# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>TABLE OF CONTENTS</td>
<td>2</td>
</tr>
<tr>
<td>ABSTRACT</td>
<td>4</td>
</tr>
<tr>
<td>INTRODUCTION</td>
<td>5</td>
</tr>
<tr>
<td>NARRATIVE</td>
<td>6</td>
</tr>
<tr>
<td>PROCEDURE</td>
<td>7</td>
</tr>
<tr>
<td>PROGRAM MODULES</td>
<td>8</td>
</tr>
<tr>
<td>ADDMETER</td>
<td>8</td>
</tr>
<tr>
<td>CHECKMETER</td>
<td>8</td>
</tr>
<tr>
<td>CHECKREADS</td>
<td>9</td>
</tr>
<tr>
<td>READBarcode</td>
<td>9</td>
</tr>
<tr>
<td>INITIALIZE</td>
<td>9</td>
</tr>
<tr>
<td>INITDIALOG</td>
<td>9</td>
</tr>
<tr>
<td>INITFUELS</td>
<td>10</td>
</tr>
<tr>
<td>INITGoal</td>
<td>10</td>
</tr>
<tr>
<td>DoCALCULATIONS</td>
<td>10</td>
</tr>
<tr>
<td>CALCULATIONS</td>
<td>11</td>
</tr>
<tr>
<td>AddROUTINE</td>
<td>12</td>
</tr>
<tr>
<td>AddFUEL</td>
<td>12</td>
</tr>
<tr>
<td>Download</td>
<td>13</td>
</tr>
<tr>
<td>READSOUT</td>
<td>14</td>
</tr>
<tr>
<td>HEADINGS1</td>
<td>14</td>
</tr>
<tr>
<td>HEADINGS2</td>
<td>14</td>
</tr>
<tr>
<td>PURGEElect</td>
<td>14</td>
</tr>
<tr>
<td>PURGESteam</td>
<td>15</td>
</tr>
<tr>
<td>PURGEGas</td>
<td>15</td>
</tr>
<tr>
<td>PURGereadings</td>
<td>15</td>
</tr>
<tr>
<td>PURGEmeters</td>
<td>15</td>
</tr>
<tr>
<td>PURGEfueLS</td>
<td>15</td>
</tr>
<tr>
<td>RESULTS AND CONCLUSIONS</td>
<td>16</td>
</tr>
<tr>
<td>APPENDIX A STRUCTURE CHART</td>
<td>17</td>
</tr>
<tr>
<td>APPENDIX B FLOWCHARTS</td>
<td>20</td>
</tr>
<tr>
<td>APPENDIX C PROGRAM CODE</td>
<td>49</td>
</tr>
<tr>
<td>APPENDIX D LIST OF VARIABLES</td>
<td>81</td>
</tr>
<tr>
<td>APPENDIX E</td>
<td>87</td>
</tr>
</tbody>
</table>
ABSTRACT

A portable Laser-Wand bar-code scanner was programmed using Universal Data Language (UDL) to read bar-coded identification characters of labels on electrical, steam, and gas meters, accept reading input, and perform calculations based on the input with minimal action from the user. The user interacts with the program mainly by making selections from lists of choices available in the menus displayed and by answering simple questions. The first time the program is placed in operation the user identifies all meters to the Laser-Wand unit. Thereafter, all the user is required to do is to input the readings shown on the meter. After all readings have been input, the user can activate the procedure to perform calculations by making a simple menu selection. All program procedures can be activated directly or indirectly from the menus displayed. Calculations and readings accumulated in the laser scanner unit can be downloaded to an ASCII file via RS 232 cable connection between an IBM-compatible personal computer and the Laser-Wand home-base unit.
INTRODUCTION

The Facilities Engineering Division at the Corpus Christi Army Depot oversees the Energy Program for the installation through an Energy Officer. Among the duties of the Energy Officer is a meter-reading-survey each month of all the electrical, steam, and gas meters placed in the field throughout the installation. Currently, the Army Energy Officer, accompanied by the Naval Air Station Energy Officer¹, walks to each meter and takes readings manually. The readings then form the basis the Navy uses for billing the Corpus Christi Army Depot for the cost of utilities used during the month.

Currently, the Army Energy Officer inputs the meter readings taken in the field into a spreadsheet. The spreadsheet is programmed to perform calculations to convert the readings which are in kilowatt-hours, cubic feet, or pounds into Billion British Thermal Units (BBTU’s). These calculations form the basis of a report submitted to higher management listing the current monthly energy used, the cumulative energy used to date for each type of utility each month, and the percent used to date in comparison to the goal set at the beginning of the fiscal year.

