2006-2307: DESIGNING, BUILDING, AND TESTING AN ADVANCED INDUSTRIAL-GRADE THREE-PHASE DIGITAL POWER METER

Bobbie Meredith, Middle Tennessee State University

BOBBIE JO MEREDITH, Mrs. Meredith is currently a graduate student, Computer Engineering Technology major in the Department of Engineering Technology and Industrial Studies at Middle Tennessee State University. She earned her BS degree in Computer Engineering Technology with minor in Computer Science from MTSU in 2005. She is currently working as an Electronics Test Engineer for Schneider Electric, Power Monitoring and Control Division. Her interests include the design, building, programming, and testing of microprocessor and microcontroller based systems.

Saleh Sbenaty, Middle Tennessee State University

SALEH M. SBENATY, Dr. Sbenaty is currently a Professor of Engineering Technology at Middle Tennessee State University. He received the BS degree in EE from Damascus University and the MS and Ph.D. degrees in EE from Tennessee Technological University. He is actively engaged in curriculum development for technology education. He has written and co-authored several industry-based case studies. He is also conducting research in the area of mass spectrometry, power electronics, lasers, and instrumentation.

Jason Thurmond, Schneider Electric, Power Monitoring and Control

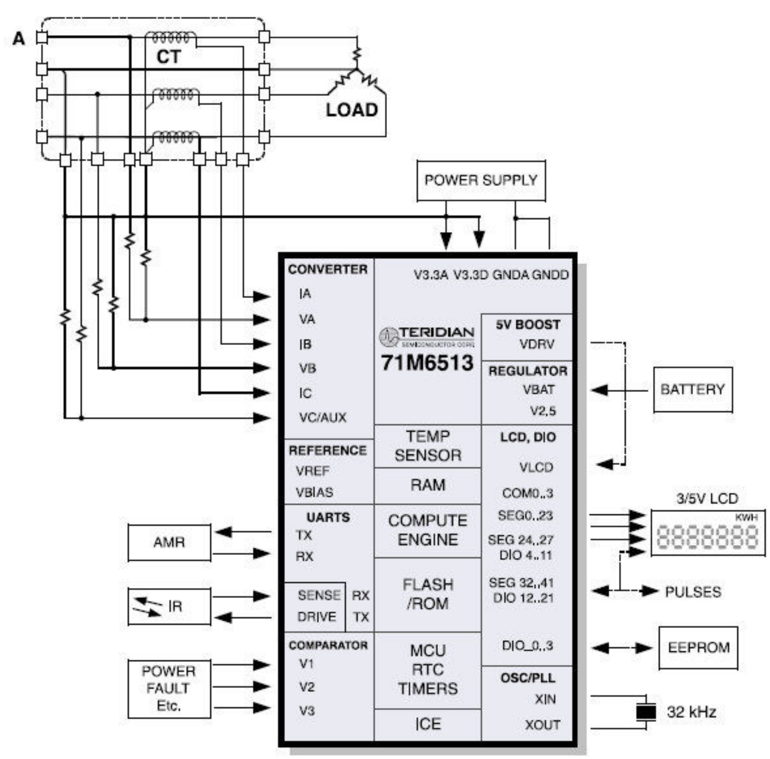
JASON THURMOND, With more than a decade of engineering and management experience, Jason Thurmond has developed a key sense of what it takes to develop Power Quality Analyzers and Power Metering equipment. Jason presently serves as Engineering Manager for Schneider Electric's Power Monitoring and Control organization. He holds a Bachelors of Science in Computer Engineering from Middle Tennessee State University. He has been involved in evaluation and implementation studies for several IEC Power Quality and Safety standards and has multiple patents related to Power Monitoring devices. He is an official member of IEC TC85.

Designing, Building, and Testing an Advanced Industrial-Grade Three-Phase Digital Power Meter

