

Komutların yazılış biçimlerini açıklarken bazı tanımlama harfleri kullanacağız. Önce bu harflerin anlamlarını verelim:

f = Dosya kayıtcısı (kaydedicisi) (File Register)
d = Gönderilen yer (destination) iki durum söz konusudur ;
 d=0Hedef W kayıtcısıdır.
 d=1Hedef dosya (F) kayıtcısıdır .
k = Sabit veya adres etiketi
b = Bit tanımlayıcı
b = Binary sayıları belirleyen harf (Örneğin b'00001111' gibi)
d = Desimal sayıları belirleyen harf (Örneğin d'16' gibi)
h = Hexadesimal sayıları belirleyen harf (Örneğin h'0A' gibi)

1.3.1. Byte Yönlendirmeli Komutlar

Bu grup komutlarda hedef kayıtcı W veya bir dosya kayıtcısı (f) olabilir. Kayıtcılar arası veri transferleri ve işlemleri bu komutlarla gerçekleşir. Şimdi bu komutları görelim:

ADDWF	W ve f kayıtcılarını toplar.
Dizim:	[etiket] ADDWF f,d
Operandlar:	0≤f≤127 d∈{0,1}
İşlem:	(W)+(f) → (hedef)
Etkilenen bayraklar:	C,DC,Z
Kodlama:	00 0111 dfff ffff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
Tamam: W kayıtcısının içeriğini f kayıtcısına ekler. Eğer d=0 ise, sonuç W kayıtcısının içerisinde depolanır. d=1 ise sonuç f kayıtcısının içerisinde geri saklanır.
Hafıza alanı (Byte): 1
Saat pulsı: 1

Örnek: ADDWF FSR, 0

Komuttan önce;
 W = h'17'
 FSR= h'C2' olsun.
Komuttan sonra;
d=0 olduğundan sonucun W de olduğuna dikkat ediniz.
 W = h'D9'
 FSR= h'C2' olur.

ANDWF	W'yi f ile mantıksal AND'le.
Dizim:	[etiket] ANDWF f,d
Operandlar:	0≤f≤127 d∈{0,1}
İşlem:	(W).AND.(f) → (hedef)

Etkilenen bayraklar: Z

Kodlama: 00 0101 dfff fff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: W kayıtcısı W kayıtcısı ile mantıksal AND'ler. AND mantığında girişlerden birinin 0 olması sonucu 0 yapacaktır. Sonucun 1 olması için tüm girişlerin 1 olması gerekir. Eğer d=0 ise, sonuç W kayıtcısı içinde saklanır. Eğer d=1 ise, sonuç f kayıtcısı içinde geri saklanır.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: ANDWF FSR, 1

Komuttan önce;

W = h'17'

FSR = h'C2' olsun.

Komuttan sonra;

h'17' = b'0001 0111'

AND h'C2' = b'1100 0010'

b'0000 0010' = h'02' ve ;

d=1 olduğundan sonucun f (burada FSR) de olduğuna dikkat ediniz.

W = h'17'

FSR = h'02'

CLRF F kayıtcısını sil. (Clear f)

Dizim: [etiket] CLRF f

Operandlar: 0 ≤ f ≤ 127

İşlem: 00h → (f) 1 → Z

Etkilenen bayraklar: Z

Kodlama: 00 0001 1fff fff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının içeriği silinir ve Z biti kurulur.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: CLRF SAY

Komuttan önce;

SAY = h'5A'

Komuttan sonra;

SAY = h'00'

Z = 1

CLRW W Kayıtcısını Sil.

Dizim: [etiket] CLRW

Operandlar: Yok

İşlem: 00h → (W) 1 → Z

Etkilenen bayraklar: Z
Kodlama: 00 0001 0000 0011
MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtlı adresi) tanımlar.
Tanım: W kayıtlı silinir. Z biti set edilir.
Hafıza alanı (Bayt): 1
Saat pulsı: 1
Örnek: CLRW

Komuttan önce;
W = h'5A'
Komuttan sonra;
W = h'00'
Z = 1

COMF F kayıtlısını tümle (tersini al).

Dizim: [etiket] COMF f,d

Operandlar: $0 \leq f \leq 127$

d=[0,1]

İşlem:

(f) → (hedef)

Etkilenen bayraklar: Z

Kodlama: 00 1001 dfff ffff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtlı adresi) tanımlar.

Tanım: F kayıtlısının içeriği tümlenmiştir. Eğer d=0 ise, sonuç W kayıtlısının içerisinde depolanır, d=1 ise, sonuç f kayıtlısının içerisinde geri saklanır.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: COMF test,0

Komuttan önce;
test = h'13' olsun.

Komuttan sonra;
d=0 olduğundan sonucun W'de olduğuna dikkat ediniz.

test = h'13' = b'0001 0011'

terslenirse; b'1110 1100' = h'EC' ve test = h'13' W = h'EC' olur.

