

## ELE 205 example using Indexed Addressing

### Source file "loop\_sum.asm" :

```
* Fred Vetter  ELE 205
* Sum 5 numbers stored in an array
* at $D000 through $D004 using a loop
*
* Assemble with:
*   as11 loop_sum.asm -l s cre > loop_sum.lst

    ORG $C000

    LDAB #$05   ; How many numbers to add in ACCB (loop counter)
    LDX  #$D000 ; starting address of array in IX
    CLRA       ; clear ACCA - sum stored here
LOOP:
    ADDA $00,X ; add the number stored at [X]+offset to ACCA
    INX       ; increment pointer to address in the array
    DECB     ; decrement the loop counter ACCB
    BNE LOOP  ; branch if more numbers to add (Z = 0)
    SWI

* end of file
```

### List file "loop\_sum.lst" :

```
Freeware assembler ASxx.EXE Ver 1.03.
0001          * Fred Vetter  ELE 205
0002          * Sum 5 numbers stored in an array
0003          * at $D000 through $D004 using a loop
0004          *
0005          * Assemble with:
0006          *   as11 loop_sum.asm -l s cre > loop_sum.lst
0007
0008 c000          ORG $C000
0009
0010 c000 c6 05    LDAB #$05   ; How many numbers to add in ACCB (loop counter)
0011 c002 ce d0 00 LDX  #$D000 ; starting address of array in IX
0012 c005 4f      CLRA       ; clear ACCA - sum stored here
0013          LOOP:
0014 c006 ab 00    ADDA $00,X ; add the number stored at [X]+offset to ACCA
0015 c008 08      INX       ; increment pointer to address in the array
0016 c009 5a      DECB     ; decrement the loop counter ACCB
0017 c00a 26 fa   BNE LOOP  ; branch if more numbers to add (Z = 0)
0018 c00c 3f      SWI
0019
0020          * end of file

LOOP          c006
LOOP          c006 *0013 0017
```

Number of errors 0