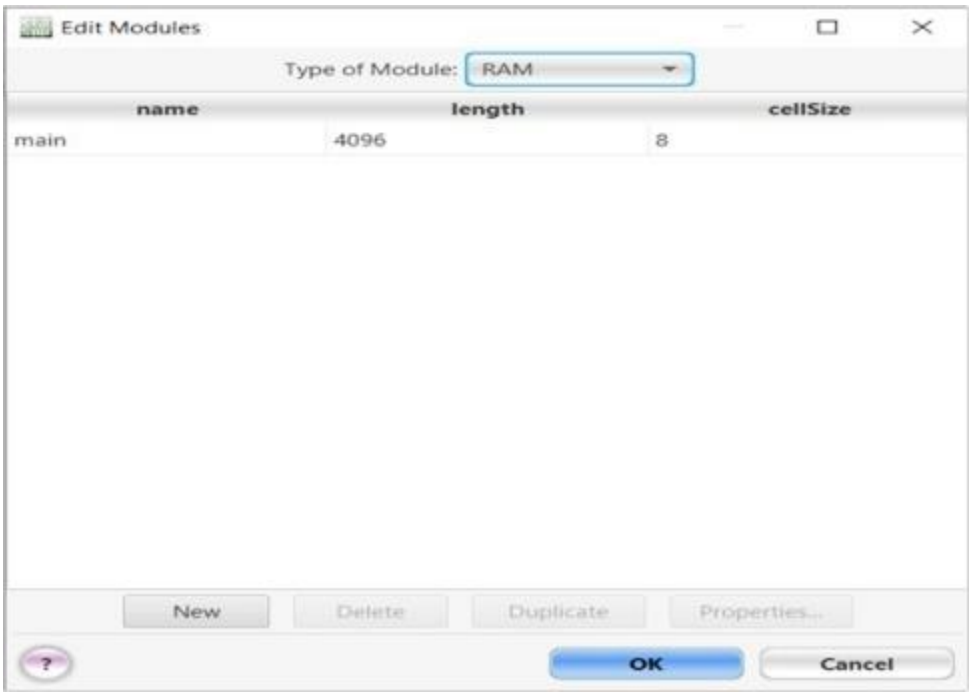
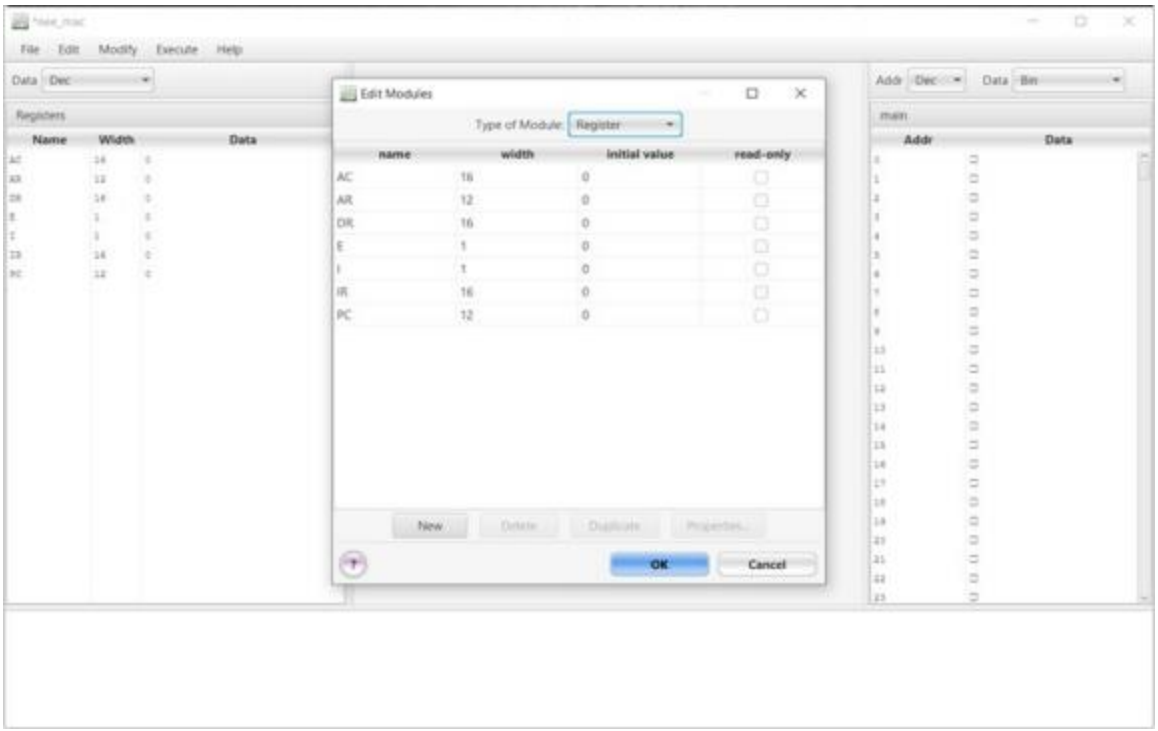
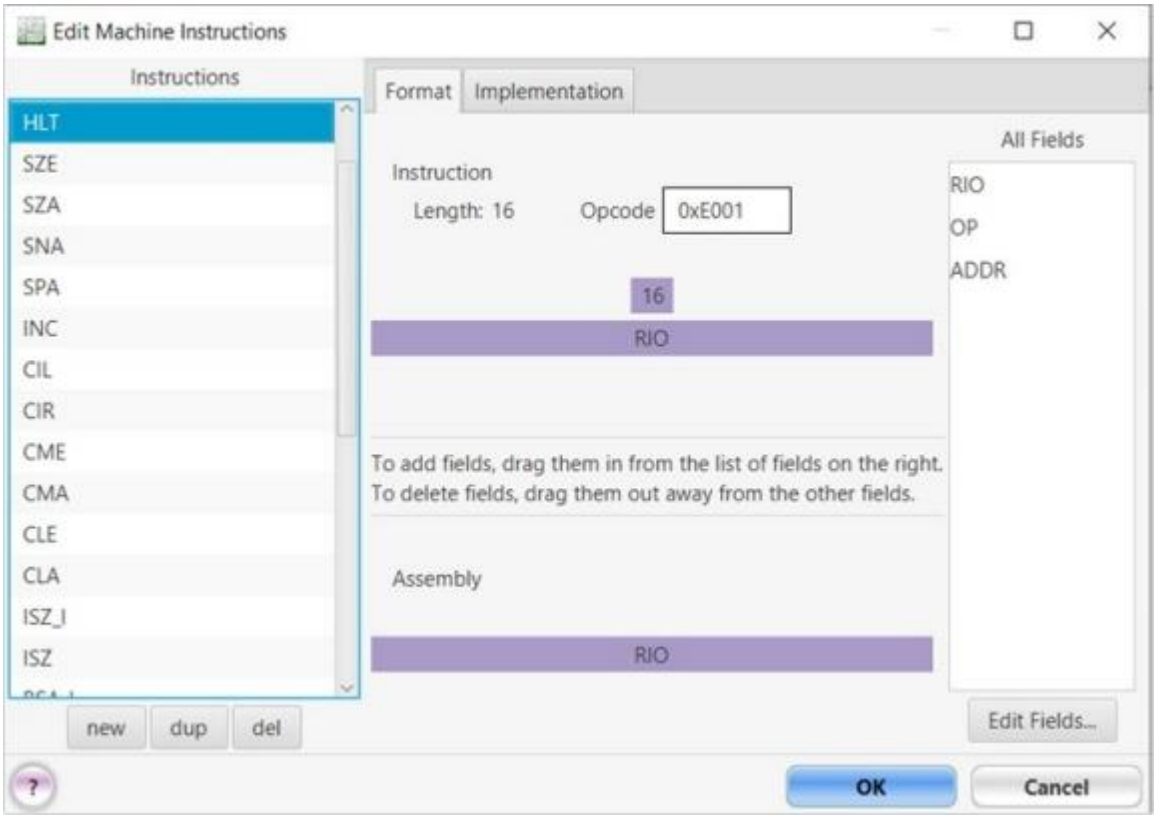
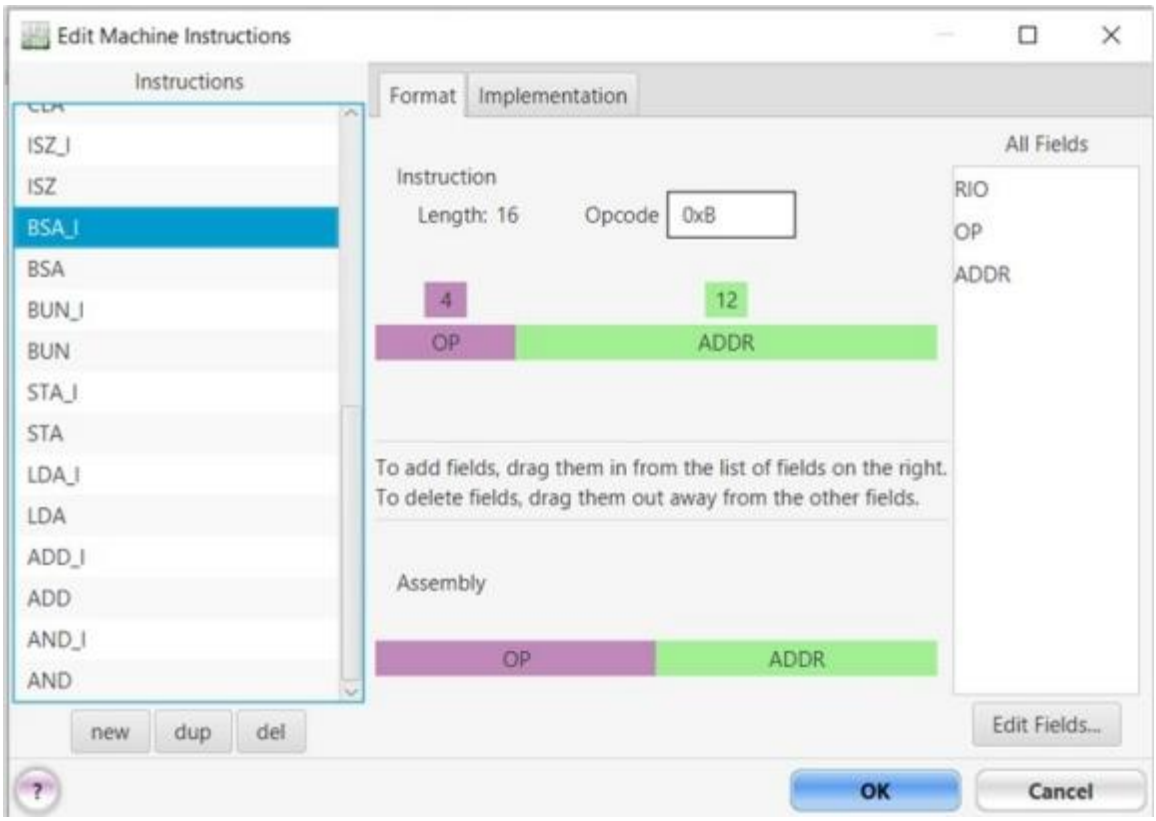
---------COMPUTER SYSTEM ARCHITECTURE PRACTICALS-----