I. Abstract

The current paper describes the design, construction, and testing of an advanced digital three-phase power meter for industrial applications. The project is the result of a very close collaboration between the author, a senior Computer Engineering Technology major, her faculty advisor at Middle Tennessee State University (MTSU), and the design engineers at the Power Logic Group of Square-D, a Division of Schneider Electric in LaVergne, Tennessee. At the time of writing this paper, the author was completing her Internship at Square-D, where she was gaining the hands-on experience necessary for a successful employment after graduation and learning state-of the-art and real-world applications. The meter initial requirements include the measurements of AC voltages in the range from 0 to 55 V_{RMS} and currents in the range from 0 to10 A per phase. In addition, the meter should be capable of measuring power (real, reactive, and apparent) and energy consumption per phase and total. The meter should also have an output pulse at the TTL level, with a set pulse weight. The output is for the total kW.hr and kVAR.hr. The meter also takes four-pulse inputs (from devices developed by Square-D), the result of converting 18-36 V to the TTL level, and display the four inputs on the onboard display. The above-mentioned data should be displayed on a seven-segment LCD display. The design, building, and testing of the above-specified power meter is presented. The research also reports on the ease of use, development tool quality, code efficiency, and other significant characteristics of the TDK development board. Discussions of the results and the benefits of this collaborative project are also presented.

71M6513 / 71M6513H Block Diagram



II. Introduction

The research project is built around the Teridian 71M6513 Power Metering chip. The project involved working with a TDK Semiconductor Corporation (TSC) 71M6513/71M6513H Demo Board.

The Teridian 71M6513 is a highly integrated System On a Chip, SOC, with an MCU core, RTC, FLASH and LCD driver. Teridian's Single Converter Technology™ with a 21bit 2nd order delta-sigma ADC, 6 analog inputs, digital temperature compensation, precision voltage reference and 32 bit computation engine supports a wide range of polyphase metering applications with very few low cost external components. A 32 kHz crystal time base for the entire system and internal battery backup support for RAM and RTC further reduce system cost.

The Teridian 71M6513 is mounted on a TDK Semiconductor Corporation (TSC) 71M6513/71M6513H Demo Board that is used for evaluating the 71M6513/71M6513H device for 3-phase electronic power metering applications. It incorporates a 71M6513 or 71M6513H integrated circuit, peripheral circuitry such as a serial EEPROM, emulator port, and on-board power supply as well as a companion Debugging Board that allows a connection to a PC through a RS232 port. The demo board allows the evaluation of the 71M6513 or 71M6513H power meter controller chip for measurement accuracy and overall system use.

The demo board is pre-loaded with codes containing a low level API library exerciser (LAPIE – 6513_demo.hex) in the FLASH memory of the 71M6513/71M6513H IC. This embedded application is developed to exercise all low-level API library calls to directly manage the peripherals, flash programming, and CPU management (clock, timing, power savings, etc.).

Below are the main components of this project. Figure 1 shows (from right) the demo board and the debugging board.

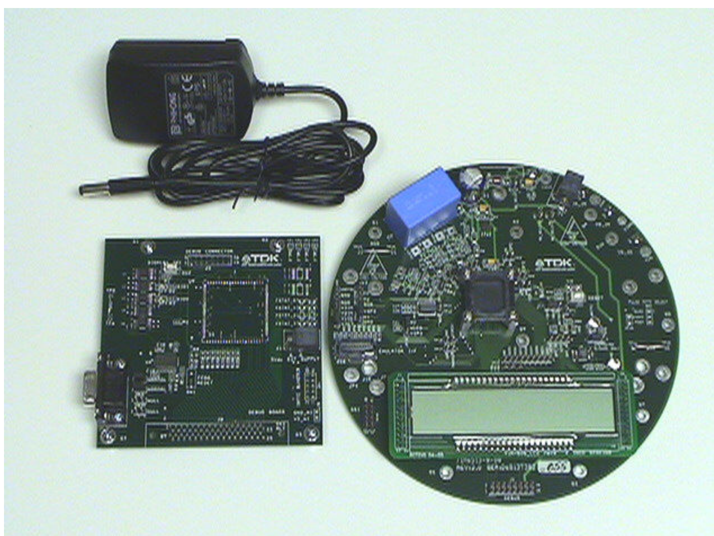


Figure 1. TDK Demo and Debugging Boards.

III. Methodology

Work on this project began by reading and understanding the literature provided along with the TDK development board. This included valuable information on the 71M6513 chip, the demo board, and on the software. From reviewing the functionality of the board and its components, it was found that the pre-loaded demo code performed some of the specified requirements while code for the remaining requirements would have to be developed using the Keil Software Environment. There were four major areas of work and study. These included:

1. Familiarization with the Keil Software Environment.
2. Familiarization with the original demo code.
3. Obtaining a flash download board and altering the original demo code to fit the specific requirements.
4. Energy Pulse Input specifications and requirements.

