

BUN CIK
 CMA
 INC
 CIK, STA PT1 I
 ISZ PT1
 LDA PT2
 ADD EKS
 STA PT2
 ISZ CTR
 BUN DON
 HLT
 PT1, HEX 100200
 PT2, HEX 309
 EKS, DEC 50
 CTR, DEC 10
 END

CIR
 CIL
 INC
 SPA
 SNA
 SZA
 SZE
 HLT
 INP
 OUT
 SKI
 SKO
 ION
 IOF

2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer eşlenik biti dizisi elde edilmektedir. Kodun assembly karşılığı

```

int NBR = 1380;
int PAR = 0;
for ( int i = 0; i < 8; i++ )
    PAR[ i ] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatör
  
```

3) [25] Herhangi bir anda bir işlemciye saklayıcıların P bazı bellek gözlerinin de $M[300] = 80$, $M[500] = 300$, kabul edelim. Load komutu belirtilen adresleme modu yüklemektedir. Buna göre aşağıdaki beş işlemin her biri belirtip AC'ye yüklenecek değerleri bulunuz.

a) Load #300

b) Load 200(XR)

c) Load R3

4) [25] Gereken sayıda 32K x 8 RAM ve kod çöz tasarımı blok halinde veriniz. Tasarımda adres, gösteriniz.

32

2

BIMU2060 - BİLGİSAYAR ORGANİZASYONU VE TASARIM
Prof. Dr. Ahmet SERTBAŞ – Araş. Gör. Dr. Ergün GÜMÜŞ – Araş. Gör.

SORULAR

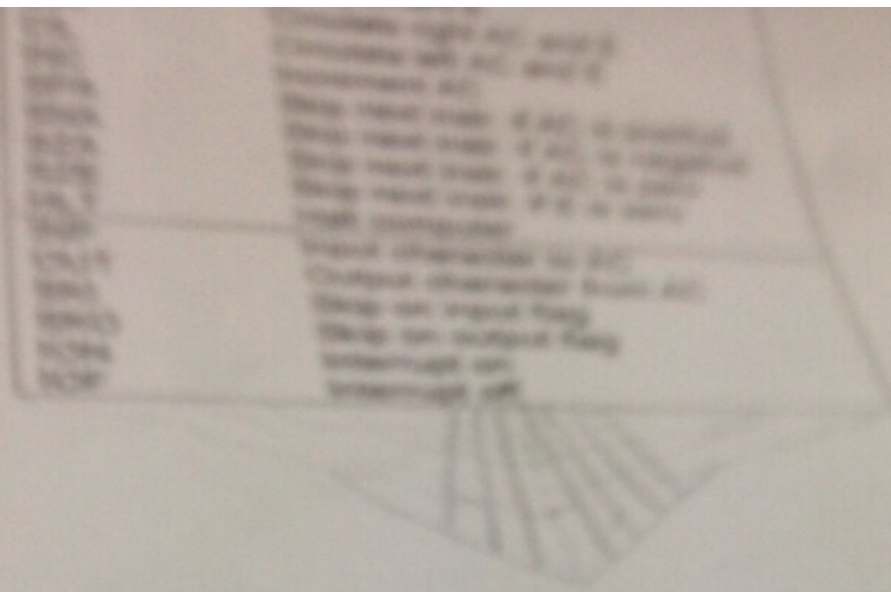
1) [25] Aşağıdaki assembly kodunun C dilindeki karşılığını yazınız. C kodunu gösteriniz.