DECF f kayıtlısının değerini 1 azalt.

Dizim: [etiket] DECF f,d

Operandlar: $0 \leq f \leq 127$ d=[0,1]

İşlem: (f) - 1 → (hedef)

Etkilenen bayraklar: Z

Kodlama: 00 0011 dfff ffff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının değerini 1 azaltır. Eğer d=0 ise, sonuç W kayıtcısının içerisinde depolanır, d=1 ise, sonuç f kayıtcısının içerisinde geri saklanır.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: DECF SAY, 1

Komuttan önce;

SAY = h'01'

Z = 0 olsun.

Komuttan sonra;

d=1 olduğundan sonucun f (burada SAY) de olduğuna dikkat ediniz.

SAY = h'00'

Z = 1

DECFSZ f kayıtcısının değerini 1 azalt , sonuç 0 ise bir sonraki komuta atla.

Dizim: [etiket] DECFSZ f,d

Operandlar: $0 \leq f \leq 127$ d ∈ {0,1}

İşlem: (f)-1 → (hedef); sonuç=0 ise atla

Etkilenen bayraklar: Yok

Kodlama: 00 1011 dfff ffff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının içeriği 1 azaltılır.

Aynı zamanda sorgulama

yapar. Kayıtcı içeriği 0 olursa program bir sonraki komuta atlar, değilse alttaki komut işlenir.

Eğer d=0 ise, sonuç W kayıtcısının içerisinde depolanır, d=1 ise, sonuç f kayıtcısının içerisinde geri saklanır.

Hafıza alanı (Bayt): 1

Saat pulsı: 1(2)

Sonuç 0'ın 2 saat pulsı , değilse 1 saat pulsı alır. Yani atlama olmadığı durumda uygulanması 1 , atlama durumunda 2 saat pulsı alır.

Örnek: DON DECFSZ CNT,1

GOTO LOOP

DEVAM
• } programın devamı
•

Komuttan önce;

PC = Adres DON

Komuttan sonra;

CNT= CNT-1

Eğer CNT= 0

PC= (Adres DEVAM)

Eğer CNT≠0,

PC={Adres DON+1} yani GOTO komutunun olduğu satır işlenir.

INCF f kayıtcısının değerini 1 artır.
Dizim: [etiket] INCF f,d
Operandlar: $0 \leq f \leq 127$ d ∈ {0,1}
İşlem: $(f) + 1 \rightarrow (\text{hedef})$
Etkilenen bayraklar: Z
Kodlama: 00 1010 dfff ffff
MSB tarafındaki 5 bit komut kodu, d biti hedef biti, f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
Tanın: f kayıtcısının değerini 1 artırır. Eğer d=0 ise, sonuç W kayıtcısının içerisinde depolanır. d=1 ise, sonuç f kayıtcısının içerisinde geri saklanır.
Hafıza alanı (Bayt): 1
Saat pulsı: 1
Örnek: INCF CNT,1

Komuttan önce;
CNT= h'FF'
Z=0

Komuttan sonra;
d=1 olduğundan sonucun f (burada CNT) de olduğuna dikkat ediniz. Ayrıca FF+1=00 olacağına da dikkat ediniz.

CNT= h'00'
Z= 1

INCFSZ f kayıtcısının değerini 1 artır, sonuç 0 ise bir sonraki komuta atla.
Dizim: [etiket] INCFSZ f,d
Operandlar: $0 \leq f \leq 127$ d ∈ {0,1}
İşlem: $(f)+1 \rightarrow (\text{hedef})$, sonuç=0 ise atla
Etkilenen bayraklar: Yok
Kodlama: 00 1111 dfff ffff
MSB tarafındaki 5 bit komut kodu, d biti hedef biti, f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
Tanın: f kayıtcısının içeriği 1 arttırılır. Aynı zamanda sorgulama yapar. Kayıtcı içeriği 0 olursa program bir sonraki komuta atlar, değilse alttaki komut işlenir. Eğer d=0 ise, sonuç W kayıtcısının içerisinde depolanır, d=1 ise, sonuç f kayıtcısının içerisinde geri saklanır.
Hafıza alanı (Bayt): 1
Saat pulsı: 1(2)
Sonuç 0'sa 2 saat pulsı, değilse 1 saat pulsı alır. Yani atlama olmadığı durumda uygulanması 1, atlama durumunda 2 saat pulsı alır.
Örnek: DON DECFSZ CNT,1

GOTO LOOP

DEVAM
• } programın devamı
•

Komuttan önce;
PC = Adres DON
Komuttan sonra;
CNT = CNT+1
Eğer CNT= 0 ise,
PC= {Adres DEVAM}
Eğer CNT≠0 ise,
PC={Adres DON+1} Yani GOTO komutunun bulunduğu satır işleme sokulur.

IORWF f ile W kayıtlarını mantıksal OR işlemine tabi tut.

