



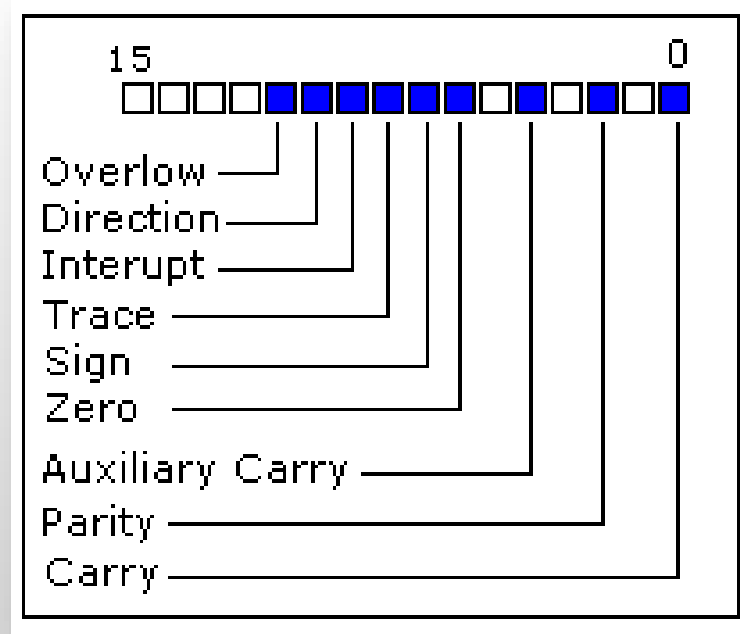
Bölüm 6: Aritmetik ve Mantık

Mikroişlemciler



Aritmetik ve Mantık Komutları

- Sonuçları işlemci durum bayraklarını etkiler.
- İşlemci Durum Bayrağı:
 - 16 bit bulunur, her biri bir bayrak, 1 ve 0 değerini alabilir.





Aritmetik ve Mantık Komutları

- İşlemci Durum Bayrağı:
 - 16 bit bulunur, her biri bir bayrak, 1 ve 0 değerini alabilir.
- Taşıma Bayrağı (*CF - Carry flag*):
 - İşaretsiz bir taşma olduğunda 1'e ayarlanır.
 - Örneğin, $255 + 1$, sonuç $0...255$ aralığında değil.
 - Taşma olmadığında 0'a ayarlanır.
- Sıfır Bayrağı (*ZF - Zero flag*):
 - Sonuç sıfır ise 1'e ayarlanır.
 - Sıfır olmayan bir sonuç için 0'a ayarlanır.



Aritmetik ve Mantık Komutları

- İşaret Bayrağı (*SF - Sign flag*):
 - Sonuç negatif ise 1'e ayarlanır.
 - Sonuç pozitif olduğunda 0'a ayarlanır.
 - Aslında bu bayrak, en önemli bitin (*MSB*) değerini alır.
- Taşma Bayrağı (*OF - Overflow flag*):
 - İmzalı bir taşma olduğunda 1'e ayarlanır.
 - Örneğin, $100 + 50$ eklediğinizde, sonuç $-128...127$ aralığında değil.
- Çiftlik Bayrağı (*PF - Parity flag*):
 - Sonuçta tek sayıda bit varsa 1'e ayarlanır,
 - Çift sayıda bit varsa 0'a ayarlanır.
 - Sonuç bir kelime ise yalnızca düşük (*low*) 8 bite bakılır.



Aritmetik ve Mantık Komutları

- Yardımcı Bayrağı (*AF - Auxiliary flag*):
 - Düşük nibble (4 bit) için işaretsiz bir taşma olduğunda 1'e ayarlanır.
- Kesme Etkin Bayrağı (*IF - Interrupt enable flag*):
 - 1'e ayarlandığında CPU, harici aygıtlardan gelen kesmelere tepki verir.
- Yön Bayrağı (*DF - Direction flag*):
 - Bazı komutlar tarafından veri zincirlerini işlemek için kullanılır;
 - 0'a ayarlandığında işlem ileri doğru yapılır,
 - 1'e ayarlandığında işlem geriye doğru yapılır.



Komutlar Üç Gruba Ayrılır

- Birinci Grup:
 - Artırma ve Azaltma
- İkinci Grup:
 - Çarpma ve Bölme
- Üçüncü Grup:
 - Tek Değişkenli İşlemler



Birinci Grup: Artırma ve Azaltma

- ADD (Toplama): İki değeri toplar ve sonucu hedefe yazar.
- SUB (Çıkarma): Bir değeri diğerinden çıkarır ve sonucu hedefe yazar.
- CMP (Karşılaştırma): İki değeri karşılaştırır, ancak sonucu saklamaz.
- AND (VE): İki değeri mantıksal olarak AND işlemine tabi tutar.
- TEST (Sınama): İki değeri bit seviyesinde sınar.
- OR (VEYA): İki değeri mantıksal olarak OR işlemine tabi tutar.
- XOR (Dışlayıcı VEYA): İki değeri mantıksal olarak XOR işlemine tabi tutar.



