

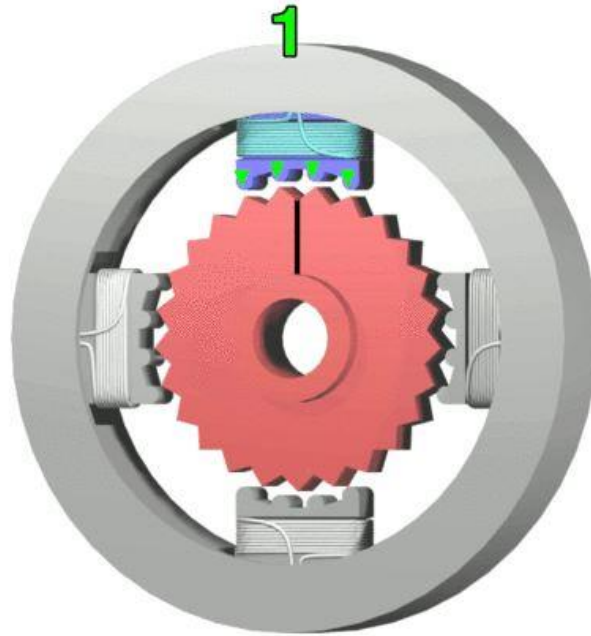
ADIM MOTORUNUN MİKROBİLGİSAYARLARLA DENETİMİ

1. GENEL TANITIM

Adım motorları, küçük momentli sayısal uygulamalarda son derece yaygınlaşmakta olan sayısal motorlardır. Hem açık hem de kapalı çevrimli kontrol sistemlerinde, en az donanımla yüksek doğrulukta kullanılabilirler. Bazı adım motorları istenilen konuma bir derece veya daha küçük hatayla konumlandırılabilir.

Çok sayıda adım motoru tasarımı olmasına rağmen, bu motorlar iki değişik temel tür altında toplanabilir: değişken relüktanslı veya hibrid yapılı.

Bir motorun iç yapısı çok farklı gözükse bile, motorun uç davranışını bu iki türün birisi cinsinden tanımlamak her zaman mümkündür.



Şekil – 1. Bir adım motorunun iç yapısı.

Adım motorunun esas özelliği, anahtarlamalı uyarma değişimlerini, rotor konumunun tam olarak belirlenmiş artımlarına dönüştürmektir. Rotorun doğru konumlanması, genellikle, motorun durağan ve dönen parçaları üzerindeki demir dişlerin manyetik olarak aynı hizaya getirilmesiyle sağlanır. Hibrid motor için manyetik akının ana kaynağı sürekli bir mıknatıstır ve bir veya daha fazla sargıdaki doğru akımlar, akıyı alternatif yollara yönlendirir.

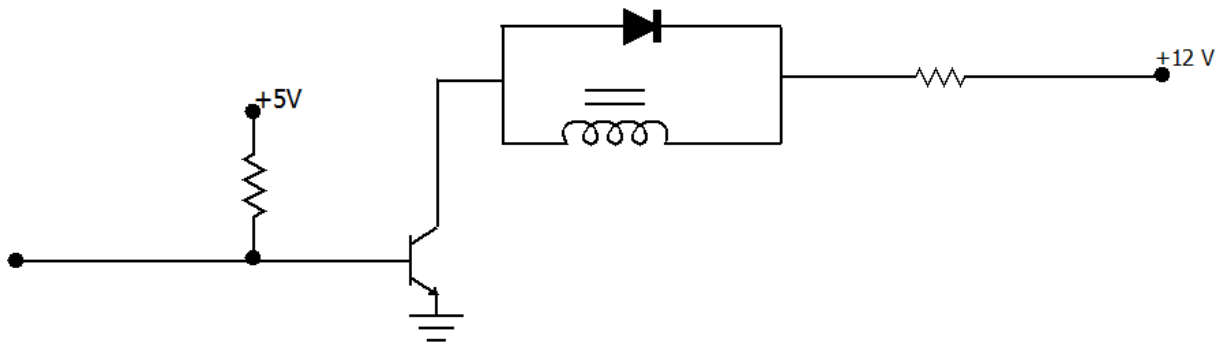
Değişken relüktanslı adım motorunun iki değişik şekli vardır. Her iki durumda da manyetik alan sadece sargı akımları tarafından üretilir.

Bir adım motorunun genel yapısı şekil-1’de gösterilmiştir. Bu motor, kendi göreceli konumunu kontrol etmek için dört ayrı sargı içerir. Sargılardan akan akımlar, rotoru konumlandırmak için manyetik alanlar oluşturur. Rotor, alan sargılarında üretilen manyetik alanlar tarafından hareket ettirilen bir sürekli mıknatıstır. Rotor bir kez son konumuna konumlandırıldığında, bu sargılardan oluşturulan manyetik alanlar, rotoru o konumda tutar. Gösterilen örnekte, rotor saat ibreleri yönünde hareket eder.

2. AKIM SÜRÜCÜSÜ

Alan akımlarını üretmek, bir akım sürücüsü gerektirir. Şekil-2’de adım motorunun dört sargısından biri için tipik bir akım sürücüsü gösterilmiştir. Sürücü, sargıdan bir akım akıtmak ve sargıyı belirli bir konumda tutmak için akım miktarını azaltma yeteneğine sahiptir. Rotoru hareket ettirmek, sabit bir konumda tutmaktan daha fazla akım gerektirir.

Dönme yönü, sargılar içinden sırayla akan akımların oluşturduğu yapıyla belirlenir. Sürekli dönme, bu uyarma dizisini tekrarlayarak üretebilir. Bu dizi, dönmenin hızını ve yönünü belirler.



