

# 國立台南大學

## 資訊工程學系

### 組合語言與系統程式

SIC/XE 單步執行模擬器

SIC/XE One-Step Execution Simulator

S10259002 黃詩豪

中華民國 104 年 06 月 11 日

# 目錄

一、	開發目的與動機（Develop Purpose and Motive）	2
二、	開發環境（Development Environment）	2
2.1	作業系統	2
2.2	整合開發環境	2
2.3	編譯器	2
三、	系統主要功能描述（Description of Requirement Operations）	2
四、	系統架構與實作（Structure and Implementation）	3
4.1	系統架構	3
4.2	物件導向實作	4
4.3	圖形化介面實作	9
五、	系統測試與除錯（Test and Debug）	11
5.1	目標檔案測試	11
5.2	錯誤發現與修正	26
六、	軟體使用說明（User Guide）	26
七、	開發心得	30
八、	參考文獻	31
附錄 A	指令表格（Opcode Table）	31
附錄 B	定址模式（Addressing mode）	33

## 一、 開發目的與動機 (Develop Purpose and Motive)

在組合語言與系統程式 (System Software) 一門課中，學習到 SIC 及 SIC/XE 兩個虛擬機器，更瞭解組合語言與硬體間溝通的概念。由組合語言經過組譯器產生目的碼 (Object Code)，然後產生目標程式 (Object Program)，在目標程式中有 Head、Text、Modification、End Record...等，將目標程式透過載入器 (Loader) 並依照前述的記錄將程式載入到記憶體中，而且會依照載入的記憶體位置對程式進行位置的修正，然後 CPU 按照程式計數器 (Program Counter) 開始執行。這一過程中使我感到興趣，對其運作的方式想更進一步瞭解，因此我將以 C/C++ 程式語言開發一個 SIC/XE 的單步執行模擬器，透過此模擬器顯示目標程式執行的過程中，每個暫存器及記憶體的變化。

## 二、 開發環境 (Development Environment)

### 2.1 作業系統

- Windows 7 旗艦版 Service Pack 1 (64 位元作業系統)

### 2.2 整合開發環境

- Eclipse Luna Service Release 2 (4.4.2, C/C++ CDT for Luna)
- Qt Creator 3.3.1 (opensource) Based on Qt5.4.1 (MSVC 2010, 32bit)

### 2.3 編譯器

- MinGW 4.9.1 32bit

## 三、 系統主要功能描述 (Description of Requirement Operations)

- **選擇目標程式。**圖形化介面讓使用者以開啟檔案的方式，選擇欲載入的目標程式。
- **指定目標程式載入位置。**模擬 Modification Record 功能之使用者輸入目標程式特定載入記憶體位置。
- **單步執行一個指令。**使用者按下指定按鈕，程式執行目標程式中的一個指令，指令執行後顯示暫存器與記憶體內容。
- **輸出目標程式完整執行結果之檔案。**使用者將載入的目標程式執行結果儲存為文字檔，檔案中顯示執行步驟、指令內容、暫存器內容。

#### 四、系統架構與實作 (Structure and Implementation)

##### 4.1 系統架構



## 4.2 物件導向實作

程式中定義一個定義檔案 (Specification.h) 及五個類別，定義檔案中主要是放程式中會使用到的結構 (Struct)、函式 (Function)；類別分別是 NIXBPE、Register、Loader、MemoryManager、Executor，以下我將說明 Specification.h 與類別之中重要的屬性與函式。

### • Specification 定義

Type define structure :

名稱	屬性	說明
myHeadRd	string name int startAddr int size	目標程式中的 Head Record 的內容，檔案名稱以字串儲存，程式載入位置轉為非負整數儲存至 startAddr，程式大小以非負整數且以 Byte 為單位儲存到 size。
myTextRd	int startAddr int size string record	目標程式中的 Text Record 的內容，將 Text Record 的起始位置轉為非負整數儲存放到 startAddr，大小也以非負整數存到 size，記錄儲存到 record 屬性。
myModificationRd	int startAddr int length	目標程式中的 Modification Record 內容，要修改的資料起始位置轉成非負整數儲存到 startAddr，以及要修改的數量存到 length 屬性。
myEndRd	int executionAddr	將 End Record 中記錄程式中，第一個指令執行的位置儲存到 executionAddr。
myObjectCode	int format int opNum string opStr string data	在本系統架構中的 Executor 類別中，儲存現在指令的格式，指令 opcode 轉為非負整數存到 opNum，然後將 opcode 指令的 Mnemonic 儲存到 opStr。最後是指令的資料，格式 1 則不設定 data；格式 2 則設定 data 為兩暫存器編號；格式 3 則設定 data 為 disp；格式 4 則設定 data 為 addr。

Function Define :

回傳	名稱	參數	說明
string	LookUpTable	string str int num bool type int offset int *format int *opNum	用來查找 opcode 的 Mnemonic 與 Format。前三個參數是代表使用字串 (type = true) 還是正整數 (type = false) 的方式查詢。offset 參數是扣除指令中最左邊的 byte 當中 n、i 位元的偏移量。format 指標儲存格式，opNum 儲存將 opcode 轉為正整數。最後回傳 opcode 的 Mnemonic 字串。

### • NIXBPE 類別

Private variables : bool n, i, x, b, p, e

變數形態	名稱	說明
bool	n	n 位元的狀態，True 與 False。
bool	i	i 位元的狀態，True 與 False。
bool	x	x 位元的狀態，True 與 False。
bool	b	b 位元的狀態，True 與 False。
bool	p	p 位元的狀態，True 與 False。
bool	e	e 位元的狀態，True 與 False。

Public method :

回傳	名稱	參數	說明
void	decodeNI	char	對傳入的字元，轉為 2 進位碼，然後將最右邊兩個字元轉為布林值，存入 n、i 屬性。
void	decodeXBPE	char	將傳入的字元，轉為 2 進位碼，然後由左至右，將字元轉為布林值存入 x、b、p、e 屬性。

## • Register 類別

Private variables :

變數形態	名稱	說明
string	A	暫存器 A 的內容，以 16 進位碼字串儲存。
string	X	暫存器 X 的內容，以 16 進位碼字串儲存。
string	L	暫存器 L 的內容，以 16 進位碼字串儲存。
string	B	暫存器 B 的內容，以 16 進位碼字串儲存。
string	S	暫存器 S 的內容，以 16 進位碼字串儲存。
string	T	暫存器 T 的內容，以 16 進位碼字串儲存。
string	F	暫存器 F 的內容，以 16 進位碼字串儲存。
string	PC	暫存器 PC 的內容，以 16 進位碼字串儲存。
string	SW	暫存器 SW 的內容，以 16 進位碼字串儲存。