NAME – TANU KAUSHIK ROLL NUMBER - 20211461 SUBJECT - CSA

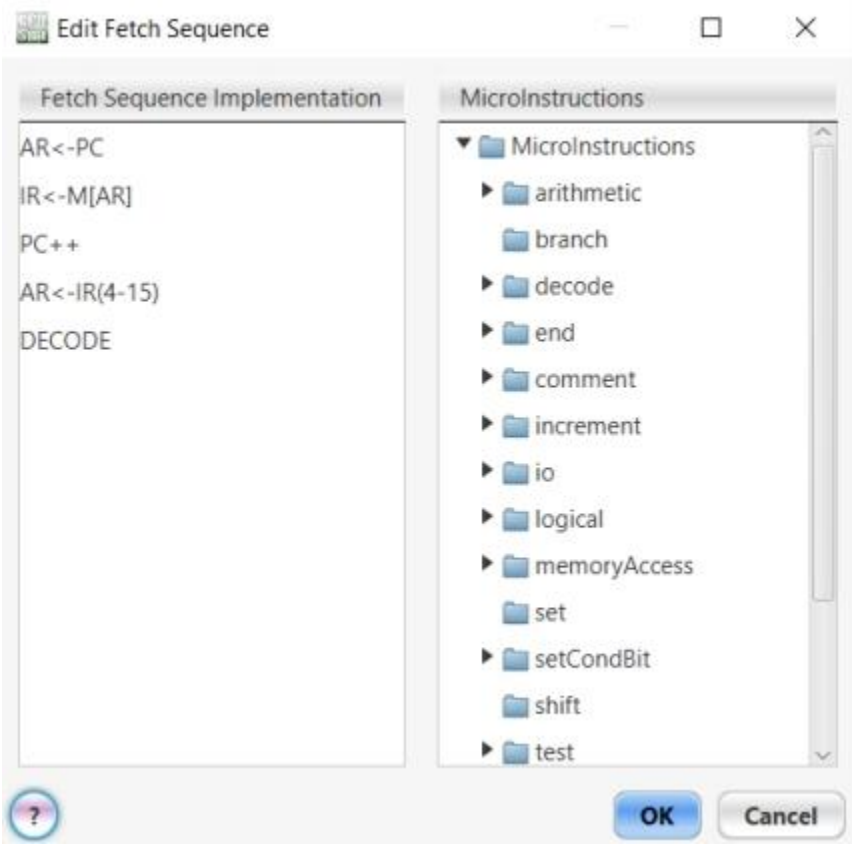


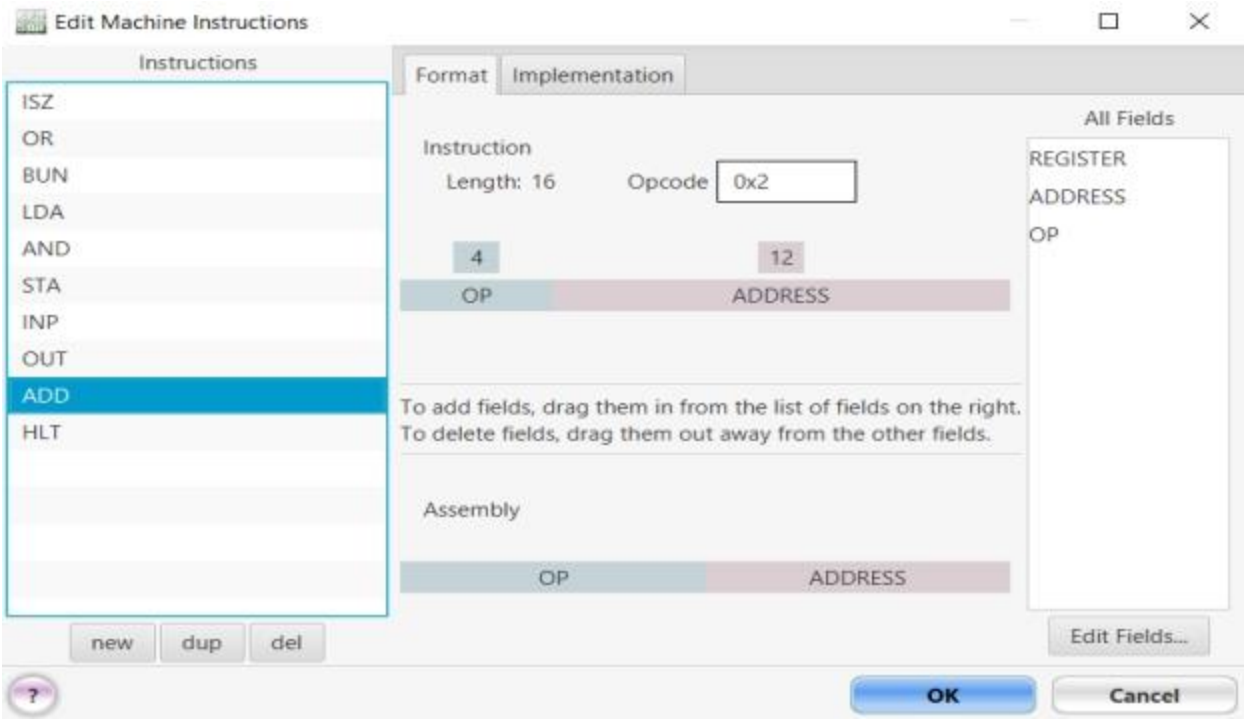


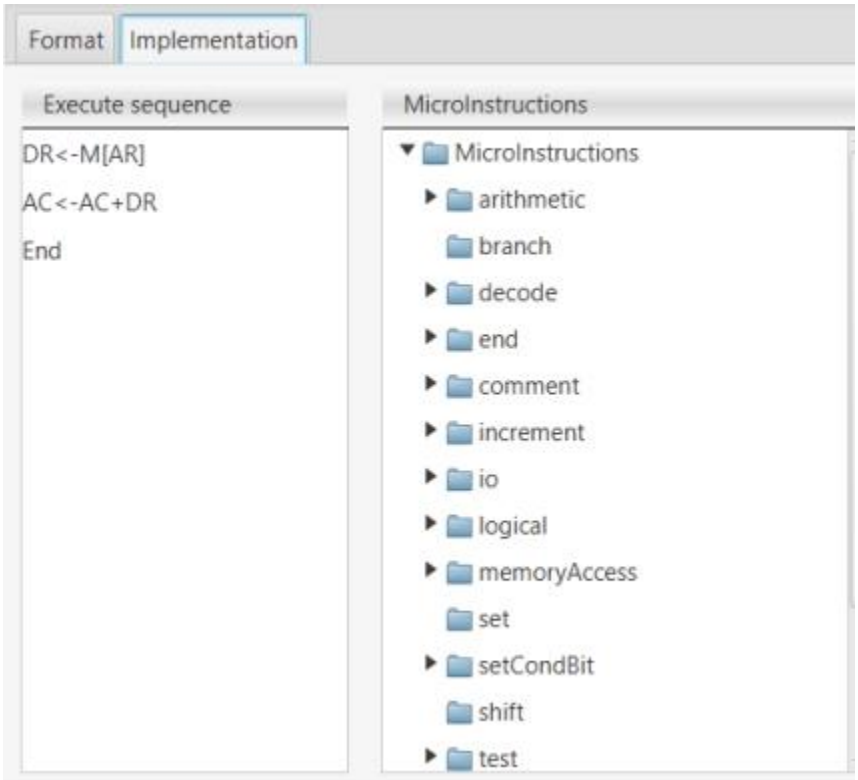

Edit Machine Instructions
Instructions
HLT
SZE
SZA
SNA
SPA
INC
CIL
CIR
CME
CMA
CLE
CLA
ISZ_I
ISZ
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE001
16
RIO
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
RIO
All Fields
RIO
OP
ADDR
Edit Fields...
OK
Cancel


Edit Machine Instructions
Instructions
ISZ_I
ISZ
BSA_I
BSA
BUN_I
BUN
STA_I
STA
LDA_I
LDA
ADD_I
ADD
AND_I
AND
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0x8
4
12
OP
ADDR
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
OP
ADDR
All Fields
RIO
OP
ADDR
Edit Fields...
OK
Cancel

2. Fetch Sequence


Edit Fetch Sequence
Fetch Sequence Implementation
AR<-PC
IR<-M[AR]
PC++
AR<-IR(4-15)
DECODE
MicroInstructions
MicroInstructions
arithmetic
branch
decode
end
comment
increment
io
logical
memoryAccess
set
setCondBit
shift
test
?
OK
Cancel


Edit Machine Instructions
Instructions
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
HLT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0x2
4
12
OP
ADDRESS
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
OP
ADDRESS
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel



OR × ISZ × ADD ×

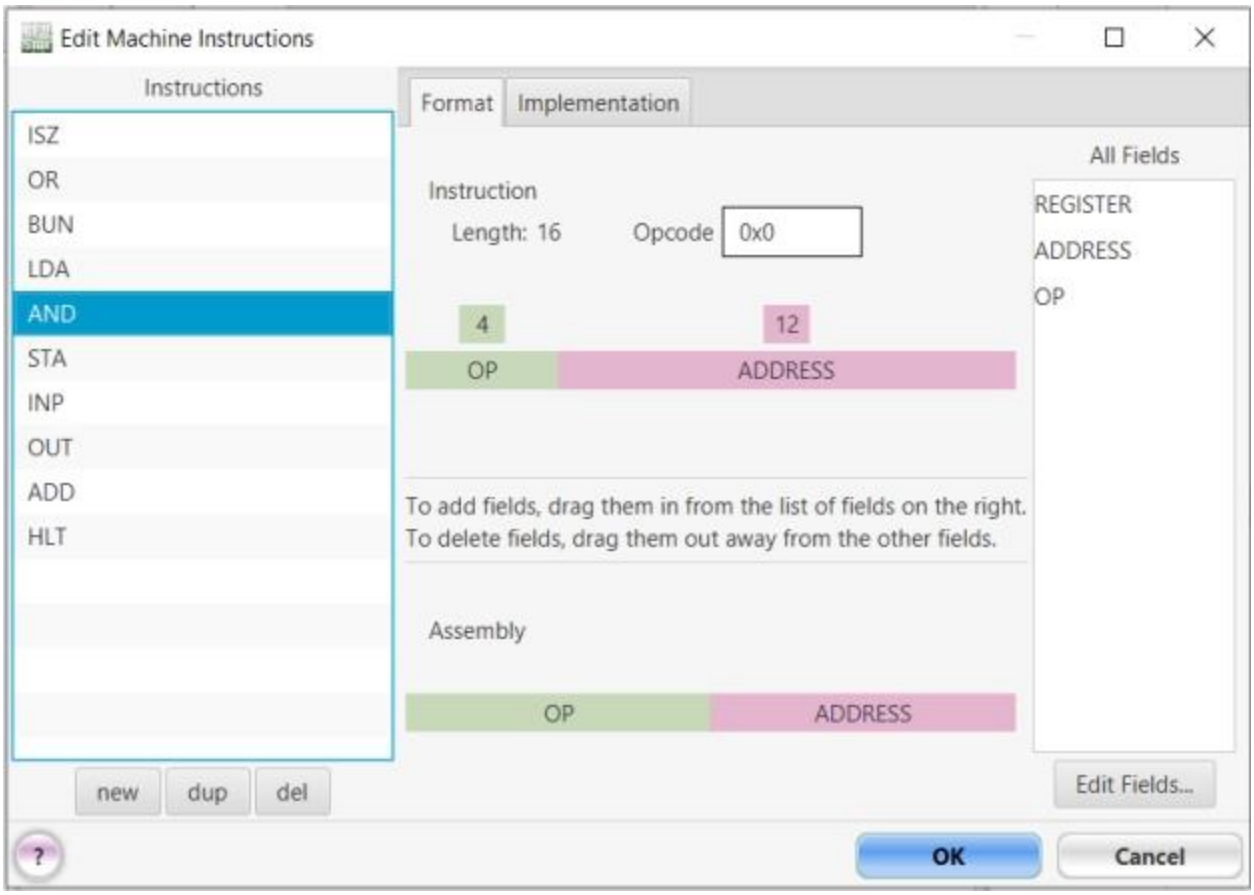
```
1 START: INP
2 STA NUM
3 INP
4 ADD NUM
5 OUT
6 HLT
7
8 NUM: .data 1 0
9
```

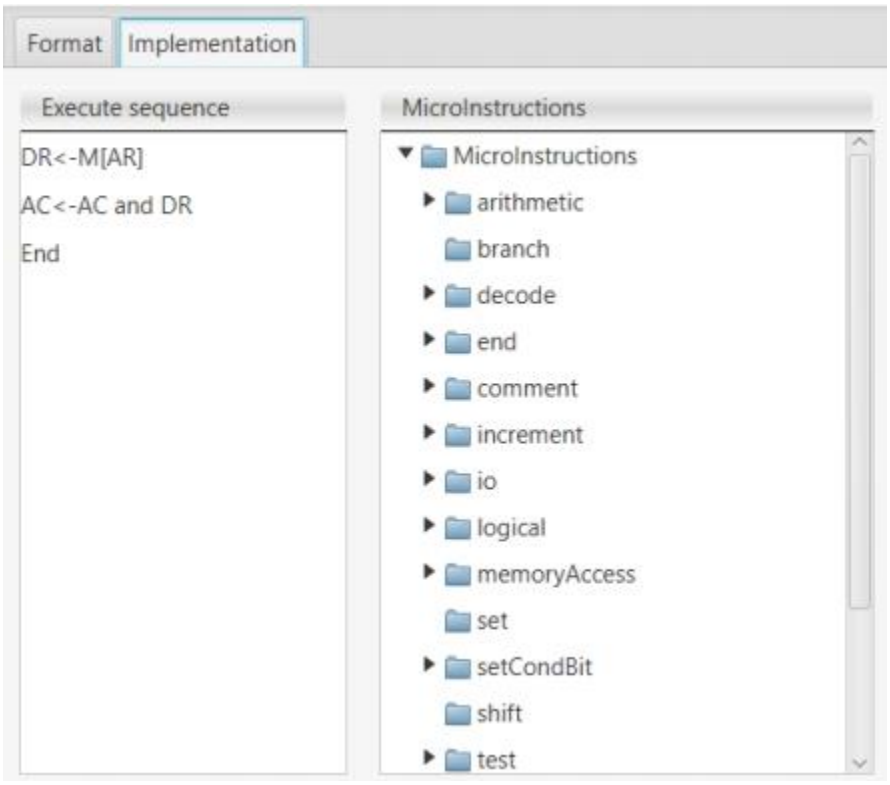
```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 12
Enter Inputs, the first of which must be an Integer: 22
Output:  34
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]
```

ADD × SUB ×

```
1 START: INP
2 STA NUM
3 INP
4 CMA
5 INC
6 ADD NUM
7 OUT
8 HLT
9
10 NUM: .data 1 0
11
```

```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 10
Enter Inputs, the first of which must be an Integer: 6
Output:  4
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT-BIT]
```





AND ×

```
1 INP
2 STA NUM
3 INP
4 AND NUM
5 OUT
6 HLT
7
8 NUM: .data 1 0
```

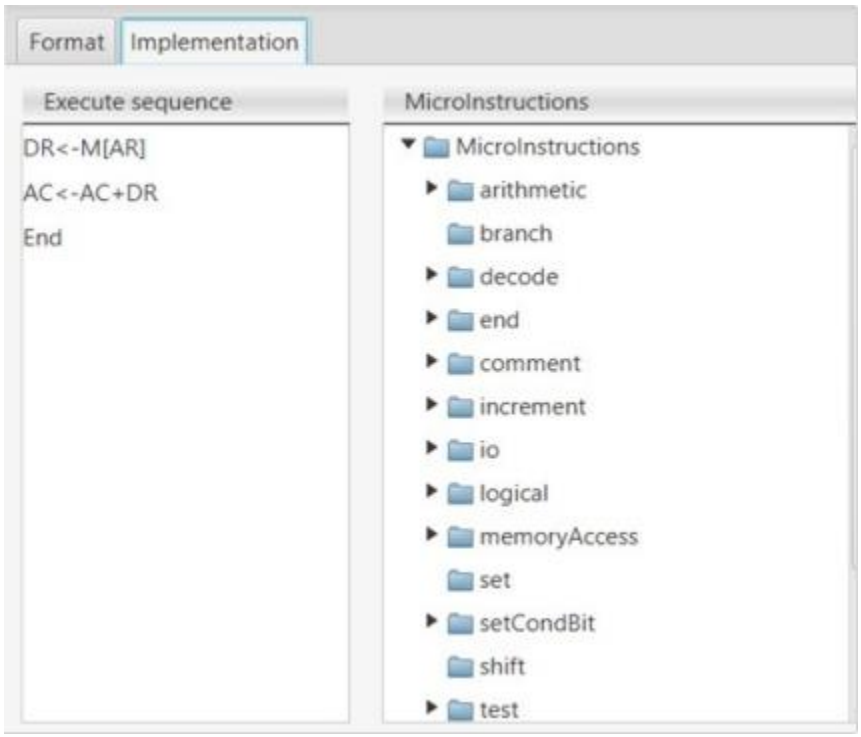
```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 5
Enter Inputs, the first of which must be an Integer: 6
Output:  4
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT-BIT]
```

MULTIPLY ×

```
1 START: INP
2 STA NUM
3 INP
4 MULTIPLY NUM
5 OUT
6 HLT
7
8 NUM: .data 1 0
9
```

```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 6
Enter Inputs, the first of which must be an Integer: 8
Output:  48
```

7.