ORG 100H
LDA YRM
CMA
INC
STA YRM
DON, BSA KON
ISZ PT1
ISZ PT2
ISZ PT3
ISZ YRM
BUN DON
HLT
KON, DEC 62
LDA PT2 I
CMA
INC
ADD PT1 I
SNA
BUN POZ
LDA PT2 I
STA PT3 I
BUN KON I
POZ, LDA PT1 I
STA PT3 I
BUN KON I
PT1, HEX 300
PT2, HEX 500
PT3, HEX 700
YRM, DEC 20
END

YRM aldıkları
akümülatörle kıyasla

→ konaya git ve
o'sa o'tla adresle
saka la

Sembol	Tanım
AND	AND
ADD	Add
LDA	Load
STA	Store
BUN	Branch
BSA	Branch
ISZ	Incr
CLA	Clear
CLE	Clear
CMA	Com
CME	Com
CIR	Circ
CIL	Circ
INC	Incr
SPA	Skip
SNA	Skip
SZA	Skip
SZE	Skip
HLT	Hal
INP	Inp
OUT	Out
SKI	Sk
SKO	Sk
ION	In
IOF	In



Yukarıda verilen 2'per bit EX-OR işlemi 8 bitlik bir assembly karpışına TBO komutlarını kullanarak yazınız.

// 16 bitlik bir sayı
// 8 bitlik bir sayı

1); // ^ operatörü EX-OR işlemidir.

saklayıcıların PC = 600, R3 = 550, XR = 100; RAM'deki M[500] = 300, M[550] = 500 değerlerine sahip olduğunu tesleme moduna göre AC'nin içeriğine kaynaktaki veriyi lemin her birinde hangi adresleme modunun kullanıldığını izle.

c) Load R3

d) Load (R3)+

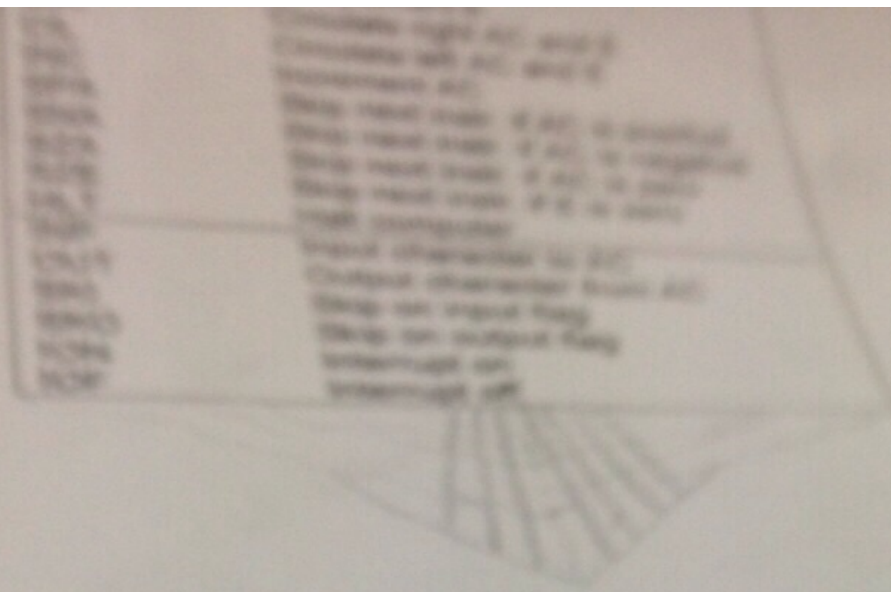
e) Load @550

e kod çözücü (decoder) kullanarak 128K x 16 RAM'da adres, veri giriş/çıkış ve gerekli kontrol işaretleri

$A_{15} - A_0$

$2^{10} \rightarrow 2^{15}$

23



Yukarıda verilen 2'per bit EX-OR'lanarak sonucu 8 bitlik bir assembly karpışına TBO komutlarını kullanarak yazınız.