Public method :

回傳	名稱	參數	說明
string	Store	int	回傳暫存器內容。傳入暫存器編碼，回傳暫存器儲存的字串。
void	Load	string, int	載入暫存器。將傳入的字串，存入指定的暫存器編碼。

## • Loader 類別

Private variables :

變數形態	名稱	說明
myHeadRd	HeadRecord	為一個 myHeadRd struct 的變數。
vector<myTextRd>	TextRecord	以 vector 的方式儲存 myTextRd struct。
vector<myModificationRd>	ModificationRecord	以 vector 的方式儲存 myModificationRd struct。
myEndRd	EndRecord	為一個 myEndRd struct 變數。
string	LoadAddr	儲存使用者自行決定目標程式所載入到的記憶體位置。

Public method :

回傳	名稱	參數	說明
bool	LoadObjectProgram	string, string	第一個參數是檔案名稱，第二參數是指定檔案所載入的記憶體位置。先將檔案中的每一行文字讀入，儲存到一個字串陣列，然後將此陣列中每一字串進行解析，分為 H、T、M、E，最後依照每一記錄中的定義，儲存到私有屬性結構中。如果程中載入目標程式，就回傳 True，否則回傳 False。

### • MemoryManager 類別

Private variables :

變數形態	名稱	說明
int	StartLoc	以正整數儲存目標程式的起始位置。
int	LoadAddr	以正整數儲存使用者指定目標程式所載入的位置。
int	ObjectProgramSize	以正整數儲存目標程式的大小。
string	MemoryBlock	以字串的方式紀錄目標程式載入記憶體的內容。

Public method :

回傳	名稱	參數	說明
bool	LoadInMemory	Loader *loader	將 Loader 載入的目標程式內容，讀入記憶體區塊中，參數是 loader 物件指標。先處理 Text Record 中的內容，計算第 i 個記錄與第 i+1 個記錄的位置差大小，並將位置差扣第 i 個記錄的 size，即為保留的記憶體空間，本系統將保留的記憶體空間以 "*" 符號代表。 然後處理目標程式中 M Record 所要修改的位置內容。 最後成功載入記憶體中，則回傳 True。
void	Write	int loc string data	傳入記憶體位置 (Byte 為單位)，以及要寫入記憶體中的資料 (data)。寫入記憶體前會先扣除使用者指定程式載入位置，然後乘以 2，因為 1 Byte 等於 2 個字元。
string	Read	int loc int bytes	傳入記憶體位置 (Byte 為單位)，寫入記憶體前會先扣除使用者指定程式載入位置，然後乘以 2，並以此 loc 將要讀取的資料量 (bytes) 回傳字串。
string	Output		無傳入參數。每列以 25 Bytes 的大小，以及該列資料在記憶體中的位置。



• **Executor 類別**

Private variables :

變數形態	名稱	說明
MemoryManager	*memBlock	MemoryManager 類別指標，指向 MemoryManager 物件。
NIXBPE	*nixbpe	NIXBPE 類別指標，指向 NIXBPE 物件。
Registers	*regs	Registers 類別指標，指向 Registers 物件
int	program_counter	以正整數記錄 Program Counter 內容，且 PC 暫存器內容也是由此變數指定。
string	cur_byte	記錄當前步驟抓取的一個 Byte，用來查詢 opcode、format、決定 n，i 位元內容。
myObjectCode	cur_objectCode	用來記錄當前解碼出來的 object code 內容。

Private method :

回傳	名稱	參數	說明
void	operators	int op int format	傳入 opcode 的十進制數值，以及此 opcode 的 format。先依據 format、n、i、x、b、p、e 與 cur_objectCode，決定定址模式，計算出 Target Address，然後再計算出 Operand。 最後以 op 參數執行指令的功能。
void	fetchByte		從記憶體中，根據 PC 暫存器的值，抓取 1 Byte，儲存到 cur_byte。
string	decode		先從 cur_byte 先解碼指令的 opcode 與 format，以及 n，i 位元。依據 format 再從記憶體中取出資料，然後將資料存入 cur_objectCode structure 的各個屬性。
void	execute		執行 Executor 類別中的私有方法 operators，將目前解碼後的 opNum 與 format 傳入，便完成執行現在的指令。

Public method :

回傳	名稱	參數	說明
(建構子)	Executor	MemoryManager *memBlock NIXBPE *nixbpe Registers *regs int firstOpAddr	載入 MemoryManager、NIXBPE、Registers 物件。 firstOpAddr 表示程式第一個指令起始位置，用來初始化 PC 內容。

bool	oneStep	string *oplInstructStr	以 fetch→decode→execute 的流程進行單步執行的方式。參數 oplInstructStr 指標是用來圖形化界面輸出用。然後成功執行此步驟，則回傳 True，當 PC 為 0xFFFFFFFF 則回傳 False。
bool	SaveFileResult	string fileName string *errorMsg	傳入欲儲存的檔案的檔名與位置。第二個參數 errorMsg 指標輸出在寫入檔案過程中，出現的錯誤。最後如果成功寫入檔案就回傳 True，否則 False。

### 4.3 圖形化介面實作



圖 1 系統介面圖

介面物件說明：

物件名稱	物件類別	說明
LoadAddrLineEdit	QLineEdit	載入記憶體位置輸入框，輸入至多六位 16 進制字元，並僅允許數字、A-F、a-f 字元。
OpenFileTextEdit	QTextEdit	記錄選擇的檔案所在位置與檔案名稱。
CurlInstructContentLable	QLabel	顯示現在執行的指令 object code。
MemeryDisplayPlainTextEdit	QPlainTextEdit	顯示所載入的目標程式所使用的記憶體資訊。
RegsPCContentLable	QLabel	將 PC 暫存器內容輸出顯示在此標籤。
RegsAContentLable	QLabel	將 A 暫存器內容輸出顯示在此標籤。
RegsXContentLable	QLabel	將 X 暫存器內容輸出顯示在此標籤。
RegsLContentLable	QLabel	將 L 暫存器內容輸出顯示在此標籤。
RegsBContentLable	QLabel	將 B 暫存器內容輸出顯示在此標籤。
RegsSContentLable	QLabel	將 S 暫存器內容輸出顯示在此標籤。
RegsTContentLable	QLabel	將 T 暫存器內容輸出顯示在此標籤。
RegsSWContentLable	QLabel	將 SW 暫存器內容輸出顯示在此標籤。
SelectFileBtn	QPushButton	選擇欲開啟的目標程式所在目的，並在選擇之後，將字串存入 OpenFileTextEdit 物件中。另外，成功取得檔案所在位置與名稱後，會將程式載入至 Loader、MemoryManager、Executor 物件中，然後等待使用者按下「下一指令」按鈕。
OneStepBtn	QPushButton	「下一指令」按鈕，當觸發 click 事件，會執行 Executor 物件中的 OneStep function，然後會更新暫存器與記憶體顯示區塊的內容，並且在記憶體顯示區塊中標示出現在執行的指令，當程式執行完畢，按此按鈕則顯示「程式執行完畢」。
RemoveObpBtn	QPushButton	移除現在載入的目標程式。
FullExecBtn	QPushButton	將載入的目標程式的完整執行步驟，輸出成文字檔案，以供使用者檢查每一步驟的暫存器變化。