Dizim: [etiket] IORWF f,d
Operandlar: $0 \leq f \leq 127$ d ∈ {0,1}
İşlem: (W).OR.(f) → (W)
Etkilenen bayraklar: Z
Kodlama: 00 0100 dfff ffff
MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtlı adresi) tanımlar.
Tanım: f ile W kayıtlarını mantıksal OR işlemine tabi tutar. OR mantığında girişlerden birinin 1 olması sonucu 1 yapacaktır. Sonucun 0 olması için tüm girişlerin 0 olması gerekir. Eğer d=0 ise, sonuç W kayıtlısının içerisinde depolanır, d=1 ise, sonuç f kayıtlısının içerisinde geri saklanır.
Hafıza alanı (Bayt): 1
Saat pılsı: 1
Örnek: IORWF RESULT, 0
Komuttan önce;
RESULT = h'13'
W = h'91'
Komuttan sonra;
d=0 olduğundan sonucun W'de olduğuna dikkat ediniz.
h'13' = b'0001 0011'
OR h'93' = b'1001 0011'
b'1001 0011' = h'93' ve ; RESULT = h'13' W = h'93' olur.

MOVF F kayıtlı içeriğini hedefe taşı.

Dizim: [etiket] MOVF f,d
Operandlar: $0 \leq f \leq 127$ d ∈ {0,1}
İşlem: (f) → (hedef)
Etkilenen bayraklar: Z
Kodlama: 00 1000 dfff ffff
MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtlı adresi) tanımlar.
Tanım: F kayıtlı içeriğini belirtilen hedefe taşır. Eğer d=0 ise, f içeriği W kayıtlısına taşır. d=1 ise, f içeriği f kayıtlısına taşır. d=1 durumu, status kayıtlısında Z bayrağını etkileyeceğinden dolayı, bir file kayıtlısının içeriğinin 0 olup olmadığının testinde kullanılabilir.
Hafıza alanı (Bayt): 1

Saat palsı: 1
Örnek: MOVF FSR, 0
Komuttan sonra;
d=0 olduğundan sonucun W' de olduğuna dikkat ediniz.

W=FSR olur.

MOVWF W'nin içeriğini f kayıtcısına taşı.

Dizim: [etiket] MOVWF f

Operandlar: 0≤f≤127

İşlem: (W)→(f)

Etkilenen bayraklar: Yok

Kodlama: 00 0000 1fff fff

MSB tarafındaki 5 bit komut kodu , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: W kayıtcısının içeriği f kayıtcısına taşınır.

Hafıza alanı (Bayt): 1

Saat palsı: 1

Örnek: MOVWF OPTION

Komuttan önce;

OPTION = h'FF'

W = h'4F'

Komuttan sonra;

Bu komutta hedefin (d) belirtilmediğine dikkat ediniz.

OPTION = h'4F'

W = h'4F'

NOP İşlem yok

Dizim: [etiket] NOP

Operandlar: Yok

İşlem: Yok

Etkilenen bayraklar: Yok

Kodlama: 00 0000 0xx0 0000

Tamam: 1 saat palsı zaman gecikmesi sağlar.

Hafıza alanı (Bayt): 1

Saat palsı: 1

Örnek: NOP

1 saat palsı zaman gecikmesi sağlar.

RLF f kayıtcısını elde biti (Carry) aracılığıyla sola döndür

Dizim: [etiket] RLF f,d

Operandlar: 0≤f≤127 d= [0,1]

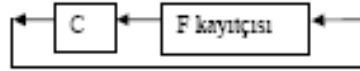
İşlem: Aşağıdaki tabloya bakınız.

Etkilenen bayraklar: C

Kodlama: 00 1101 dfff fff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının içeriği elde biti (C) içinden bir bit sola döndürülür. C biti f kayıtcısının MSB bitini taşır. Eğer d=0 ise, sonuç W kayıtcısının içerisine yerleştirilir. Eğer d=1 ise, sonuç f kayıtcısının içerisinde geri depolanır.



Hafıza alanı (Bayt): 1

Saat pılsı: 1

Örnek: RLF REG1,0

Komuttan önce;

REG1 = b'1110 0110'

C = 0

Komuttan sonra;

REG1 = b'1110 0110'

W = b'1100 1100'

C = 1 olur.

RRF f kayıtcısına elde biti (Carry) aracılığıyla sağa döndür

Dizin: [etiket] RRF f,d

Operandlar: $0 \leq f \leq 127$ d ∈ [0,1]

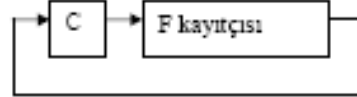
İşlem: Aşağıdaki tanıma bakınız.

Etkilenen bayraklar: C

Kodlama: 00 1100 dfff ffff

MSB tarafındaki 5 bit komut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının içeriği elde biti (C) içinden bir bit sağa döndürülür. C biti f kayıtcısının LSB bitini taşır. Eğer d=0 ise, sonuç W kayıtcısının içerisine yerleştirilir. Eğer d=1 ise, sonuç f kayıtcısının içerisinde geri depolanır.



