Alexandria University

Final Project – Phase I

SIC/XE assembler

Zahraa Emara – 4558

Mayar El Mahdy – 4639

Mariam Beltagy – 5109

Sarah Yosry – 4582

1. **Requirement specification**
2. **Design**
3. **Main Data structures**
4. **Algorithms Description**

It’s implemented in the (Fixed Controller) class. It scans the source file Instruction by instruction.

First the instruction is read and stored in a string, then this string is divided into words and each word is identified either as a label, an opcode or an operand and stored in its specified array accordingly as demonstrated below, if a comment is detected the dividing step is skipped and instead it’s stored in the comments array.

For example if the instruction **PROG START 1000** is read the algorithm stores it in the arrays as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| index | Label Array | Opcode Array | Operands Array |
| 0 | **PROG** | **START** | **1000** |

But, if the instruction **ADDR A,X** is read to preserve the sequence of instructions an empty space is stored in the label array as follows:

|  |  |  |  |
| --- | --- | --- | --- |
| index | Label Array | Opcode Array | Operands Array |
| 1 | NULL | **ADDR** | **A,X** |

After scanning the entire file and populating the 4 main arrays we start validating the first elements in all arrays if there’s an error a relevant message is put in the error array then the instruction is printed in the list file and then the second elements is validated then printed and so on.

1. **Assumptions**
2. **Sample Runs**