

# Temel Bilgisayarın Buyrukları

(53)

## a) Bellek Adreslemeli Buyruklar

Sembol	Onaltılı Kod	Açıklama
	I=0 I=1	
AND	0XXX 8XXX	$AC \leftarrow AC \wedge M[AR]$
ADD	1XXX 9XXX	$AC \leftarrow AC + M[AR], E \leftarrow Cout$
LDA	2XXX AXXX	$AC \leftarrow M[AR]$
STA	3XXX BXXX	$M[AR] \leftarrow AC$
BUN	4XXX CXXX	$PC \leftarrow AR$ Şartsız Dallon
BSA	5XXX DXXX	$M[AR] \leftarrow PC, PC \leftarrow AR+1$ Dallon ve geri dönüş adresini sakla
ISZ	6XXX EXXX	$M[AR] \leftarrow M[AR] + 1, M[AR] + 1 = 0$ ise $PC \leftarrow PC + 1$ Artır ve eğer sıfır ise atla

## b) Yazma Adreslemeli Buyruklar

Sembol	Onaltılı Kod	Açıklama
CLA	7800	$AC \leftarrow 0$ Akümülatörü Sil
CLE	7400	$E \leftarrow 0$ Elderi Sil
CMA	7200	$AC \leftarrow \overline{AC}$ Akümülatörün Tümleyenini Al
CME	7100	$E \leftarrow \overline{E}$ Eldenin Tümleyenini Al
CIR	7080	$AC \leftarrow shr AC, AC(15) \leftarrow E, E \leftarrow AC(0)$ Dairesel Sağa Kaydır
CIL	7040	$AC \leftarrow shl AC, AC(0) \leftarrow E, E \leftarrow AC(15)$ Dairesel Solu Kaydır
INC	7020	$AC \leftarrow AC + 1$
SPA	7010	$AC(15) = 0$ ise $PC \leftarrow PC + 1$
SNA	7008	$AC(15) = 1$ ise $PC \leftarrow PC + 1$
SZA	7004	$AC = 0$ ise $PC \leftarrow PC + 1$
SZE	7002	$E = 0$ ise $PC \leftarrow PC + 1$
HLT	7001	$S \leftarrow 0$ Programı Durdur

## c) Giriş - Çıkış Adreslemeli Buyruklar

Sembol	Onaltılı Kod	Açıklama
INP	F800	$AC(7-0) \leftarrow INPR, FGI \leftarrow 0$
OUT	F400	$OUTR \leftarrow AC(7-0), FGO \leftarrow 0$
SKI	F200	$FGI = 1$ ise $PC \leftarrow PC + 1$
SKO	F100	$FGO = 1$ ise $PC \leftarrow PC + 1$
ION	F080	$IEN \leftarrow 1$ Kesmeyi Aktif Yap
IOF	F040	$IEN \leftarrow 0$ Kesmeyi Pasif Yap

# Temel Bilgisayarın Programlanması

Temel Bilgisayarın komut kümesini kullanarak aşağıdaki işleri yapan altprogramları gerçekleştir