Hafıza alanı (Bayt): 1

Saat pılsı: 1

Örnek: RRF REG1,0

Komuttan önce;

REG1 = b'1110 0110'

C = 0

Komuttan sonra;

REG1 = b'1110 0110'

W = b'0111 0011'

C = 0

SUBWF F kayıtcısından W kayıtcısını çıkart.

Dizim: [etiket] SUBWF f,d
 Operandlar: $0 \leq f \leq 127$ d=[0,1]
 İşlem: $(f) - (W) \rightarrow (\text{hedef})$
 Etkilenen bayraklar: C,D,C,Z
 Kodlama: 00 0010 dfff fff
 MSB tarafındaki 5 bit konut kodu , d biti hedef biti , f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
 Tanım: W kayıtcısı f kayıtcısından çıkartılır. d=1 ise, sonuç f kayıtcısı içerisinde geri saklanır.
 Hafıza alanı (Bayt): 1
 Saat pulsu: 1
 Örnek 1: SUBWF REG1,1

Komuttan önce;
 REG1 = 3
 W = 2
 C = ?
 Z = ?
 Komuttan sonra;
 REG1 = 1
 W = 2
 C = 1, sonuç pozitif
 Z = 0

Örnek 2: Komuttan önce;
 REG1 = 2
 W = 2
 C = ?
 Z = ?

Komuttan sonra;
 REG1 = 0
 W = 2
 C = 1, sonuç sıfır
 Z = 1

Örnek 3: Komuttan önce;
 REG1 = 1
 W = 2
 C = ?
 Z = ?

Komuttan sonra;
 REG1 = FF
 W = 2
 C = 0, sonuç negatif
 Z = 1

SWAPF f yi takas et

Dizim: [etiket] SWAPF f,d

Operandlar: $0 \leq k \leq 255$ $d \in \{0,1\}$

İşlem: $(f \langle 3:0 \rangle) \rightarrow (\text{hedef} \langle 7:4 \rangle), (f \langle 7:4 \rangle) \rightarrow (\text{hedef} \langle 3:0 \rangle)$
Yani f kayıtcısının LSB tarafındaki 4 biti (1.Dijiti) ile MSB tarafındaki 4 biti (2.dijiti) yer değiştirir. Eğer $d=0$ ise, sonuç W kayıtcısının içerisine yerleştirilir. Eğer $d=1$ ise, sonuç f kayıtcısının içerisinde geri depolanır.

Etkilenen bayraklar: Yok

Kodlama: 00 1110 dfff ffff

MSB tarafındaki 5 bit komut kodu, d biti hedef biti, f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: f kayıtcısının yüksek ve alçak baytları dönüştürülür. $d=0$ ise, sonuç W kayıtcısı içine yerleştirilir. Eğer $d=1$ ise, sonuç f kayıtcısı içine yerleştirilir.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: SWAPF ISIK, 0

Komuttan önce;
ISIK = h'A5' olsun.

Komuttan sonra;
ISIK = h'A5'
W = h'5A' olur.

XORWF f ile W 'nin içeriğini mantıksal EXOR'la.

Dizim: [etiket] XORWF f,d

Operandlar: $0 \leq k \leq 127$ $d \in \{0,1\}$

İşlem: $(W) .XOR. (f) \rightarrow (\text{hedef})$

Etkilenen bayraklar: Z

Kodlama: 00 0110 dfff ffff

MSB tarafındaki 5 bit komut kodu, d biti hedef biti, f ile belirtilen LSB tarafındaki 5 bit ise dosya adresini (f kayıtcı adresi) tanımlar.

Tamam: W kayıtcısının içeriği, f kayıtcısı ile mantıksal EXOR'lanır. EXOR mantığında aynı bitlerde çıkış 0, farklı bitlerde çıkış 1 olur. Eğer $d=0$ ise, sonuç W kayıtcısı içinde depolanır, $d=1$ ise sonuç f kayıtcısı içine geri yerleştirilir.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: XORWF REG, 1

Komuttan önce;
REG = h'AF
W = h'B5

Komuttan sonra;
h'AF' = b'1010 1111'
EXOR h'B5' = b'1011 0101'
b'0001 1010' = h'1A' ve;
REG = h'1A
W = h'B5

1.3.2. Bit Yönlendirmeli Komutlar

Bit Yönlendirmeli komutlarda dikkat edilirse hedef (d) belirtilmemiştir. Sonuç her zaman f kayıtçısının içersindedir. Şimdi bu komutları görelim;

BCF	F kayıtçısının belirtilen bitini sıfırla.
Dizim:	[etiket] BCF f,b
Operandlar:	$0 \leq f \leq 127$
İşlem:	$0 \rightarrow (f < b >)$
Etkilenen bayraklar:	Yok
Kodlama:	<u>01 00bb bfff fff</u>
MSB tarafındaki 4 bit komut kodu (opcode) , 3 tane b biti hedef biti belirler , f ile belirtilen LSB tarafındaki 7 bit ise dosya adresini (f kayıtçı adresi) tanımlar.	
Tamam: f içindeki 'b.' biti silinir.	
Hafıza alanı (Bayt):	1
Saat pulsü:	1

Örnek: BCF TEST,7

Komuttan önce;

TEST = h'C7' olsun.