// 16 bitlik bir sayı
// 8 bitlik bir sayı

1); // ^ operatörü EX-OR işlemidir.

saklayıcıların PC = 600, R3 = 550, XR = 100; RAM'deki M[500] = 300, M[550] = 500 değerlerine sahip olduğunu tesleme moduna göre AC'nin içeriğine kaynaktaki veriyi lemin her birinde hangi adresleme modunun kullanıldığını izle.

c) Load R3

d) Load (R3)+

e) Load @550

e kod çözücü (decoder) kullanarak 128K x 16 RAM'da adres, veri giriş/çıkış ve gerekli kontrol işaretleri

$A_{15} - A_0$

$2^{10} - 2^{15}$

23

SORULAR

1) [25] Aşağıdaki assembly kodunun C dilindeki karşılığını yazınız. C kodunda bellek gözünü M gösteriniz.

ORG 100H ; 100H adresinden başları
LDA CTR ; CTR'ye değeri yükleniyor
CMA ; CTR'in komplementi alınıyor
INC ; CTR arttırılıyor
STA CTR ; CTR'in değeri CTR'ye yazılıyor
CLA ; Akümülatör temizleniyor
INC ; PC arttırılıyor
CMA ; CTR'in komplementi alınıyor
INC ; CTR arttırılıyor
STA EKS ; EKS'ye değeri yazılıyor
DON, LDA PT1 ; PT1
ADD PT2 ; PT2
SNA ; SNA
BUN ÇIK ; ÇIK
CMA ; CMA
INC ; INC
ÇIK, STA PT1 ; PT1
ISZ PT1 ; PT1
LDA PT2 ; PT2
ADD EKS ; EKS
STA PT2 ; PT2
ISZ CTR ; CTR
BUN DON ; DON
HLT ; HLT
PT1, HEX 100200
PT2, HEX 309
EKS, DEC 50
CTR, DEC 10
END

Sembol	Tanımlama
AND	AND memory word to AC
ADD	Add memory word to AC
LDA	Load AC from memory
STA	Store content of AC into memory
BUN	Branch unconditionally
BSA	Branch and save return address
ISZ	Increment and skip if zero
CLA	Clear AC
CLE	Clear E
CMA	Complement AC
CME	Complement E
CIR	Circulate right AC and E
CIL	Circulate left AC and E
INC	Increment AC
SPA	Skip next instr. if AC is positive
SNA	Skip next instr. if AC is negative
SZA	Skip next instr. if AC is zero
SZE	Skip next instr. if E is zero
HLT	Halt computer
INP	Input character to AC
OUT	Output character from AC
SKI	Skip on input flag
SKO	Skip on output flag
ION	Interrupt on
IOF	Interrupt off

2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer biti EX-OR'lanarak selenik biti dizisi elde edilmektedir. Kodun assembly karşılığını TBO komutlarını kullanarak yazınız.

```

NBR = 1380; // 16 bitlik bir sayı
PAR = 0; // 8 bitlik bir sayı
for (int i = 0; i < 8; i++)
    PAR[i] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatörü EX-OR işlemidir