With the debugging board, a computer with a hyper-terminal, and the demo code pre-loaded on the board, one could look at a number of values including energy per phase and total, RMS voltages and currents, date and time, and operating hours. In order to alter this code and add the remaining functions it was necessary to obtain the Keil Compiler and uVision2 environment, as well as a flash download board from the Teridian Semiconductor Company.

1. The Keil Software Environment

The Keil Software Company has a whole range of development tools for the Intel 8051 processor family. The kit that was used to develop codes for this project comes with a compiler, linker, and assembler. The development environment is the uVision2 IDE, or Integrated Development Environment, version 2.40 for Windows. This IDE combines project management, source code editing, and program debugging. With the tools provided one can compile C code, assemble files, link and locate modules and libraries, create HEX files, and debug code. This software can be used with almost any derivative of the 8051 Microprocessor including the embedded 8051 in the Teridian 71M6513. Figure 2 below gives and example of what the uVision2 IDE looks like during development.

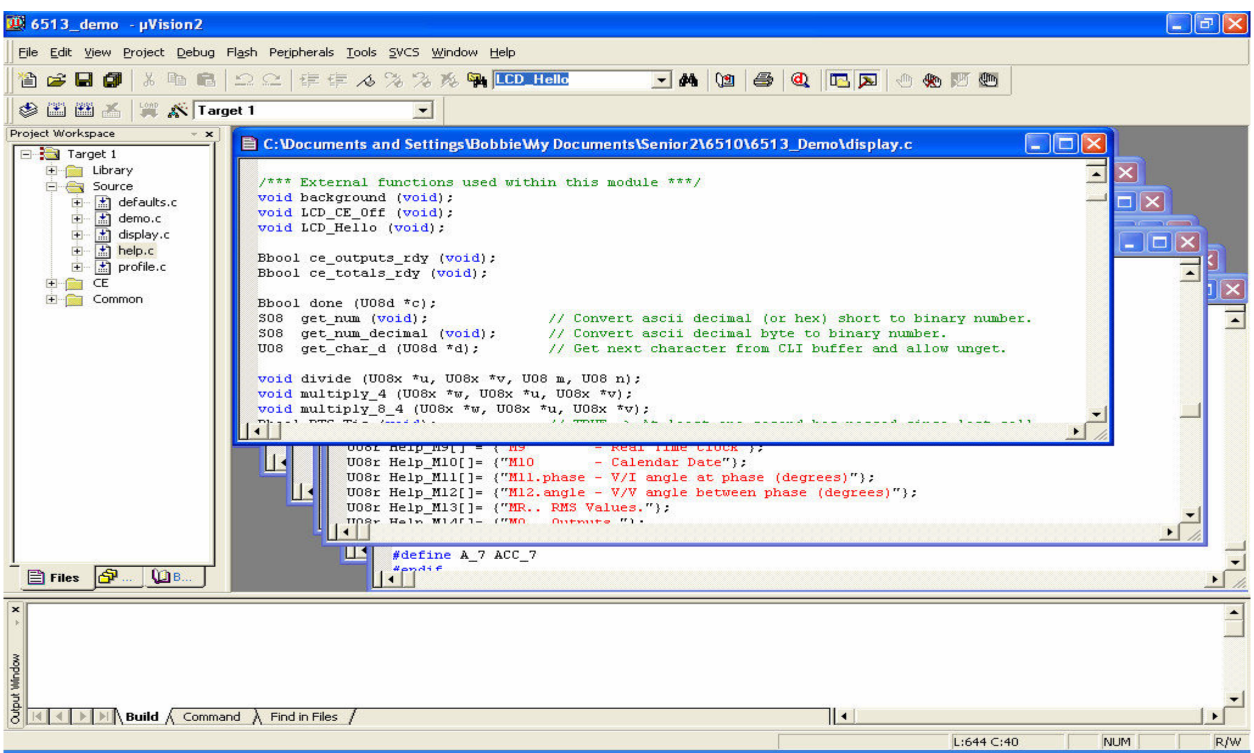


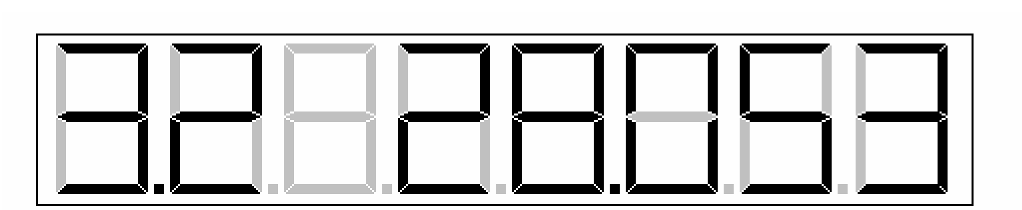
Figure 2. The Keil Software Environment.

