Milestone 2

Instruction Fetch and Decode Cycle (T0–T2)

T0: AR←PC //(AR ld = 1)(s=010)

T1: IR←M[AR], PC←PC+1 //(PC inc = 1)(IR ld = 1)(Memory Read= 1)(S=110)

T2: D0,...,D7←Decode IR (12-14), AR←IR(0-6) //(AR←IR(0-6))(DECODE E = 1)(S=101)

Instruction Execution Cycle (T3–Tn)

AND `0XXX`

D = D0

T3D0: DR←M[AR], (DRld = 1, s=110, MEMORY Read= 1)

T4D0: AC←AC^DR, SC←0,(ACld = 1, C=0101 , s=011)

ADD `1XXX`

Adds memory Word to AC

D = D1

T3D1: AC←AC+M[AR], SC←0,(ACld = 1, C=0100 , s=110, MEMORY Read= 1)

LDA `2XXX`

Load Memory Word to AC

D = D2

T3D2: AC←M[AR], SC←0,(ACld=1, MEMORY Read= 1, S=110, C=0010)

STA `3XXX`

Store content of AC in memory

D = D3

T3D3: M[AR]←AC, SC←0, (MEMORY WRITE = 1, S=011)

BUN `4XXX`

Branch unconditionally

D = D4

T3D4: PC←AR, SC←0,(PCld = 1, S=001)

MUL `5XXX`

multiplies values in the memory with the AC value

D = D5

T3D5: AC←AC*M[AR], SC←0, (ACld = 1, S=110, C=1001, MEMORY READ=1)

ISZ `6XXX`

Increment and skip if zero

D = D6

T3D6: TR←M[AR], (TRld = 1, S=110, MEMORY READ=1)

T4D6: TR←TR+1 (TR inc=1)

T5D6:M[AR]←TR, if(TR=0) then PC←PC+1,SC←0,(S=111,MEMORY WRITE = 1,PC inc = TR')

Program

```
int A, B, C, i; // variables

for( i=0; i<3; i++)
{
   C += A&B;
   A += A * B;
}
```

`A→b`

`B→c`

`C→d`

`i→e`

Address	Instruction	Correct Machine Code
0	LDA A	`200b`
1	AND B	`000c`
2	ADD C	`100d`
3	STA C	`300d`
4	LDA A	`200b`
5	MUL B	`500c`
6	ADD A	`100b`
7	STA A	`300b`
8	ISZ i	`600e`
9	BUN 0	`4000`
10	HLT	`0000`

11	A = 16	`0010`
12	B = 2	`0002`
13	C = 0	`0000`
14	i = -3	`FFFD`

After testing the program above the final values where:

Variable	Final Value
A	`432`
B	`2`
C	`0`
i	`0`