İkinci Grup: Çarpma ve Bölme

- MUL (Çarpma): İki değeri çarpar.
- IMUL (İşaretli Çarpma): İki işaretli değeri çarpar.
- DIV (Bölme): Bir değeri diğerine böler ve sonucu hedefe yazar.
- IDIV (İşaretli Bölme): İki işaretli değeri böler ve sonucu hedefe yazar.



Üçüncü Grup: Tek Değişkenli İşlemler

- INC (Artırma): Bir değeri bir artırır.
- DEC (Azaltma): Bir değeri bir azaltır.
- NOT (Mantıksal NOT): Bir değer bitlerini tersine çevirir.
- NEG (Negatif Alma): Bir değeri negatif hale getirir.



Birinci Grup: ADD, SUB, CMP, AND, TEST, OR, XOR

- İşlenenler:
 - REG, memory
 - memory, REG
 - REG, REG
 - memory, immediate
 - REG, immediate

- REG: AX, BX, CX, DX, AH, AL, BL, BH, CH, CL, DH, DL, DI, SI, BP, SP.
- memory: [BX], [BX+SI+7], variable, gibi..
- immediate: 5, -24, 3Fh, 10001101b, gibi..



Birinci Grup: ADD, SUB, CMP, AND, TEST, OR, XOR

- İki işlenenli işlemler sonrasında, sonuç her zaman ilk işlenende saklanır.
- CMP ve TEST komutları sadece bayrakları etkiler ve bir sonuç saklamaz.
- Etkilenen Bayraklar:
 - CF, ZF, SF, OF, PF, AF.
- ADD (Toplama)/SUB (Çıkarma): İkinci işleneni birinci işlenene ekler/çıkartır.
- CMP (Karşılaştırma): İkinci işleneni birinciden çıkarır, sonucu saklamaz.
 - sadece bayrakları etkiler.
- AND (VE)/OR (VEYA): İki işlenenin bitleri arasında VE/VEYA işlemi yapar.
- TEST (Test): AND ile aynıdır, sadece bayrakları etkiler.
- XOR (Dışlayan VEYA): İki işlenenin tüm bitleri arasında XOR işlemi yapar.



Mantıksal İşlem Kuralları

- AND:
 - $1 \text{ AND } 1 = 1$, $1 \text{ AND } 0 = 0$,
 - $0 \text{ AND } 1 = 0$, $0 \text{ AND } 0 = 0$.
- OR:
 - $1 \text{ OR } 1 = 1$, $1 \text{ OR } 0 = 1$,
 - $0 \text{ OR } 1 = 1$, $0 \text{ OR } 0 = 0$.
- XOR:
 - $1 \text{ XOR } 1 = 0$, $1 \text{ XOR } 0 = 1$,
 - $0 \text{ XOR } 1 = 1$, $0 \text{ XOR } 0 = 0$.



İkinci Grup: MUL, IMUL, DIV, IDIV

- İşlenenler:
 - REG
 - memory
- REG: AX, BX, CX, DX, AH, AL, BL, BH, CH, CL, DH, DL, DI, SI, BP, SP.
- memory: [BX], [BX+SI+7], variable, gibi..



İkinci Grup: MUL, IMUL, DIV, IDIV

- MUL ve IMUL komutları yalnızca CF ve OF bayraklarını etkiler.
- Sonuç, işlenen boyutunu aştığında bu bayraklar 1'e ayarlanır.
- DIV ve IDIV komutlarında bayraklar tanımsızdır.



MUL, IMUL

- MUL (İşaretsiz Çarpma):
 - İşlenen bir byte ise: $AX = AL * \text{işlenen}$.
 - İşlenen bir kelime ise: $(DX AX) = AX * \text{işlenen}$.
- IMUL (İşaretili Çarpma):
 - İşlenen bir byte ise: $AX = AL * \text{işlenen}$.
 - İşlenen bir kelime ise: $(DX AX) = AX * \text{işlenen}$.



DIV, IDIV

- DIV (İşaretsiz Bölme):
 - İşlenen bir byte ise: $AL = AX / \text{işlenen}$, $AH = \text{kalan (modulus)}$.
 - İşlenen bir kelime ise: $AX = (DX AX) / \text{işlenen}$, $DX = \text{kalan (modulus)}$.
- IDIV (İşaretli Bölme):
 - İşlenen bir byte ise: $AL = AX / \text{işlenen}$, $AH = \text{kalan (modulus)}$.
 - İşlenen bir kelime ise: $AX = (DX AX) / \text{işlenen}$, $DX = \text{kalan (modulus)}$.



Üçüncü Grup: INC, DEC, NOT, NEG

- İşlenenler:
 - REG
 - memory
- REG: AX, BX, CX, DX, AH, AL, BL, BH, CH, CL, DH, DL, DI, SI, BP, SP.
- memory: [BX], [BX+SI+7], variable, gibi..



INC, DEC, NOT, NEG

- INC ve DEC komutları yalnızca ZF, SF, OF, PF, AF bayraklarını etkiler.
- NOT komutu hiçbir bayrağı etkilemez!
 - İşlenenin her bir bitini ters çevirir.
- NEG komutu yalnızca CF, ZF, SF, OF, PF, AF bayraklarını etkiler.
 - İşleneni negatif yapar (ikili tümleme).
 - Her bir bitini ters çevirir ve ardından 1 ekler.
 - Örneğin, 5 -5'e ve -2 2'ye dönüşecektir.