## 五、系統測試與除錯 (Test and Debug)

### 5.1 目標檔案測試

#### i. 第一個目標檔案 (TEST11.EX)

HTEST11000000000002A

T0000001E75000B05000053A00B57A013B8503B2FF54F00005445535420535452494E

T00001E0147

E000000

測試結果：

STEP :===== 1	STEP :===== 24
CURRENT INSTRUCTION :LDT 00B	CURRENT INSTRUCTION :STCH 013
PC : 000003 A : 000000 X : 000000	PC : 00000C A : 000053 X : 000005
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 2	STEP :===== 25
CURRENT INSTRUCTION :LDX 000	CURRENT INSTRUCTION :TIXR 50
PC : 000006 A : 000000 X : 000000	PC : 00000E A : 000053 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 3	STEP :===== 26
CURRENT INSTRUCTION :LDCH 00B	CURRENT INSTRUCTION :JLT FF5
PC : 000009 A : 000054 X : 000000	PC : 000006 A : 000053 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 4	STEP :===== 27
CURRENT INSTRUCTION :STCH 013	CURRENT INSTRUCTION :LDCH 00B
PC : 00000C A : 000054 X : 000000	PC : 000009 A : 000054 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040

STEP :===== 5	STEP :===== 28
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 013
PC :00000E A :000054 X:000001	PC :00000C A :000054 X:000006
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 6	STEP :===== 29
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC :000006 A :000054 X:000001	PC :00000E A :000054 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 7	STEP :===== 30
CURRENT INSTRUCTION :LDCH 00B	CURRENT INSTRUCTION :JLT FF5
PC :000009 A :000045 X:000001	PC :000006 A :000054 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 8	STEP :===== 31
CURRENT INSTRUCTION :STCH 013	CURRENT INSTRUCTION :LDCH 00B
PC :00000C A :000045 X:000001	PC :000009 A :000052 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 9	STEP :===== 32
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 013
PC :00000E A :000045 X:000002	PC :00000C A :000052 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 10	STEP :===== 33
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC :000006 A :000045 X:000002	PC :00000E A :000052 X:000008
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040

STEP :===== 11

CURRENT INSTRUCTION : LDCH 00B

PC : 000009 A : 000053 X : 000002

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 12

CURRENT INSTRUCTION : STCH 013

PC : 00000C A : 000053 X : 000002

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 13

CURRENT INSTRUCTION : TIXR 50

PC : 00000E A : 000053 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 14

CURRENT INSTRUCTION : JLT FF5

PC : 000006 A : 000053 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 15

CURRENT INSTRUCTION : LDCH 00B

PC : 000009 A : 000054 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 16

CURRENT INSTRUCTION : STCH 013

PC : 00000C A : 000054 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 34

CURRENT INSTRUCTION : JLT FF5

PC : 000006 A : 000052 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 35

CURRENT INSTRUCTION : LDCH 00B

PC : 000009 A : 000049 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 36

CURRENT INSTRUCTION : STCH 013

PC : 00000C A : 000049 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 37

CURRENT INSTRUCTION : TIXR 50

PC : 00000E A : 000049 X : 000009

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 38

CURRENT INSTRUCTION : JLT FF5

PC : 000006 A : 000049 X : 000009

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 39

CURRENT INSTRUCTION : LDCH 00B

PC : 000009 A : 00004E X : 000009

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 17	STEP :===== 40
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 013
PC : 00000E A : 000054 X: 000004	PC : 00000C A : 00004E X: 000009
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 18	STEP :===== 41
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC : 000006 A : 000054 X: 000004	PC : 00000E A : 00004E X: 00000A
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 19	STEP :===== 42
CURRENT INSTRUCTION :LDCH 00B	CURRENT INSTRUCTION :JLT FF5
PC : 000009 A : 000020 X: 000004	PC : 000006 A : 00004E X: 00000A
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 20	STEP :===== 43
CURRENT INSTRUCTION :STCH 013	CURRENT INSTRUCTION :LDCH 00B
PC : 00000C A : 000020 X: 000004	PC : 000009 A : 000047 X: 00000A
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 21	STEP :===== 44
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 013
PC : 00000E A : 000020 X: 000005	PC : 00000C A : 000047 X: 00000A
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 22	STEP :===== 45
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC : 000006 A : 000020 X: 000005	PC : 00000E A : 000047 X: 00000B
L : FFFFFFFF B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0

STEP :===== 23	STEP :===== 46
CURRENT INSTRUCTION : LDCH 00B	CURRENT INSTRUCTION : JLT FF5
PC : 000009 A : 000053 X : 000005	PC : 000011 A : 000047 X : 00000B
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
	STEP :===== 47
	CURRENT INSTRUCTION : RSUB 000
	PC : FFFFFFFF A : 000047 X : 00000B
	L : FFFFFFFF B : 000000 S : 000000
	T : 00000B SW : 0000C0

ii. 第二個目標檔案 (TEST12.EX)

HTEST1200000000002A

T0000000B5445535420535452494E47

T0000161475000B05000053AFE157AFE9B8503B2FF54F0000

E000016

測試結果：

STEP :===== 1	STEP :===== 24
CURRENT INSTRUCTION : LDT 00B	CURRENT INSTRUCTION : STCH FE9
PC : 000019 A : 000000 X : 000000	PC : 000022 A : 000053 X : 000005
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 2	STEP :===== 25
CURRENT INSTRUCTION : LDX 000	CURRENT INSTRUCTION : TIXR 50
PC : 00001C A : 000000 X : 000000	PC : 000024 A : 000053 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 3	STEP :===== 26
CURRENT INSTRUCTION : LDCH FE1	CURRENT INSTRUCTION : JLT FF5
PC : 00001F A : 000054 X : 000000	PC : 00001C A : 000053 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040