```

3) Herhangi bir anda bir işlemci içindeki saklayıcıların PC = 600, R3 = 55

FULLY ASSOCIATIVE MEMORY

C editi

OR6 100H

100 LDA YRM

AC = -YRM = -20 over

101 CMA

102 INC

103 STA YRM

104 ~~STA~~ ~~YRM~~

105 JSZ PT1 PT1 = 301

106 JSZ PT2 PT2 = 501

107 JSZ PT3 PT3 = 501

108 JSZ YRM YRM = -19

109 BUN DON

110 HLT

111 KON, DEC 62

112 LDA PT2 I → AC = 500 dan

AC = 501

113 CMA

114 INC → AC = -500

AC = -501

115 ADD PT1 I → -500 + 300

-501 + 301 = -200

116 SNA = Sonucu negatif = 200

117 BUN P02

118 LDA PT2 I → AC = 500

119 STA PT3 I PT3 = 500 olsun

120 BUN KON I → 105. yi 121 ile denek

121 P02, LDA PT1 I

122 STA PT3 I

123 BUN KON I

124 P02, LDA PT1 I

125 STA PT3 I

126 BUN KON I

127 P02, LDA PT1 I

128 STA PT3 I

129 BUN KON I

130 P02, LDA PT1 I

131 STA PT3 I

132 BUN KON I

133 P02, LDA PT1 I

134 STA PT3 I

135 BUN KON I

136 P02, LDA PT1 I

137 STA PT3 I

138 BUN KON I

139 P02, LDA PT1 I

140 STA PT3 I

141 BUN KON I

142 P02, LDA PT1 I

143 STA PT3 I

144 BUN KON I

145 P02, LDA PT1 I

146 STA PT3 I

147 BUN KON I

148 P02, LDA PT1 I

149 STA PT3 I

150 BUN KON I

151 P02, LDA PT1 I

152 STA PT3 I

153 BUN KON I

154 P02, LDA PT1 I

155 STA PT3 I

156 BUN KON I

157 P02, LDA PT1 I

158 STA PT3 I

159 BUN KON I

160 P02, LDA PT1 I

161 STA PT3 I

162 BUN KON I

163 P02, LDA PT1 I

164 STA PT3 I

165 BUN KON I

166 P02, LDA PT1 I

167 STA PT3 I

168 BUN KON I

169 P02, LDA PT1 I

170 STA PT3 I

171 BUN KON I

172 P02, LDA PT1 I

173 STA PT3 I

174 BUN KON I

175 P02, LDA PT1 I

176 STA PT3 I

177 BUN KON I

178 P02, LDA PT1 I

179 STA PT3 I

180 BUN KON I

181 P02, LDA PT1 I

182 STA PT3 I

183 BUN KON I

184 P02, LDA PT1 I

185 STA PT3 I

186 BUN KON I

187 P02, LDA PT1 I

188 STA PT3 I

189 BUN KON I

190 P02, LDA PT1 I

191 STA PT3 I

192 BUN KON I

193 P02, LDA PT1 I

194 STA PT3 I

195 BUN KON I

196 P02, LDA PT1 I

197 STA PT3 I

198 BUN KON I

199 P02, LDA PT1 I

200 STA PT3 I

201 BUN KON I

202 P02, LDA PT1 I

203 STA PT3 I

204 BUN KON I

205 P02, LDA PT1 I

206 STA PT3 I

207 BUN KON I

208 P02, LDA PT1 I

209 STA PT3 I

210 BUN KON I

211 P02, LDA PT1 I

212 STA PT3 I

213 BUN KON I

214 P02, LDA PT1 I

215 STA PT3 I

216 BUN KON I

217 P02, LDA PT1 I

218 STA PT3 I

219 BUN KON I

220 P02, LDA PT1 I

221 STA PT3 I

222 BUN KON I

223 P02, LDA PT1 I

224 STA PT3 I

225 BUN KON I

226 P02, LDA PT1 I

227 STA PT3 I

228 BUN KON I

229 P02, LDA PT1 I

230 STA PT3 I

231 BUN KON I

232 P02, LDA PT1 I

233 STA PT3 I

234 BUN KON I

235 P02, LDA PT1 I

236 STA PT3 I

237 BUN KON I

238 P02, LDA PT1 I

239 STA PT3 I

240 BUN KON I

241 P02, LDA PT1 I

242 STA PT3 I

243 BUN KON I

244 P02, LDA PT1 I

245 STA PT3 I

246 BUN KON I

247 P02, LDA PT1 I

248 STA PT3 I

249 BUN KON I