Dizi Elemanları Toplama

```
org 100h
jmp start
vec1 db 1, 2, 5, 6
vec2 db 3, 5, 6, 1
vec3 db ?, ?, ?, ?
start:
lea si, vec1
lea bx, vec2
lea di, vec3
mov cx, 4
```



Dizi Elemanları Toplama

sum:

```
mov al, [si]
```

```
add al, [bx]
```

```
mov [di], al
```

```
inc si
```

```
inc bx
```

```
inc di
```

```
loop sum
```

```
ret
```



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	26
DX	00	00
CS	07 00	
IP	01 00	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 00

07100:	EB	235	6	
07101:	0C	012	♀	
07102:	01	001	⊖	
07103:	02	002	⊖	
07104:	05	005	♣	
07105:	06	006	♣	
07106:	03	003	♥	
07107:	05	005	♣	
07108:	06	006	♣	
07109:	01	001	⊖	
0710A:	00	000	NULL	
0710B:	00	000	NULL	
0710C:	00	000	NULL	
0710D:	00	000	NULL	
0710E:	BE	190	↓	
0710F:	02	002	⊖	
07110:	01	001	⊖	
07111:	BB	187	↑	
07112:	06	006	♣	
07113:	01	001	⊖	
07114:	BF	191	↑	
07115:	0A	010	NEWL	

07 00: 01 00

```
JMP 010Eh
ADD [BP + SI], AX
ADD AX, 00306h
ADD AX, 00106h
ADD [BX + SI], AL
ADD [BX + SI], AL
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	26
DX	00	00
CS	07 00	
IP	01 0E	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 0E

07100:	EB	235	6	
07101:	0C	012	♀	
07102:	01	001	⊙	
07103:	02	002	⊙	
07104:	05	005	♣	
07105:	06	006	♣	
07106:	03	003	♥	
07107:	05	005	♣	
07108:	06	006	♣	
07109:	01	001	⊙	
0710A:	00	000	NULL	
0710B:	00	000	NULL	
0710C:	00	000	NULL	
0710D:	00	000	NULL	
0710E:	BE	190	↓	
0710F:	02	002	⊙	
07110:	01	001	⊙	
07111:	BB	187	↑	
07112:	06	006	♣	
07113:	01	001	⊙	
07114:	BF	191	↑	
07115:	0A	010	NEWL	

07 00: 01 0E

```
MOU SI, 00102h
MOU BX, 00106h
MOU DI, 0010Ah
MOU CX, 00004h
MOU AL, [SI]
ADD AL, [BX]
MOU [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	26
DX	00	00
CS	07 00	
IP	01 11	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	01 02	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 11

```
07100: EB 235 6
07101: 0C 012 ♀
07102: 01 001 ♂
07103: 02 002 ♂
07104: 05 005 ♀
07105: 06 006 ♀
07106: 03 003 ♀
07107: 05 005 ♀
07108: 06 006 ♀
07109: 01 001 ♂
0710A: 00 000 NULL
0710B: 00 000 NULL
0710C: 00 000 NULL
0710D: 00 000 NULL
0710E: BE 190 ↓
0710F: 02 002 ♂
07110: 01 001 ♂
07111: BB 187 ↱
07112: 06 006 ♀
07113: 01 001 ♂
07114: BF 191 ↱
07115: 0A 010 NEWL
```

07 00: 01 11

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	01	06
CX	00	26
DX	00	00
CS	07 00	
IP	0114	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0102	
DI	0000	
DS	07 00	
ES	07 00	

0700:0114

07100:	EB	235	6	
07101:	0C	012	♀	
07102:	01	001	⊖	
07103:	02	002	⊖	
07104:	05	005	♣	
07105:	06	006	♣	
07106:	03	003	♥	
07107:	05	005	♣	
07108:	06	006	♣	
07109:	01	001	⊖	
0710A:	00	000	NULL	
0710B:	00	000	NULL	
0710C:	00	000	NULL	
0710D:	00	000	NULL	
0710E:	BE	190	↓	
0710F:	02	002	⊖	
07110:	01	001	⊖	
07111:	BB	187	↑	
07112:	06	006	♣	
07113:	01	001	⊖	
07114:	BF	191	↑	
07115:	0A	010	NEWL	

0700:0114

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	01	06
CX	00	04
DX	00	00
CS	07 00	
IP	011A	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0102	
DI	010A	
DS	07 00	
ES	07 00	

0700:011A

```
07117: B9 185 |  
07118: 04 004 |  
07119: 00 000 NULL  
0711A: 8A 138 è  
0711B: 04 004 |  
0711C: 02 002 |  
0711D: 07 007 BEEP  
0711E: 88 136 è  
0711F: 05 005 |  
07120: 46 070 F  
07121: 43 067 C  
07122: 47 071 G  
07123: E2 226 |  
07124: F5 245 J  
07125: C3 195 |  
07126: 90 144 É  
07127: 90 144 É  
07128: 90 144 É  
07129: 90 144 É  
0712A: 90 144 É  
0712B: 90 144 É  
0712C: 90 144 É
```