STEP :===== 4	STEP :===== 27
CURRENT INSTRUCTION :STCH FE9	CURRENT INSTRUCTION :LDCH FE1
PC :000022 A :000054 X:000000	PC :00001F A :000054 X:000006
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000000	T :00000B SW :000040
STEP :===== 5	STEP :===== 28
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH FE9
PC :000024 A :000054 X:000001	PC :000022 A :000054 X:000006
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 6	STEP :===== 29
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC :00001C A :000054 X:000001	PC :000024 A :000054 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 7	STEP :===== 30
CURRENT INSTRUCTION :LDCH FE1	CURRENT INSTRUCTION :JLT FF5
PC :00001F A :000045 X:000001	PC :00001C A :000054 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 8	STEP :===== 31
CURRENT INSTRUCTION :STCH FE9	CURRENT INSTRUCTION :LDCH FE1
PC :000022 A :000045 X:000001	PC :00001F A :000052 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 9	STEP :===== 32
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH FE9
PC :000024 A :000045 X:000002	PC :000022 A :000052 X:000007
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040

STEP :===== 10

CURRENT INSTRUCTION :JLT FF5

PC : 00001C A : 000045 X : 000002

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 11

CURRENT INSTRUCTION :LDCH FE1

PC : 00001F A : 000053 X : 000002

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 12

CURRENT INSTRUCTION :STCH FE9

PC : 000022 A : 000053 X : 000002

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 13

CURRENT INSTRUCTION :TIXR 50

PC : 000024 A : 000053 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 14

CURRENT INSTRUCTION :JLT FF5

PC : 00001C A : 000053 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 15

CURRENT INSTRUCTION :LDCH FE1

PC : 00001F A : 000054 X : 000003

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 33

CURRENT INSTRUCTION :TIXR 50

PC : 000024 A : 000052 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 34

CURRENT INSTRUCTION :JLT FF5

PC : 00001C A : 000052 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 35

CURRENT INSTRUCTION :LDCH FE1

PC : 00001F A : 000049 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 36

CURRENT INSTRUCTION :STCH FE9

PC : 000022 A : 000049 X : 000008

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 37

CURRENT INSTRUCTION :TIXR 50

PC : 000024 A : 000049 X : 000009

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 38

CURRENT INSTRUCTION :JLT FF5

PC : 00001C A : 000049 X : 000009

L : FFFFFFFF B : 000000 S : 000000

T : 00000B SW : 000040

STEP :===== 16	STEP :===== 39
CURRENT INSTRUCTION :STCH FE9	CURRENT INSTRUCTION :LDCH FE1
PC :000022 A :000054 X:000003	PC :00001F A :00004E X:000009
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 17	STEP :===== 40
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH FE9
PC :000024 A :000054 X:000004	PC :000022 A :00004E X:000009
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 18	STEP :===== 41
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC :00001C A :000054 X:000004	PC :000024 A :00004E X:00000A
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 19	STEP :===== 42
CURRENT INSTRUCTION :LDCH FE1	CURRENT INSTRUCTION :JLT FF5
PC :00001F A :000020 X:000004	PC :00001C A :00004E X:00000A
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 20	STEP :===== 43
CURRENT INSTRUCTION :STCH FE9	CURRENT INSTRUCTION :LDCH FE1
PC :000022 A :000020 X:000004	PC :00001F A :000047 X:00000A
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040
STEP :===== 21	STEP :===== 44
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH FE9
PC :000024 A :000020 X:000005	PC :000022 A :000047 X:00000A
L :FFFFFF B :000000 S:000000	L :FFFFFF B :000000 S:000000
T :00000B SW :000040	T :00000B SW :000040

STEP :===== 22	STEP :===== 45
CURRENT INSTRUCTION :JLT FF5	CURRENT INSTRUCTION :TIXR 50
PC : 00001C A : 000020 X : 000005	PC : 000024 A : 000047 X : 00000B
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
STEP :===== 23	STEP :===== 46
CURRENT INSTRUCTION :LDCH FE1	CURRENT INSTRUCTION :JLT FF5
PC : 00001F A : 000053 X : 000005	PC : 000027 A : 000047 X : 00000B
L : FFFFFFFF B : 000000 S : 000000	L : FFFFFFFF B : 000000 S : 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
	STEP :===== 47
	CURRENT INSTRUCTION :RSUB 000
	PC : FFFFFFFF A : 000047 X : 00000B
	L : FFFFFFFF B : 000000 S : 000000
	T : 00000B SW : 0000C0

### iii. 第三個目標檔案 (TEST2.EX)

HTEST2 00000000103B  
 T0000001B75000B0500001720124B10103457A019B8503B2FF40B20034F0000  
 T00001E0B5445535420535452494E47  
 T001034075390001E4F0000  
 M00000A05  
 M00103505  
 E000000

### 測試結果：

STEP :===== 1	STEP :===== 36
CURRENT INSTRUCTION :LDT 00B	CURRENT INSTRUCTION :RSUB 000
PC : 001003 A : 000000 X : 000000	PC : 00100D A : 000053 X : 000005
L : FFFFFFFF B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040

STEP :===== 2	STEP :===== 37
CURRENT INSTRUCTION :LDX 000	CURRENT INSTRUCTION :STCH 019
PC : 001006 A : 000000 X : 000000	PC : 001010 A : 000053 X : 000005
L : FFFFFFFF B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 3	STEP :===== 38
CURRENT INSTRUCTION :STL 012	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000000 X : 000000	PC : 001012 A : 000053 X : 000006
L : FFFFFFFF B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 4	STEP :===== 39
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000000 X : 000000	PC : 001009 A : 000053 X : 000006
L : 00100D B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 5	STEP :===== 40
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :+JSUB 02034
PC : 002038 A : 000054 X : 000000	PC : 002034 A : 000053 X : 000006
L : 00100D B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 6	STEP :===== 41
CURRENT INSTRUCTION :RSub 000	CURRENT INSTRUCTION :+LDCH 0101E
PC : 00100D A : 000054 X : 000000	PC : 002038 A : 000054 X : 000006
L : 00100D B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040
STEP :===== 7	STEP :===== 42
CURRENT INSTRUCTION :STCH 019	CURRENT INSTRUCTION :RSub 000
PC : 001010 A : 000054 X : 000000	PC : 00100D A : 000054 X : 000006
L : 00100D B : 000000 S : 000000	L : 00100D B : 000000 S : 000000
T : 00000B SW : 000000	T : 00000B SW : 000040