$$C \leftarrow A + B$$

TPL, \_\_\_\_\_  
LDA A  
ADD B  
STA C  
BUN TPL, I

$$C \leftarrow A \wedge B$$

VE, \_\_\_\_\_  
LDA A  
AND B  
STA C  
BUN VE, I

$$C \leftarrow A \oplus B$$

$$\leftarrow A' \wedge B \vee A \wedge B'$$

$$\leftarrow ((A' \wedge B)' \wedge (A \wedge B')')$$

EXR, \_\_\_\_\_  
LDA A  
CMA  
AND B  
CMA  
STA C  
LDA B  
CMA  
AND A  
CMA  
AND C  
CMA  
STA C  
BUN EXR, I

$$C \leftarrow A - B$$

$$\leftarrow A + \bar{B} + 1$$

CKR, \_\_\_\_\_  
LDA B  
CMA  
INC  
ADD A  
STA C  
BUN CKR, I

$$C \leftarrow A \vee B$$

$$\leftarrow (A' \wedge B')'$$

VYA, \_\_\_\_\_  
LDA A  
CMA  
STA C  
LDA B  
CMA  
AND C  
CMA  
STA C  
BUN VYA, I

$$F \leftarrow A \vee B \wedge C$$

$$\leftarrow (A' \wedge (B \wedge C'))'$$

LOJ, \_\_\_\_\_  
LDA A  
CMA  
STA F  
LDA B  
AND C  
CMA  
AND F  
CMP  
STA F  
BUN LOJ, I

$$F \leftarrow A - B - C$$

$$\leftarrow A - (B + C)$$

ISL, \_\_\_\_\_  
LDA B  
ADD C  
CMA  
INC  
ADD A  
STA F  
BUN ISL, I

$A > B$  ise YR1'e 8 bit  
 $A = B$  ise YR2'e 8 bit  
 $A < B$  ise YR3'e 8 bit

LDA B  
CMA  
INC  
ADD A  
SPA  
BUN YR3  
SZA  
BUN YR1  
BUN YR2



B ← A-1 işlemini yapan DEC isimli altprogramı üç farklı yolla yazınız. (61)

DEC, \_\_\_\_\_  
LDA A  
CMA  
INC  
CMA  
STA B  
BUN DEC, I

DEC, \_\_\_\_\_  
CLA  
CMA  
ADD A  
STA B  
BUN DEC, I

DEC, \_\_\_\_\_  
CLA  
INC  
CMA  
INC  
ADD A  
STA B  
BUN DEC, I

BSA MOV  
HEX 100  
HEX 200  
DEC -16  
HLT

PT1, \_\_\_\_\_  
PT2, \_\_\_\_\_  
CTR, \_\_\_\_\_

Herhangi bir dil ile yazılmış kod parçaları

$F = 0$  → CLA  
STA F

$F = F + A + B$

→ LDA F  
ADD A  
ADD B  
STA F

$A = A - B$

→ LDA B  
CMA  
INC  
ADD A  
STA A

$B = A + B$

→ LDA B  
ADD A  
STA B

MOV, HEX 0  
LDA MOV, I  
STA PT1  
ISZ MOV  
LDA MOV, I  
STA PT2  
ISZ MOV  
LDA MOV, I  
STA CTR  
ISZ MOV  
LOP, LDA PT1, I  
STA PT2, I  
ISZ PT1  
ISZ PT2  
ISZ CTR  
BUN LOP  
BUN MOV, I

Adres	Komut
100	CLA
101	ADD 106
102	BUN 104
103	HLT
104	AND 107
105	BUN 103
106	A2DC
107	7EA9

Yanda verilen program temel bilgisayarın bellek biriminde saklıdır.

Her buyruk çalıştıktan sonra onaltılık olarak AC, PC, IR yazarsanız işeriklerini yazınız.

Adres	Komut	AC	PC	IR
100	CLA	0000	101	7800
101	ADD 106	A3E5	102	1106
102	BUN 104	A3E5	104	4104
103	HLT	8041	104	7001
104	AND 107	8041	105	0107
105	BUN 103	8041	103	4103
106	A3E5			
107	9C4B			

A3E5	→	1010	0011	1110	0101
9C4B	→	1001	1100	0100	1011
		<u>1000</u>	<u>0000</u>	<u>0100</u>	<u>0001</u>
		8	0	4	1

## Altprogram kullanan bir program

Adres	Program	Adres	Program
	ORG 100	109	SH4, HEX 0
100	LDA X	10A	CIL
101	BSA SH4	10B	CIL
102	STA X	10C	CIL
103	LDA Y	10D	CIL
104	BSA SH4	10E	AND MSK
105	STA Y	10F	BUN SH4
106	HLT	110	MSK, HEX FFF0
107	X, HEX 1234		END
108	Y, HEX 4321		