250 P02, LDA PT1 I

251 STA PT3 I

252 BUN KON I

253 P02, LDA PT1 I

254 STA PT3 I

255 BUN KON I

256 P02, LDA PT1 I

257 STA PT3 I

258 BUN KON I

259 P02, LDA PT1 I

260 STA PT3 I

261 BUN KON I

262 P02, LDA PT1 I

263 STA PT3 I

264 BUN KON I

265 P02, LDA PT1 I

266 STA PT3 I

267 BUN KON I

268 P02, LDA PT1 I

269 STA PT3 I

270 BUN KON I

271 P02, LDA PT1 I

272 STA PT3 I

273 BUN KON I

274 P02, LDA PT1 I

275 STA PT3 I

276 BUN KON I

277 P02, LDA PT1 I

278 STA PT3 I

279 BUN KON I

280 P02, LDA PT1 I

281 STA PT3 I

282 BUN KON I

283 P02, LDA PT1 I

284 STA PT3 I

285 BUN KON I

286 P02, LDA PT1 I

287 STA PT3 I

288 BUN KON I

289 P02, LDA PT1 I

290 STA PT3 I

291 BUN KON I

292 P02, LDA PT1 I

293 STA PT3 I

294 BUN KON I

295 P02, LDA PT1 I

296 STA PT3 I

297 BUN KON I

298 P02, LDA PT1 I

299 STA PT3 I

300 BUN KON I

301 P02, LDA PT1 I

302 STA PT3 I

303 BUN KON I

304 P02, LDA PT1 I

305 STA PT3 I

306 BUN KON I

307 P02, LDA PT1 I

308 STA PT3 I

309 BUN KON I

310 P02, LDA PT1 I

311 STA PT3 I

312 BUN KON I

313 P02, LDA PT1 I

314 STA PT3 I

315 BUN KON I

316 P02, LDA PT1 I

317 STA PT3 I

318 BUN KON I

319 P02, LDA PT1 I

320 STA PT3 I

321 BUN KON I

322 P02, LDA PT1 I

323 STA PT3 I

324 BUN KON I

325 P02, LDA PT1 I

326 STA PT3 I

327 BUN KON I

328 P02, LDA PT1 I

329 STA PT3 I

330 BUN KON I

331 P02, LDA PT1 I

332 STA PT3 I

333 BUN KON I

334 P02, LDA PT1 I

335 STA PT3 I

336 BUN KON I

337 P02, LDA PT1 I

338 STA PT3 I

339 BUN KON I

340 P02, LDA PT1 I

341 STA PT3 I

342 BUN KON I

343 P02, LDA PT1 I

344 STA PT3 I

345 BUN KON I

346 P02, LDA PT1 I

347 STA PT3 I

348 BUN KON I

349 P02, LDA PT1 I

350 STA PT3 I

351 BUN KON I

352 P02, LDA PT1 I

353 STA PT3 I

354 BUN KON I

SORULAR

1) [25] Aşağıdaki assembly kodunun C dilindeki karşılığını yazınız. C kodunda bellek gözetimi M[] ile gösteriniz.

```
ORG 100H
LDA CTR
CMA
INC
STA CTR
CLA
INC
CMA
INC
STA EKS
DON, LDA PT1
ADD PT2
SNA
BUN CIK
CMA
INC
CIK, STA PT1
ISZ PT1
LDA PT2
ADD EKS
STA PT2
ISZ CTR
BUN DON
HLT
PT1, HEX 100
PT2, HEX 309
EKS, DEC 50
CTR, DEC 10
END
```

Sembol	Tanım
AND	AND memory word to AC
ADD	Add memory word to AC
LDA	Load AC from memory
STA	Store content of AC into memory
BUN	Branch unconditionally
BSA	Branch and save return address
ISZ	Increment and skip if zero
CLA	Clear AC
CLE	Clear E
CMA	Complement AC
CME	Complement E
CIR	Circulate right AC and E
CIL	Circulate left AC and E
INC	Increment AC
SPA	Skip next instr. if AC is positive
SNA	Skip next instr. if AC is negative
SZA	Skip next instr. if AC is zero
SZE	Skip next instr. if E is zero
HLT	Halt computer
INP	Input character to AC
OUT	Output character from AC
SKI	Skip on input flag
SKO	Skip on output flag
ION	Interrupt on
IOF	Interrupt off