0700:011A

```
MOV SI, 00102h  
MOV BX, 00106h  
MOV DI, 0010Ah  
MOV CX, 00004h  
MOV AL, [SI]  
ADD AL, [BX]  
MOV [DI], AL  
INC SI  
INC BX  
INC DI  
LOOP 011Ah  
RET  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	01
BX	01	06
CX	00	04
DX	00	00
CS	07 00	
IP	01 1C	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	01 02	
DI	01 0A	
DS	07 00	
ES	07 00	

07 00: 01 1C

```
07117: B9 185 H
07118: 04 004
07119: 00 000 NULL
0711A: 8A 138 e
0711B: 04 004
0711C: 02 002
0711D: 07 007 BEEP
0711E: 88 136 e
0711F: 05 005
07120: 46 070 F
07121: 43 067 C
07122: 47 071 G
07123: E2 226 r
07124: F5 245 J
07125: C3 195 t
07126: 90 144 e
07127: 90 144 e
07128: 90 144 e
07129: 90 144 e
0712A: 90 144 e
0712B: 90 144 e
0712C: 90 144 e
```

07 00: 01 1C

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	04
BX	01	06
CX	00	04
DX	00	00
CS	07 00	
IP	011E	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0102	
DI	010A	
DS	07 00	
ES	07 00	

0700:011E

```
07117: B9 185 |  
07118: 04 004 |  
07119: 00 000 NULL  
0711A: 8A 138 è  
0711B: 04 004 |  
0711C: 02 002 |  
0711D: 07 007 BEEP  
0711E: 88 136 è  
0711F: 05 005 |  
07120: 46 070 F  
07121: 43 067 C  
07122: 47 071 G  
07123: E2 226 |  
07124: F5 245 J  
07125: C3 195 |  
07126: 90 144 é  
07127: 90 144 é  
07128: 90 144 é  
07129: 90 144 é  
0712A: 90 144 é  
0712B: 90 144 é  
0712C: 90 144 é
```

0700:011E

```
MOV SI, 00102h  
MOV BX, 00106h  
MOV DI, 0010Ah  
MOV CX, 00004h  
MOV AL, [SI]  
ADD AL, [BX]  
MOV [DI], AL  
INC SI  
INC BX  
INC DI  
LOOP 011Ah  
RET  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	04
BX	01	07
CX	00	04
DX	00	00
CS	07 00	
IP	0122	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0103	
DI	010A	
DS	07 00	
ES	07 00	

0700:0122

```
07117: B9 185 |  
07118: 04 004 |  
07119: 00 000 NULL  
0711A: 8A 138 |  
0711B: 04 004 |  
0711C: 02 002 |  
0711D: 07 007 BEEP  
0711E: 88 136 |  
0711F: 05 005 |  
07120: 46 070 F  
07121: 43 067 C  
07122: 47 071 G  
07123: E2 226 |  
07124: F5 245 J  
07125: C3 195 |  
07126: 90 144 |  
07127: 90 144 |  
07128: 90 144 |  
07129: 90 144 |  
0712A: 90 144 |  
0712B: 90 144 |  
0712C: 90 144 |
```

0700:0122

```
MOV SI, 00102h  
MOV BX, 00106h  
MOV DI, 0010Ah  
MOV CX, 00004h  
MOV AL, [SI]  
ADD AL, [BX]  
MOV [DI], AL  
INC SI  
INC BX  
INC DI  
LOOP 011Ah  
RET  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	04
BX	01	07
CX	00	04
DX	00	00
CS	07 00	
IP	0123	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0103	
DI	010B	
DS	07 00	
ES	07 00	

0700:0123

```
07117: B9 185 H
07118: 04 004 D
07119: 00 000 NULL
0711A: 8A 138 e
0711B: 04 004 D
0711C: 02 002 S
0711D: 07 007 BEEP
0711E: 88 136 e
0711F: 05 005 S
07120: 46 070 F
07121: 43 067 C
07122: 47 071 G
07123: E2 226 T
07124: F5 245 J
07125: C3 195 T
07126: 90 144 E
07127: 90 144 E
07128: 90 144 E
07129: 90 144 E
0712A: 90 144 E
0712B: 90 144 E
0712C: 90 144 E
```

0700:0123

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	04
BX	01	07
CX	00	03
DX	00	00
CS	07 00	
IP	011A	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0103	
DI	010B	
DS	07 00	
ES	07 00	

0700:011A

```
07117: B9 185 |  
07118: 04 004 |  
07119: 00 000 NULL  
0711A: 8A 138 è  
0711B: 04 004 |  
0711C: 02 002 |  
0711D: 07 007 BEEP  
0711E: 88 136 è  
0711F: 05 005 |  
07120: 46 070 F  
07121: 43 067 C  
07122: 47 071 G  
07123: E2 226 |  
07124: F5 245 J  
07125: C3 195 |  
07126: 90 144 É  
07127: 90 144 É  
07128: 90 144 É  
07129: 90 144 É  
0712A: 90 144 É  
0712B: 90 144 É  
0712C: 90 144 É
```

0700:011A