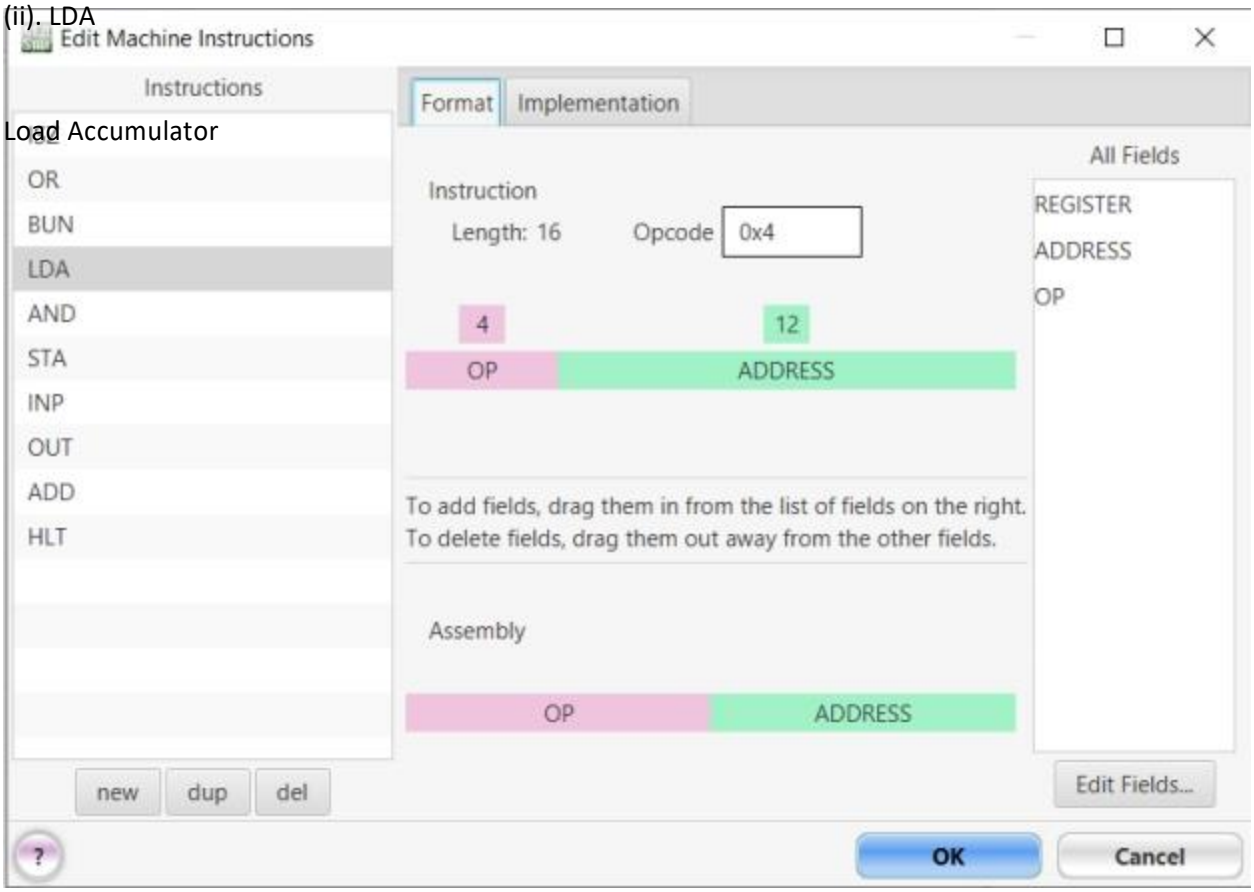


OR × ISZ × ADD ×

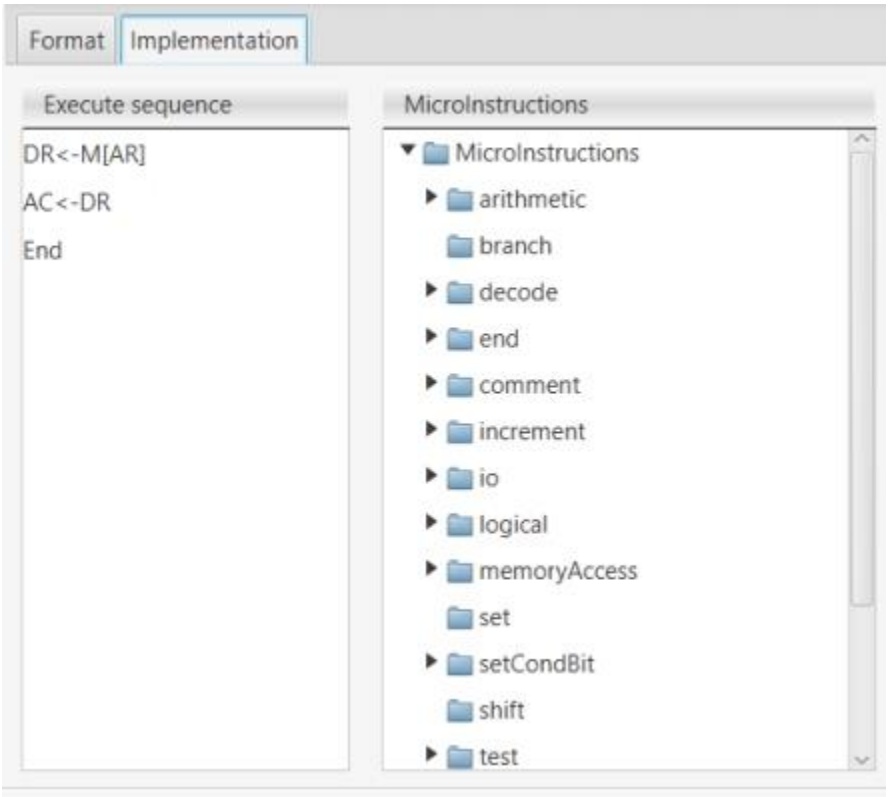
```
1 START: INP
2 STA NUM
3 INP
4 ADD NUM
5 OUT
6 HLT
7
8 NUM: .data 1 0
9
```

```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 12
Enter Inputs, the first of which must be an Integer: 22
Output:  34
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]
```

(ii). LDA

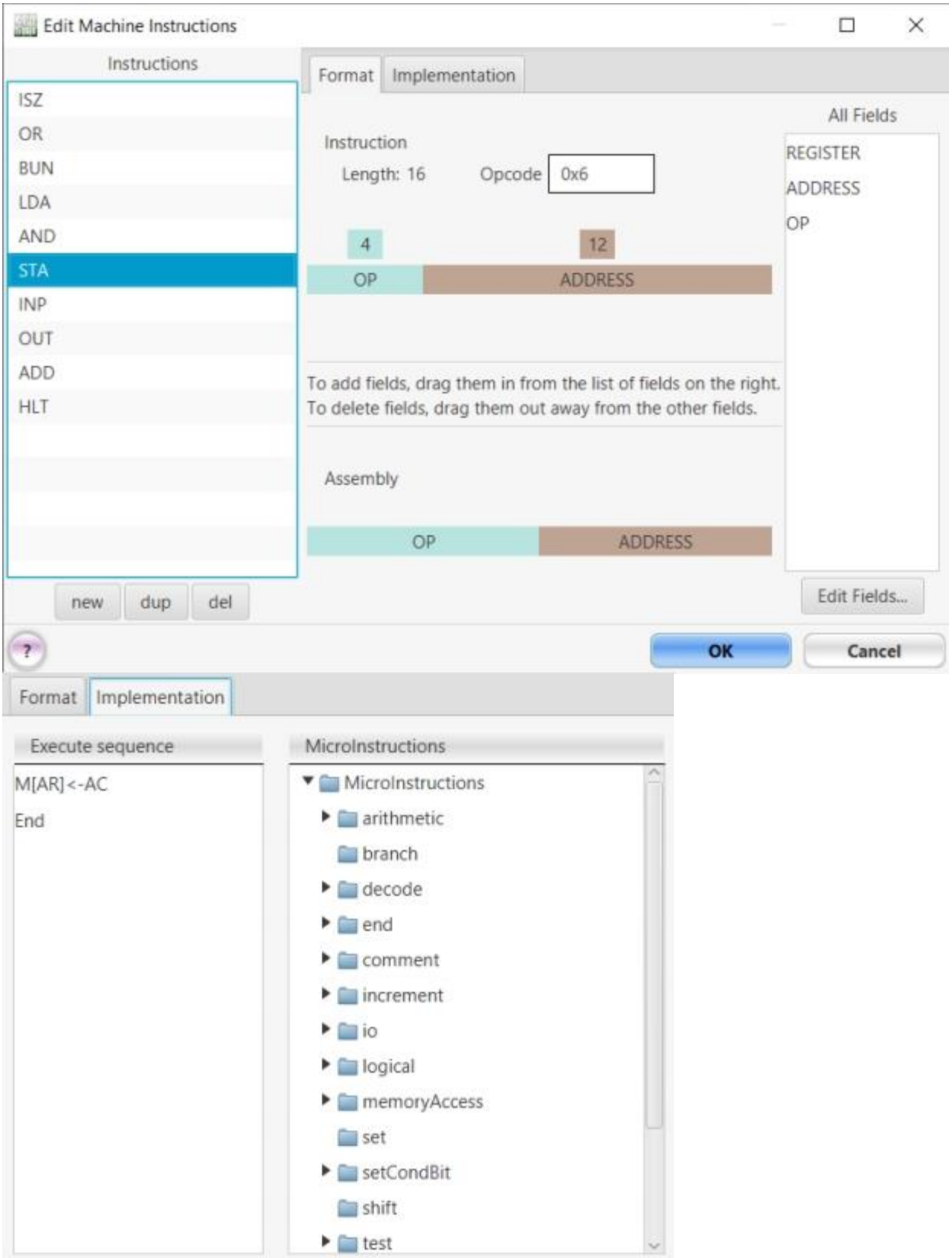


Load Accumulator



(iii). STA

Store Accumulator


Edit Machine Instructions
Instructions
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
HLT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0x6
4
12
OP
ADDRESS
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
OP
ADDRESS
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel
Format
Implementation
Execute sequence
M[AR]<-AC
End
MicroInstructions
MicroInstructions
arithmetic
branch
decode
end
comment
increment
io
logical
memoryAccess
set
setCondBit
shift
test

LDA × STA ×

```
1 INP
2 STA NUM
3 OUT
4 HLT
5
6 NUM: .data 1 0
```

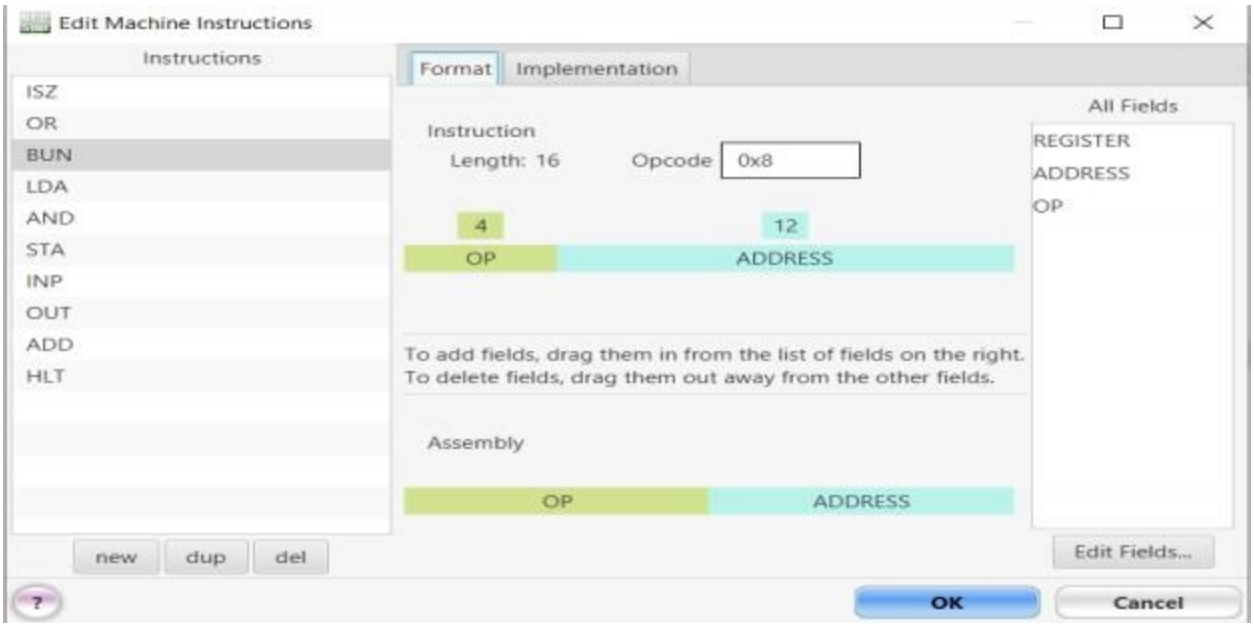
Addr Hex Data Dec

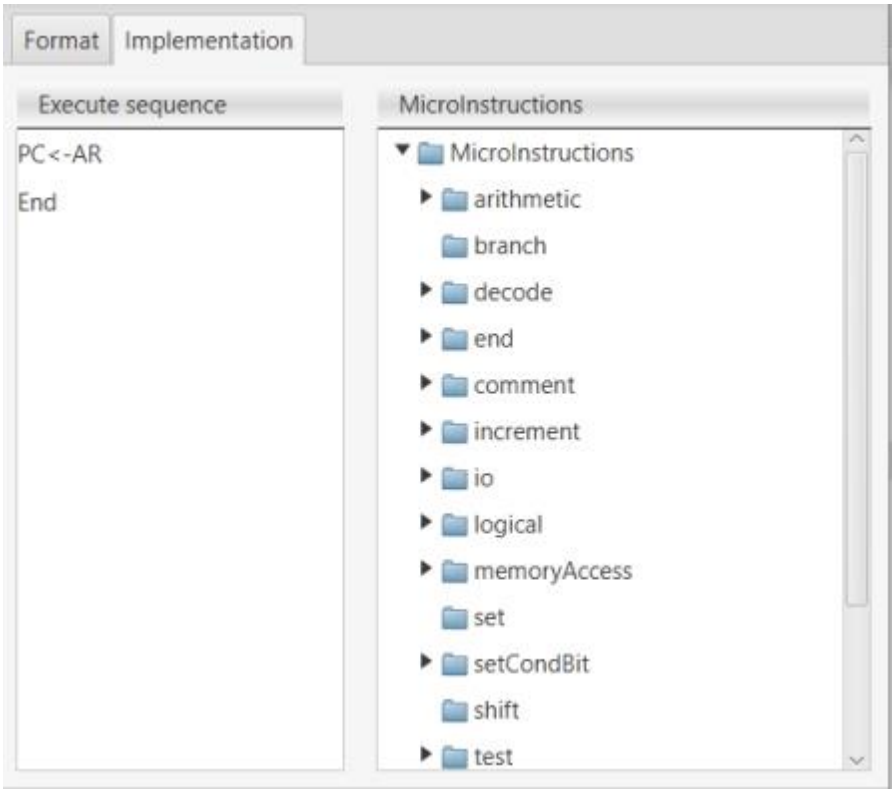
M

Addr	Data
024	62464
025	57345
026	68
027	0
028	0
029	0
02A	0

```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 68
Output:  68
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT-BIT]
```

(iv). BUN

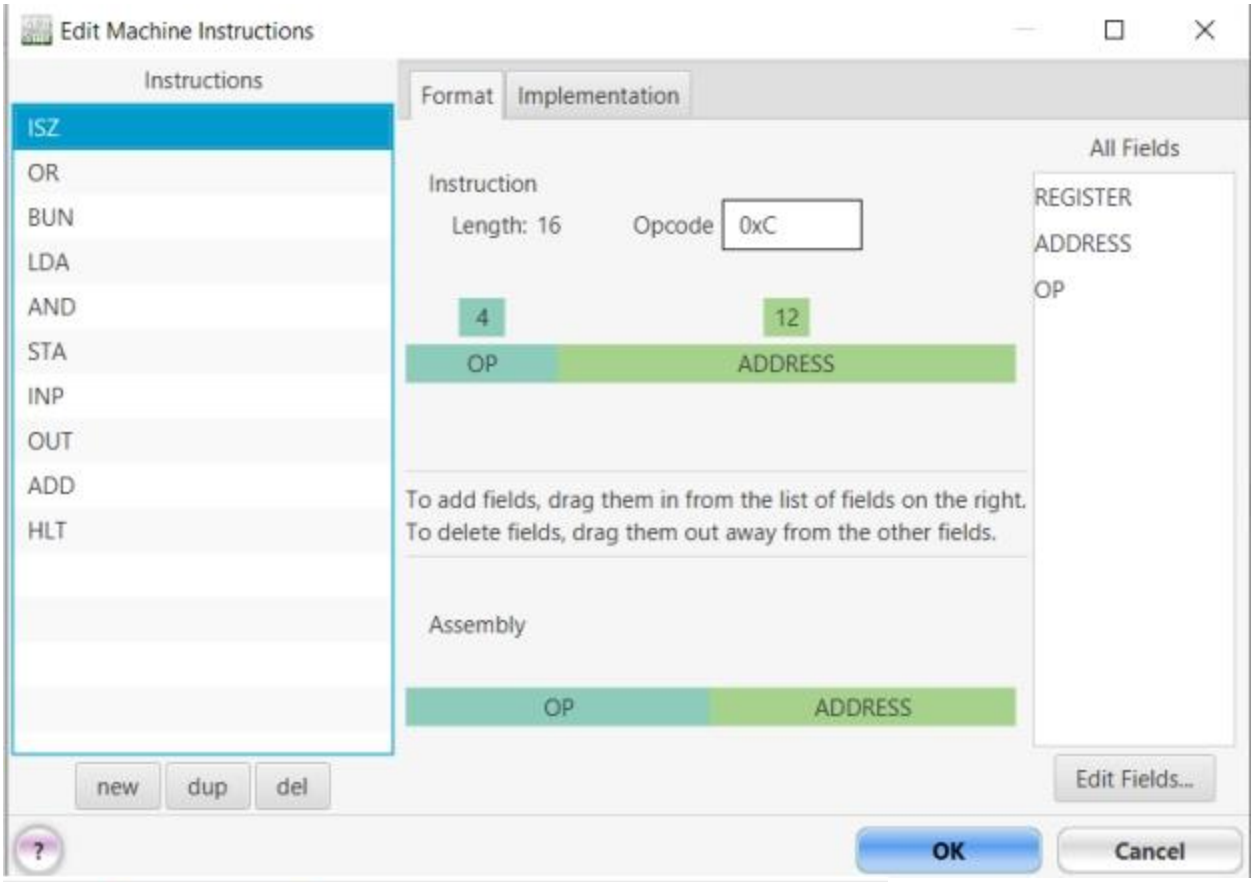


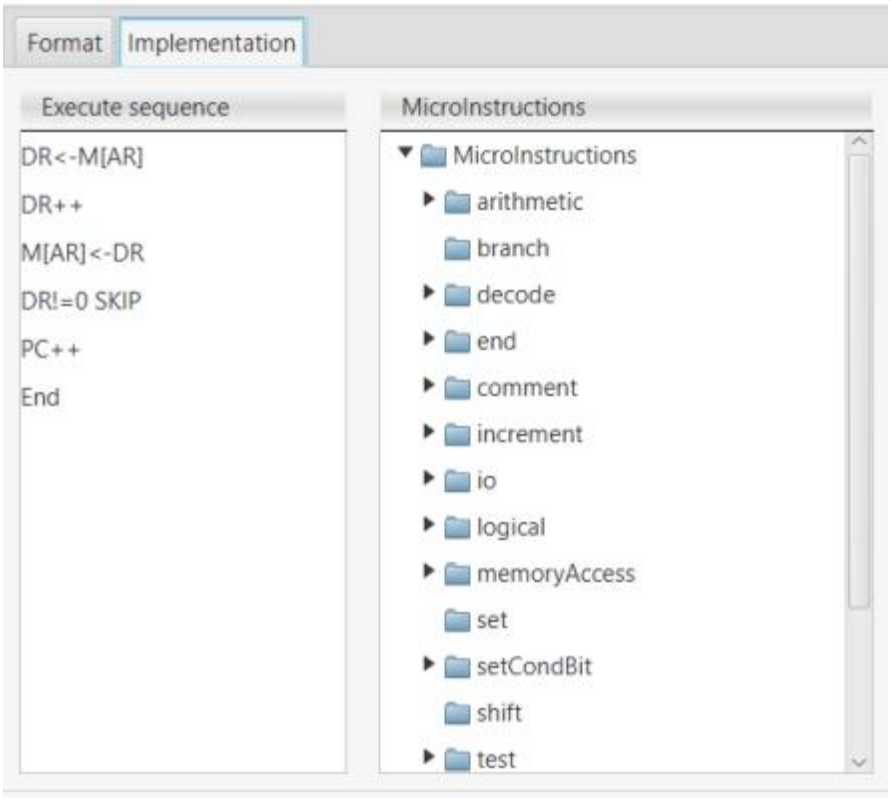



```
EXECUTING...
Enter Inputs, the first of which must be an Integer: 5
Output:  5
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT-BIT]
```