2. Provided Demo Code

The provided demo code performs some of the specified requirements. The demo code executes commands in two ways, both of which require the use of the debugging and demo board simultaneously. First, there is a push button on the debugging board labeled DISPLAY_SEL. With the demo code loaded into the demo board, providing that both the target and the debugging board are powered, this button scrolls through the following values:

1. Temperature in Delta
2. Frequency
3. W.h Total Consumption (in Watt.Hr)
4. W.h Total Inverse Consumption (in Watt.Hr)
5. VAR.h Total Consumption (in Volt Ampere Reactive.Hr)
6. VAR.h Total Inverse Consumption (in Volt Ampere Reactive.Hr)
7. VA.h Total Consumption (in Volt Ampere.Hr)
8. Operating Hours
9. Time
10. Date

The second way that the debugging board executes commands from the demo code is through a Command Line Interface or CLI. The commands are sent using a hyper terminal setup to communicate through the serial port of the CPU and the debugging board. The demo code then executes these commands. One of the commands in the demo code is to display different metered values depending on the designated value and phase selected. The value types are selected by a whole number. For example to view Total W.h Consumption one would select M3 (M is the display command). At this point, the screen displays the total W.h Consumption until another command is issued. To read different phases the user would enter the desired value and then a decimal and then the desired phase. For example, M3.0 would be for sum, M3.1 would be for phase A, M3.2 would be for phase B, and M3.3 would be for phase C. For example the display below shows the number and then the value in 6 digits with 3 of those at the right the decimal point.



The above indicates W.h Consumption on Phase B is 28.053 W.h. Figure 3 is the section of code that prints the help menu in the hyper-terminal. It shows which whole numbers are aligned with the indicated value types.

```
// Meter Controls.
U08r Help_M[]  = {"M       - Meter Totals Display"};
U08r Help_M0[] = {"M0      - No display update"};
U08r Help_M1[] = {"M1      - Temperature (degrees delta)"};
U08r Help_M2[] = {"M2      - Frequency (degrees) "};
U08r Help_M3[] = {"M3.phase  - Wh Total Consumption"};
U08r Help_M4[] = {"M4.phase  - Wh Total Inverse Consumption"};
U08r Help_M5[] = {"M5.phase  - VARh Total Consumption"};
U08r Help_M6[] = {"M6.phase  - VARh Total Inverse Consumption"};
U08r Help_M7[] = {"M7.phase  - VAh Total"};
U08r Help_M8[] = {"M8        - Operating Hours"};
U08r Help_M9[] = {"M9        - Real Time Clock"};
U08r Help_M10[]= {"M10       - Calendar Date"};
U08r Help_M11[]= {"M11.phase - V/I angle at phase (degrees)"};
U08r Help_M12[]= {"M12.angle - V/V angle between phase (degrees)"};
U08r Help_M13[]= {"MR.. RMS Values."};
U08r Help_M14[]= {"MO.. Outputs."};
U08r Help_M15[]= {""};
U08r Help_M16[]= {"Phase: Sum = 0; A = 1; B = 2; C = 3"};
U08r Help_M17[]= {"Angle: A:B = 1; A:C = 2"};

U08r * code MHelp[] = {
    Help_M,  Help_M0, Help_M1, Help_M2, Help_M3,
    Help_M4, Help_M5, Help_M6, Help_M7, Help_M8,
    Help_M9, Help_M10, Help_M11, Help_M12, Help_M13,
    /* Help_M14, */ Help_M15, Help_M16, NULL
```

Figure 3. Print Code.

3. Code Changes

While the demo code was used as a base for this meter project, it did not satisfy all requirements. The code was altered so that the values began at the whole number 1 with Volts RMS and ended with the Calendar date at whole number 12. The complete list is as follows:

1.Phase	VRMS
2.Phase	IRMS
3.Phase	W
4.Phase	VAR
5.Phase	VA
6.Phase	Wh
7.Phase	VARh
8.Phase	VARh
9.input#	Wh
10.	Operating Hours
11.	RTC
12.	Calendar

Values from 1 to 8 where values that could have associated phase values, i.e. (.1) for phase A. Value 9 was designated for the energy pulse input, so 9.0 would display the information for input

1 and so on to input 4. The display used was only a seven-segment LCD display and prevented the display of actual words and letters to describe the values. For voltage values, a display of V was used (U on the seven-segment display), for current, I was displayed, and for Power and Energy, P and E were displayed. These indicator letters were placed before the whole number values on 1 through 8. The phases or total were shown after the whole number by either a t, A, B, or C. For the inputs, I n 1 was displayed and the operating hours and time are designated by OPH and t.