```
MOV SI, 00102h  
MOV BX, 00106h  
MOV DI, 0010Ah  
MOV CX, 00004h  
MOV AL, [SI]  
ADD AL, [BX]  
MOV [DI], AL  
INC SI  
INC BX  
INC DI  
LOOP 011Ah  
RET  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	07
BX	01	0A
CX	00	01
DX	00	00
CS	07 00	
IP	0122	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0106	
DI	010D	
DS	07 00	
ES	07 00	

0700:0122

```
07117: B9 185 H
07118: 04 004 D
07119: 00 000 NULL
0711A: 8A 138 e
0711B: 04 004 D
0711C: 02 002 S
0711D: 07 007 BEEP
0711E: 88 136 e
0711F: 05 005 S
07120: 46 070 F
07121: 43 067 C
07122: 47 071 G
07123: E2 226 r
07124: F5 245 J
07125: C3 195 t
07126: 90 144 E
07127: 90 144 E
07128: 90 144 E
07129: 90 144 E
0712A: 90 144 E
0712B: 90 144 E
0712C: 90 144 E
```

0700:0122

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	07
BX	01	0A
CX	00	01
DX	00	00
CS	07 00	
IP	0123	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0106	
DI	010E	
DS	07 00	
ES	07 00	

0700:0123

```
07117: B9 185 H
07118: 04 004 D
07119: 00 000 NULL
0711A: 8A 138 e
0711B: 04 004 D
0711C: 02 002 S
0711D: 07 007 BEEP
0711E: 88 136 e
0711F: 05 005 S
07120: 46 070 F
07121: 43 067 C
07122: 47 071 G
07123: E2 226 T
07124: F5 245 J
07125: C3 195 T
07126: 90 144 E
07127: 90 144 E
07128: 90 144 E
07129: 90 144 E
0712A: 90 144 E
0712B: 90 144 E
0712C: 90 144 E
```

0700:0123

```
MOV SI, 00102h
MOV BX, 00106h
MOV DI, 0010Ah
MOV CX, 00004h
MOV AL, [SI]
ADD AL, [BX]
MOV [DI], AL
INC SI
INC BX
INC DI
LOOP 011Ah
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Dizi Elemanları Toplama

emulator: add-2.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	07
BX	01	0A
CX	00	00
DX	00	00
CS	07 00	
IP	0125	
SS	07 00	
SP	FFFE	
BP	0000	
SI	0106	
DI	010E	
DS	07 00	
ES	07 00	

0700:0125

```
07117: B9 185 |  
07118: 04 004 |  
07119: 00 000 NULL  
0711A: 8A 138 |  
0711B: 04 004 |  
0711C: 02 002 |  
0711D: 07 007 BEEP  
0711E: 88 136 |  
0711F: 05 005 |  
07120: 46 070 F  
07121: 43 067 C  
07122: 47 071 G  
07123: E2 226 |  
07124: F5 245 J  
07125: C3 195 |  
07126: 90 144 |  
07127: 90 144 |  
07128: 90 144 |  
07129: 90 144 |  
0712A: 90 144 |  
0712B: 90 144 |  
0712C: 90 144 |
```

0700:0125

```
MOV SI, 00102h  
MOV BX, 00106h  
MOV DI, 0010Ah  
MOV CX, 00004h  
MOV AL, [SI]  
ADD AL, [BX]  
MOV [DI], AL  
INC SI  
INC BX  
INC DI  
LOOP 011Ah  
RET  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
NOP  
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal

; first number '9':

```
mov  ah, 09h
```

; second number '5':

```
mov  al, 05h
```

; $al = al + ah = 09h + 05h = 0eh$

```
add  al, ah
```

; clear tens byte of bcd

```
xor  ah, ah
```

; adjust result to bcd form, ; $ah = 1, al = 4 \rightarrow '14'$

```
aaa
```




Binary Coded Decimal

emulator: bcd_aaa.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	09	00
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 02	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 02

07100:	B4	180	
07101:	09	009	TAB
07102:	B0	176	///
07103:	05	005	+
07104:	02	002	0
07105:	C4	196	-
07106:	32	050	2
07107:	E4	228	Σ
07108:	37	055	7
07109:	90	144	É
0710A:	90	144	É
0710B:	90	144	É
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 02

```
MOU AH, 09h
MOU AL, 05h
ADD AL, AH
XOR AH, AH
AAA
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal

emulator: bcd_aaa.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	09	05
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 04	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 04

07100:	B4	180	
07101:	09	009	TAB
07102:	B0	176	
07103:	05	005	▲
07104:	02	002	⊖
07105:	C4	196	—
07106:	32	050	2
07107:	E4	228	Σ
07108:	37	055	7
07109:	90	144	É
0710A:	90	144	É
0710B:	90	144	É
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 04

```
MOV AH, 09h
MOV AL, 05h
ADD AL, AH
XOR AH, AH
AAA
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal

The screenshot shows an emulator window titled "emulator: bcd_aaa.com_". The interface includes a menu bar (file, math, debug, view, external, virtual devices, virtual drive, help) and a toolbar with buttons for Load, reload, step back, single step, run, and a step delay slider set to 0 ms. On the left, a "registers" panel displays the state of various registers:

Register	H	L
AX	09	0E
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 06	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