(v).ISZ

Increment-Skip-ifZero

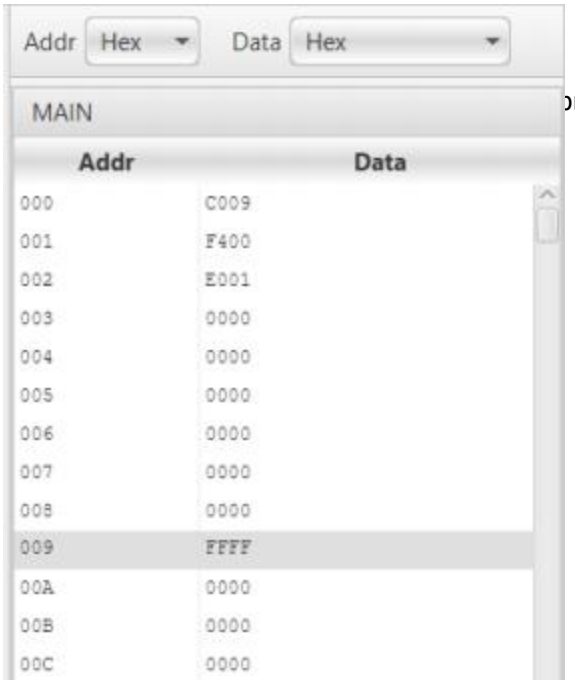

Edit Machine Instructions
Instructions
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
HLT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xC
4
12
OP
ADDRESS
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
OP
ADDRESS
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel


Format
Implementation
Execute sequence
DR<-M[AR]
DR++
M[AR]<-DR
DR!=0 SKIP
PC++
End
MicroInstructions
MicroInstructions
arithmetic
branch
decode
end
comment
increment
io
logical
memoryAccess
set
setCondBit
shift
test

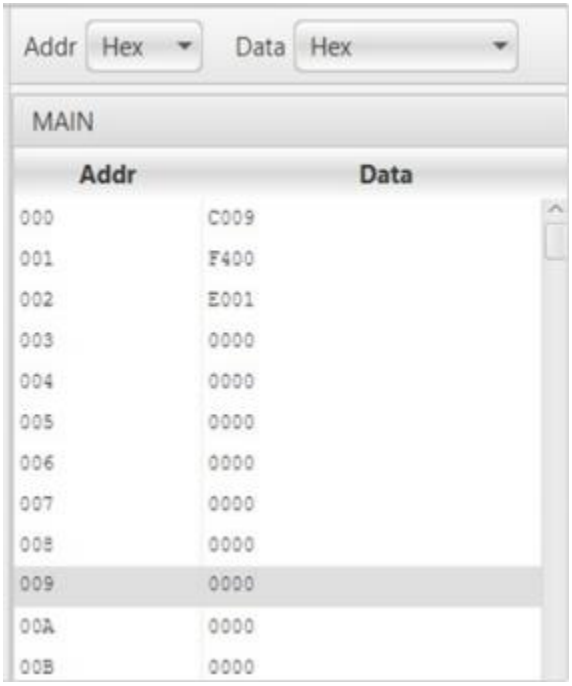
OR × ISZ ×

```
1 ISZ 009
2 OUT
3 HLT
```

First we assembled and loaded the program



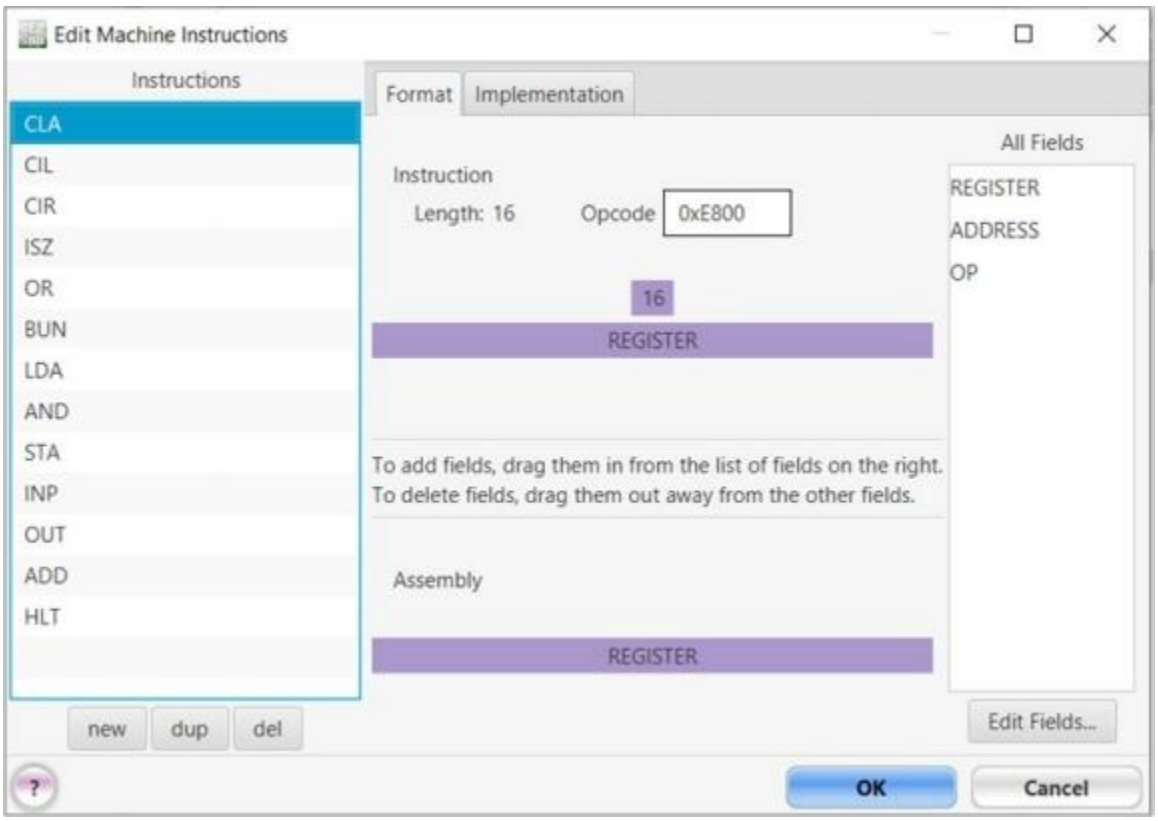
program - the data is incremented and is 0000.

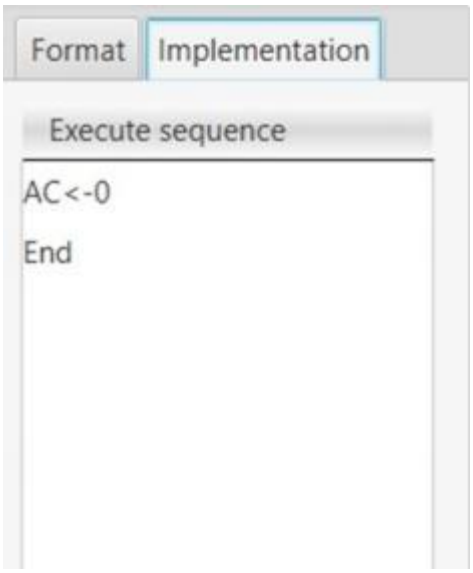



```
EXECUTING...
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]
```

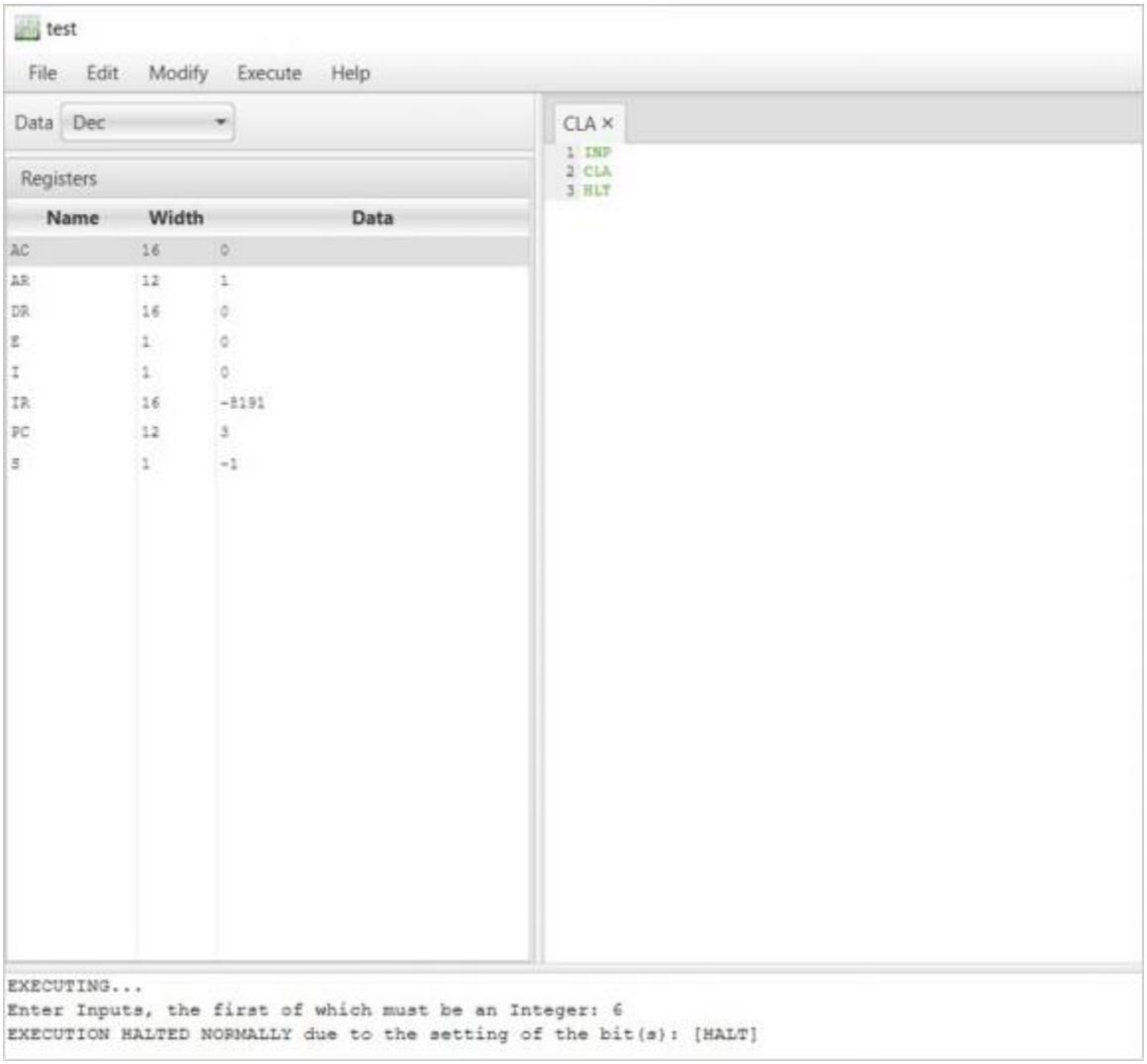
9.

