Dec1604
Honeywell
Dr.Zambreno
Katherine Gresback/Team Leader
David Orona/Weekly Report Compiler
Jonathan Yeoh
Eric Riedl/Webmaster

dec1604@iastate.edu

Revised: 3/3/2016

Honeywell Binary LMS File Interpretation

Design Document

Contents

1 Introduction

1.1 Project statement

The primary objective of this project is to create a MATLAB mex function that processes LMS SCADA DAQ binary files into MATLAB directly. Currently, the LMS file is being converted into some kind of file first using Siemens software; the process may takes hours depending on the file's size.

1.2 Purpose

The purpose is to make the process of the conversion of binary files to Matlab files more efficient. The binary files are generated during a testing process of a plant at Honeywell. The files are then taken to a computer running Siemens software and converted to .mat files. Once in the .mat file it can be loaded into Matlab to run analysis on the file.

1.3 Goals

The goal is to write a complete MATLAB function that would take the ldsf files, and load them directly without having the LMS file to pass through the Siemens software. A more direct process will shorten the time spent on converting the files so that they are ready to analyze.

2 Deliverables

We feel a fulfilled project consists of a runnable MATLAB function with:

1) Associated C++ source codes, so they can be edited for user preferences
2) Sample output from the function based on the example LMS data the team was given (this should be identical to the sample output Honeywell has provided us)

Along with the source code and perfected process, the team also plans to demonstrate a presentation of our program. In this presentation, we plan to highlighting all of the methods utilized and overall showing that our program is truly a solution to Honeywell's time constraint issue. There more than likely will be additional deliverables as the project progresses, including details we feel need some form of concentration or details Honeywell itself wants to know more about. However, as engineers, we will react and respond accordingly.

3 Design

3.1 System specifications

- Honeywell estimated that the reduction time of the process should be around 20% or 30% and that's depending on the size of the LMS file.
- In the MATLAB function, we would be using C++ scripts to do the file conversion as well.

3.1.1 Non-functional

Note: non-functional requirements describe how the system works

- The associated C++ code provided must be able to compile and linked to the MATLAB with the MEX built-in functions; at the same time, they should be able to pass the desired output to the MATLAB function properly

3.1.2 Functional

Note:functional requirements describe what the system should do

- The MATLAB function should be able to compile and run in MATLAB background without error. The MATLAB function should also be capable of converting the LMS SCADA DAQ binary file successfully into the desired .mat files with shorter time compared to the current process time in Honeywell.
- The output of the data conversion must be display properly in a readable manner

3.2 PROPOSED DESIGN/METHOD

We are going to write a program to allow a user to call a function in Matlab and take the binary ldsf file and directly convert that into an uploadable .mat file so then further analysis of the data can begin. Right now, we had decided to study the C++ codes provided by the Honeywell to get a better understanding on what kind of input and output file we are dealing with, and what kind of functions and iterations did they used to do the data conversion. Since we had never deal with C++ language before, we would be applying a lot of trial and error analysis on our code for the early stage; and as we move on, we should be able to get used to C++ programming.

3.3 DESIGN ANALYSIS

We have not started writing the source code for this project. As a group, we each are learning C++ for this project. Once we learn more about C++, we will be able to create a design or outline of our source code.