Temel bilgisayarın komut kümesini kullanarak

a) Bir karakter alma işlemini yapan altprogram

b) Bir karakter basma işlemini yapan altprogram

CIF, —  
 LOP, SKI  
 BUN LOP  
 INP  
 OUT  
 STA CHR  
 BUN CIF, I

CHR, —  
 COF, —  
 LDA CHR  
 LOP, SKO  
 BUN LOP  
 OUT  
 BUN COF, I

İki karakter alma işlemi yapan bir altprogram

IN2, —  
 FST, SKI  
 BUN FST  
 INP  
 OUT  
 BSA SH4  
 BSA SH4  
 SCD, SKI  
 BUN SCD

INP  
 OUT  
 BUN IN2, I  
 MSK, HEX FFF0  
 SH4, —  
 CIL  
 CIL  
 CIL  
 CIL  
 AND MSK  
 BUN SH4, I



# Bir Kesme Servis Programı

64

Adres Program

0	ZRO, BUN SRV
1	
...	
100	CLA
101	ION
102	LDA X
103	ADD Y
104	STA Z
...	

→ Dönüş adresi burada saklanır  
→ Servis yordamına şartsız daller.

→ Kesme burada oluşur.  
→ Kesmeden sonra program buraya döner.

200	SRV, STA SAC
	CIR
	STA SE
	SKI
	BUN NXT
	INP
	OUT
	STA PT1, I
	ISZ PT1
NXT, SKO	
BUN	EXT
LDA	PT2, I
OUT	
ISZ	PT2
EXT, LDA	SE
CIL	
LDA	SAC
ION	
BUN ZRO, I	

SAC, —  
SE, —  
PT1, —  
PT2, —

Kesme Servis Yordamı

SAC Akümülatörün saklandığı adres

SE Elden saklandığı adres

PT1 Giriş Buffer Göstergesi

PT2 Çıkış Buffer Göstergesi

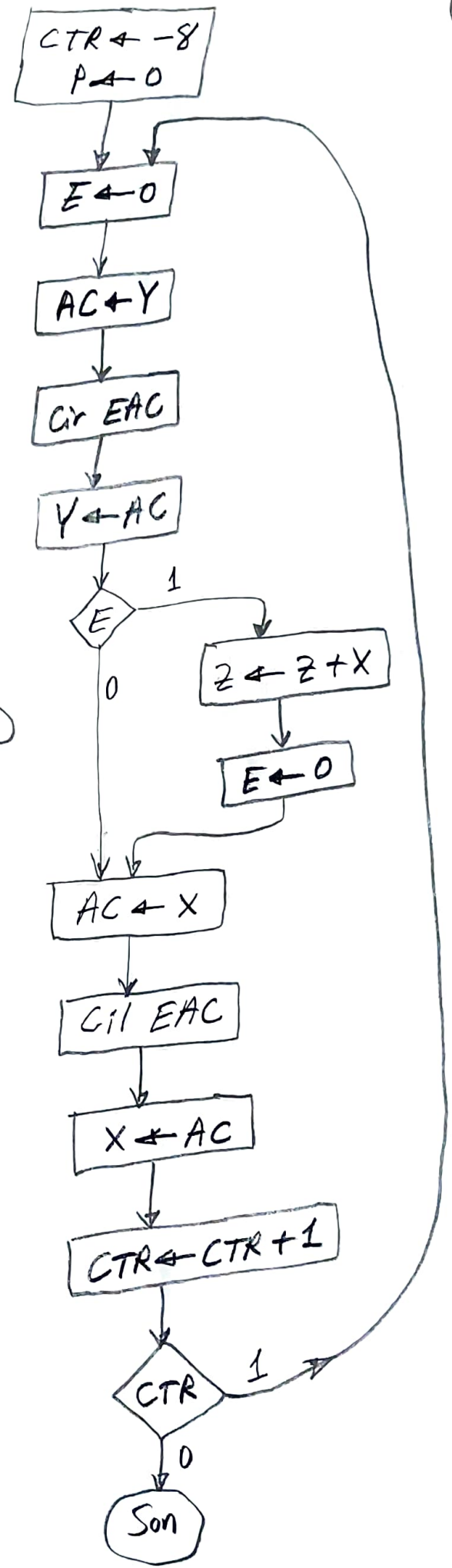
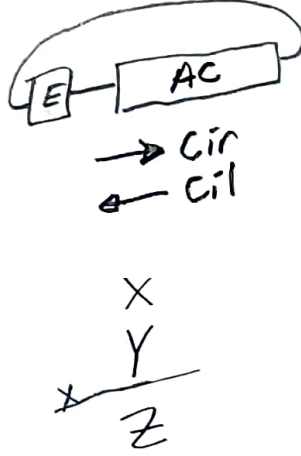
→ Daha önce çalıştırılan programa geri dsn.