STEP :===== 8	STEP :===== 43
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 019
PC : 001012 A : 000054 X: 000001	PC : 001010 A : 000054 X: 000006
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 9	STEP :===== 44
CURRENT INSTRUCTION :JLT FF4	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000054 X: 000001	PC : 001012 A : 000054 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 10	STEP :===== 45
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000054 X: 000001	PC : 001009 A : 000054 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 11	STEP :===== 46
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :+JSUB 02034
PC : 002038 A : 000045 X: 000001	PC : 002034 A : 000054 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 12	STEP :===== 47
CURRENT INSTRUCTION :RSub 000	CURRENT INSTRUCTION :+LDCH 0101E
PC : 00100D A : 000045 X: 000001	PC : 002038 A : 000052 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 13	STEP :===== 48
CURRENT INSTRUCTION :STCH 019	CURRENT INSTRUCTION :RSub 000
PC : 001010 A : 000045 X: 000001	PC : 00100D A : 000052 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040

STEP :===== 14	STEP :===== 49
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 019
PC : 001012 A : 000045 X: 000002	PC : 001010 A : 000052 X: 000007
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 15	STEP :===== 50
CURRENT INSTRUCTION :JLT FF4	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000045 X: 000002	PC : 001012 A : 000052 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 16	STEP :===== 51
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000045 X: 000002	PC : 001009 A : 000052 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 17	STEP :===== 52
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :+JSUB 02034
PC : 002038 A : 000053 X: 000002	PC : 002034 A : 000052 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 18	STEP :===== 53
CURRENT INSTRUCTION :RSUB 000	CURRENT INSTRUCTION :+LDCH 0101E
PC : 00100D A : 000053 X: 000002	PC : 002038 A : 000049 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 19	STEP :===== 54
CURRENT INSTRUCTION :STCH 019	CURRENT INSTRUCTION :RSUB 000
PC : 001010 A : 000053 X: 000002	PC : 00100D A : 000049 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040

STEP :===== 20	STEP :===== 55
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 019
PC : 001012 A : 000053 X: 000003	PC : 001010 A : 000049 X: 000008
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 21	STEP :===== 56
CURRENT INSTRUCTION :JLT FF4	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000053 X: 000003	PC : 001012 A : 000049 X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 22	STEP :===== 57
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000053 X: 000003	PC : 001009 A : 000049 X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 23	STEP :===== 58
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :+JSUB 02034
PC : 002038 A : 000054 X: 000003	PC : 002034 A : 000049 X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 24	STEP :===== 59
CURRENT INSTRUCTION :RSub 000	CURRENT INSTRUCTION :+LDCH 0101E
PC : 00100D A : 000054 X: 000003	PC : 002038 A : 00004E X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 25	STEP :===== 60
CURRENT INSTRUCTION :STCH 019	CURRENT INSTRUCTION :RSub 000
PC : 001010 A : 000054 X: 000003	PC : 00100D A : 00004E X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040



STEP :===== 26	STEP :===== 61
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 019
PC : 001012 A : 000054 X: 000004	PC : 001010 A : 00004E X: 000009
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 27	STEP :===== 62
CURRENT INSTRUCTION :JLT FF4	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000054 X: 000004	PC : 001012 A : 00004E X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 28	STEP :===== 63
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000054 X: 000004	PC : 001009 A : 00004E X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 29	STEP :===== 64
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :+JSUB 02034
PC : 002038 A : 000020 X: 000004	PC : 002034 A : 00004E X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 30	STEP :===== 65
CURRENT INSTRUCTION :RSUB 000	CURRENT INSTRUCTION :+LDCH 0101E
PC : 00100D A : 000020 X: 000004	PC : 002038 A : 000047 X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 31	STEP :===== 66
CURRENT INSTRUCTION :STCH 019	CURRENT INSTRUCTION :RSUB 000
PC : 001010 A : 000020 X: 000004	PC : 00100D A : 000047 X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040

STEP :===== 32	STEP :===== 67
CURRENT INSTRUCTION :TIXR 50	CURRENT INSTRUCTION :STCH 019
PC : 001012 A : 000020 X: 000005	PC : 001010 A : 000047 X: 00000A
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 000040
STEP :===== 33	STEP :===== 68
CURRENT INSTRUCTION :JLT FF4	CURRENT INSTRUCTION :TIXR 50
PC : 001009 A : 000020 X: 000005	PC : 001012 A : 000047 X: 00000B
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
STEP :===== 34	STEP :===== 69
CURRENT INSTRUCTION :+JSUB 02034	CURRENT INSTRUCTION :JLT FF4
PC : 002034 A : 000020 X: 000005	PC : 001015 A : 000047 X: 00000B
L : 00100D B : 000000 S: 000000	L : 00100D B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
STEP :===== 35	STEP :===== 70
CURRENT INSTRUCTION :+LDCH 0101E	CURRENT INSTRUCTION :LDL 003
PC : 002038 A : 000053 X: 000005	PC : 001018 A : 000047 X: 00000B
L : 00100D B : 000000 S: 000000	L : FFFFFFFF B : 000000 S: 000000
T : 00000B SW : 000040	T : 00000B SW : 0000C0
	STEP :===== 71
	CURRENT INSTRUCTION :RSUB 000
	PC : FFFFFFFF A : 000047 X: 00000B
	L : FFFFFFFF B : 000000 S: 000000
	T : 00000B SW : 0000C0

## 5.2 錯誤發現與修正

- 錯誤：  
選擇目標程式與儲存程式完整執行檔時，中文路徑的錯誤  
修正方法：  
將從 `QFileDialog::getOpenFileName` 方法中取得的檔案字串，利用 `QString` 類別中 `toLocal8Bit()` 與 `toString()` 方法，使得原先讀取的字串重新編碼，即可正確開啟或儲存檔案。
- 錯誤：  
載入目標程式時，未考慮 Directive 中的 RESB、RESW 宣告保留的記憶體空間  
修正方法：  
利用 Text Record 中的起始位置與大小，將第  $n + 1$  與第  $n$  個 Text Record 的起始位置相減，再扣除第  $n$  個記錄的大小，即可得到第  $n$  到  $n + 1$  個記錄所佔用的記憶體大小。
- 錯誤：  
未將 PC 暫存器內容以 End Record 初始化  
修正方法：  
在 Executor 物件中，將 Loader 物件中所記錄的 End Record 結構的起始位置指定給 PC 暫存器。