¹ The Naval Air Station is the installation landlord and the provider of utility services to the Corpus Christi Army Depot (CCAD). CCAD pays all utility bills directly to the Navy.
NARRATIVE

The Laser-Wand bar-code reader was programmed to automate and facilitate the data collection, calculations, and reporting requirements of the energy program through the bar-coding of all utility meters. A self-adhesive label consisting of a bar-coded meter number was designed to identify each meter at the installation. A sample of the label design is enclosed in Appendix E. The label will be posted next to the meter where it will be accessible for reading by the laser bar-code scanner.

The program developed consists of easy-to-follow sets of menus implemented to minimize keystroke input. The program allows the user to do the following:

- Add a meter record to a meter file
- Read a meter bar-code
- Perform calculations
- Purge system files
- Initialize meter records
- Format and download meter readings to an ASCII file in a personal computer
PROCEDURE

The Energy Management System was developed using Universal Data Language version 3.5, and a Laser-Wand programmable, hand-held, bar-code reader with 640 Kb of RAM, an alphanumeric keyboard, and 32 character display. Both the language used to program the unit and the bar-code reader are products of Hand Held Products Inc., Charlotte, North Carolina.

Software to support the development of applications using the language was loaded into an IBM compatible personal computer. An editor, a linker, a compiler, and a communication interface are provided for the language to facilitate program development. The editor provided was used to develop the Energy Management System. After the assembling, compilation, and linking of the main program and sub-programs, the machine language version was loaded using the communications interface environment provided by the language into the Laser-Wand bar-code reader via an RS232 cable connected to the communications serial port of the personal computer and the home-base unit into which the Laser Wand is plugged.

The bar-coded labels included in Appendix E were developed for this project using software and printing equipment from Intermec Corporation.
PROGRAM MODULES

The Energy Management System consists of the following procedures:

AddMeter.

This procedure is used once, when a meter is first identified to the Laser Wand. The user is prompted to enter the following information to identify each meter:

- meter bar-code
- meter type
- meter multiplier
- meter initial read
- meter location

This information is stored in the Meters file. The AddMeter procedure ensures that only one record of information is stored for each meter identified to the Laser Wand. It does this validation by calling the procedure CheckMeter. The CheckMeter procedure reads each record in the Meters file and makes a comparison between meter numbers. If a meter with the same bar-code is found, it informs the user by displaying a message. It does not allow the user to add more than one meter with the same identification number.

CheckMeter.

This procedure is used to prevent the input of a meter record more than once. The procedure is used to validate the existence of a meter before a reading is accepted as valid. This procedure is called by the AddMeter and the ReadBarcode procedures.
CheckReads.

This procedure determines whether more than one meter reading for a meter has been entered. This procedure is called each time the user tries to input a meter reading. The files where the readings are stored are checked for the presence of a previous reading. If a reading exists, the previous reading is overwritten with the new reading, and a message is displayed informing the user that a previous meter reading was replaced with the new entry.

ReadBarcode.

This procedure prompts the user to read a bar-coded number from the label placed next to each meter. The procedure calls the CheckMeter procedure to validate the existence of a record in the Meters file before allowing the user to input a reading. If there is a record, then the user is prompted to enter the reading. The date and the time of the reading become part of the readings record. The procedure files the readings in either the ElectricalReadsfile, SteamReadsfile, or the GasReadsfile according to information contained in the meter record.

Initialize.

This procedure allows for the input of initial values directly by the user. This step is required the first time the program is put into use. The procedure prompts the user to enter the number of the month to initialize, which would normally be the month previous to the month when actual meter readings will be taken. This procedure calls the InitDialog and the InitFuels procedures below depending on the type of energy records being initialized, i.e., electrical, steam, gas, or fuel energy.

InitDialog.

This procedure will be used the first time the program is placed in use. The procedure will ask the user to input several quantities. The purpose is to initialize the cumulative energy used up to the previous month. The user will be prompted to initialize the following fields for the month of interest:

- cumulative energy used to date in BBTU's
- percent used to date versus goal
- previous month's energy used

These values are important in keeping a running total of energy used since the beginning of the fiscal year to date.
InitFuels.

This procedure is similar to the InitDialog procedure with the exception that more fields need to be initialized for the fuel used. The user is prompted to enter a value for each of the following fields:

- cumulative fuel consumption to date in BBTU’s
- cumulative fuel consumption in 100 gallons units
- percent used to date versus goal
- previous month’s fuel consumption in BBTU’s
- previous month’s fuel consumption in 100 gallon units

These values are important in keeping a running total of fuel consumption since the beginning of the fiscal year to date.

InitGoal.