The main window is split into two panes. The left pane shows memory addresses and their contents, with the current instruction at 07106: 32 050 2 highlighted. The right pane shows the assembly code being executed:

```
MOV AH, 09h
MOV AL, 05h
ADD AL, AH
XOR AH, AH
AAA
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```



Binary Coded Decimal

The screenshot shows an emulator window titled "emulator: bcd_aaa.com_". The interface includes a menu bar (file, math, debug, view, external, virtual devices, virtual drive, help) and a toolbar with buttons for Load, reload, step back, single step, run, and a step delay slider set to 0 ms. On the left, a "registers" panel displays the state of various registers: AX (00 0E), BX (00 00), CX (00 09), DX (00 00), CS (07 00), IP (01 08), SS (07 00), SP (FFFE), BP (0000), SI (0000), DI (0000), DS (07 00), and ES (07 00). The main area is split into two panes. The left pane shows memory addresses from 07100 to 07115 with their corresponding hex values and some symbols. The address 07108: 37 055 ? is highlighted in blue. The right pane shows assembly instructions: MOV AH, 09h; MOV AL, 05h; ADD AL, AH; XOR AH, AH; followed by several NOP instructions. The instruction "AAA" is highlighted in blue. At the bottom, there are buttons for screen, source, reset, aux, vars, debug, stack, and flags.



Binary Coded Decimal

emulator: bcd_aaa.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	01	04
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 09	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 09

Address	Hex	Dec	Symbol
07100:	B4	180	
07101:	09	009	TAB
07102:	B0	176	
07103:	05	005	+
07104:	02	002	@
07105:	C4	196	-
07106:	32	050	2
07107:	E4	228	Σ
07108:	37	055	7
07109:	90	144	É
0710A:	90	144	É
0710B:	90	144	É
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 09

```
MOV AH, 09h
MOV AL, 05h
ADD AL, AH
XOR AH, AH
AAA
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal - Negatif Sayı

; make 5 - 8, al = 0fdh (not in binary coded decimal form)

```
mov  al, 05h
```

```
mov  bl, 08h
```

```
sub  al, bl
```

; convert to binary coded decimal, al = 7

; and 1 is borrowed from ah, like calculating 15 - 8:

```
aas
```



Binary Coded Decimal - Negatif Sayı

emulator: bcd_aas.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	12
DX	00	00
CS	07 00	
IP	01 00	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 00

07100:	B0	176	⚡
07101:	05	005	⚡
07102:	B3	179	
07103:	08	008	BACK
07104:	2A	042	*
07105:	C3	195	
07106:	3F	063	?
07107:	0C	012	♀
07108:	30	048	0
07109:	B4	180	
0710A:	0E	014	⌘
0710B:	CD	205	=
0710C:	10	016	▶
0710D:	B4	180	
0710E:	00	000	NULL
0710F:	CD	205	=
07110:	16	022	-
07111:	C3	195	
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

MOU AL, 05h
MOU BL, 08h
SUB AL, BL
AAS
OR AL, 030h
MOU AH, 0Eh
INT 010h
MOU AH, 00h
INT 016h
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...

screen source reset aux vars debug stack flags



Binary Coded Decimal - Negatif Sayı

emulator: bcd_aas.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	05
BX	00	00
CX	00	12
DX	00	00
CS	07 00	
IP	01 02	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 02

07100:	B0	176	⚡
07101:	05	005	♣
07102:	B3	179	
07103:	08	008	BACK
07104:	2A	042	*
07105:	C3	195	
07106:	3F	063	?
07107:	0C	012	♀
07108:	30	048	0
07109:	B4	180	
0710A:	0E	014	♠
0710B:	CD	205	=
0710C:	10	016	▶
0710D:	B4	180	
0710E:	00	000	NULL
0710F:	CD	205	=
07110:	16	022	-
07111:	C3	195	
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 02

```
MOU AL, 05h
MOU BL, 08h
SUB AL, BL
AAS
OR AL, 030h
MOU AH, 0Eh
INT 010h
MOU AH, 00h
INT 016h
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal - Negatif Sayı

emulator: bcd_aas.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	05
BX	00	08
CX	00	12
DX	00	00
CS	07 00	
IP	01 04	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 04

Address	Hex	Dec	Symbol
07100:	B0	176	
07101:	05	005	
07102:	B3	179	
07103:	08	008	BACK
07104:	2A	042	*
07105:	C3	195	
07106:	3F	063	?
07107:	0C	012	♀
07108:	30	048	0
07109:	B4	180	
0710A:	0E	014	∩
0710B:	CD	205	=
0710C:	10	016	▷
0710D:	B4	180	
0710E:	00	000	NULL
0710F:	CD	205	=
07110:	16	022	
07111:	C3	195	
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 04

```
MOV AL, 05h
MOV BL, 08h
SUB AL, BL
AAS
OR AL, 030h
MOV AH, 0Eh
INT 010h
MOV AH, 00h
INT 016h
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Binary Coded Decimal - Negatif Sayı

emulator: bcd_aas.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	FD
BX	00	08
CX	00	12
DX	00	00
CS	07 00	
IP	01 06	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 06