## 六、 軟體使用說明（User Guide）

1. 輸入目標程式載入位置（六碼，0 至 9，A 至 F，a 至 f）



圖 2 輸入記憶體載入位置的錯誤輸入圖

2. 按下「選擇檔案」，選取正確的目標程式，載入成功會顯示「笑臉」的圖示

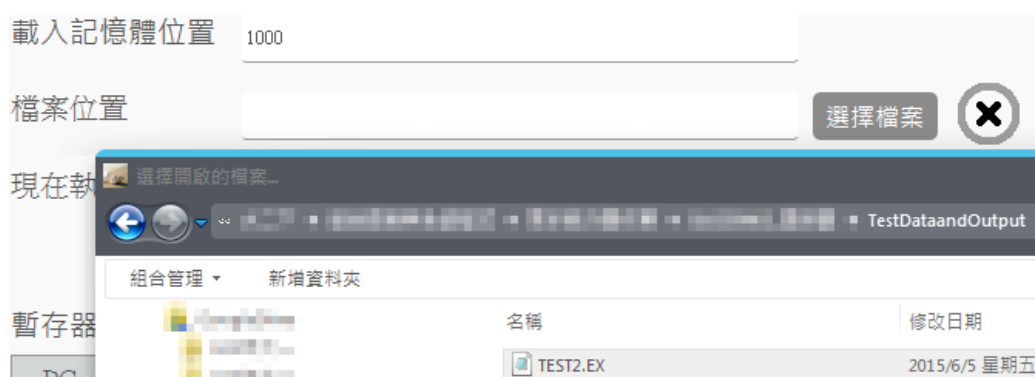


圖 3.1 選擇檔案圖

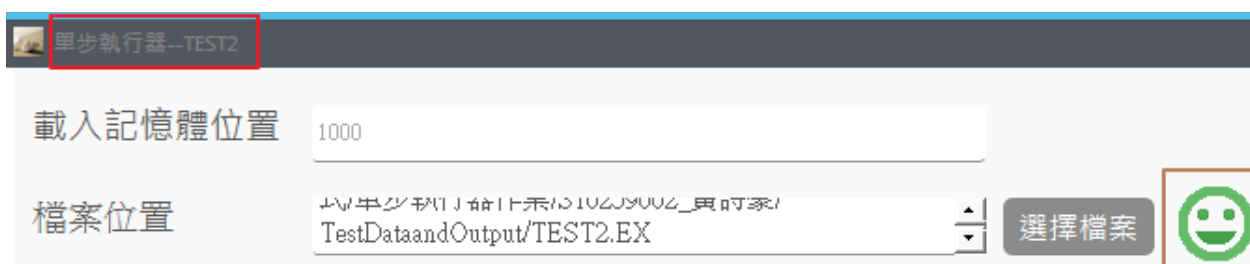


圖 3.2 檔案成功載入圖

3. 模擬執行目標程式，按下「下一指令」，會顯示當前執行指令目的碼，且在記憶體區塊會標示出該目的碼，同時更新暫存器與記憶體內容



圖 4 模擬執行指令目的碼

4. 程式執行完畢時，再按下「下一指令」會顯示「程式已執行完畢」



圖 5 目標程式執行完畢圖

5. 儲存目標程式完整執行輸出檔，按下「輸出完整執行結果」，選擇欲儲存路徑與輸入檔案名稱，於儲存成功後，可選擇是否自動開啟已儲存之檔案

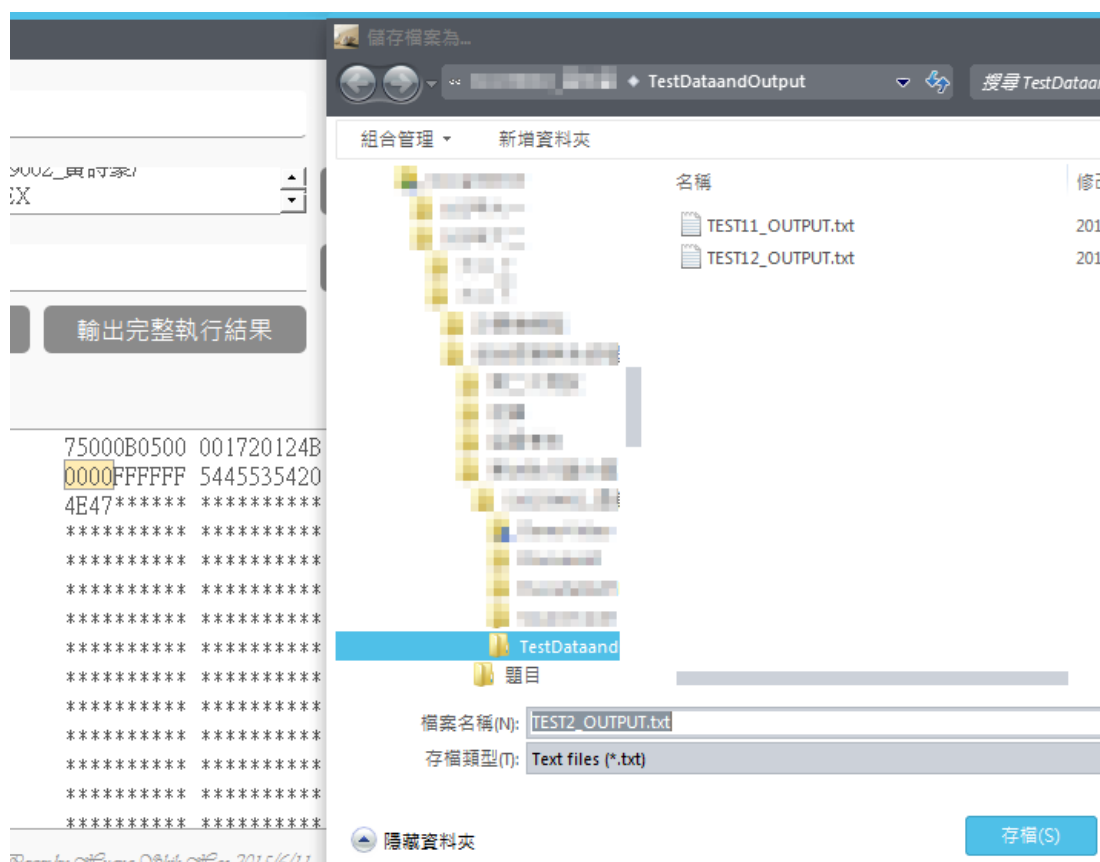


圖 6.1 儲存目標程式完整執行輸出檔



圖 6.2 自動開啟檔案圖

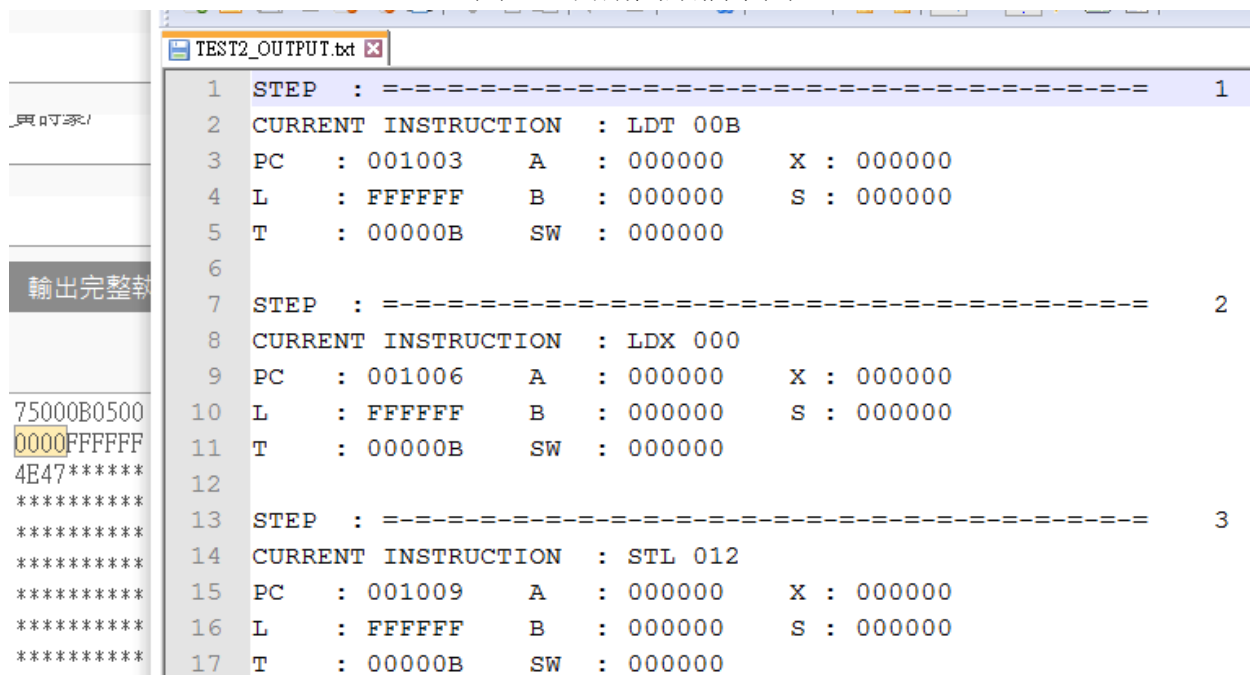


圖 6.3 輸出的執行檔案圖

6. 按下「移除 Object Program」將清除目前載入的檔案



圖 7 移除目前載入的程式

