1)

|  |  |
| --- | --- |
| **Control signal** | **Comment** |
| V09 | Get instruction from command counter |
| V13 | The decoder provides access to the desired cell |
| V14 | The control signal determines the action—reading or writing |
| V16 | Theoperationtobeexecutedisdefinedbyaspecialcontrolsignal |
|  |  |
| V11 | Command register –stores the actual command as Code and Address |
| V04 | Control signals are issued by control module |
| GetOperant |  |
| V12 | Pass address to ,the decoder |
| V13 | The decoder provides access to the desired cell |
| V14 | The control signal determines the action—reading or writing |
| V16 | The operationto be executed is defined by a special control signal |
| V18 | Registers are used to store temporary values |
| Execute |  |
| V20 | The operation to be executed is defined by a special control signal |
| V22 | heapplicationcodeisexecutedbyanarithmetic-logicunit |
| V23 | Data sent to ram |
| V17 | Dataandcommandsarestoredinonesharedmemory |
| V15 | Write |

2)

a)

Push A

Push B

Push C

Div

Add

Push D

Push E

Mul

Push F

Sub

rez

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Push A | Push B | Push C | Div | Add | Push D | Push E | mul | Push F | Sub | Sub | Pop R |
| A | B | C | B/C | A+B/C | D | E | D\*E | F | D\*E-F | (A+B/C)-(D\*E-F) |  |
|  | A | B | A |  | A+B/C | D | A+B/C | D\*E | A+B/C |  |  |
|  |  | A |  |  |  | A+B/C |  | A+B/C |  |  |  |

B)

Load B

Div C

Add A

Store G

Load D

Mul E

Sub F

Sub G

Rez

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Load B | Div C | Add A | Store G | Load D | Mul E | Sub F | Store H | Load G | Sub H | Rez |
| B | B/C | B/C+A | B/C+A | D | D\*E | D\*E-F | (D\*E-F) | B/C+A | (B/C+A)-(D\*E-F) | (B/C+A)-(D\*E-F) |

3)

|  |  |  |  |
| --- | --- | --- | --- |
| **Addressing type** | **operation** | **Command** | **R1 value** |
| Immediate | add R1, #5 | R1🡨R1+5 | 50+5 = 55 |
| Direct | add R1, (5) | R1🡨R1+M[5] | 50+100 = 150 |
| Direct register | add R1, R2 | R1🡨R1+R2 | 50+25 = 75 |
| Indirect register | add R1, (R3) | R1🡨R1+M[R3] | 50+100 = 150 |
| Indexed | add R1, (R1-R2) | R1🡨R1+M(R1-R2) | 50+201 = 251 |
| Displacement | add R1, 20(R3) | R1🡨R1+M[20+5] | 50+201 = 251 |