2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer biti EX-OR'lanarak sonuçta 8 bitlik bir eşlenik biti dizisi elde edilmektedir. Kodun assembly karşılığını TBO komutlarını kullanarak yazınız.

```
int NBR = 1380; // 16 bitlik bir sayı
int PAR = 0; // 8 bitlik bir sayı
for (int i = 0; i < 8; i++)
    PAR[i] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatörü EX-OR işlemidir.
```

3) [25] Herhangi bir anda bir işlemciye saklayıcıların PC = 600, R3 = 550, XR = 100; RAM'deki bazı bellek gözlemlerinin de M[300] = 80, M[500] = 300, M[550] = 500 değerlerine sahip olduğunu kabul edelim. Load komutu belirtilen adresleme moduna göre AC'nin içeriğine kaynaktaki veriyi yüklemektedir. Buna göre aşağıdaki beş işlemin her birinde hangi adresleme modunun kullanıldığını belirtip AC'ye yüklenecek değerleri bulunuz.

- a) Load #300 b) Load 200(XR) c) Load R3 d) Load (R3)+ e) Load @550

4) [25] Gereken sayıda 32K x 8 RAM ve kod çözücü (decoder) kullanarak 128K x 16 RAM tasarımını blok halinde veriniz. Tasarımda adres, veri giriş/çıkış ve gerekli kontrol işaretlerini gösteriniz.

CMA
INC
CIK, STA PT1
ISZ PT1
LDA PT2
ADD EKS
STA PT2
ISZ CTR
BUN DON
HLT

PT1, HEX 100200
PT2, HEX 309
EKS, DEC 50
CTR, DEC 10
END

SPA	Skip next instr. if AC
SNA	Skip next instr. if AC
SZA	Skip next instr. if AC
SZE	Skip next instr. if E
HLT	Halt computer
INP	Input character to A
OUT	Output character to A
SKI	Skip on input flag
SKO	Skip on output flag
ION	Interrupt on
IOF	Interrupt off

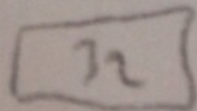
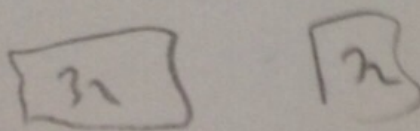
2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer biti EX-OR'lanarak eşlenik biti dizisi elde edilmektedir. Kodun assembly karşılığını TBO komutlarını

```
int NBR = 1380;           // 16 bitlik bir sayı
int PAR = 0;              // 8 bitlik bir sayı
for ( int i = 0; i < 8; i++ )
    PAR[ i ] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatörü EX-OR işlemi
```

3) [25] Herhangi bir anda bir işlemciye saklayıcıların PC = 600, R3 = 500, bazı bellek gözlerinin de M[300] = 80, M[500] = 300, M[550] = 500 kabul edelim. Load komutu belirtilen adresleme moduna göre AC'nin yüklenmektedir. Buna göre aşağıdaki beş işlemin her birinde hangi adres AC'ye yüklenecek değerleri bulunuz.

a) Load #300 b) Load 200(XR) c) Load R3 d) Load

25] Gereken sayıda 32K x 8 RAM ve kod çözücü (decoder) tasarımını blok halinde veriniz. Tasarımda adres, veri giriş/çıkış veriniz.



A
2⁵ · 2¹⁰
2¹⁵

CMA
INC
CIK, STA PT1
ISZ PT1
LDA PT2
ADD EKS
STA PT2
ISZ CTR
BUN DON
HLT

PT1, HEX 100200
PT2, HEX 309
EKS, DEC 50
CTR, DEC 10
END

SPA	Skip next instr. if AC
SNA	Skip next instr. if AC
SZA	Skip next instr. if AC
SZE	Skip next instr. if E
HLT	Halt computer
INP	Input character to A
OUT	Output character to A
SKI	Skip on input flag
SKO	Skip on output flag
ION	Interrupt on
IOF	Interrupt off

