Αναφορά Τρίτης Εργαστηριακής Άσκησης

|  |  |  |
| --- | --- | --- |
| **Main Memory** | | |
| Κώδικας εντολής | Θέση | Περιεχόμενο |
| Y | m17 | 01 |
|  | m18 | 01 |
|  | m19 | 00 |
|  | m1a | 06 |
|  | m1b | 02 |
|  | m1c | 07 |
|  | m1d | 06 |
|  | m1e | 02 |
| W | m1f | 00 |
|  | m20 | 00 |
|  | m21 | 00 |
|  | m22 | 00 |
|  | m23 | 00 |
|  | m24 | 00 |
|  | m25 | 00 |
|  | m26 | 00 |
|  | m27 |  |

|  |  |  |
| --- | --- | --- |
| **Main Memory** | | |
| Κώδικας εντολής | Θέση | Περιεχόμενο |
| **LDX #K** | m00 | 01 |
| 0 | m01 | 00 |
| **LDA $K,X** | m02 | 00 |
| 0f | m03 | 0f |
| **SHLA** | m04 | 08 |
| **ADC $K,X** | m05 | 05 |
| 17 | m06 | 17 |
| **STA $K,X** | m07 | 04 |
| 1f | m08 | 1f |
| **INX** | m09 | 02 |
| **CMPX #Y** | m0a | 03 |
| 9 | m0b | 08 |
| **BRNZ $K** | m0c | 07 |
| 02 | m0d | 02 |
| **HALT** | m0e | 09 |
| Ζ | m0f | 02 |
|  | m10 | 03 |
|  | m11 | 05 |
|  | m12 | 07 |
|  | m13 | 07 |
|  | m14 | 06 |
|  | m15 | 01 |
|  | m16 | 06 |

Σπυριδάκης Ορέστης Νεκτάριος (ΑΜ:1067541)

|  |  |  |
| --- | --- | --- |
| **Mapper** | | |
| Κώδικας εντολής | Opcode/Θέση | Περιεχόμενα |
| LDA $K,X | **00000000** | 02 |
| LDX #K | **00000001** | 07 |
| INX | **00000010** | 0b |
| CMPX #Y | **00000011** | 0e |
| STA $K,X | **00000100** | 12 |
| ADC $K,X | **00000101** | 17 |
| CRC | **00000110** | 1c |
| BRNZ $K | **00000111** | 1f |
| SHLA | **00001000** | 24 |
| HALT | **00001001** | 26 |

Λουδάρος Ιωάννης (ΑΜ:1067400)

Τμήμα Εργαστηρίου: Β05

Δηλωμένοι Registers

Program Counter: 0000

Accumulator: 0010

Βοηθητικός Καταχωρητής Χ: 0001

Αποτέλεσμα

Μετά την εκτέλεση του προγράμματος, στις θέσεις μνήμης m10-m27 υπάρχουν τα αποτελέσματα που παρήχθησαν σύμφωνα με την σχέση W[i]=Y[i]+2\*Z[i].

**Micromemory**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **LDA $K,X** | 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 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m02 |
| MDR+X->NOP,MAR | xxxxx | 000 | xxx | 101 | 000 | 001 | 0001 | xxxx | xx | X | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | m03 |
| MDR+0->ACC | xxxxx | 000 | xxx | 111 | 000 | 011 | xxxx | 0010 | xx | X | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m04 |
| PC+1->PC,MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m05 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m06 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **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 | 0000 | xx | x | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 | 1 | m00 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | x | x | 1 | 0 | 0 | 0 | x | x | x | x | m01 |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **LDX #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 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m07 |
| MDR+0->X | xxxxx | 000 | xxx | 111 | 000 | 011 | xxxx | 0001 | xx | X | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 1 | m08 |
| PC+1->PC,MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m09 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m0a |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **ΙΝΧ** | 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) |  |  |  |  |  |  |  |  |  |  |  |
| X+1->RAMF,X | xxxxx | 000 | xxx | 101 | 000 | 011 | 0001 | 0001 | 01 | X | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | m0b |
| PC+1->PC,MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0c |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m0d |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **CMPX #Y** | 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 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m0e |
| X-MDR -> NOP ,NSTATUS | xxxxx | 000 | xxx | 101 | 001 | 001 | 0010 | xxxx | xx | X | x | 1 | 0 | 1 | 1 | 1 | 1 | 0 | 1 | m0f |
| PC+1->PC, MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m10 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m11 |