07100:	B0	176	⚡
07101:	05	005	⬆
07102:	B3	179	
07103:	08	008	BACK
07104:	2A	042	*
07105:	C3	195	⌋
07106:	3F	063	?
07107:	0C	012	♀
07108:	30	048	0
07109:	B4	180	⌋
0710A:	0E	014	⌋
0710B:	CD	205	=
0710C:	10	016	⌋
0710D:	B4	180	⌋
0710E:	00	000	NULL
0710F:	CD	205	=
07110:	16	022	-
07111:	C3	195	⌋
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

MOV AL, 05h
MOV BL, 08h
SUB AL, BL
AAS
OR AL, 030h
MOV AH, 0Eh
INT 010h
MOV AH, 00h
INT 016h
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...

screen source reset aux vars debug stack flags



Binary Coded Decimal - Negatif Sayı

emulator: bcd_aas.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	FF	07
BX	00	08
CX	00	12
DX	00	00
CS	07 00	
IP	01 07	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 07

07100:	B0	176	⚡
07101:	05	005	⚡
07102:	B3	179	
07103:	08	008	BACK
07104:	2A	042	*
07105:	C3	195	
07106:	3F	063	?
07107:	0C	012	♀
07108:	30	048	0
07109:	B4	180	
0710A:	0E	014	⌘
0710B:	CD	205	=
0710C:	10	016	▶
0710D:	B4	180	
0710E:	00	000	NULL
0710F:	CD	205	=
07110:	16	022	-
07111:	C3	195	
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 07

```
MOV AL, 05h
MOV BL, 08h
SUB AL, BL
AAS
OR AL, 030h
MOV AH, 0Eh
INT 010h
MOV AH, 00h
INT 016h
RET
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenayt Dönüşümü

```
org 100h
```

```
start:
```

```
; convert celsius to fahrenheit according to  $f = c * 9 / 5 + 32$ 
```

```
mov cl, 15
```

```
mov al, 9
```

```
imul cl
```

```
mov cl, 5
```

```
idiv cl
```

```
add al, 32
```




Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	0C
DX	00	00
CS	07 00	
IP	01 00	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

0700:0100

07100:	B1	177	⌘
07101:	0F	015	*
07102:	80	176	⌘
07103:	09	009	TAB
07104:	F6	246	÷
07105:	E9	233	0
07106:	B1	177	⌘
07107:	05	005	⌘
07108:	F6	246	÷
07109:	F9	249	-
0710A:	04	004	◆
0710B:	20	032	SPA
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

0700:0100

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIV CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	0F
DX	00	00
CS	07 00	
IP	01 02	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 02

```
07100: B1 177
07101: 0F 015 *
07102: B0 176
07103: 09 009 TAB
07104: F6 246 ÷
07105: E9 233 0
07106: B1 177
07107: 05 005 *
07108: F6 246 ÷
07109: F9 249 -
0710A: 04 004 ♦
0710B: 20 032 SPA
0710C: 90 144 É
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
```

07 00: 01 02

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIV CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	09
BX	00	00
CX	00	0F
DX	00	00
CS	07 00	
IP	01 04	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 04

```
07100: B1 177
07101: 0F 015 *
07102: B0 176
07103: 09 009 TAB
07104: F6 246 ÷
07105: E9 233 0
07106: B1 177
07107: 05 005
07108: F6 246 ÷
07109: F9 249 -
0710A: 04 004
0710B: 20 032 SPA
0710C: 90 144 É
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
```

07 00: 01 04

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIU CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	87
BX	00	00
CX	00	0F
DX	00	00
CS	07 00	
IP	01 06	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 06

```
07100: B1 177
07101: 0F 015 *
07102: B0 176
07103: 09 009 TAB
07104: F6 246 ÷
07105: E9 233 0
07106: B1 177
07107: 05 005 +
07108: F6 246 ÷
07109: F9 249 -
0710A: 04 004 ♦
0710B: 20 032 SPA
0710C: 90 144 É
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
```

07 00: 01 06

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIU CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	87
BX	00	00
CX	00	05
DX	00	00
CS	07 00	
IP	01 08	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 08

```
07100: B1 177
07101: 0F 015 *
07102: B0 176
07103: 09 009 TAB
07104: F6 246 ÷
07105: E9 233 0
07106: B1 177
07107: 05 005
07108: F6 246 ÷
07109: F9 249 -
0710A: 04 004
0710B: 20 032 SPA
0710C: 90 144 É
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
```

07 00: 01 08

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIU CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	1B
BX	00	00
CX	00	05
DX	00	00
CS	07 00	
IP	01 0A	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 0A

07100:	B1	177	⏏
07101:	0F	015	*
07102:	B0	176	⏏
07103:	09	009	TAB
07104:	F6	246	÷
07105:	E9	233	0
07106:	B1	177	⏏
07107:	05	005	⚡
07108:	F6	246	÷
07109:	F9	249	-
0710A:	04	004	◆
0710B:	20	032	SPA
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 0A

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIU CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Santigrad Fahrenheit Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	3B
BX	00	00
CX	00	05
DX	00	00
CS	07 00	
IP	01 0C	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 0C

07100:	B1	177	⌘
07101:	0F	015	*
07102:	B0	176	⌘
07103:	09	009	TAB
07104:	F6	246	÷
07105:	E9	233	0
07106:	B1	177	⌘
07107:	05	005	⌘
07108:	F6	246	÷
07109:	F9	249	-
0710A:	04	004	◆
0710B:	20	032	SPA
0710C:	90	144	É
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 0C