Komuttan sonra;

h'C7'=b' 1100 0111'



0 olur ve;
TEST = h'47' olur.

BSF	F kayıtçısının belirtilen bitini kur (birle).
Dizim:	[etiket] BSF f,b
Operandlar:	$0 \leq f \leq 127$ $0 \leq b \leq 7$
İşlem:	$1 \rightarrow (f < b >)$
Etkilenen bayraklar:	Yok
Kodlama:	<u>01 01bb bfff fff</u>
MSB tarafındaki 4 bit komut kodu (opcode) , 3 tane b biti hedef biti belirler , f ile belirtilen LSB tarafındaki 7 bit ise dosya adresini (f kayıtçı adresi) tanımlar.	
Tamam: f kayıtçısı içindeki 'b.' Bit 1'e kururur.	
Hafıza alanı (Bayt):	1
Saat pulsü:	1
Örnek:	BSF TEST, 7
Komuttan önce;	
TEST = h'0A' olsun.	
Komuttan sonra;	
h'0A'=b' 0000 1010'	
↓	
1 olur ve; TEST = h'1A' olur	

BTFSC f kayıtcısının belirlenen biti 0 ise bundan sonraki komutu atla.

Dizim: [etiket] BTFSC f,b
 Operandlar: $0 \leq f \leq 127$ $0 \leq b \leq 7$
 İşlem: $F() = 0$ ise atla
 Etkilenen bayraklar: Yok
 Kodlama: 01 01bb bfff ffff
 MSB tarafındaki 4 bit komut kodu (opcode) , 3 tane b biti hedef biti belirler , f ile belirtilen LSB tarafındaki 7 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
 Tanım: f kayıtcısı içindeki "b." Bitin 0 olup olmadığı test edilir. Eğer "b." bit 0 ise, bu komutun altındaki komut işlenmez bir sonraki komuta sapılır. Aksi durumda ise sıradaki komut işlenir.
 Hafıza alanı (Bayt): 1
 Saat pulsı: 1(2)
 Test edilen bit 1 ise 1 saat pulsı , 0 ise 2 saat pulsı alır.
 Örnek: BASLA BTFSC PORTB,1 ;B portunun 1.bitini test et.
 GOTO BASLA ; 0 değilse bu satır işlenir, BASLA etiketine dallanır.
 ;Yani 0 olana kadar test etmeye devam edecek.
 BSF PORTB,1 ;0 ise bu satır işlenir ve B portunun 1. biti bu komut ile 1 yapılır. Eğer bu pine bir led bağlı ise yanar.

BTFSS f kayıtcısının belirlenen biti 1 ise bundan sonraki komutu atla.

Dizim: [etiket] BTFSS f,b
 Operandlar: $0 \leq f \leq 127$ $0 \leq b \leq 127$
 İşlem: Eğer $(f) = 1$ ise atla
 Etkilenen bayraklar: Yok
 Kodlama: 01 11bb bfff ffff
 MSB tarafındaki 4 bit komut kodu (opcode) , 3 tane b biti hedef biti belirler , f ile belirtilen LSB tarafındaki 7 bit ise dosya adresini (f kayıtcı adresi) tanımlar.
 Tanım: f kayıtcısı içindeki "b." Bitin 1 olup olmadığı test edilir. Eğer "b." bit 1 ise, bu komutun altındaki komut işlenmez bir sonraki komuta sapılır. Aksi durumda ise sıradaki komut işlenir.
 Hafıza alanı (Bayt): 1
 Saat pulsı: 1(2)
 Test edilen bit 0 ise 1 saat pulsı , 1 ise 2 saat pulsı alır.
 Örnek: BASLA BTFSS PORTA,1 ;A portunun 1.bitini test et.
 GOTO BASLA ;1 değilse bu satır işlenir, BASLA etiketine dallanır.
 ;Yani 1 olana kadar test etmeye devam edecek.
 BSF PORTB,1 ;1 ise bu satır işlenir ve B portunun 1. biti bu komut ile 1 yapılır. Eğer bu pine bir led bağlı ise yanar.

1.3.3. Sabit İşleyen Komutlar

Sabit yönlendirmeli komutlarda dikkat edilirse hedef (d) belirtilmemiştir. Sonuç her zaman W kayıtçısının içersindedir. Ayrıca bu gruptaki komutların W dışındaki kayıtçılar ile kullanılmadığına dikkat ediniz. Şimdi bu komutları görelim;

ANDLW	W ile birlikte 8 bitlik k sabitini mantıksal AND'le.
Dizim:	[etiket] ANDLW k
Operandlar:	$0 \leq k \leq 255$
İşlem:	$(W) \cdot \text{AND} (k) \rightarrow (W)$
Etkilenen bayraklar:	Z
Kodlama:	11 1001 kkkk kkkk

MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.