4. Energy Pulse Input

A major objective in this project was creating an energy pulse input to be read, calculated, and displayed by the metering chip. The energy pulse was to be brought in from a Square D power meter in the PM7 family. This family of power meters sends out an 18-39 V energy pulse signal with programmable pulse weights. The required rate of scan was to be 25 Hz. In order to read this value from the 71M6513, the voltage and current of the pulse had to be stepped down to a TTL level. This was accomplished using the circuit shown in Figure 4.

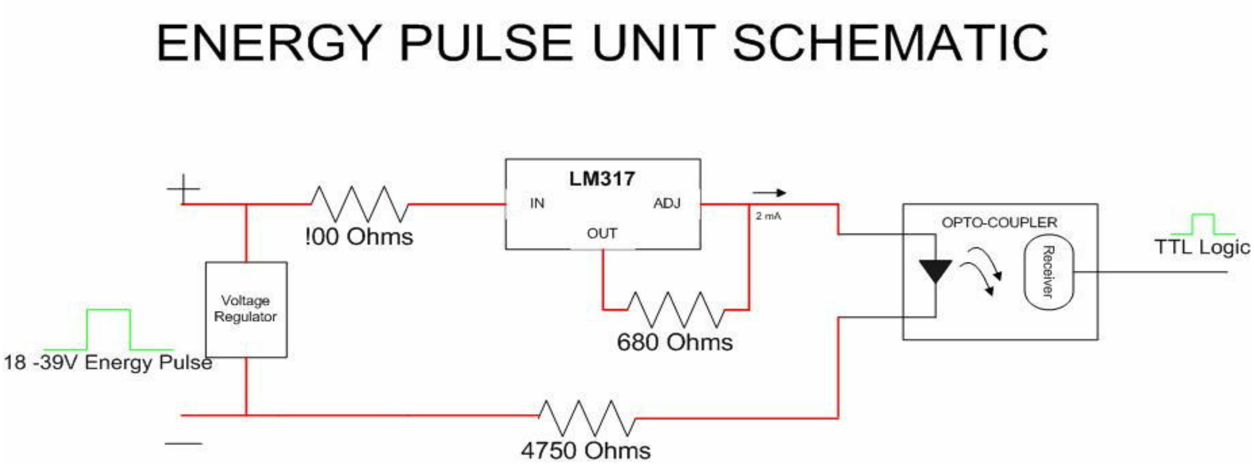


Figure 4. Energy Pulse Circuit.

In this circuit the 39 V Transient Voltage Suppressor or TVS chops the incoming voltage at 39 volts so that the circuit does not become overloaded. The LM317 and resistors bring the current down to 2mA so that it can trigger the opto-coupler without burning it. When a high signal is read in on the left side of the opto-coupler, light is emitted and then received by the diode in the right side of the opto-coupler. The single line TTL signal is then received from the right side. This opto-coupler is necessary to prevent damage to the demo board in the event that another component is damaged or faulty.

IV. Results

The Teridian 71M6513 Power Metering chip along with the demonstration board worked very well for this research project. However, when it came to actually changing the coding of the board and adding external inputs, the demo system was hard to change. These can be highlighted by the following:

1. Necessary Additions

Most of the set backs and roadblocks in this project came in gathering the necessary tools and components to create original code. The Keil Programming Kit was obtained from MTSU. This provided the necessary software tool; however, the 71M6513 Power Metering chip must have either a download board or an emulator to accept new codes. The data sheets for the 71M6513 Power Metering chip state that the flash-download board for the meter can only install updates to the code on the board or load code to an empty board.

The ADM51 is an In Circuit Emulator or ICE, which was designed to program and debug the 71M6513 Power Metering chip. This emulator is much more expensive and time was spent trying to find an ADM51 at a better price. Nearly 3 weeks into researching the prices and necessity of this emulator, Teridian Tech Support indicated that the Flash Download board, at a more economical price, could program the 71M6513. This download board does not provide any debugging capabilities but simply loads the code into the flash memory. After receiving this Flash Download Board, the demo code was altered and downloaded back to the board.