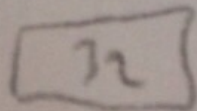
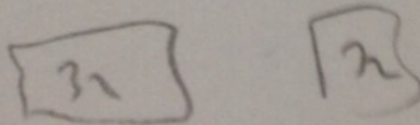
2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer biti EX-OR'lanarak eşlenik biti dizisi elde edilmektedir. Kodun assembly karşılığını TBO komutlarını

```
int NBR = 1380;           // 16 bitlik bir sayı
int PAR = 0;              // 8 bitlik bir sayı
for ( int i = 0; i < 8; i++ )
    PAR[ i ] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatörü EX-OR işlemi
```

3) [25] Herhangi bir anda bir işlemciye saklayıcıların PC = 600, R3 = 500, bazı bellek gözlerinin de M[300] = 80, M[500] = 300, M[550] = 500 kabul edelim. Load komutu belirtilen adresleme moduna göre AC'nin yüklenmektedir. Buna göre aşağıdaki beş işlemin her birinde hangi adresler AC'ye yüklenecek değerleri bulunuz.

a) Load #300 b) Load 200(XR) c) Load R3 d) Load

25] Gereken sayıda 32K x 8 RAM ve kod çözücü (decoder) tasarımını blok halinde veriniz. Tasarımda adres, veri giriş/çıkış veriniz.



A
2⁵ · 2¹⁰
2¹⁵

SORULAR

1) [25] Aşağıdaki assembly kodunun C dilindeki karşılığını yazınız. C kodunda bellek göstergesi M gösteriniz.

```

ORG 100H 100H adresinden başlandı
LDA CTR 32R değe yüklendi
CMA 2 Tamlemeşi alındı
INC 2 1 artırıldı
STA CTR 2 Değeri 32R değe yollandı
CLA 2 Akümülatör temizlendi
INC 2 1 artırıldı
CMA 2 Tamlemeşi alındı
INC 2 1 artırıldı
STA EKS 2 Aküdeki değeri EKS değeri
DON, LDA RT1 I 2
ADD PT2 I
SNA
BUN CIK
CMA
INC
CIK, STA PT1 I
ISZ PT1
LDA PT2
ADD EKS
STA PT2
ISZ CTR
BUN DON
HLT
PT1, HEX 100200
PT2, HEX 309
EKS, DEC 50
CTR, DEC 10
END
    
```

Sembol	Tanım
AND	AND memory word to AC
ADD	Add memory word to AC
LDA	Load AC from memory
STA	Store content of AC into mem
BUN	Branch unconditionally
BSA	Branch and save return addre
ISZ	Increment and skip if zero
CLA	Clear AC
CLE	Clear E
CMA	Complement AC
CME	Complement E
CIR	Circular right AC and E
CIL	Circular left AC and E
INC	Increment AC
SPA	Skip next instr. if AC is pos
SNA	Skip next instr. if AC is neg
SZA	Skip next instr. if AC is zer
SZE	Skip next instr. if E is zero
HLT	Halt computer
INP	Input character to AC
OUT	Output character from AC
SKI	Skip on input flag
SKO	Skip on output flag
ION	Interrupt on
IOF	Interrupt off

2) [25] Aşağıdaki kodda 16 bitlik NBR sayısının ardışık 2'şer biti EX-OR'lanarak soğuk bit dizisi elde edilmektedir. Kodun assembly karşılığına TBO komutlarına ko-

```

NBR = 1380; // 16 bitlik bir sayı
PAR = 0; // 8 bitlik bir sayı
for (int i = 0; i < 8; i++)
    PAR[i] = NBR[2*i] ^ NBR[2*i + 1]; // ^ operatörü EX-OR işlemidir.
    
```

5) Herhangi bir anda bir işlemci içindeki saklayıcıların PC = 600, R3 = 550, R4 = 300, M[500] = 500 değ-