## 七、開發心得

整個開發的過程中，需要先對 SIC/XE 架構、指令、定址模式進行瞭解，然後再從組譯器 (Assembler) 的角度理解組譯的過程。一開始的想法是將目標程式中的每一個 Text Record 先串起來，接著依照指令格式將記錄中切割好，儲存到我定義的資料結構中，但這樣的一個方法，會造成一個問題——無法區別指令與資料，因此詢問了指導老師，發現我對於目標程式的執行概念完全錯誤。在老師的指導與重新讀過課本內容後，才知道要使用 Program Counter (PC) 暫存器，因為 PC 暫存器的內容是指向下一個指令執行的記憶體位置。最後成功地執行載入的目標程式，也解決無法區別指令與資料的問題。

在老師給得測試檔案中，發現 TEST12.EX 檔案載入程式會出現錯誤，因為此測試檔案將資料的宣告擺在指令前面，而我未將 End Record 中記錄程式第一個可執行指令的位置初始化 PC 暫存器，發現此問題後，將此記錄初始化 PC 暫存器便能正確執行 TEST12.EX 檔案。

實作學期專案的過程中，更瞭解了目標程式的載入與執行過程，同時，學會使用 Qt 的圖形化介面 (GUI) 套件，它是能套用在 C/C++ 程式語言實作出 GUI 的套件，更方便的是，它有類似於 Java 的 WindowBuilder 插件功能的整合開發環境，Qt Creator，使得在開發 GUI 的過程中減少許多時間，更加快了本系統的開發與完整度。雖然此次的專題並不完整地模擬 SIC/XE 架構，但也學習到很多的知識。

## 八、 參考文獻

[1]. Leland L. Beck (1997). System Software An Introduction to Systems Programming. United States of America: Addison Wesley Longman, Inc.