4 Testing/Development

4.1 INTERFACE specifications

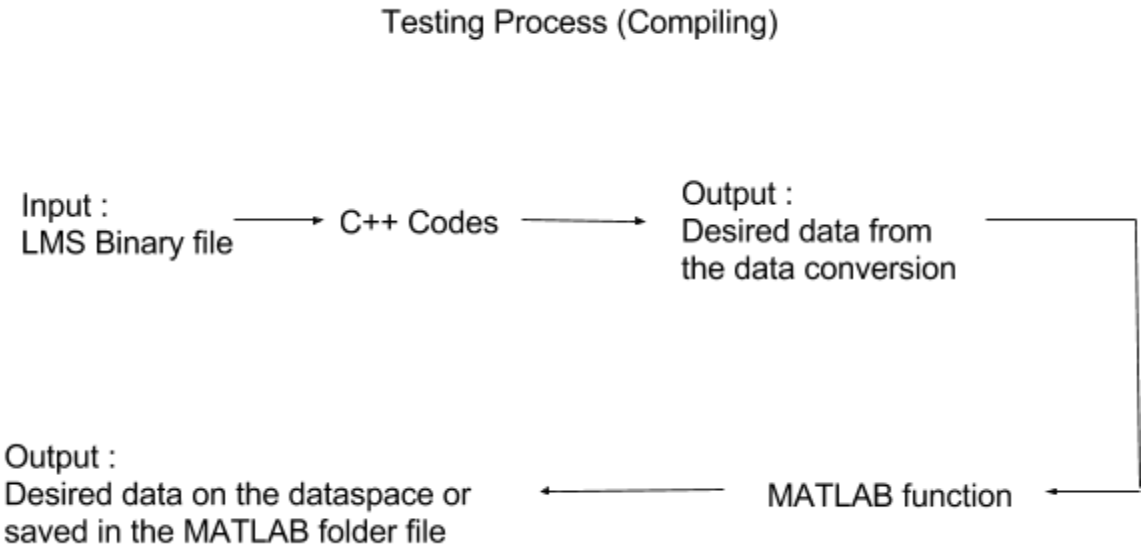
This project is solely software based. We will be integrating binary ldsf files into Matlab. We will be closely following the procedure that Honeywell currently follows. As a team, we will be transferring and converting the files provided to us. The equipment or hardware linked to where these files are being pulled from is not relevant to our task.

4.2 Hardware/software

We will be using Matlab to test our newly created function. Since Matlab is already used by our client we can assure that the expected output and ours will be the same. In the MATLAB function, we will be using the MEX functions; thus, this project does require us to use the C++ background softwares. Currently, we are using the Microsoft Visual Studio to study and compile the sample C++ script given by Honeywell.

4.2 Process

Basically, we would just use the C++ script compiler such as C++ Shell or Dev-C++ to test out our codes and get the desired output of the data conversion from LMS binary file to .txt file. After we got the C++ codes working, then we will create the MATLAB function that contain the MEX functions. The MEX functions would execute the C++ script associated for the data conversion.



5 Results

For most of the semester we ran into a number of issues with the original C++ code that Honeywell provided to us. While they were originally a major stumbling block for us we eventually worked through them with the help of Dr. Zambreno. This mostly correlated into relocating the library files into a new folder where the C++ code was able to find them, which allowed us to successfully compile the code.

Beyond compiling the code we were entirely unsuccessful, as it would crash any time we attempted to run it using the LMS input file. However, in our final week of work we held a meeting with our advisor and client, which resulted in Honeywell providing us a new, properly functioning C++ that outputs the expected text file converted from the LMS data originally supplied. The new operational code will allow us to begin creating a Dynamic Link Library based on the existing C++ code, which is the first major step towards completing our project.

6 Conclusions

We believed that we are doing well because we are progressing as the time goes on. We obtained the necessary files from Honeywell and currently trying to get the C++ files given to compile and run so that we can see how the output looks like. We are able to establish a platform for us to keep track of each team member's adjustment on the C++ code and know our project progress in developing a successful codes as a team, by using the 'source.ece.iastate.edu' website. After getting the codes working, we should be able to know why the data conversion takes so long; and by knowing that, we would come up with an effective solution in developing the MATLAB function that would load the LMS file directly into the MATLAB. So right now, we will be studying the C++ codes that is given by the Honeywell; at the same time, getting more familiar on the C++ languages.

7 References

1. Online Source
    a) www.cplusplus.com. C++ programming learning
    b) www.mathwork.com. Understanding MATLAB Mex, Mex file creation API and etc.
    c) https://source.ece.iastate.edu/projects/dec1604/. The link to the svn source website where all of our C++ codes/scripts will be uploaded and shared.

8 Appendices

If you have any large graphs, tables, or similar that does not directly pertain to the problem but helps support it, include that here. This would also be a good area to include hardware/software manuals used. May include CAD files, circuit schematics, layout etc. PCB testing issues etc. Software bugs etc.

All we have is just C++ codes and they are not compiled successfully right now. You can expected we include the codes here once we got it compile successfully in the future.