TABLE OF CONTENTS

1 TEST PLAN IDENTIFIER

N/A

2 REFERENCES

Memo titled Automatic Door Opener Request Memo2 Revised12_07_18 given by Awesome Customer Inc.

3 INTRODUCTION

The purpose of this document is to provide a method of testing to achieve all of the specified system requirements of the Automatic Door Opener (ADO) system.

4 TEST ITEMS (FUNCTIONS)

The items to be tested are the member functions in the Motor class and Button class.

Motor Class Member Functions:

- int readFromMemory(int addr, int byteOffset, int bit)
- void writeToMemory(int addr, int byteOffset, int bit, int valueToWrite)
- bool isMotorMoving(int addr)
- int getMotorPosition(int addr)
- void openDoor(int addr)
- void closeDoor(int addr)
- void initializeMotorSpeed(int addr)
- double BinToDec(int[ ] arrayOfBits)

Button Class Member Functions:

- int readFromMemory(int addr, int byteOffset, int bit)
- bool getButtonState(int addr)

5 SOFTWARE RISK ISSUES

A critical area of the software that needs to be tested is readFromMemory function in both the Motor class and Button. This function plays a highly important role in reading critical data. Another critical area of the software that needs to be tested is the writeToMemory function in both classes. This function plays an important role in writing data to memory which commands the motors to perform tasks.

6 FEATURES TO BE TESTED

The items to be tested from the users viewpoint include the user interaction with the push button. The push button is the only means of interaction with the user so one of the main features to be tested is push button functionality and ability to determine whether the button has been pressed or released.

7 FEATURES NOT TO BE TESTED

Functions that will not need to be tested due to no system requirements include:

- Function changing the motor speed.
- System operation in the case of a power shutdown.

8 APPROACH (STRATEGY)

The testing will be done by the developer and this will be carried out through unit testing and system testing.

The developer will test each function with a list of test cases and each test should return the correct outputs.

The developer will perform system testing by executing the program to ensure that the door will perform the correct operations when the push button is triggered. These operations include opening the door when it is in the closed

position, closing the door when it is in the open position, ignoring the button press when the door is moving, opening the door fully, and closing the door fully.

9 ITEM PASS/FAIL CRITERIA

Items that must pass include

Unit tests:

- All functions must be tested with the list of test cases provided.
- All of the test cases must return the correct outputs.

System tests:

- All tests of system requirements must pass.
- All failed tests must be documented.

10 SUSPENSION CRITERIA AND RESUMPTION REQUIREMENTS

N/A

11 TEST DELIVERABLES

- Unit tests
- System tests
- Failed test reports

12 REMAINING TEST TASKS

N/A

13 ENVIRONMENTAL NEEDS

Test data for the unit tests will be provided by the developer. This data will include boundary tests for functions that deal with a range of data.

14 STAFFING AND TRAINING NEEDS

No special staffing or training needs required.

15 RESPONSIBILITIES

The developer is responsible for all verification and accepting of unit tests,

system tests and failed test reports.

16 SCHEDULE

Schedule is to be determined.

17 PLANNING RISKS AND CONTINGENCIES

Risks:

- Lack of division of responsibilities

18 APPROVALS

Project sponsor: Dave Flansbaum