, 24, 18, 115, and 230 volts design are the most common. Today designs offer a number of general purpose relays that require as little as 4milliamp at 12v DC making them IC compatible to TTL and CMOS logic design (gates). These relays are available in a volt range of switching configuration (Ghoshal 1997; Craig 1998; Computerized meter 2008; Intersil 2008). Basically the function of the relay in this project is to switch on the load whenever credit is loaded and to switch off the load when the credit is finished.

Output Point

This is where the load is connected as shown in Fig. 8.

System Operation

The printer port must be set to bi-directional on the BIOS setup of the computer system and software program installed. When a user wants to recharge, he buys a card, scratches the credit and get the pin numbers in the card. He then key in the number to a specified area in the user interface and then click on “load”. The computer will check for the validity of the card and then display the credit wattage bought. As the user plugs in his load, the computer system will display the consumed power, the credit balance and the rate of sale of power. If the credit gets finished, computer switches off power to the user’s load until he loads in credit again (Craig 1998).

Below are the PIN numbers programmed for this device:

Charging rate-100W
111-N5/sec
112-N10/sec
111-896100q-N200
111-324865e-N300
111-432865a4-N500
111-8326984f-N1000
112-896100q-N200
112-324865e-N300
112-432865a4-N500
112-8326984f-N1000

Discussion

The hardware and software implementation is presented with respect to the systems functionality, respectively.

Hardware Implementation

The project made use of various discrete electronic components. Each of these components was powered with +5V. The LEDs were mounted at each testing point to test the required output at each point to ensure proper functionality. These tests were carried out for most of the components on the dial-up module. The electronic components utilized in the construction comprised of dual-in-line packaged ICs.

The components were mounted on two breadboards joined together. The mounting procedures followed the real implementation diagram of the circuit and not the logic diagrams. For every component added, the source (5V) and ground was first connected before making any further connection. Also, LEDs were connected at strategic points to test/indicate the output. Logic systems are tested, debugged and serviced by putting them into operation and subsequently comparing the results obtained against the supposed result. The testing of the breadboard circuit was done on a-one-step-at-a-time basis. This was done by sending a clock signal from the PC to the connected modules and observing the output of the LEDs and then comparing them with the supposed result. This was done for the entire circuit (Ghoshal 1997).

The complete circuit diagram is shown in Fig. 9, the pictorial view is shown in Fig. 10, and the inside view is shown in Fig. 11.

Software Implementation

Software packages are a useful tool for implementing automated systems. They supports the performance by providing established application functionality based on a predefined operation sequence. The packaged software designed and built for this application is such that can best fit the needs of the system functionality. The software implementation is done by outlining an initial performance definition to the level required. The implemented software when integrated with the hardware serves as a driver to it. The software was tested with the hardware secured on the breadboard. On the Vero board the various components (ICs) were soldered with IC sockets as a means of precaution and protection against erratic performance. The Vero board implementation now provided a permanent connection for the temporary circuit on the breadboard. This increased the reliability of the system.

Circuit Diagram

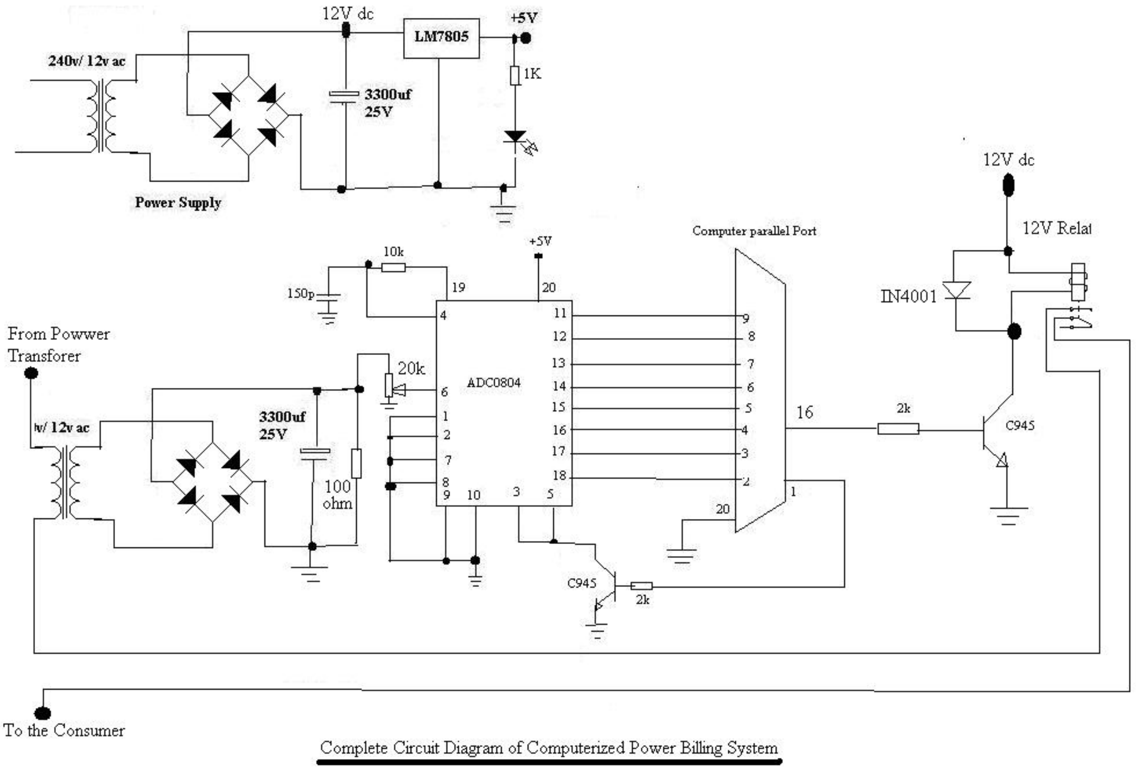

Fig. 9. Complete circuit diagram of computerized power billing system.