This procedure is called by the DoCalculations procedure when the cumulative goal for each month since the beginning of the fiscal year is zero. The field in the array tabla[GOAL, month] is checked to see if the contents are greater than zero. If the content of the field in the array tabla[GOAL, month] month is equal to zero, then the InitGoal procedure prompts the user to enter the annual goal for the given type of energy. The procedure divides the annual goal by twelve and assigns each month its cumulative goal starting from the month of October. Dividing the cumulative consumption at a given month by the cumulative goal for the same month established at the beginning of the fiscal year allows the Energy Officer to take appropriate measures to keep consumption in line with the goal.

DoCalculations.

This procedure is called from the main menu after all the meters have been read and the quantity of fuel consumed during the month are available to the user.

The procedure opens the files where the readings are accumulated for the month and begins the calculations. Three other files are used to store the results of calculations. These are: Electfile where the electrical usage by month since the beginning of the fiscal year are stored, Steamfile where the steam consumption by month since October are stored, and theGasfile for gas consumption. Each of these files is opened and its contents read into the two-dimensional array tabla[] prior to processing of the current month’s readings. The following data is stored in the array tabla:
<table>
<thead>
<tr>
<th></th>
<th>Oct</th>
<th>Nov</th>
<th>Dec</th>
<th>Jan</th>
<th>Feb</th>
<th>Mar</th>
<th>Apr</th>
<th>May</th>
<th>Jun</th>
<th>Jul</th>
<th>Aug</th>
<th>Sep</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cum Goal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cum Cons</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>% Goal</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Curr Cons</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1. Rows and columns of data stored in the array tabla[]

A nested repeat until loop is executed until the readings stored in each file (i.e. ElectReadfile, SteamReadfile, and GasReadfile) have been processed and transformed into Btu units. The program keeps a running total of the energy consumed during the month.

After the meter readings for each type of energy are processed, the procedure updates the tabla[] array with consumption figures for the current month, and files the information before processing the readings and consumption figures for the other utilities.

Calculations.

The following calculations are performed by this procedure.

Electrical energy consumed:

\[
KWH\ used\ this\ month = \left(\text{Current}\ reading\ in\ KWH - \text{Previous}\ month\ reading\ in\ KWH\right) \times \text{meter\ multiplier}
\]

\[
BBTU's\ used = \frac{\left(KWH\ used \times 1.01\right) \times 0.003413}{1000.0}
\]

Steam energy consumed:

\[
LBS\ of\ steam\ used = \left(\text{Current}\ reading - \text{Previous}\ reading\right) \times \text{meter\ multiplier}
\]

\[
BBTU's\ of\ steam\ used = \left[\frac{\left(LBS\ of\ steam \times 1.01\right)}{1000.0}\right] \times 0.00134
\]

Energy Management Data Collection System
for Corpus Christi Army Depot
Natural Gas energy consumed:

\[
\text{Cubic feet of gas consumed} = (\text{Current reading} - \text{Previous reading}) \times \text{meter multiplier}
\]

\[
\text{Btu's of gas consumed} = \frac{\text{Gas consumed in ft}^3 \times 0.001031}{1000.0}
\]

**AddRoutine.**

This routine is called from the DoCalculations procedure to add the current month's consumption to the previous month's consumption for each energy type. An array named `factabla[]` is also updated with the consumption figures. This array adds the month's consumption figures for each type of energy. The same type of variables are updated as in the `tabla` array with the exception that `factabla[]` is used to maintain the combined total for electrical, steam, gas, and fuel energy used in the installation boilers to generate steam.

After the `tabla[]` and `factabla[]` arrays are updated the procedure writes the information to the respective file.

**AddFuel.**

This procedure allows the user to enter the quantity of fuel consumed in the operation of equipment to test engines, fuel used in the operation of vehicles, and fuel drained from aircraft coming in for repairs that ends up used in plant boilers for the generation of steam. The first and second type of fuel are placed under the category called 'mobilities' fuels while the latter is placed under the category of 'facilities' fuels.

The procedure opens the `totsmobfuels`, `totfachfuels`, and `totfp4fuels` files. Any information contained in these files is read into the arrays `mobtabla`, `factabla`, and `fp4tabla` respectively. The following table shows the fields of information stored in the arrays `mobtabla` and `fp4tabla`.
Table 2. Rows and columns of data stored in arrays factbla[], mobjtabla[], and jpc4tabla[]

The procedure checks each array for a value greater than zero in the cell [1,10], or at the beginning of the fiscal year. If the value in the cell is zero, or if it is the beginning of the fiscal year the program requires the user to set the annual goal limit for consumption of ‘facilities’, or ‘mobilities’ fuels. The procedure divides this value by 12 and allocates an equal amount (cumulative) to each month².

For each type of fuel, the procedure transforms the quantities entered by the user into 100 gal units³ and BBTU’s. For each of the fields depicted above, the procedure adds the current consumption to the previous month’s consumption and stores the cumulative quantity in the current month in a manner similar as was described earlier with electrical, steam, and gas utilities.

