**ΑΣΚΗΣΗ 1**

Accumulator: 1000

Program Counter: 1001

Βοηθητικός Καταχωρητής Χ: 1010

LDA $K : Φόρτωσε τον Accumulator με το περιεχόμενο της διεύθυνσης Κ (της κύριας μνήμης)

1.PC+1->PC,MAR

2. MDR + 0->Χ

3. Χ + 0->MAR

4. MDR + 0->ACC

5. PC + 1 ->PC, MAR

6. NEXT(PC)

ADD $K : Πρόσθεσε στον Accumulator το περιεχόμενο της διεύθυνσης Κ

1. PC + 1->PC , MAR

2. MDR + 0->X

3. X + 0->NOP, MAR

4. MDR + ACC->ACC

5. PC + 1->PC, MAR

6. NEXT(PC)

STA $K : Αποθήκευσε το περιεχόμενο του Accumulator στη θέση μνήμης με διεύθυνση Κ

1. PC + 1->PC , MAR

2. MDR + 0->X

3. X + 0->NOP,MAR

4. ACC + 0->NOP, MWE~

5. PC + 1->PC, MAR

6. NEXT(PC)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **LDA $K** | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | **ADDRESS** |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m02 |
| MDR+0->Χ | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | 1010 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m03 |
| Χ+0->MAR | XXXXX | 000 | XXX | 100 | 000 | 001 | 1010 | XXXX | XX | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | m04 |
| MDR+0->ACC | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | 1000 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m05 |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m06 |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m07 |

Ως don’t care bits χαρακτηρίζουμε σε όλες τις εντολές τα bra και con, διότι είναι πάντα 000 (οι εντολές είναι σειριακές και έτσι ξέρουμε πάντα σε ποια διεύθυνση είμαστε). Επίσης στις εντολές NEXT(PC) τα bits των I[5:3] των Α,Β,DD είναι αδιάφορα. Τέλος όταν στο Ι[8:6] έχουμε 001 δεν γίνεται εγγραφή στο register file και δεν έχει σημασία το περιεχόμενο του PortB, άρα και εκεί υπάρχουν αδιάφορα bits.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **BOOTSTRAP** | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | **ADDRESS** |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| SW+0->PC,MAR | xxxxx | 000 | xxx | 111 | 000 | 011 | xxxx | 0100 | 00 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | m00 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | x | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m01 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ADD $K | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | ADDRESS |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m08 |
| MDR+0->X | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | 1010 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m09 |
| X+0->NOP,MAR | XXXXX | 000 | XXX | 100 | 000 | 001 | 1010 | XXXX | XX | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | m0a |
| MDR+ACC->ACC | XXXXX | 000 | XXX | 101 | 000 | 011 | XXXX | 1000 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m0b |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0c |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m0d |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STA $K | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | ADDRESS |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0e |
| MDR+0->X | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | 1010 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m0f |
| X+0->NOP,MAR | XXXXX | 000 | XXX | 100 | 000 | 001 | 1010 | XXXX | XX | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 1 | m10 |
| ACC+0->NOP,MWE | XXXXX | 000 | XXX | 100 | 000 | 001 | 1000 | XXXX | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | m11 |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m12 |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m13 |

Ο βοηθητικός καταχωρητής (Χ) μπορεί να παραληφθεί σε όλες τις μικροεντολές. Δεν είναι απαραίτητος, καθώς έτσι δεν θα χρειαστούμε τρίτο καταχωρητή από το Register File της ALU. Άρα οι εντολές :

Στο LDA $K, το MDR + 0 -> X και το X + 0 -> MAR θα γίνει MDR +0 -> MAR

Στο STA $K, το MDR + 0 -> X και το X + 0 -> NOP, MAR θα γίνει MDR +0 -> NOP, MAR

Στο ADD $K, το MDR + 0 -> X και το X + 0 -> NOP, MAR θα γίνει MDR +0 -> NOP, MAR

Οπότε ο καταχωρητής δεν είναι απαραίτητος

Οι 40αδες θα ως διαμορφωθούν ως εξής:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **BOOTSTRAP** | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | **ADDRESS** |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| SW+0->PC,MAR | xxxxx | 000 | xxx | 111 | 000 | 011 | xxxx | 0100 | 00 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | m00 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | x | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m01 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **LDA $K** | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | **ADDRESS** |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m02 |
| MDR+0->ΜΑR | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | XXXX | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m03 |
| MDR+0->ACC | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | 1000 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m05 |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m06 |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m07 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ADD $K | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | ADDRESS |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m08 |
| MDR+0->NOP,MAR | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | XXXX | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m09 |
| MDR+ACC->ACC | XXXXX | 000 | XXX | 101 | 000 | 011 | XXXX | 1000 | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m0b |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0c |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m0d |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STA $K | BRA | BIN | CON | I | I | I | APORT | BPORT | DDATA | SH~ | SELB | MWE~ | MARCLK | MSTATUS | LDS~ | PCE~ | CARRYE~ | MDE~ | DDATAE~ | ADDRESS |
|  | (4:0) | (2:0) | (2:0) | (2:0) | (5:3) | (8:6) | (3:0) | (3:0) | (1:0) |  |  |  |  |  |  |  |  |  |  |  |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0e |
| MDR+0->NOP,MAR | XXXXX | 000 | XXX | 111 | 000 | 011 | XXXX | XXXX | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m0f |
| ACC+0->NOP,MWE | XXXXX | 000 | XXX | 100 | 000 | 001 | 1000 | XXXX | XX | 1 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | m11 |
| PC+1->PC,MAR | XXXXX | 000 | XXX | 101 | 000 | 011 | 1001 | 1001 | 01 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m12 |
| NEXT(PC) | XXXXX | 000 | XXX | XXX | XXX | 001 | XXXX | XXXX | XX | X | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | m13 |

|  |  |  |
| --- | --- | --- |
| **Mapper** | | |
| Κώδικας εντολής | Opcode/Θέση | Περιεχόμενα |
| LDA $K | **m00** | 02 |
| ADD $K | **m01** | 08 |
| STA $K | **m02** | 0e |

|  |  |  |  |
| --- | --- | --- | --- |
| **Main Memory** | | |  |
| Κώδικας εντολής | Θέση | Περιεχόμενο |  |
| LDA $06 | **m00** | 00 | Opcode LDA |
| **m01** | 06 | Έντελο εντολής LDA |
| ADD $07 | **m02** | 01 | Opcode ADD |
| **m03** | 07 | Έντελο εντολής ADD |
| STA $08 | **m04** | 02 | Opcode STA |
| **m05** | 08 | Έντελο εντολής STA |
|  | **m06** | f0 |  |
|  | **m07** | ff |  |
|  | **m08** | 03 | //περιοχή δεδομένων |
|  | **m09** | 02 |  |
|  | **m0a** | 01 |  |

Γ) Για να υπολογιστεί η πράξη Γi = Αi+Bi (i=10) χρειάζονται 30 εντολές. Πρέπει να γίνουν 10 πράξεις και για κάθε πράξη απαιτούνται 3 μάκροεντολές (10\*3=30).

Δ) Το πρόβλημα που δημιουργείται είναι η εκτέλεση αλλεπάλληλων πράξεων, ώστε να φτάσουμε στο ζητούμενο αποτέλεσμα. Για να ξεπεράσουμε το πρόβλημα και να γλιτώσουμε χρόνο-μνήμη, μπορούμε να χρησιμοποιήσουμε ένα loop ώστε να μειωθεί ο αριθμός των εντολών.