2. Code Problems

The demo code that was provided with the board was not very susceptible to small changes. Most of the display functions and display mode setups had to be rewritten in order for the meter to function as desired. It is desired that the meter display the values in a cycle, continuously. The way that the code is written, there are two modes, CLI and IDLE. The code bounces back and forth between these two modes checking to see if one is active. The code will not go into the display portion unless the meter receives the CLI command or the scroll command from the push button. This inhibited me from getting the meter to display what I wanted it to.

Another problem that arose in the display was that the onboard display is a seven-segment LCD display. Therefore, only numerical characters can be displayed with some reference to alpha characters, i.e. P or E can be represented properly with the seven-segments. In future work on this project it is hoped that a character display will be added so that one can display “Energy: W.h xxx.xxx” and so on.

3. Digital I/O Pins

With the energy pulse input requirements, it is necessary to have 4 digital inputs to bring pulses into the board. The 71M6513 Power Metering chip has 22 digital I/O pins, which can be configured for either I/O or LCD control. Depending on the type of LCD used, a different number of I/O can be required to run it. With the seven-segment LCD display provided, there were 15 I/O pins being used by the display. The first I/O pin was labeled DIO21, then DIO20, and down to DIO0. This meant that DIO21 through DIO7 were being used on this demo board for LCD control leaving only DIO6 through DIO0 for digital inputs and outputs. The only DIO that can be connected to the board for use are DIO21 through DIO16. However, these were used by the LCD and so cannot be used for the digital inputs as desired. There are ways to work around this problem with more time, including multiplexing the inputs to these pins or speaking with a Teridian representative about other options for DIO.

V. Conclusions and Discussions

From the above research, the following conclusions can be drawn:

1. Ease of Use

The Teridian 71M6513 Power Metering chip and demonstration board is simple to use. With the correct equipment and knowledge, this demo board can be converted into a working multi-use meter.

2. Development Tool Quality

With the debugging board required to send commands and the download board required to load and change code, this can be troublesome to use for developing a power/energy meter. The parts required to develop and build this power meter were not very flexible. There was only one company, Signum Systems, which sold either the Flash Download Board or the ADM 51 ICE. The Keil Compiler was the most complete set of software for this Microprocessor derivative, and was the most recommended.

3. Code Efficiency

The code was efficient in the demo code applications. When trying to change the code from what the demo program did to what was needed, there were many problems in setting up the modes and counters. When altering the contents of a function, by perhaps adding a for-loop or while-loop, the timing of the outputs was also altered. There were few comments in the demo code to explain what might be happening in any of the functions, so when there was strange behavior there was no way to tell what a function was supposed to do.

4. Benefits to the Student, Faculty, and the Hosting Company

This research project and the internship experience have greatly benefited the senior undergraduate student by exposing her to real-world applications to what she studied in the classroom. Working with various engineers at Square D and communicating with the technical support staff from TDK and Teridian have increased her knowledgebase, communication skills, and hands-on experience. The faculty advisor also benefited from this joint project by collaborating with Square D staff, a leader company in power control and measurement. Square D also benefited at the end by evaluating this new board for their use, preparing students for the work place, and finally by actually hiring the intern who worked on this project and who graduated last December and started a full-time job there as an electronic test engineer.

V. References

1. "71M6511/71M6511H and 75M6513/75M6513H Power Meter IC Family: Software User's Guide" Teridian Semiconductor Corporation. 25, October 2005.
2. "75M6513/75M6513H 3-Phase Energy Meter IC: Data Sheet." Teridian Semiconductor Corporation. September 2005
3. "75m6513/75m6513H Demo Board: User's Manual." Teridian Semiconductor Corporation. October 2005.
4. "Meters." Teridian Semiconductor Corporation. Updated: 2005. http://www.tdksemiconductor.com/products/product_category_detail.cfm?category_key=27

Appendix A. Project Budget

Project Budget

Expenses	
TDK 71M6513 Demo Board with Debugging Board and Software	-$500.00
Flash Download Board	-$200.00
Keil Compiler Kit	-$900.00
Presentation Materials	-$50.00
Digital Input Parts	-$30.00
	-$1,680.00
Credits	
Demo Board From Square D	$500.00
Flash Download Board From TDK Tech Support	$200.00
Keil Compiler Kit from MTSU	$3,000.00
	$3,700.00
Future Expenses	
ADM51 ICE	-$1,200.00