Download.

This procedure called from the Laser Wand’s main menu is performed to download all meter readings and calculations stored in all files in the scanner. The procedure is performed after all meters have been read and calculations have been carried out.

To download the files from the Laser Wand, the scanner’s home base unit must first be connected to the serial port of an IBM compatible personal computer using an RS 232 communications cable. Additionally, prior to the data being sent from the scanner, the user must access UDL’s main menu in the host computer and select the command Wandcom before any transfer of data can take place. Wandcom is a language-supplied set of commands instructing the computer how to handle the stream of data received over the communication port.

The main purpose of procedure Download is to access the files resident in the Laser Wand barcode reader, format the data, and send the data over the communications cable to the host computer. Therefore, the sending and receiving of data is a process of running the Download procedure from the scanner and selecting the Wandcom command from the UDL environment.

² This is the same process as was done to enter the goals for the year in the tabla array, with the exception that new fields are required to store the 100 gallon units.

³ This is a requirement set by management.
The procedure first sets the communications port parameters between the home base and the host computer before opening and accessing data files. Data contained in files is downloaded to an ASCII file in the computer following the format coded in the procedure. The report's format was coded to conform closely with the current reporting requirements. These requirements call for the listing of consumption figures for each type of energy by month in table format.

The procedure sends formatting characters to serve as headings before opening and downloading readings from each file. The procedure *ReadsOut* is called from within to carry out the actual downloading of data to an ASCII file.

Procedures named *Headings1* and *Headings2* download the names of the month to serve as headings for the data to follow. The data consists of the current and cumulative consumption figures to date for each type of energy tracked. This data is stored in the *Elecfile*, *Steamfile*, *Gasfile*, *totmobfuels*, *totjpfuels*, and *totfacfuels* files. Each file is accessed in sequence and its contents sent via a communications port to a computer file.

### *ReadsOut*.

*ReadsOut* is a short procedure called from within the procedure *Download* to write a line of output to an ASCII file in the host computer connected to the Laser-Wand bar-code reader. The data output are the meter readings which were input last.

### *Headings1*.

*Headings1* is a short procedure called from within the procedure *Download* to write a line of output labeling each column with the names of the month from October to March.

### *Headings2*.

*Headings2* is a short procedure called from within the procedure *Download* to write a line of output labeling each column with the names of the month from April to September.

### *PurgeElect*.

*PurgeElect* is a short procedure accessed from the second menu which deletes all records maintained in the file *Elecfile*. The purging of data in this file must be accomplished once a year before the beginning of the next fiscal year.
PurgeSteam.

PurgeSteam is a short procedure accessed from the second menu which deletes all records maintained in the file Steamfile. The purging of data in this file must be accomplished once a year before the beginning of the next fiscal year.

PurgeGas.

PurgeGas is a short procedure accessed from the second menu which deletes all records maintained in the file Gasfile. The purging of data in this file must be accomplished once a year before the beginning of the next fiscal year.

PurgeReadings.

PurgeReadings is a short procedure accessed from the second menu which deletes all records maintained in the files ElectReadsfile, SteamReadsfile, and GasReadsfile. The purging of data in these files must be accomplished after the calculations are performed and the readings and calculations are downloaded to the host computer and before the next set of readings are collected the following month.

PurgeMeters.

PurgeMeters is a short procedure accessed from the second menu which deletes all records maintained in the file Meters. This procedure can be performed when the user wishes to re-establish the file by re-entering every meter record anew.

PurgeFuels.

Purge fuels is a short procedure accessed from the second menu which deletes all records maintained in the files totmofuels, totjpfuels, and totfacfuels. The purging of data in these files must be accomplished once a year before the beginning of the next fiscal year.
RESULTS AND CONCLUSIONS

The objective of this project was twofold: to facilitate the collection of meter readings and to automate the calculations performed to estimate the monthly energy consumption at the Corpus Christi Army Depot. The program developed for use with the laser scanner does both with minimal input from the user.

Implementation of this project will do away with the procedure of collecting meter readings manually using pad and pencil. All actions necessary to read, collect, validate, calculate, and store readings and data can be performed at the touch of a button or are done automatically. The only requirement for a procedure to be activated and an action to take effect is for the user to make a selection from the set of menus displayed. Prompts in the form of questions coded at key points in the program allow the user to correct a mistake by re-entering data.

An additional benefit gained by developing this project has been the knowledge and understanding acquired of laser bar-code scanners and data collection systems in general. The knowledge obtained is already proving helpful in developing other projects using bar-code technology for the tracking of helicopter parts through the repair process at the Corpus Christi Army Depot.
APPENDIX H

REFERENCES