Tamam-W kayıtçısının içeriği 8-bitlik sabit 'k' ile mantıksal AND'lenir. Sonuç,W kayıtçısına geri konur.

Hafıza alanı (Bayt): 1
Saat palsi: 1
Örnek: ANDLW h'5F'

Komuttan önce;
W = h'A3'
Komuttan sonra;
h'A3' = b'1010 0011'
AND h'5F' = b'0101 1111'
b'0000 0011' = h'03' ve ;
W = h'03' olur.

ADDLW	W ile birlikte 8 bitlik k sabitini toplar.
Dizim:	[etiket] ADDLW k
Operandlar:	$0 \leq k \leq 255$
İşlem:	$(W) + k \rightarrow (W)$
Etkilenen bayraklar:	C,D,C,Z
Kodlama:	11 111x kkkk kkkk

MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat edin.

Hafıza alanı (Bayt): 1
Saat palsi: 1
Örnek: ADDLW h'15'
Komuttan önce;
W = h'10'
Komuttan sonra; W = h'25' olur.

IORLW	W ile birlikte kapsayan OR yazımı				
Dizim:	[etiket] IORLW k				
Operandlar:	$0 \leq k \leq 255$				
İşlem:	$(W).OR.(k) \rightarrow (W)$				
Etkilenen bayraklar:	Z				
Kodlama:	<table><tr><td>1</td><td>1000</td><td>kkkk</td><td>kkkk</td></tr></table>	1	1000	kkkk	kkkk
1	1000	kkkk	kkkk		
MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.					
Tanım: W kayıtcısının içeriği 8-bitlik sabit 'k' ile OR'lanır. Sonuç, W kayıtcısına geri konur.					
Hafıza alanı (Bayt):	1				
Saat pulsü:	1				
Örnek:	IORLW h'35'				
Komuttan önce;					
W= h'9A'					
Komuttan sonra;					
h'9A'=b'1001 1010'					
OR h'35'=b'0011 0101'					
b'1011 1111'=h'BF' ve ; W= h'BF' ve Z=1 olur.					
MOVLW	W kayıtcısına bir sayı/sabit yükle.				
Dizim:	[etiket] MOVLW k				
Operandlar:	$0 \leq k \leq 255$				
İşlem:	$k \rightarrow (W)$				
Etkilenen bayraklar:	Yok				
Kodlama:	<table><tr><td>1</td><td>00xx</td><td>kkkk</td><td>kkkk</td></tr></table>	1	00xx	kkkk	kkkk
1	00xx	kkkk	kkkk		
MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.					
Tanım: 8-bitlik sabit 'k', W kayıtcısına yüklenir.					
Hafıza alanı (Bayt):	1				
Saat pulsü:	1				
Örnek:	MOVLW h'5A'				
Komuttan önce;					
W= h'9A' olsun ;					
Komuttan sonra;					
W= h'5A' olur.					
RETLW	Altprogramdan W'ye bir sayı/sabit yükle ve geri dön.				
Dizim:	[etiket] RETLW k				
Operandlar:	$0 \leq k \leq 255$				
İşlem:	$(k) \rightarrow (W)$				
TOS \rightarrow (PC)					
Etkilenen bayraklar:	Yok				
Kodlama:	<table><tr><td>1</td><td>01xx</td><td>kkkk</td><td>kkkk</td></tr></table>	1	01xx	kkkk	kkkk
1	01xx	kkkk	kkkk		
MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.					

Tanım: Altprogramdan W'ye bir k sayı/sabiti yükleyerek dönmek için kullanılır. Dönüş TOS (yığın üstü- yani geri dönüş adresinin bulunduğu hafıza alanı) değerinin PC (program sayacı) ye aktarılmasıyla yapılır. Bu komut daha çok çevrim tablolarında kullanılır. Herde detaylı işlenecektir.

Hafıza alanı (Bayt): 1

Saat palsi: 2

Örnek: RETLW h'3F'

Komuttan önce W'nin içeriği ne olursa olsun komuttan sonra W=h'3F' yapılır ve anaprograma geri döndürülür.

SUBLW Bir k sayı/sabitten W'yi çıkart.

Dizim: [etiket] SUBLW k

Operandlar: $0 \leq k \leq 255$

İşlem: $k - (W) \rightarrow (W)$

Etkilenen bayraklar: C, DC, Z

Kodlama: [1 110x kkkk kkkk]

MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.

Tanım: W kayıtcısı (2'nin tümleyeni metodu ile) sekiz bitlik 'k' yazımından çıkartılır. Sonuç, W kayıtcısı içerisine yerleştirilir.