[2]. Qt Documentation. <http://doc.qt.io/>

## 附錄 A 指令表格 (Opcode Table)

Mnemonic	Format	Opcode	Effect	Notes
<b>ADD m</b>	3/4	18	$A \leftarrow (A) + (m..m+2)$	
<b>ADDF m</b>	3/4	58	$F \leftarrow (F) + (m..m+5)$	X F
<b>ADDR r1, r2</b>	2	90	$r2 \leftarrow (r2) + (r1)$	X
<b>AND m</b>	3/4	40	$A \leftarrow (A) \& (m..m+2)$	
<b>CLEAR r1</b>	2	B4	$r1 \leftarrow 0$	X
<b>COMP m</b>	3/4	28	$(A) : (m..m+2)$	C
<b>COMPF m</b>	3/4	88	$(F) : (m..m+5)$	X F C
<b>COMPR r1, r2</b>	2	A0	$(r1) : (r2)$	X C
<b>DIV m</b>	3/4	24	$A \leftarrow (A) / (m..m+2)$	
<b>DIVF m</b>	3/4	64	$F \leftarrow (F) / (m..m+5)$	X F
<b>DIVR r1, r2</b>	2	9C	$r2 \leftarrow (r2) / (r1)$	X
<b>FIX</b>	1	C4	$A \leftarrow (F)$ [convert to integer]	X F
<b>FLOAT</b>	1	C0	$F \leftarrow (A)$ [convert to floating]	X F
<b>HIO</b>	1	F4	Halt I/O channel number(A)	P X
<b>J m</b>	3/4	3C	$PC \leftarrow m$	
<b>JEQ m</b>	3/4	30	$PC \leftarrow m$ if CC set to =	
<b>JGT m</b>	3/4	34	$PC \leftarrow m$ if CC set to >	
<b>JLT m</b>	3/4	38	$PC \leftarrow m$ if CC set to <	
<b>JSUB m</b>	3/4	48	$L \leftarrow (PC); PC \leftarrow m$	
<b>LDA m</b>	3/4	00	$A \leftarrow (m..m+2)$	
<b>LDB m</b>	3/4	68	$B \leftarrow (m..m+2)$	X
<b>LDCH m</b>	3/4	50	$A[\text{rightmost byte}] \leftarrow (m)$	
<b>LDF m</b>	3/4	70	$F \leftarrow (m..m+5)$	X F
<b>LDL m</b>	3/4	08	$L \leftarrow (m..m+5)$	
<b>LDS m</b>	3/4	6C	$S \leftarrow (m..m+2)$	X
<b>LDT m</b>	3/4	74	$T \leftarrow (m..m+2)$	X
<b>LDX m</b>	3/4	04	$X \leftarrow (m..m+2)$	



<b>LPS m</b>	3/4	D0	Load processor status from information beginning at address m(see Section 6.2.1)	P X
<b>MUL m</b>	3/4	20	$A \leftarrow (A) * (m..m+2)$	
<b>MULF m</b>	3/4	60	$F \leftarrow (F) * (m..m+5)$	X F
<b>MULR r1,r2</b>	2	98	$r2 \leftarrow (r2) * (r1)$	X
<b>NORM</b>	1	C8	$F \leftarrow (F)$ [normalized]	X F
<b>OR m</b>	3/4	44	$A \leftarrow (A)   (m..m+2)$	
<b>RD m</b>	3/4	D8	$A[\text{rightmost byte}] \leftarrow$ data from device specified by (m)	P
<b>RMO r1,r2</b>	2	AC	$r2 \leftarrow (r1)$	X
<b>RSUB</b>	3/4	4C	$PC \leftarrow (L)$	
<b>SHIFTL r1,n</b>	2	A4	$r1 \leftarrow (r1)$ ; left circular shift n bits. {In assembled instruction, $r2 = n-1$ }	X
<b>SHIFTR r1,n</b>	2	A8	$r1 \leftarrow (r1)$ ; right shift n bits, with vacated bit positions set equal to leftmost bit of (r1). {In assembled instruction, $r2 = n-1$ }	X
<b>SIO</b>	1	F0	Start I/O channel number (A); address of channel program is given by (S)	P X
<b>SSK m</b>	3/4	EC	Protection key for address m $\leftarrow (A)$ {see Section 6.2.4}	P X
<b>STA m</b>	3/4	0C	$m..m+2 \leftarrow (A)$	
<b>STB m</b>	3/4	78	$m..m+2 \leftarrow (B)$	X
<b>STCH m</b>	3/4	54	$m \leftarrow (A)$ [rightmost byte]	
<b>STF m</b>	3/4	80	$m..m+5 \leftarrow (F)$	X F
<b>STI m</b>	3/4	D4	Interval timer value $\leftarrow (m..m+2)$ {see Section 6.2.1}	P X
<b>STL m</b>	3/4	14	$m..m+2 \leftarrow (L)$	
<b>STS m</b>	3/4	7C	$m..m+2 \leftarrow (S)$	X
<b>STSW m</b>	3/4	E8	$m..m+2 \leftarrow (SW)$	P
<b>STT m</b>	3/4	84	$m..m+2 \leftarrow (T)$	X
<b>STX m</b>	3/4	10	$m..m+2 \leftarrow (X)$	
<b>SUB m</b>	3/4	1C	$A \leftarrow (A) - (m..m+2)$	
<b>SUBF m</b>	3/4	5C	$F \leftarrow (F) - (m..m+5)$	X F
<b>SUBR r1,r2</b>	2	94	$r2 \leftarrow (r2) - (r1)$	X
<b>SVC n</b>	2	B0	Generate SVC interrupt. {In assembled instruction, $r1=n$ }	X
<b>TD m</b>	3/4	E0	Test device specified by (m)	P C

<b>TIO</b>	1	F8	Test I/O channel number (A)	P X C
<b>TIX m</b>	3/4	2C	$X \leftarrow (X) + 1; (X) : (m..m+2)$	C
<b>TIXR r1</b>	2	B8	$X \leftarrow (X) + 1; (X) : (r1)$	X C
<b>WD m</b>	3/4	DC	Device specified by (m) $\leftarrow$ (A) [rightmost byte]	P

## 附錄 B 定址模式 (Addressing mode)

Addressing type	Flag bits n i x b p e	Assembler language notation	Calculation of target address TA	Operand	Notes
Simple	1 1 0 0 0 0	op c	disp	(TA)	D
	1 1 0 0 0 1	+op m	addr	(TA)	4 D
	1 1 0 0 1 0	op m	(PC) + disp	(TA)	A
	1 1 0 1 0 0	op m	(B) + disp	(TA)	A
	1 1 1 0 0 0	op c,X	disp + (X)	(TA)	D
	1 1 1 0 0 1	+op m,X	addr + (X)	(TA)	4 D
	1 1 1 0 1 0	op m,X	(PC) + disp + (X)	(TA)	A
	1 1 1 1 0 0	op m,X	(C) + disp + (X)	(TA)	A
	0 0 0 - - -	op m	b/p/e/disp	(TA)	D S
	0 0 1 - - -	op m,X	b/p/e/disp + (X)	(TA)	D S
Indirect	1 0 0 0 0 0	op @c	disp	((TA))	D
	1 0 0 0 0 1	+op @m	addr	((TA))	4 D
	1 0 0 0 1 0	op @m	(PC) + disp	((TA))	A
	1 0 0 1 0 0	op @m	(D) + disp	((TA))	A
Immediate	0 1 0 0 0 0	op #c	disp	TA	D
	0 1 0 0 0 1	+op #m	addr	TA	4 D
	0 1 0 0 1 0	op #m	(PC) + disp	TA	A
	0 1 0 1 0 0	op #m	(E) + disp	TA	A