Software Demo Release Note Version 1.0

RsLogix 500 Modbus Extraction Tool

Xutong Qiu
Entegris Software Team

Release Date: July 28, 2023

1 Introduction

This document provides the release notes for the Demo release of the RsLogix 500 Modbus Extraction Tool. It outlines the details of this demo release, including supported features, user guide, and software structure. Note that in order to run the software on your local computer, a RsLogix 500 Pro must be installed and activated.

2 How to Use

Below is an example of how to use this software:

- **Click** "Load File" to load a new RSS file.
- **Click** "Display Database" to display all the addresses with their information in the original RSS file.
- **Click** "Perform Mapping" to find and apply mappings to the current database.
- **Click** "Display Database" to display all the addresses with their information in the modified database.
- **Click** "Find Invalid Mapping" to display all the invalid mapping in the RSS file.
- **Close** the software. A pop-up window shows up and asks the user whether they want to save all changes to the original RSS file. **Choose** Yes or No.

3 Supported Features

- Loading File with the format and availability check
  - The software checks whether the file is an RSS file.
  - The software checks whether the file is used by another application.
- Displaying data in a data grid that is easy to read and examine
  - A search bar is provided to help users quickly locate a data entry in the grid by typing in the address of the entry.
  - All modified data entries will be displayed in the grid to allow the users to examine the results after mapping is performed.
- Performing all one-to-one mapping
  - The software checks and identifies the logic to determine whether a mapping is one-to-one or many-to-one. It stores the logic in the corresponding data entry so that when many-to-one mapping is implemented, the corresponding logic can be easily accessed.
  - In this release, all many-to-one mapping is carried out by mapping the first address the software sees to the target address.
  - In this release, all tag names are modified to avoid duplication by adding/replacing the last character with an underscore character.
- Displaying invalid register mappings
  - The software finds all invalid register mappings(i.e. many-to-one mapping) and displays them on the data grid.

4 Software Structure

Throughout the progression of this project, the structure of this software has been continually refined and revamped. This series of modifications and redesigns aimed to achieve an architecture that is not only comprehensible but also easy to maintain.

At this stage, given the knowledge I have learned, it is believed the current structural design meets the project requirements. It has reached a point where no major redesigns are foreseen for the remainder of the project. Explained below is a brief overview of the current structure.

4.1 Form1

This is the UI that users see when they open the software. Currently, it has six buttons.

- **Load File**: loading an RSS file into the project
- **Display Database**: After a file is loaded, the database of it can be displayed using this button.
- **Perform Mapping**: Finding and applying all Modbus mapping found in the loaded file. Note that by clicking this button, the original file is not modified although users can see the resulting database. The original RSS file will only be modified if the save message window pops up and the user clicks "Yes".
- **Find Invalid Mapping**: All invalid register mapping(i.e. many-to-one mapping) will be displayed in the grid. The display of invalid coil mapping is currently not supported.
- **Load Ref Table**: This button is reserved for future use and has not been used.
- **Search**: When clicked, the software searches for the data entry with the address matching the user input typed in the text box.

4.2 PLC_DB

An instance of this class stores all data of an RSS file. It also provides functions for the outside classes to interact with the database. Within a database, each data entry is an instance of the data entry class.

4.3 IOHandler

This module handles all input and output events of the software, such as loading files, applying changes to the original RSS file, and writing to CSV files.

4.4 Parser

This module interprets PLC logic and converts them into a parse tree that can be understood by other classes of this software. Each node in the parse tree comes from a node class that stores information about an instruction block, including its arguments and subbranches.

4.5 RegLogicAnalyzer

This module analyzes the logic of a parse tree and extracts any register mapping information in it. This is done by pattern matching. In other words, this module needs to be updated if more patterns are found in future RSS files.

4.6 CusComparer

This class defines custom comparers that are used in the software to sort and compare data entries.

4.7 AddressSolver

This module parses a given address as a string and extracts or trims it based on the need to provide proper information to the caller class.

5 Feedback

Please feel free to test the software and share your feedback if you encounter any malfunctions or crashes. The process of debugging is extensive and demanding, aimed at releasing stable and robust software. Your feedback plays a crucial role in this process and is greatly appreciated.