Hafıza alanı (Bayt): 1

Saat palsi: 1

Örnek 1:

SUBLW h'02'

Komuttan önce;

W = 1

C = ?

Z = ?

Komuttan sonra;

W = 1

C = 1, sonuç pozitif

Z = 0

Örnek 2: Komuttan önce;

W = 2

C = ?

Z = ?

Komuttan sonra;

W = 0

C = 1, sonuç sıfır

Z = 1

Örnek 3: Komuttan önce;

W = 3

C = ?

Z = ?

Komuttan sonra;

W = FF

C = 0, sonuç negatif
Z = 1

XORLW Bir k sayı/sabit ile W'yi mantıksal EXOR'la.

Dizim: [etiket] XORLW k
Operandlar: $0 \leq k \leq 255$
İşlem: $(W) \text{ .XOR. } k \rightarrow (W)$
Etkilenen bayraklar: Z
Kodlama: **11 1010 kkkk kkkk**

MSB tarafındaki 5 bit komut kodu, k ile belirtilen LSB tarafındaki 8 bit ise 1 baytlık sabit değeri tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.

Tanım: W kayıtcısının içeriği, sekiz bitlik 'k' yazının ile mantıksal EXOR'lanır. EXOR mantığında aynı bitlerde çıkış 0, farklı bitlerde çıkış 1 olur. Sonuç W kayıtcısı içine yerleştirilir.

Hafıza alanı (Bayt): 1
Saat pulsu: 1
Örnek: XORLW h'AF'

Komuttan önce;
W = h'B5' olsun.
Komuttan sonra;
h'AF' = b'1010 1111'
EXOR h'B5' = b'1011 0101'
b'0001 1010' = h'1A' ve ;
W = h'1A' olur.

1.3.4. Kontrol Komutları

Kontrol komutları program akışını belirleyen komutlardır. Altprogram çağırma, şartsız dallanma ve uyku moduna geçme gibi olaylar bu komutlarla gerçekleşir. Bu komutlarda hedef genelde bir adres değeridir. Şimdi bu komutları görelim;

CALL Altprogram çağır .

Dizim: [etiket] CALL k
Operandlar: $0 \leq k \leq 2047$
İşlem: $(PC)+1 \rightarrow TOS$,
PC: Program sayacı (program counter)
TOS: Yığın üstü (Top Of Stack)
 $k \rightarrow (PC<10:0>)$,
 $(PCLATH<4:3>) \rightarrow (PC<12:11>)$
Etkilenen bayraklar: Yok
Kodlama: **10 0kkk kkkk kkkk**

MSB tarafındaki 3 bit komut kodu, k ile belirtilen LSB tarafındaki 11 bit ise altprogram başlangıç adresini tanımlar. Burada hedef (d) in kodlanmadığına dikkat ediniz.

Tamam: Altprogram çağırır. İlk olarak, geri-dönüş adresi (PC+1) yığına itilir. Onbir bitlik altprogramın başlangıç adresi, PC bitleri <10:0> içerisine yüklenir. PC'nin üst bitleri PC<12:11>, PCLATH'in <4:3> bitlerinden yüklenir. CALL mutlaka RETURN komutu ile birlikte kullanılmalıdır. RETURN komutu ilerde anlatılacaktır.

Hafıza alanı (Bayt): 1

Saat pulsı: 2

Örnek: DON CALL SAYAC

DON'un bir etiket olduğuna aynı zamanda CALL komutunun şu anki adresini tuttuğuna dikkat ediniz. SAYAC ise altprogramın başlangıç adresini tutan etikettir. Bu durumda ;

Komuttan önce;

PC = {Adres DON}

Komuttan sonra;

PC = {Adres SAYAC}

TOS = {Adres DON} olur.

CLRWD Bekçi köpeği zamanlayıcısını (Watchdog Timer) sil.

Dizim: [etiket] CLRWD

Operandlar: Yok

İşlem: 00h → WDT

0 → WDT ön-bölücüsü

1 → $\overline{T0}$

1 → \overline{PD}

Etkilenen bayraklar: T0, PD

Kodlama: 00 0000 0110 0100

Tamam: CLRWD komutu, Watchdog Timer'ı reset eder. Bu komut aynı zamanda WDT'nin ön bölücüsünün de resetlenmesine sebep olur. T0 ve PD durum bitleri de set edilir. Bu bitlerin işlevleri için status kayıtçısı konusuna bakınız.

Hafıza alanı (Bayt): 1

Saat pulsı: 1

Örnek: CLRWD

Komuttan önce;

WDT sayacı = ?