Η σύγκριση των δύο αριθμών γίνεται μέσω της πράξης της αφαίρεσης. Ενημερώνουμε τα macroFlags με τις τιμές των microFlags

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **STA $K,X** | 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 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m12 |
| MDR+X->NOP,MAR | xxxxx | 000 | xxx | 101 | 000 | 001 | 0001 | xxxx | xx | X | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | m13 |
| ACC+0->NOP, MWE~ | xxxxx | 000 | xxx | 100 | 000 | 001 | 0010 | xxxx | xx | x | x | 0 | 0 | 0 | 1 | x | 1 | x | x | m14 |
| PC+1->PC, MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m15 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m16 |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **ADC $K,X** | 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 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m17 |
| MDR+X->NOP,MAR | xxxxx | 000 | xxx | 101 | 000 | 001 | 0001 | xxxx | xx | X | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | m18 |
| MDR+ACC->ACC, MSTATUS, CARRYE~ | xxxxx | 000 | xxx | 101 | 000 | 011 | 0010 | 0010 | xx | x | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 0 | 1 | m19 |
| PC+1->PC, MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m1a |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m1b |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **CRC** | 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) |  |  |  |  |  |  |  |  |  |  |  |
| Z + 1-> NOP,MSTATUS | xxxxx | 000 | xxx | 111 | 000 | 001 | xxxx | xxxx | 01 | X | x | 1 | 0 | 1 | 1 | 1 | 1 | 1 | 0 | m1c |
| PC+1->PC,MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m1d |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m1e |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **BRNZ $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) |  |  |  |  |  |  |  |  |  |  |  |
| BRmacroZ[setZ],PC+1->PC,MAR | 00011 | 011 | 011 | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m1f |
| MDR+0->PC,MAR | xxxxx | 000 | xxx | 111 | 000 | 011 | xxxx | 0000 | xx | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | m20 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m21 |
| [setZ]: PC+1->PC, MAR (+2) | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | x | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m22 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m23 |

Στην πρώτη μικροεντολή της BRNZ ελέγχουμε το macroZ ώστε να δούμε πως πρέπει να εξελιχθεί το πρόγραμμα. Ανάλογα την τιμή του είτε περνάμε το Κ στο pc και κάνουμε το branch, είτε προσπερνάμε το Κ, και συνεχίζουμε την εκτέλεση σειριακά.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| **SHLA** | 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) |  |  |  |  |  |  |  |  |  |  |  |
| ACC + 0 -> ACC,SRAMU,SH~ | xxxxχ | 000 | xxx | 011 | 000 | 111 | xxxx | 0010 | xx | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | m24 |
| PC+1->PC, MAR | xxxxx | 000 | xxx | 101 | 000 | 011 | 0000 | 0000 | 01 | X | 1 | 1 | 1 | 0 | 1 | 1 | 1 | 1 | 0 | m25 |
| NEXT(PC) | xxxxx | 000 | xxx | xxx | xxx | 001 | xxxx | xxxx | xx | X | Χ | 1 | 0 | 0 | 0 | Χ | Χ | Χ | Χ | m26 |

**HALT**: Τέλος εκτέλεσης του προγράμματος.  
Γενικά θα μπορούσαμε να φτιάξουμε την Halt χρησιμοποιώντας το παρακάτω μικροπρόγραμμα.

PC+0→PC,MAR  
NEXT(PC)

Για οικονομία χώρου στην μικρομνήμη όμως , μπορούμε να παραλείψουμε το «PC+0→PC,MAR» και απλά όποτε καλείται το Halt να πηγαίνουμε τον microcounter κάπου που ήδη υπάρχει το next(pc). Αυτό που καταφέρνουμε συνεπώς είναι σε κάθε κύκλο ρολογιού απλά να κάνει refresh

**Ζητούμενο Μακροπρόγραμμα**

LDX #0 //Bάλε στον X το 0, τον χρησιμοποιούμε σαν counter  
LDA $(0f,X) //Φόρτωσε το Χιοστό στοιχείο του πίνακα Z

SHLA //Πολλαπλασίασε το με το δύο  
ADC $(18,X) //Πρόσθεσε το με το Χιοστό στοιχείο του πίνακα Υ

STA $(20,X) //Αποθήκευσε το αποτέλεσμα της πράξης στο

INX //Αύξησε το περιεχόμενο του Χ κατά 1  
CMPX #8 //Δες αν είναι ίσο με 8

BRNZ $02 //Αν είναι προχώρα, αλλιώς συνέχισε την εκτέλεση από την διεύθυνση 02

HALT //Τέλος προγράμματος