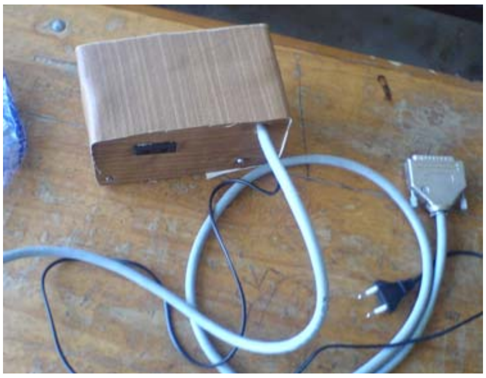

Fig. 10. Pictorial view of the device.

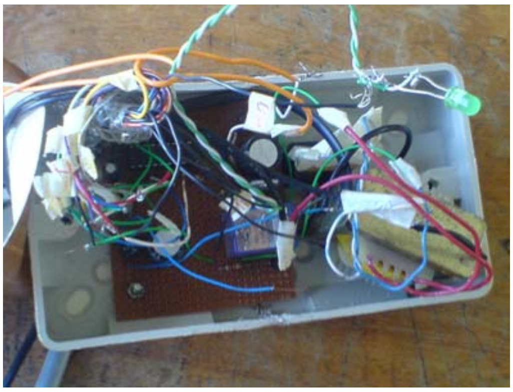

Fig. 11. Inside view of the device.

Recommendation

This project work serves as a model which can be adapted for a number of devices that needs to be controlled. It is therefore recommended that sensors be utilized to monitor changes in the environment which can trigger off certain programmed responses. High speed components of high reliability can be utilized to reduce the components delay to ensure correct synchronization of commands and clock signals. It will also enhance effective handshake with the computer's frequency and speed. Finally, solar power supply system is also recommended to serve as a back up to the main supply system for real time applications.

Conclusion

The design and construction of a computer based billing system with a software user interface, signal conditioning and data acquisition hardware has been implemented. Although a model, it can be developed and adapted to present use. Seamlessly integrating this setup to a typical building or establishment involves a careful analysis and planning of the electrical wiring and lighting infrastructure. There is great potential for this system to be used in numerous industries.

References

Axelson, J. 1996. Parallel port complete. Lakeview Research, Madison, WI, USA.

Byron, G.S. 2002. Shaum's outline of theory and problems of programming with Visual Basic. McGraw Hill, New York, NY, USA.

Computerized meter, 2008. Available: http://www.computerizedmeter.com/html/amrc.htm

Craig, P.1998. Interfacing the PC. Available: http://www. senet.com.au/vcpeakock

IEEE. 2008. Handbook of electricity metering. Available: www.ieeeexplore.com

Hall, D.V. 1992. Microprocessors and interfacing - programming and hardware. 2nd ed., Macmillan McGraw- Hill, Singapore.

Intersil. 2008. Available: http://www.intersil.com/data/fn/fn3094.pdf

Ghoshal, K. 1997. Distribution automation: SCADA integration is key. IEEE Computer Applications in Power, pp. 31-5.

Millman, J. 1979. Microelectronics: digital and analog circuits and systems. McGraw-Hill, Singapore, pp 215-6.

Seshanna, P.; Nashtara, I.; and Sajeed, A.H. 2006. Automated industrial load measurement system. AU J.T. 10(1): 23-8.

Appendix

```
[Bootstrap]
SetupTitle=Install
SetupText=Copying Files, please stand by.
CabFile=Billing setup.CAB
Spawn=Setup1.exe
Uninstal=st6unst.exe
TmpDir=msftqws.pdw
Cabs=1

[Bootstrap Files]
File1=@VB6STKIT.DLL,$(WinSysPathSysFile),,,6/18/98 12:00:00 AM,102912,6.0.81.69
File2=@COMCAT.DLL,$(WinSysPathSysFile),$(DLLSelfRegister),,5/31/98 12:00:00 AM,22288,4.71.1460.1
File3=@stdole2.tlb,$(WinSysPathSysFile),$(TLBRegister),,8/4/04 1:00:00 PM,17920,3.50.5014.0
File4=@asycfilt.dll,$(WinSysPathSysFile),,,8/4/04 1:00:00 PM,65024,5.1.2600.2180
File5=@olepro32.dll,$(WinSysPathSysFile),$(DLLSelfRegister),,8/4/04 1:00:00 PM,83456,5.1.2600.2180
File6=@oleaut32.dll,$(WinSysPathSysFile),$(DLLSelfRegister),,8/4/04 1:00:00 PM,553472,5.1.2600.2180
File7=@msvbvm60.dll,$(WinSysPathSysFile),$(DLLSelfRegister),,8/4/04 1:00:00 PM,1392671,6.0.96.90

[IconGroups]
Group0=Computerized Power Billing Machine
PrivateGroup0=True
Parent0=$(Programs)

[Computerized Power Billing Machine]
Icon1=""Billing setup.exe""
Title1=Computerized Power Billing Machine
StartIn1=$(AppPath)

[Setup]
Title=Computerized Power Billing Machine
DefaultDir=$(ProgramFiles)\Project1
AppExe=Billing setup.exe
AppToUninstall=Billing setup.exe

[Setup1 Files]
File1=@INPOUT32.DLL,$(WinSysPath),,$(Shared),7/22/03 11:04:08 PM,32768,1.0.0.1
File2=@Billing setup.exe,$(AppPath),,,11/15/07 9:34:22 AM,40960,1.0.0.0
```