Komuttan sonra;

WDT sayacı = h'00'

WDT ön-bölücü değeri = 0

T0 = 1

PD = 1

GOTO Adres'e git

Dizim: [etiket] GOTO k

Operandlar: 0 ≤ k ≤ 2047

İşlem: $k \rightarrow (PC<10:0>)$
 (PCLATH <4:3>) \rightarrow (PC<12:11>)
 Etkilenen bayraklar: Yok
 Kodlama:

10	1kkk	kkkk	kkkk
----	------	------	------

 Tanım: GOTO, koşulsuz bir sıpma komutudur. K nın belirlediği 11-bit'lik adres PC bitlerinin <10:0> içerisine yüklenir. PC'nın üst bitleri (PC<12:11>) , PCLATH<4:3>'ten yüklenir ve oluşan adrese sapılır.
 Hafıza alanı (Bayt): 1
 Saat pılsı: 2
 Örnek: GOTO SAYAC
 Komuttan sonra:
 $PC = \{Adres\ SAYAC\}$ olur.

RETFIE Kesme altprogramından geri dön.

 Dızım: [etiket] RETFIE
 Operandlar: Yok
 İşlem: $TOS \rightarrow (PC)$; 1 \rightarrow GIE TOS da geri dönüş adresi bulunur.
 Etkilenen bayraklar: Yok
 Kodlama:

00	0000	0000	1001
----	------	------	------

 Tanım: Kesme altprogramından geri dönmek için kullanılır. Yığında (TOS) bulunan geri dönüş adresi PC'ye yüklenir. INTCON kesme kayıtısının GIE (Genel kesme yetkisi) biti set edilir. Bu, iki döngülik bir komuttur.
 Hafıza alanı (Bayt): 1
 Saat pılsı: 2
 Örnek: RETFIE
 Komuttan sonra:
 $PC = TOS$
 $GIE = 1$

RETURN Altprogramdan geri dön.

 Dızım: [etiket] RETURN
 Operandlar: Yok
 İşlem: $TOS \rightarrow (PC)$
 Etkilenen bayraklar: Yok
 Kodlama:

00	0000	0000	0000
----	------	------	------

 Tanım: Altprogramdan geri dönmeyi sağlar. Yığında (TOS) bulunan geri dönüş adresi PC'ye yüklenir. Bu, iki döngülik bir komuttur.
 Hafıza alanı (Bayt): 1
 Saat pılsı: 2
 Örnek: RETURN
 Komuttan sonra:
 $PC = TOS$; yani PC'ye geri dönüş adresi yüklenir ve ana programa yüklenir.

SLEEP	Uyku (Standby) moduna gir.
Dizim:	[etiket] SLEEP
Operandlar:	Yok
İşlem:	00h → WDT
0 → $\overline{\text{WDT}}$ ön-bölücüsü	
1 → $\overline{\text{TO}}$	
0 → $\overline{\text{PD}}$	
Etkilenen bayraklar:	$\overline{\text{TO}}$, $\overline{\text{PD}}$
Kodlama:	00 0000 0110 0011
<p>Tanım: Güç yok durum biti PD (Power down) sıfırlanır. Süre aşımı TO (Time-out) durum biti ise 1'e kurulur. Watchdog zamanlayıcısı ve ön-bölücüsü silinir. Osilatörün durmasıyla işlemci SLEEP moduna girer. PIC bu durumda çok az güç harcar. Arada bir kontrol gereken güvenlik işlerinde, ya da belirli sürelerde yapılacak işler bittiğinde PIC uyuma moduna sokulur.</p>	
Hafıza alanı (Bayt):	1
Saat pilsı:	1
Örnek:	SLEEP

1.4. Sayı ve Karakterlerin Yazılışı

PIC assembly komutlarında sayılar heksadesimal, binary veya desimal formda kullanılabilir. Değişik kaynaklarda kullanılan sayı ve karakter gösteriliş biçimleriyle karşılaştığımızda bunları okuyabilmemiz için aşağıda örnekler verilmiştir.

1.4.1. Heksadesimal Sayılar

Heksadesimal sayılar '0x', '0' veya 'h' harfleriyle başlamalıdır. Örneğin, STATUS kayıtcısına 03 adresini atamak için aşağıda gösterilen yazılış biçimleri kullanılabilir.

```
STATUS EQU 0x03
STATUS EQU 03h
STATUS EQU h'03'
```

MOVLW komutu ile W kayıtcısı içerisine yüklenecek olan FF heksadesimal sabiti ise aşağıdaki gibi yazılabilir:

```
MOVLW 0 x FF veya MOVLW h' FF'
Biz hexadesimal formatı h' xx' şeklinde kullanacağız.
```

1.4.2. Binary Sayılar

Binary sayılar b harfi ile başlamalıdır. Örneğin 00001010 binary sayısını W kayıtcısı içerisine yüklemek için aşağıdaki gibi yazılmalıdır.

```
MOVLW b'00001010'
```

Referans: MİKRODENETLEYİCİ PROGRAMLAMA, MESLEKİ EĞİTİM VE ÖĞRETİM SİSTEMİNİN GÜÇLENDİRİLMESİ PROJESİ, ELEKTRİK ELEKTRONİK TEKNOLOJİSİ