```
MOU CL, 0Fh
MOU AL, 09h
IMUL CL
MOU CL, 05h
IDIV CL
ADD AL, 020h
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

- org 100h
- start:
- ; convert fahrenheit to celsius according to: $c = (f - 32) * 5 / 9$
- mov cl, 59
- sub cl, 32
- mov al, 5
- imul cl
- mov cl, 9
- idiv cl



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	0D
DX	00	00
CS	07 00	
IP	01 00	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

0700:0100

```
07100: B1 177
07101: 3B 059 ;
07102: 80 128 Ç
07103: E9 233 0
07104: 20 032 SPA
07105: B0 176
07106: 05 005
07107: F6 246 ÷
07108: E9 233 0
07109: B1 177
0710A: 09 009 TAB
0710B: F6 246 ÷
0710C: F9 249 ·
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
```

0700:0100

```
MOU CL, 03Bh
SUB CL, 020h
MOU AL, 05h
IMUL CL
MOU CL, 09h
IDIU CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	00
BX	00	00
CX	00	1B
DX	00	00
CS	07 00	
IP	01 05	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

0700:0105

```
07100: B1 177 ;
07101: 3B 059 ;
07102: 80 128 Ç
07103: E9 233 0
07104: 20 032 SPA
07105: B0 176 ;
07106: 05 005 ✖
07107: F6 246 ÷
07108: E9 233 0
07109: B1 177 ;
0710A: 09 009 TAB
0710B: F6 246 ÷
0710C: F9 249 ·
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
...
```

0700:0105

```
MOV CL, 03Bh
SUB CL, 020h
MOV AL, 05h
IMUL CL
MOV CL, 09h
IDIV CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	05
BX	00	00
CX	00	1B
DX	00	00
CS	07 00	
IP	01 07	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 07

07100:	B1	177	;
07101:	3B	059	;
07102:	80	128	Ç
07103:	E9	233	0
07104:	20	032	SPA
07105:	B0	176	;
07106:	05	005	*
07107:	F6	246	÷
07108:	E9	233	0
07109:	B1	177	;
0710A:	09	009	TAB
0710B:	F6	246	÷
0710C:	F9	249	.
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

MOU CL, 03Bh
SUB CL, 020h
MOU AL, 05h
IMUL CL
MOU CL, 09h
IDIU CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	87
BX	00	00
CX	00	1B
DX	00	00
CS	07 00	
IP	01 09	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 09

```
07100: B1 177 ;
07101: 3B 059 ;
07102: 80 128 Ç
07103: E9 233 0
07104: 20 032 SPA
07105: B0 176 ;
07106: 05 005 *
07107: F6 246 ÷
07108: E9 233 0
07109: B1 177 ;
0710A: 09 009 TAB
0710B: F6 246 ÷
0710C: F9 249 .
0710D: 90 144 É
0710E: 90 144 É
0710F: 90 144 É
07110: 90 144 É
07111: 90 144 É
07112: 90 144 É
07113: 90 144 É
07114: 90 144 É
07115: 90 144 É
...
```

MOU CL, 03Bh
SUB CL, 020h
MOU AL, 05h
IMUL CL
MOU CL, 09h
IDIU CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	87
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 0B	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 0B

07100:	B1	177	;
07101:	3B	059	;
07102:	80	128	Ç
07103:	E9	233	0
07104:	20	032	SPA
07105:	B0	176	;
07106:	05	005	±
07107:	F6	246	÷
07108:	E9	233	0
07109:	B1	177	;
0710A:	09	009	TAB
0710B:	F6	246	÷
0710C:	F9	249	-
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

07 00: 01 0B

```
MOV CL, 03Bh
SUB CL, 020h
MOV AL, 05h
IMUL CL
MOV CL, 09h
IDIU CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...
```

screen source reset aux vars debug stack flags



Fahrenayt Santigrad Dönüşümü

emulator: celsi.com_

file math debug view external virtual devices virtual drive help

Load reload step back single step run step delay ms: 0

registers

	H	L
AX	00	0F
BX	00	00
CX	00	09
DX	00	00
CS	07 00	
IP	01 0D	
SS	07 00	
SP	FF FE	
BP	00 00	
SI	00 00	
DI	00 00	
DS	07 00	
ES	07 00	

07 00: 01 0D

07100:	B1	177	█
07101:	3B	059	;
07102:	80	128	Ç
07103:	E9	233	0
07104:	20	032	SPA
07105:	B0	176	█
07106:	05	005	♣
07107:	F6	246	÷
07108:	E9	233	0
07109:	B1	177	█
0710A:	09	009	TAB
0710B:	F6	246	÷
0710C:	F9	249	.
0710D:	90	144	É
0710E:	90	144	É
0710F:	90	144	É
07110:	90	144	É
07111:	90	144	É
07112:	90	144	É
07113:	90	144	É
07114:	90	144	É
07115:	90	144	É

MOV CL, 03Bh
SUB CL, 020h
MOV AL, 05h
IMUL CL
MOV CL, 09h
IDIU CL
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
NOP
...

screen source reset aux vars debug stack flags



SON