(i).CLA

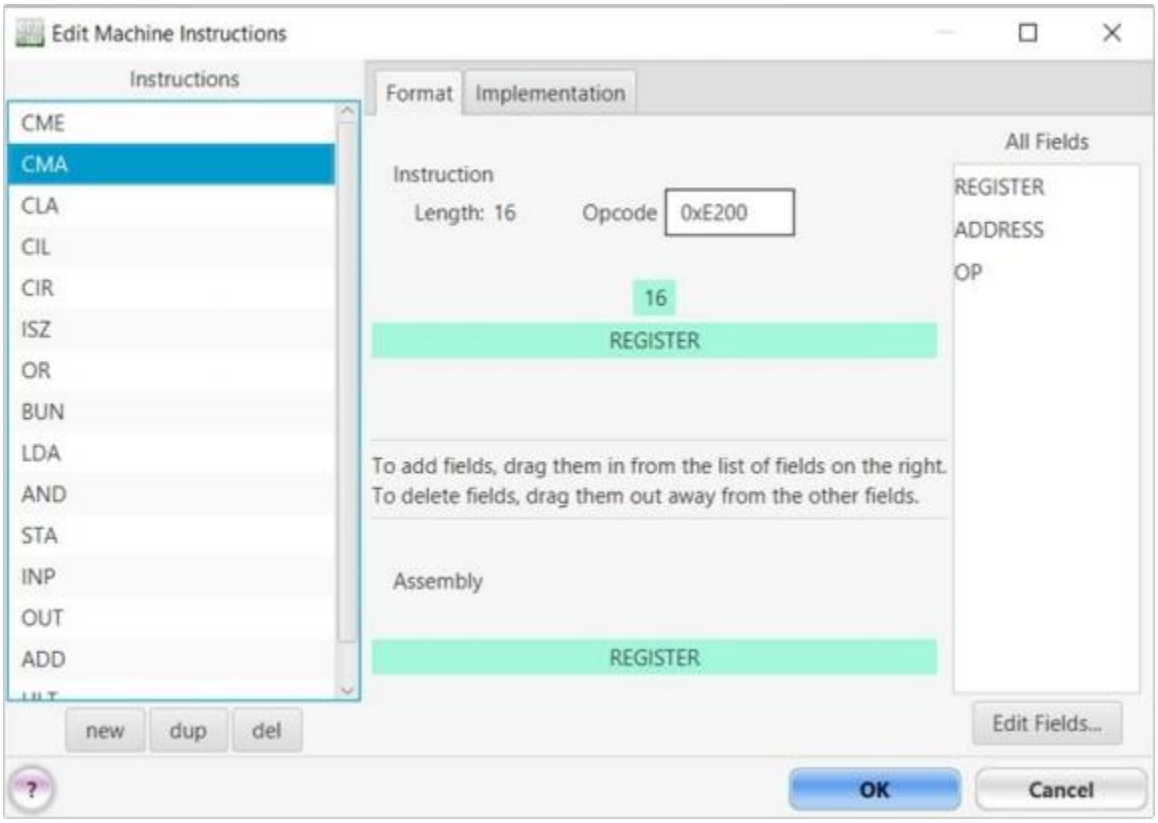


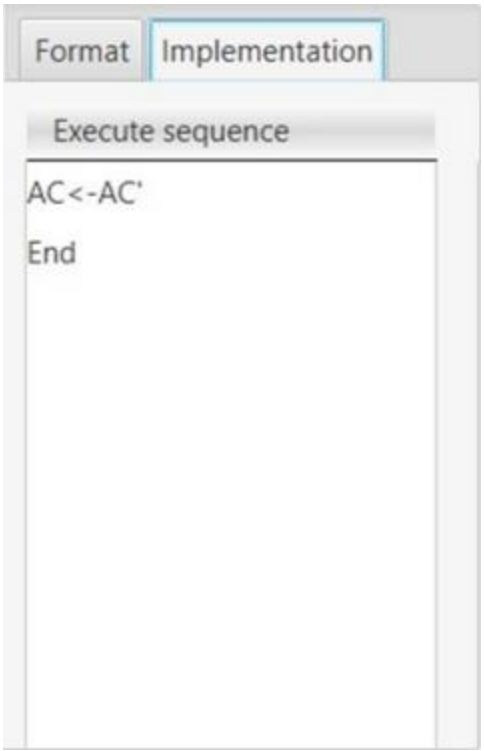


Accumulator is clear after execution

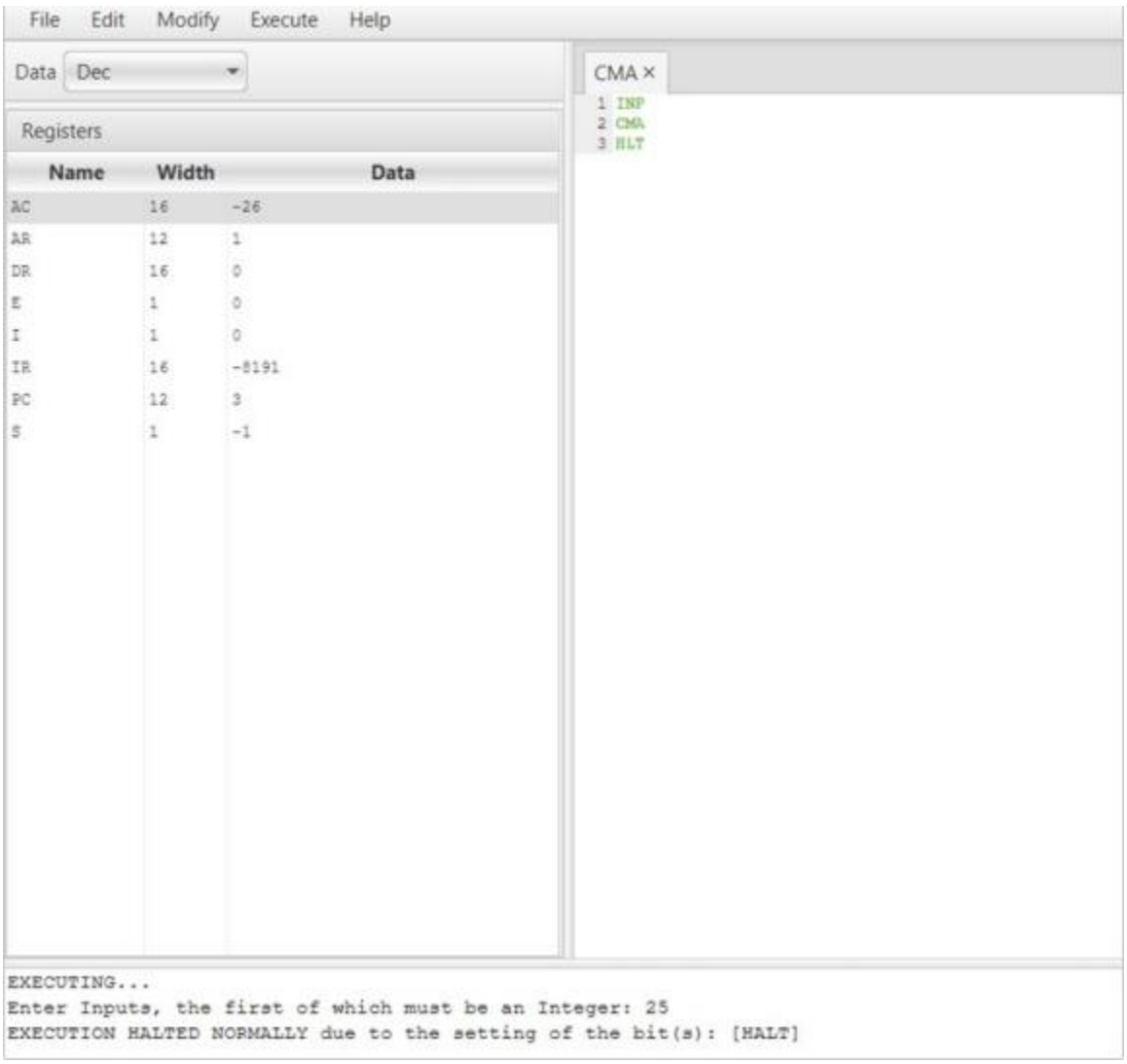

test
File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 0
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 3
S 1 -1
CLA ×
1 INP
2 CLA
3 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 6
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

(ii).CMA

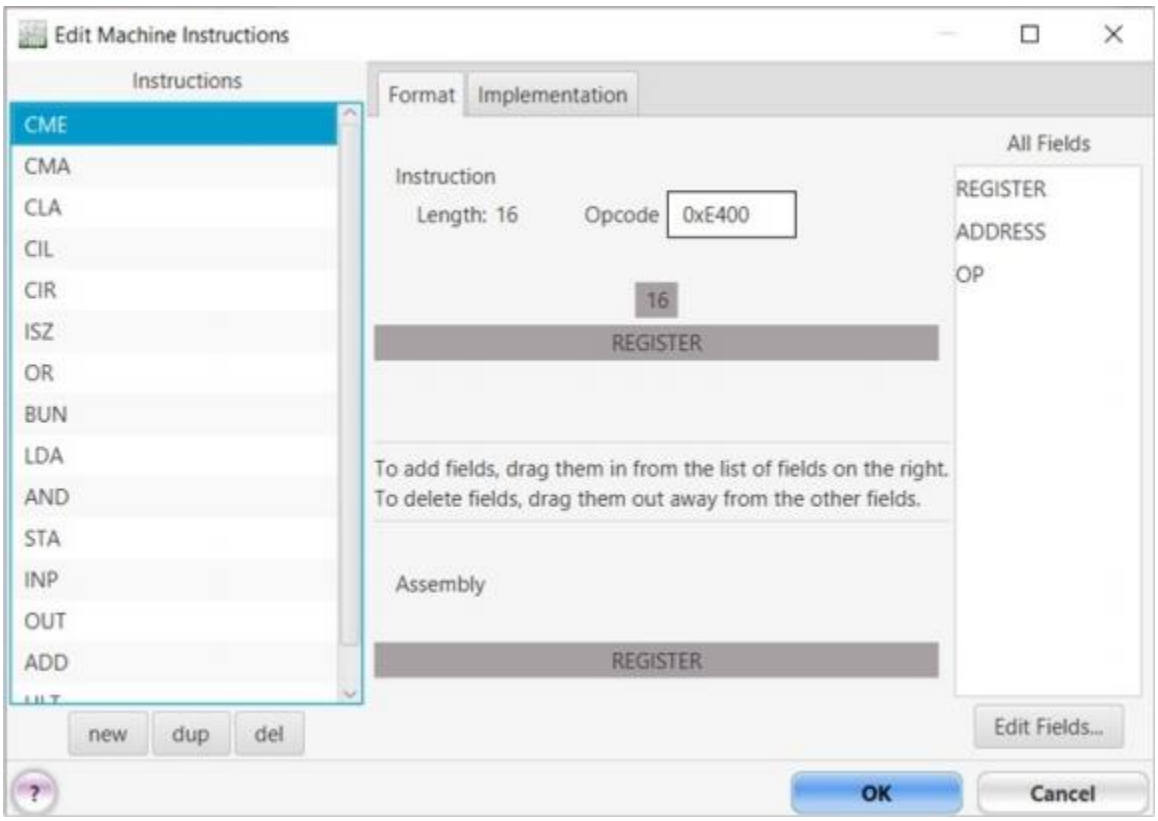


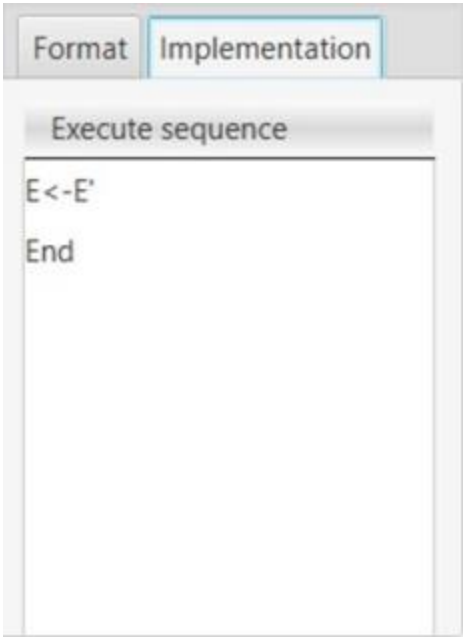


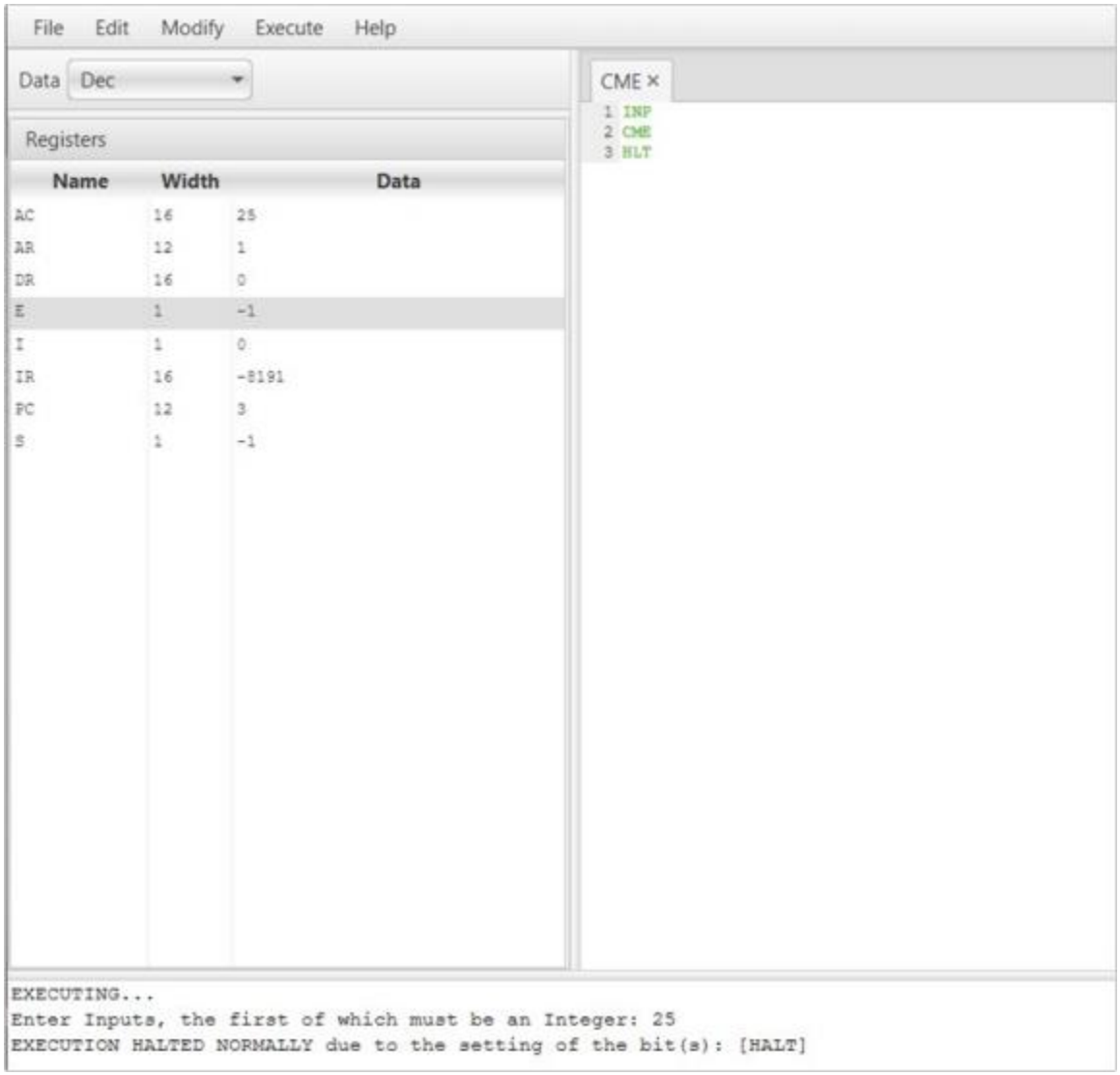
Accumulator holds the compliment of input value


File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 -26
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 3
S 1 -1
CMA ×
1 INP
2 CMA
3 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 25
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

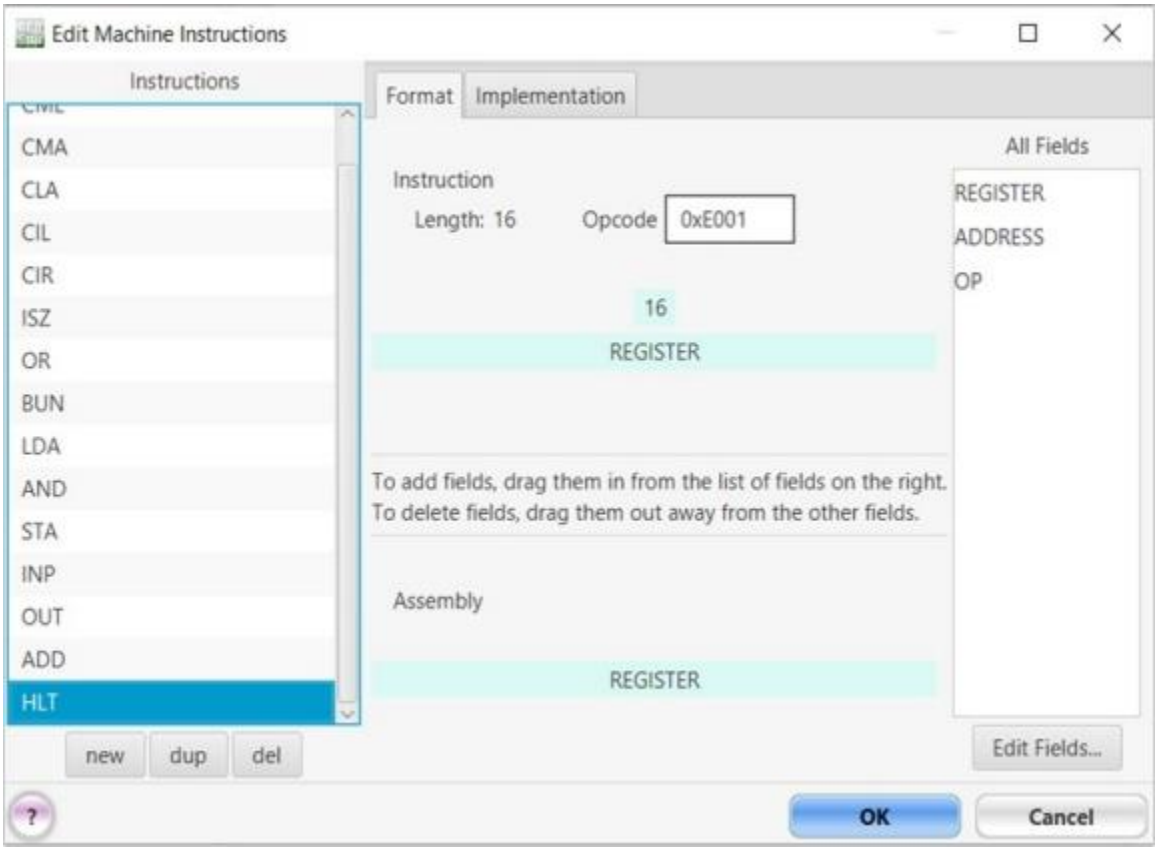
(iii).CME

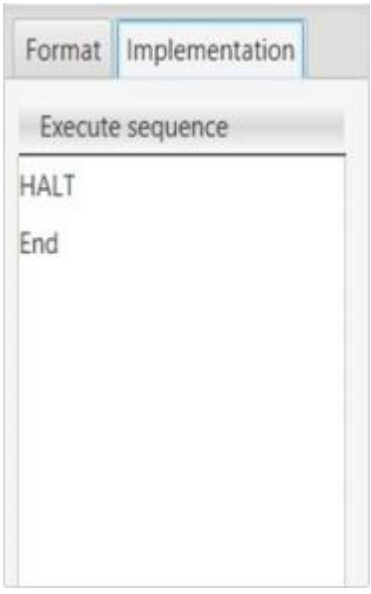

Edit Machine Instructions
Instructions
CME
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE400
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
OK
Cancel