İki pozitif sayının çarpımını  
ile ilgili program

ORG 100  
LOP, CLE  
LOA Y  
CIR  
STA Y  
SZE  
BUN ONE  
BUN ZRO  
ONE, LOA X  
ADD Z  
STA Z  
CLE  
ZRO, LOA X  
CIL  
STA X  
ISZ CTR  
BUN LOP  
HLT

CTR, DEX -8  
X, HEX C7  
Y, HEX 5A  
Z, HEX 0

$Z = X * Y$   
↓  
Sonuç  
↓  
Çarpılan



iki gift dayarlikli sayinin toplanmasi

(66)

LDA AL      AL, —  
ADD BL      AH, —  
STA CL      BL, —  
CLA          BH, —  
CIL          CL, —  
ADD AH      CH, —  
ADD BH  
STA CH  
HLT

AH AL

BH BL

+

CH CL

$F \leftarrow A * B$   
Tekrarli toplama  
mantigina gsr  
altprogrami yaz  
 $27 * 13$  isin  
13 kez 27'nin toplamı

CRP, CLE  
CLA  
INC  
CIL  
CIL  
CIL  
CIL  
CMA  
INC  
STA C  
LOA A  
STA D  
LOA B  
STA E  
CLA  
STA F  
LOP, CLE  
LOA E  
CIL  
STA E  
SZE  
BUN ONE  
BUN ZRO

ONE, LDA D  
ADD F  
STA F  
CLE  
ZRO, LDA D  
CIL  
STA D  
ISZ C  
BUN LOP  
BUN CRP, I

CRP, LDA A  
SZA  
BUN NZR  
BUN ZRO  
NZR, CMA  
INC  
STA C  
CLA  
LOP, ADD B  
ISZ C  
BUN LOP  
ZRO, STA F  
BUN CRP, I

$F \leftarrow A * B$

A, —  
B, —  
C, —  
D, —  
E, —  
F, —



## Veri bloğunun taşınması ile ilgili program

BSA MVE  
HEX 100  
HEX 200  
DEC -16  
HLT

PT1, \_\_\_\_\_  
PT2, \_\_\_\_\_  
CTR, \_\_\_\_\_  
MVE, \_\_\_\_\_

LDA MVE, I  
STA PT1  
ISZ MVE  
LDA MVE, I  
STA PT2  
ISZ MVE  
LDA MVE, I  
STA CTR  
ISZ MVE  
LOP, LDA PT1, I  
STA PT2, I  
ISZ PT1  
ISZ PT2  
ISZ CTR  
BUN LOP  
BUN MVE, I

Veriler, veri sayısı ve  
adresler  
korunuyor.

## 100 sayının toplanmasını yapan sembolik program (67)

ORG 100  
100, LDA ADS  
STA PTR  
LDA NBR  
STA CTR  
CLA  
LOP, ADD PTR, I  
ISZ PTR  
ISZ CTR  
BUN LOP  
STA SUM  
HLT

ADS, HEX 150  
PTR, HEX 0  
NBR, DEC -100  
CTR, HEX 0  
SUM, HEX 0

ORG 150  
150, DEC 75 ilk veri

DEC 23 Son veri  
END → Sembolik  
programın  
sonu

A adresindeki  
veri bozulmuyor.  
B ← içindeki bit sayısı (A)

SAY, \_\_\_\_\_

CLE

CLA

STA B

LDA A

SZA

BUN 601

BUN 602

601, CIL

SZE

BUN 603

BUN 601

603, CLE

ISZ B

SZA

BUN 601

602, BUN SAY, I

B ← Bitset Ters Çevir (A) (68)

C Sayı 9

G geçici bellek adresi

A adresindeki veri  
bozulmuyor.

TRS, \_\_\_\_\_

LDA A

STA G

CLE

CLA

INC

CIL

CIL

CIL

CIL

CMA

INC

STA C

LOP, LDA G

CIL

STA G

LDA B

CIR

STA B

ISZ C

BUN LOP

BUN TRS, I