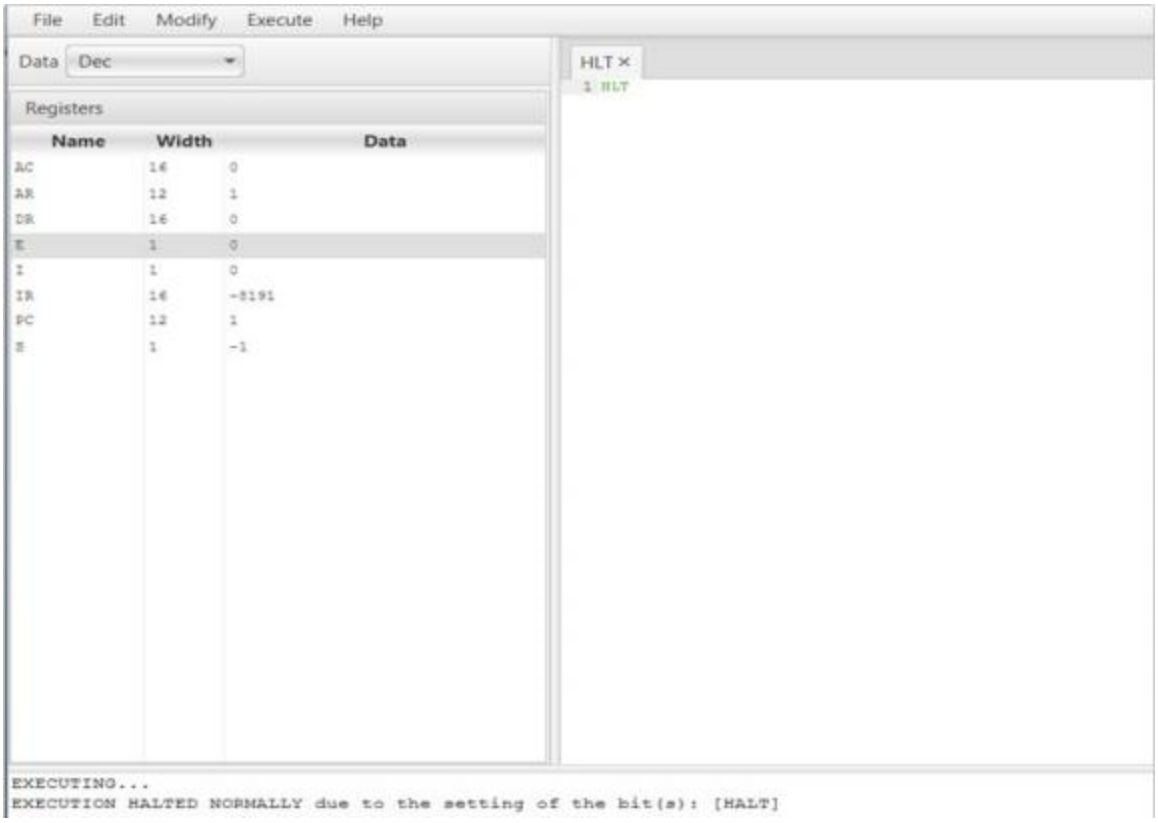

Format
Implementation
Execute sequence
E<-E'
End


File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 25
AR 12 1
DR 16 0
E 1 -1
I 1 0
IR 16 -8191
PC 12 3
S 1 -1
CME ×
1 INP
2 CME
3 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 25
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

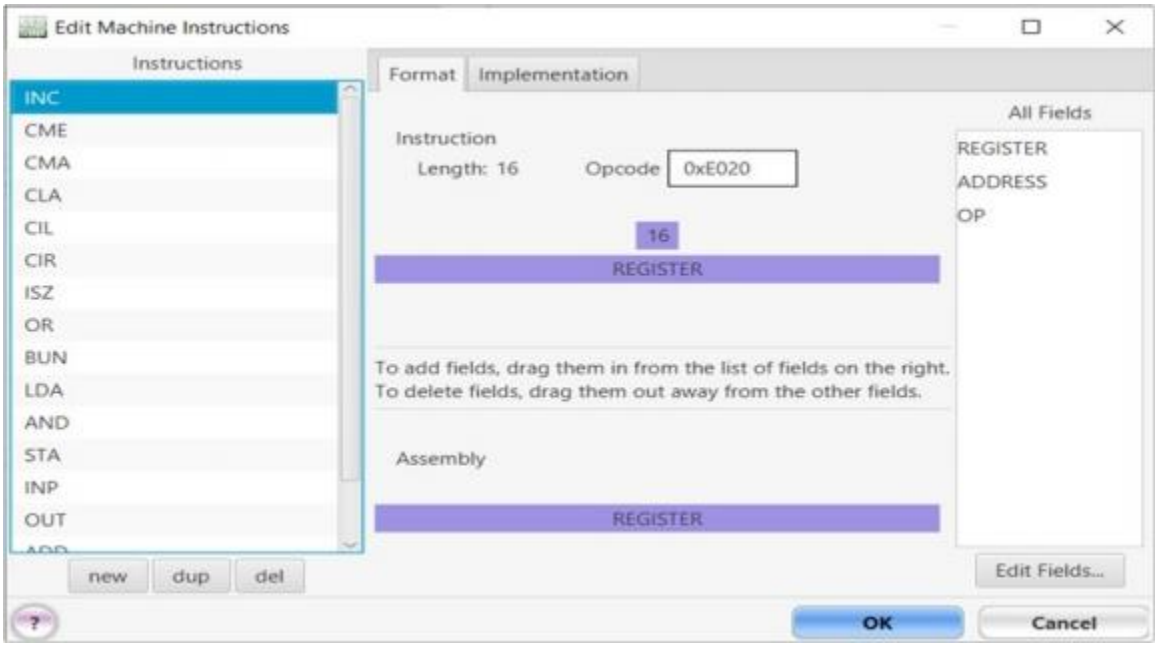
(iv).HLT


Edit Machine Instructions
Instructions
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
HLT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE001
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel

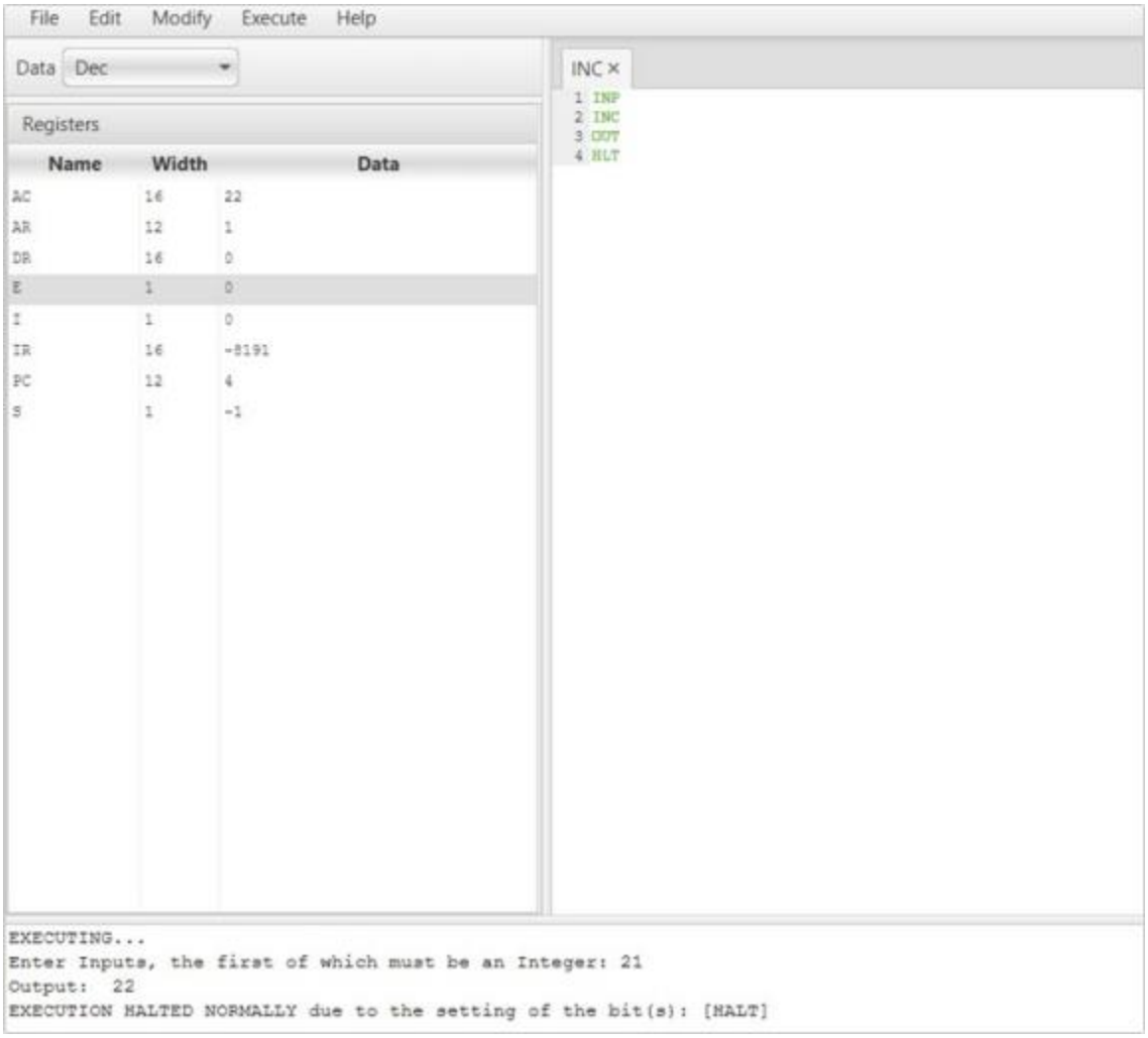

Format
Implementation
Execute sequence
HALT
End


File
Edit
Modify
Execute
Help
Data
Dec
Registers
Name
Width
Data
AC 16 0
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 1
S 1 -1
HLT ×
1 HLT
EXECUTING...
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

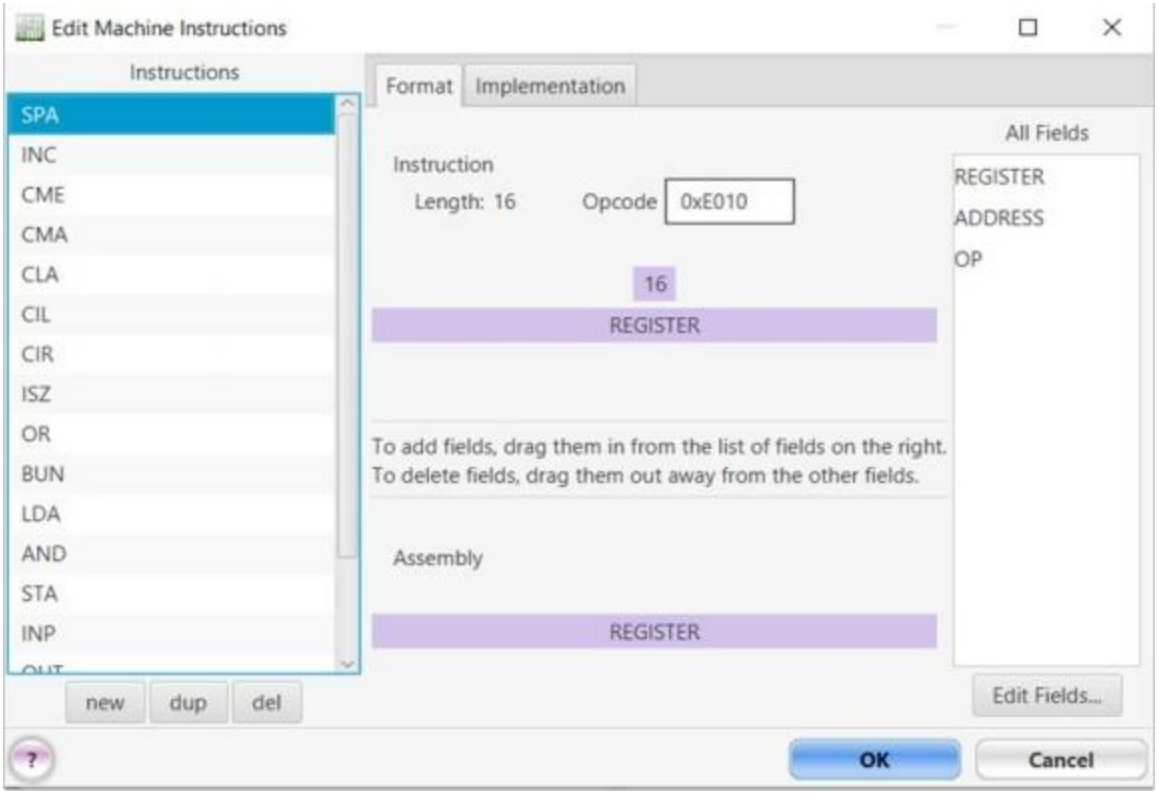
9.

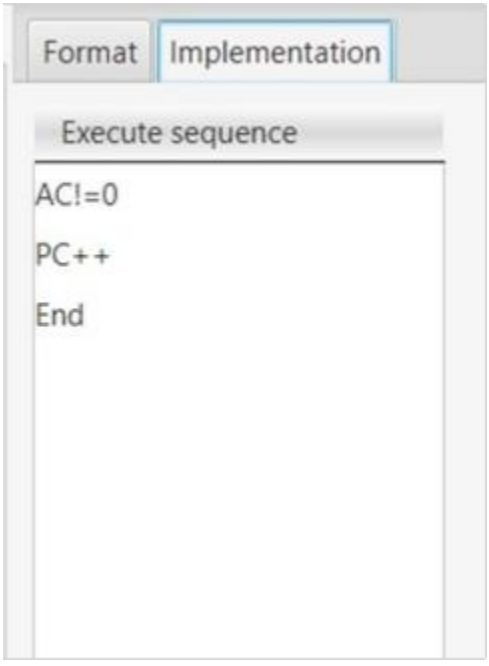

Edit Machine Instructions
Instructions
INC
CME
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE020
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
OK
Cancel

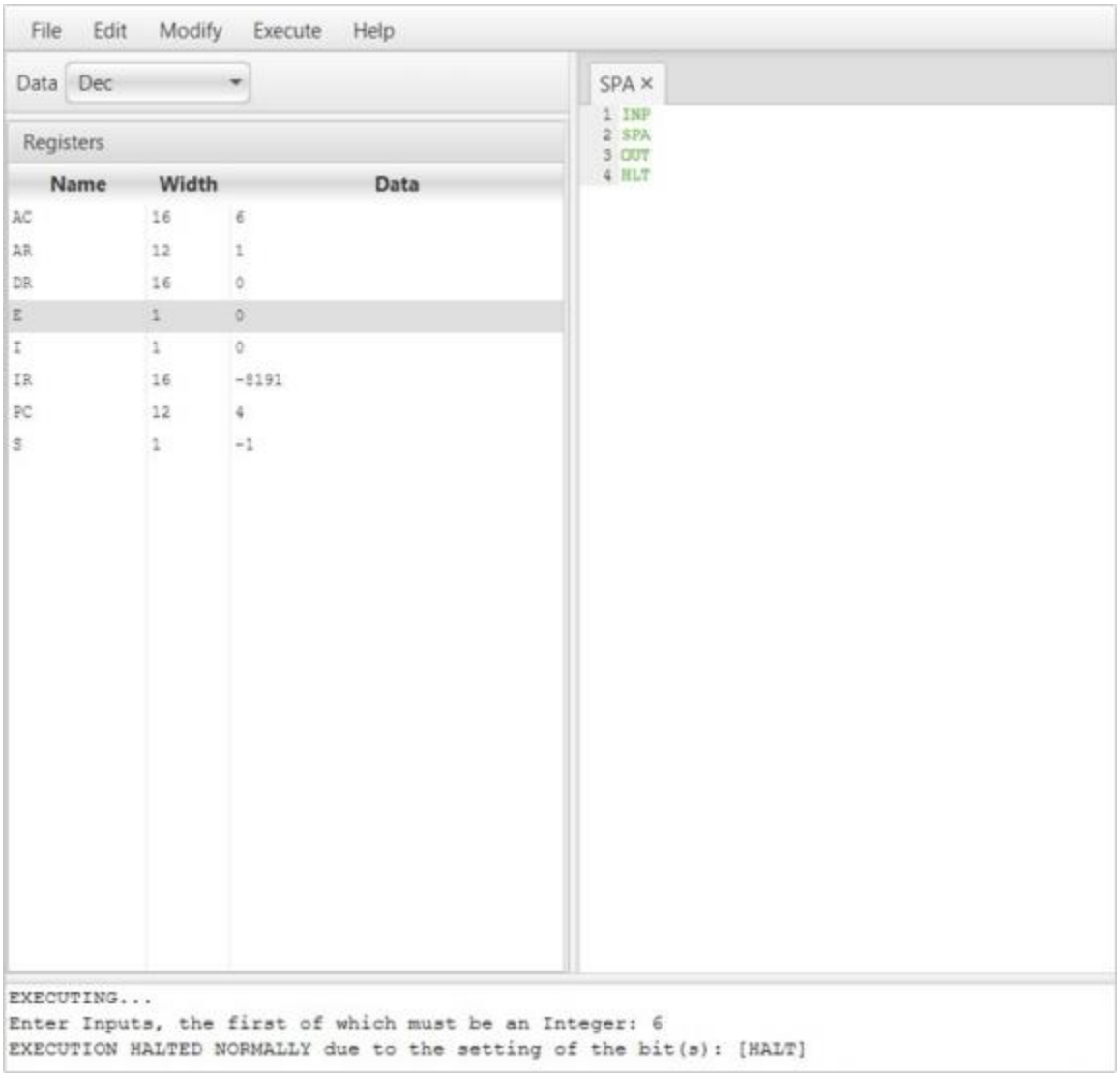

Format
Implementation
Execute sequence
AC++
End


File
Edit
Modify
Execute
Help
Data
Dec
Registers
Name
Width
Data
AC 16 22
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 4
S 1 -1
INC ×
1 INP
2 INC
3 OUT
4 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 21
Output: 22
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

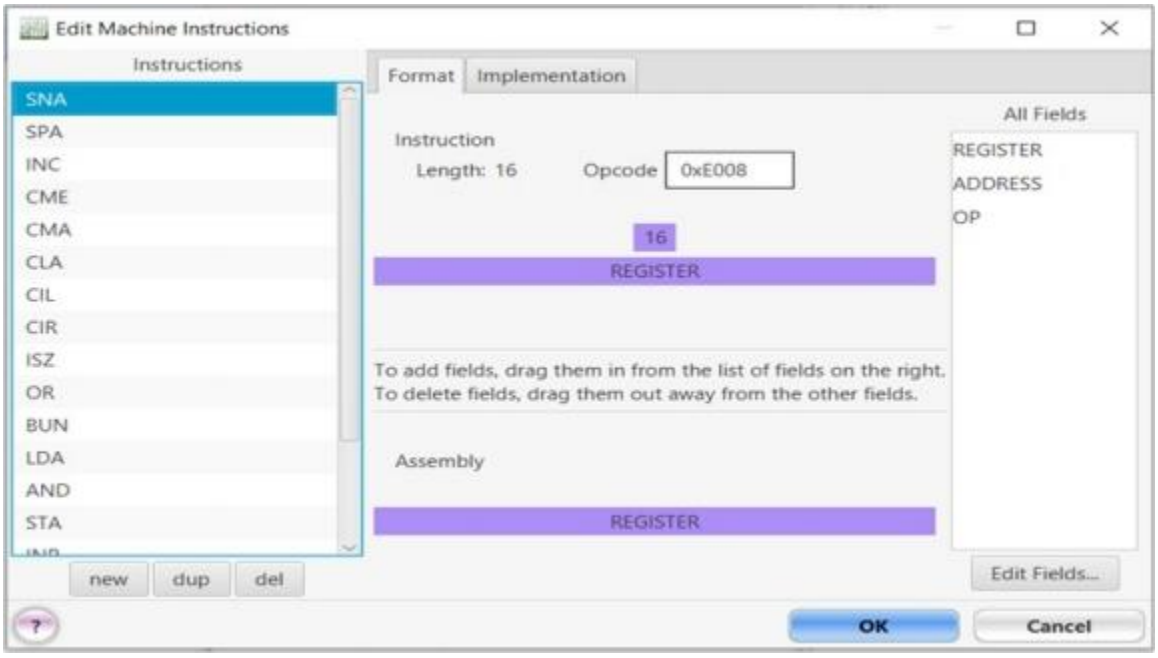
(ii).SPA

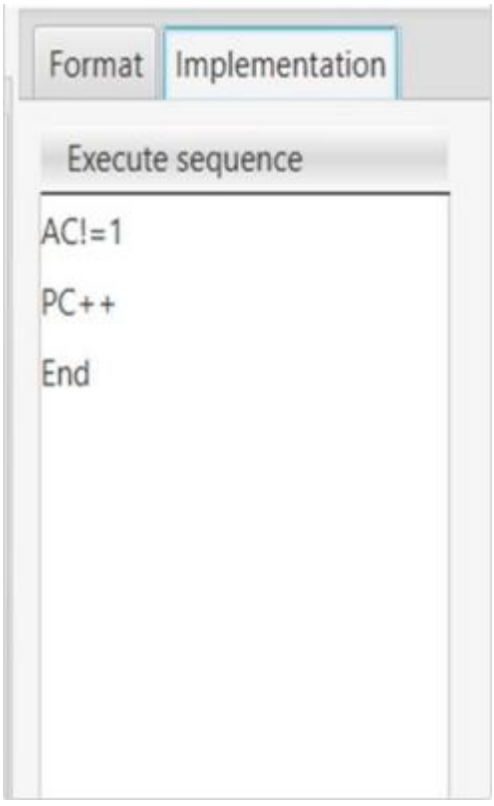

Edit Machine Instructions
Instructions
SPA
INC
CME
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
INP
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE010
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
OK
Cancel

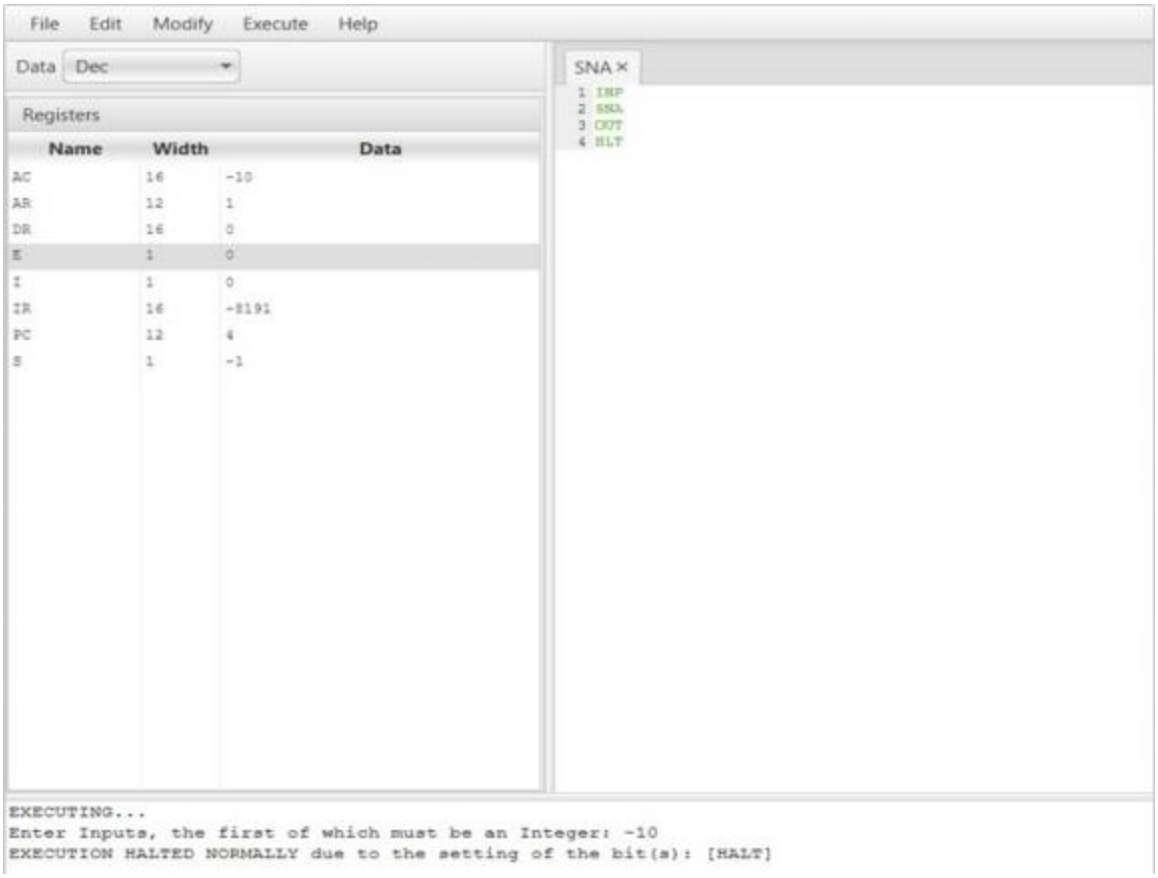

Format
Implementation
Execute sequence
AC!=0
PC++
End

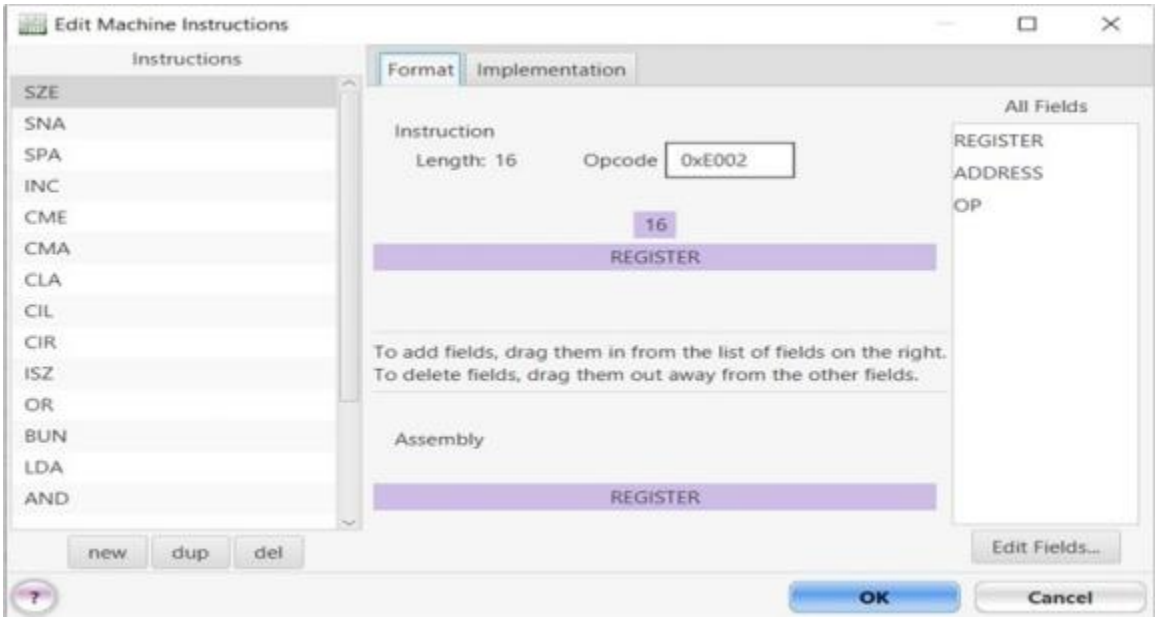

File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 6
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 4
S 1 -1
SPA ×
1 INP
2 SPA
3 OUT
4 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 6
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

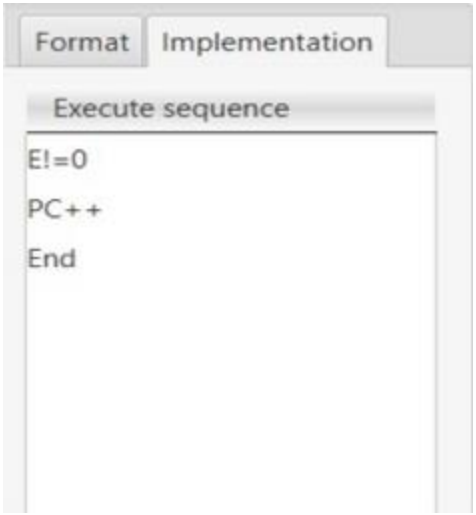
(iii).SNA

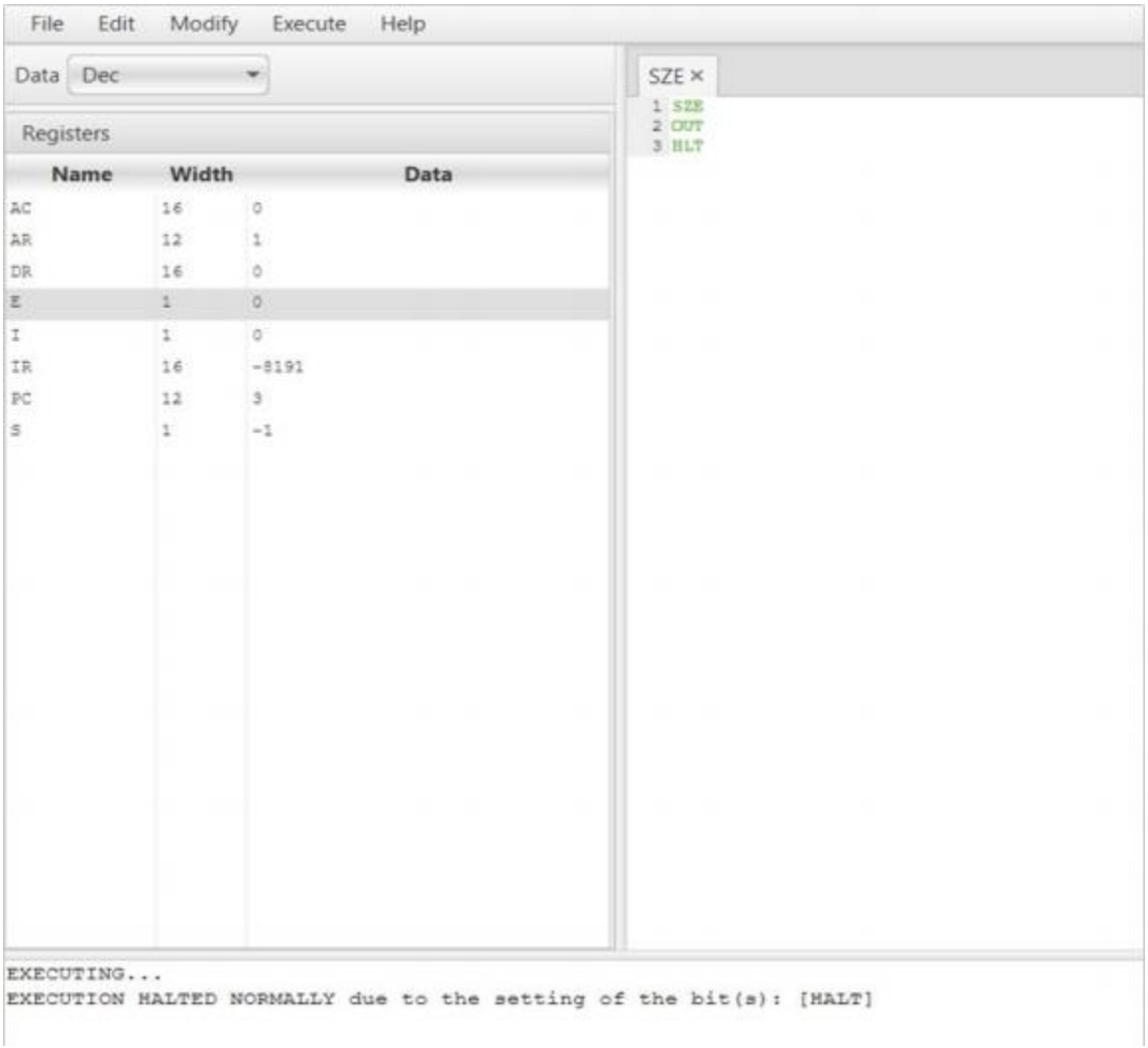

Edit Machine Instructions
Instructions
SNA
SPA
INC
CME
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE008
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
OK
Cancel


Format
Implementation
Execute sequence
AC!=1
PC++
End


File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 -10
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 4
S 1 -1
SNA ×
1 INP
2 SNA
3 OUT
4 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: -10
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

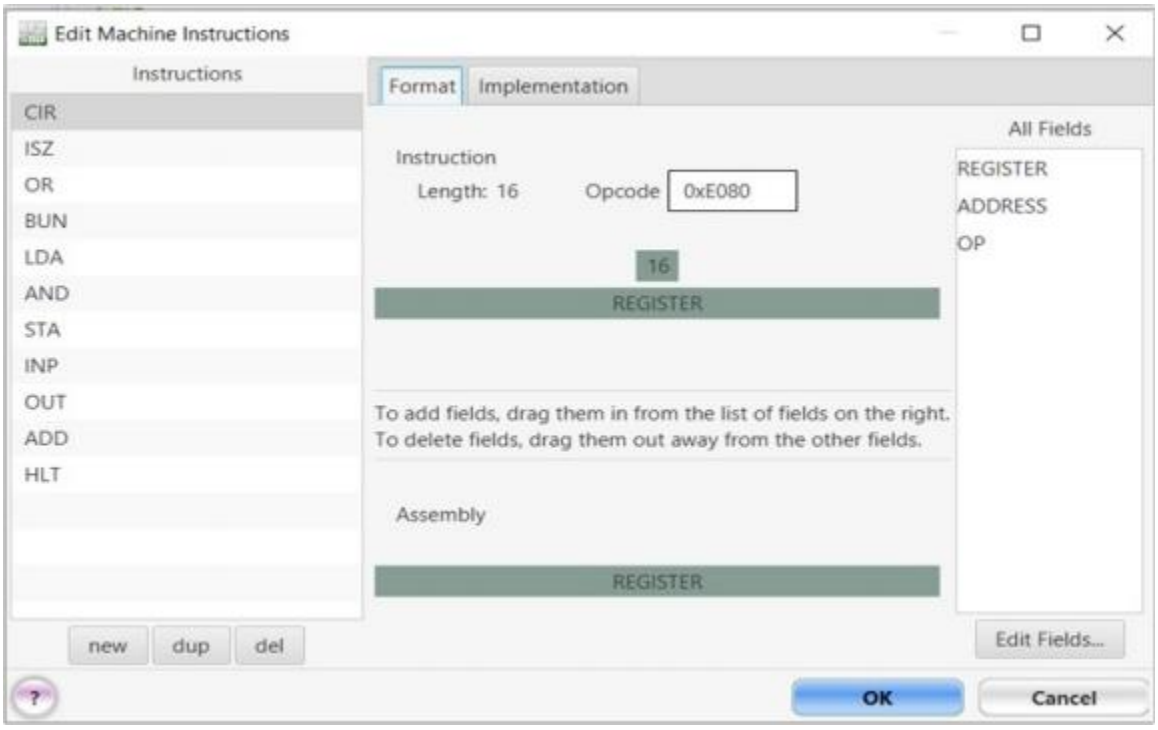

Edit Machine Instructions
Instructions
SZE
SNA
SPA
INC
CME
CMA
CLA
CIL
CIR
ISZ
OR
BUN
LDA
AND
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode 0xE002
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel

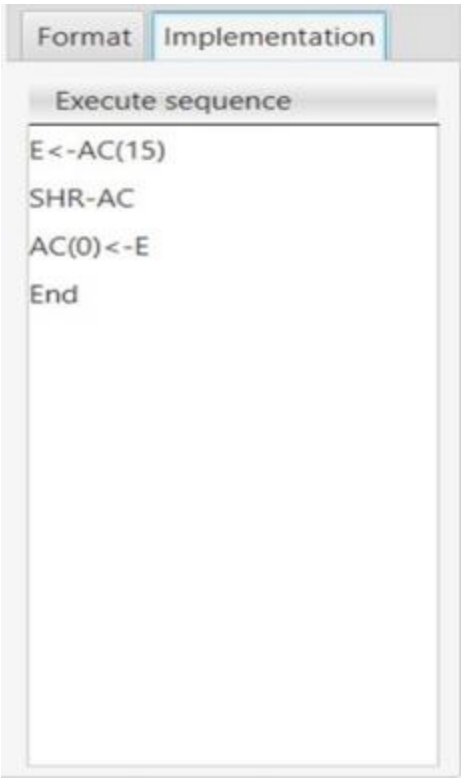

Format
Implementation
Execute sequence
E!=0
PC++
End


File
Edit
Modify
Execute
Help
Data
Dec
Registers
Name
Width
Data
AC 16 0
AR 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 3
S 1 -1
SZE ×
1 SZE
2 OUT
3 HLT
EXECUTING...
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]

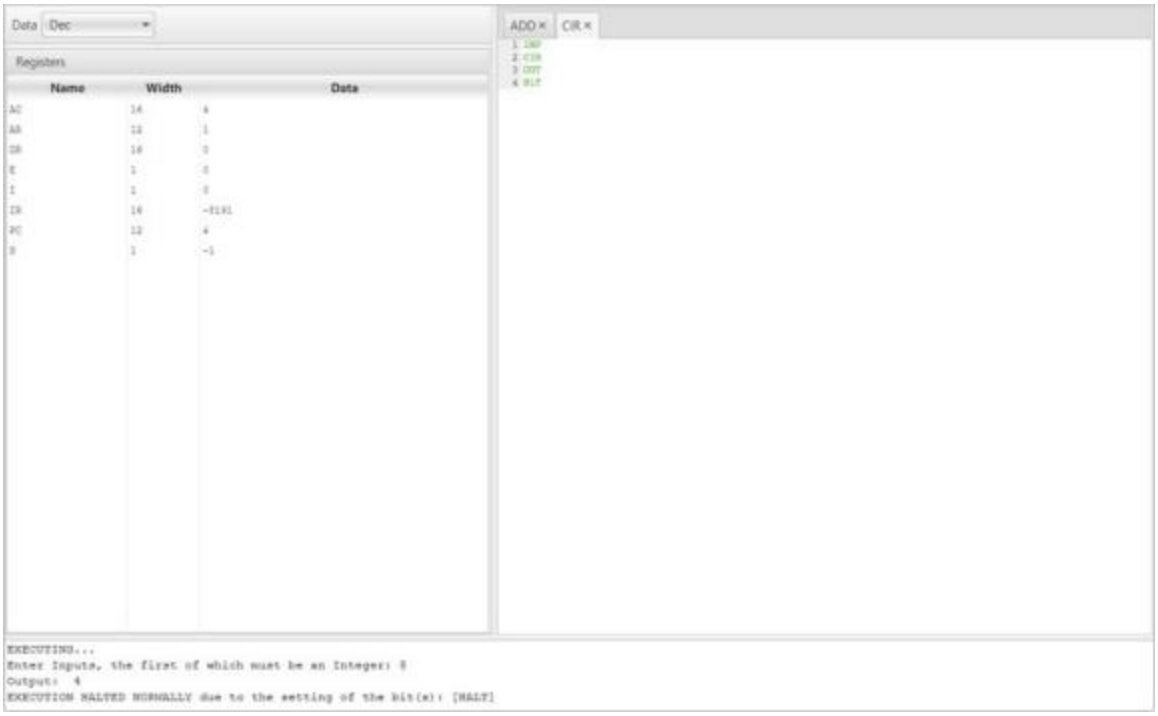
10.

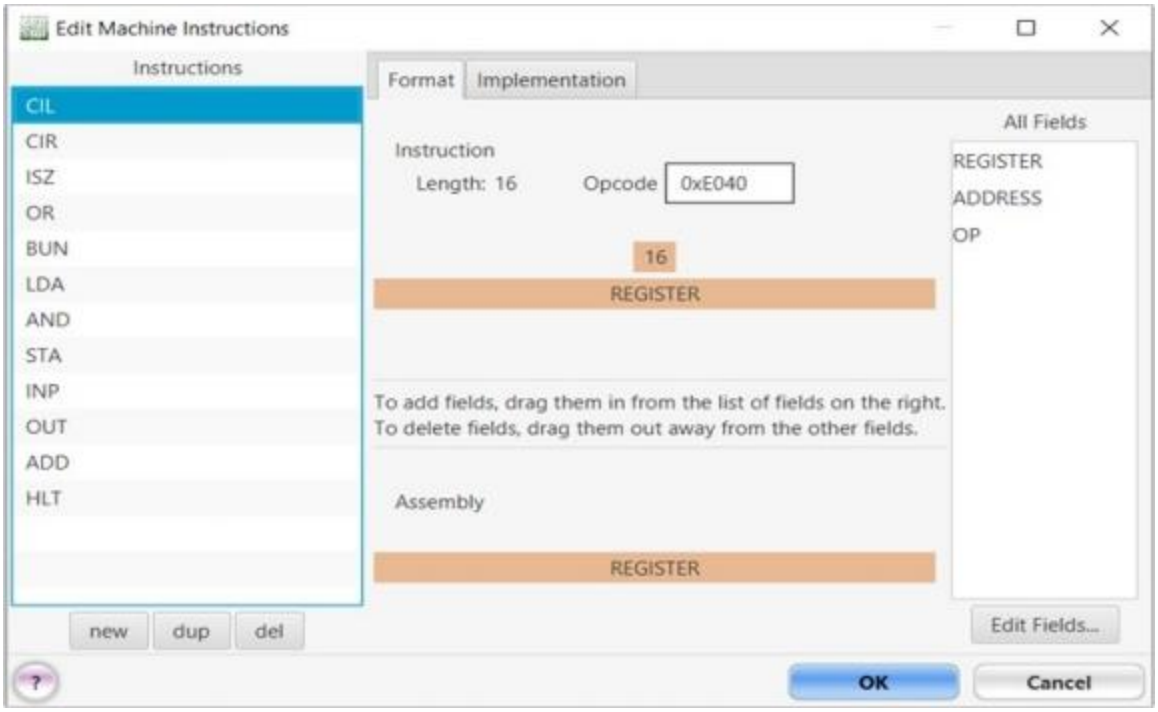
(i). CIR

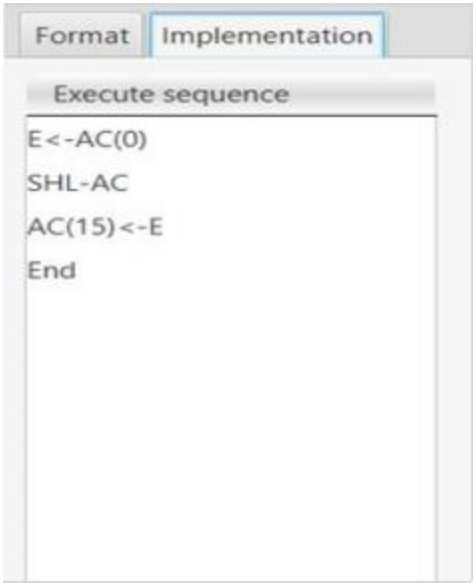


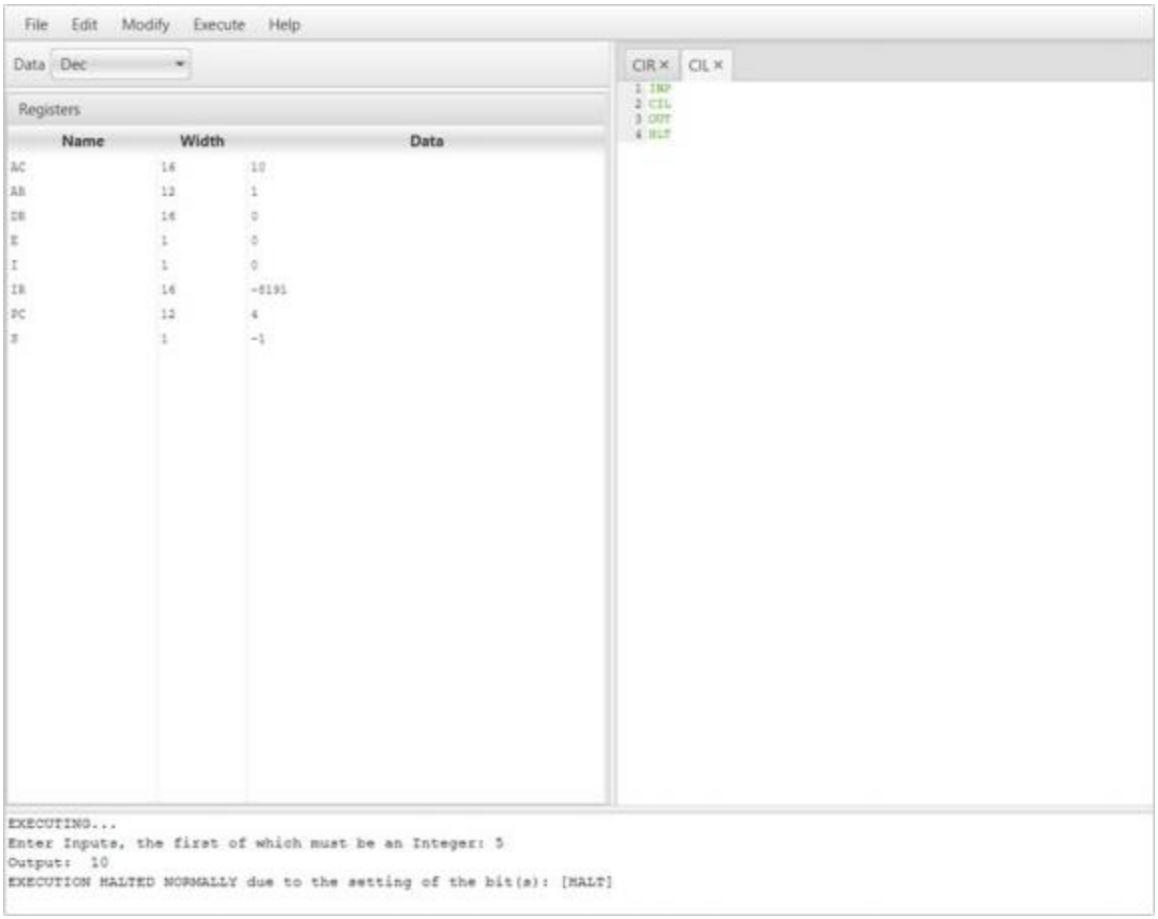


Changes in Registers after executing the program


Data
Dec
Registers
Name
Width
Data
ADD
CIR
EXECUTING...
Enter Inputs, the first of which must be an Integer: 8
Output: 4
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]


Edit Machine Instructions
Instructions
CIL
CIR
ISZ
OR
BUN
LDA
AND
STA
INP
OUT
ADD
HLT
new
dup
del
Format
Implementation
Instruction
Length: 16
Opcode
0xE040
16
REGISTER
To add fields, drag them in from the list of fields on the right.
To delete fields, drag them out away from the other fields.
Assembly
REGISTER
All Fields
REGISTER
ADDRESS
OP
Edit Fields...
?
OK
Cancel


Format
Implementation
Execute sequence
E<-AC(0)
SHL-AC
AC(15)<-E
End


File Edit Modify Execute Help
Data Dec
Registers
Name Width Data
AC 16 10
Ah 12 1
DR 16 0
E 1 0
I 1 0
IR 16 -8191
PC 12 4
T 1 -1
CIR × CIL ×
1 INP
2 CIL
3 OUT
4 HLT
EXECUTING...
Enter Inputs, the first of which must be an Integer: 5
Output: 10
EXECUTION HALTED NORMALLY due to the setting of the bit(s): [HALT]