Şekil – 2. Adım sargı sürücüsü. Sargıdan akım akıp akmayacağını belirler.

3. ADIM DENETİMİ

Adım motorunun denetimi, dört ayrı akım sürücüsü ve bir adım kontrol işareti gerektirir. Eğer adım aralığı 1.8 dereceye eşitse, rotoru bir devir döndürmek için 200 adım veya darbe gereklidir. Tablo-1'deki ikili bit yapısı sürücülere gönderilerek, rotorun hızı ve yönü kolayca kontrol edilebilir. Bu bit yapısı adım motorunun dört sargısından ikisinin aynı anda enerjilenmesine sebep olur. Bit yapısını gösterilen yönde hareket ettirmek, motor içindeki manyetik alanı döndürür. Bu yapı ne kadar hızlı hareket ettirilirse, motor da o kadar hızlı döner.

Saat yönü	Saat yönünün tersi
1 1 0 0	0 0 1 1
0 1 1 0	0 1 1 0
0 0 1 1	1 1 0 0
1 0 0 1	1 0 0 1

Tablo - 1. Adım motoru kontrol bit yapıları.

Adım motorunun denetimi için yapılan yazılım, hem dönmenin yönünü, hem de döndürülecek adım sayısını kontrol etmek için gereklidir. Ayrıca rotorun bulunduğu konumu hatırlama yeteneğine de sahip olmalıdır. Motora ardı ardına uygulanan kontrol bit yapıları arasında belirli bir gecikme olması gerekir. Dolayısıyla bu gecikme süresi bir minimum değere (genel olarak 1ms) sahiptir. Adım motorunun ve şekil-3'deki arayüzün denetimi amacıyla Intel 80286 için yazılım geliştirilmiştir. Şekil-4'de bu yazılımın akış diyagramı verilmiştir. Burada hız ayarı, motora gönderilen bit dizileri arasındaki “bekleme süresi” değiştirilerek yapılmaktadır.

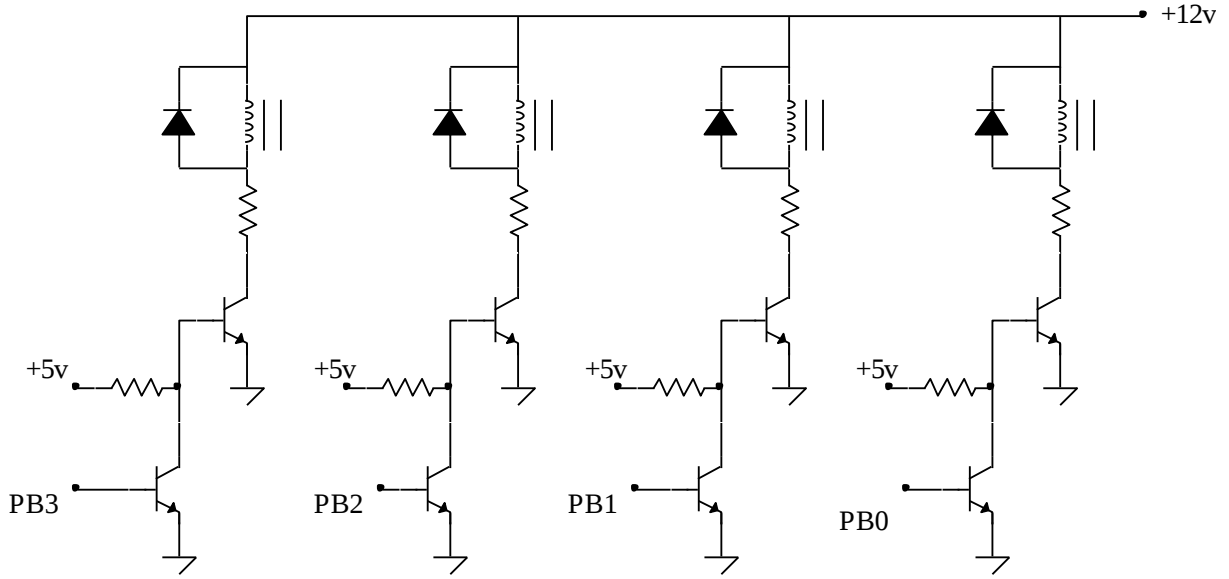
4. KONUMSAL GERİBESLEME

Adım motorunun konumu, belirli bir uygulamada kritik olabilir veya olmayabilir. Örneğin, bir yazıcıda sayfa adım motoruyla hareket ettiriliyorsa, konum önemli değildir. Sayfanın bir satır oynatılması, sadece adım motorunun her bir satır için 20 defa darbelenmesi anlamına gelir. 10 defa darbelenmek, sayfayı yarım satır hareket ettirir.

Diğer yandan, eğer adım motoru, bir yazıcıda yazma kafasını hareket ettirirse, konumsal geribesleme gerçekten önemli olabilir. Bu geribeslemeyi elde etmenin bir yolu, bir başlangıç konumu algılamaktır. Yazıcılarda başlangıç konumu normalde sayfanın sol kenarıdır. Bu, çoğu kez bir led ve bir fototransistör içeren optik bir cihazla sağlanır. Kafa başlangıç konumuna getirildiğinde, ışık ışını kırılır ve bu da donanım ve yazılımla algılanabilir.

Bu başlangıç konumu bir kez bilinirse, yazıcı yazılımı, başka herhangi bir geribesleme olmaksızın, gerekli adımlarla, herhangi bir yazma konumuna atlanabilir. Başka bir

geribesleme olmayınca hatalar oluşabileceği kabul edilir. Fakat günümüz modern adım motorlarında bu hatalar çok nadir olmaktadır.



Şekil – 3. TTL uyumlu devre ve dört sargılı adım motoru için bir arayüz.

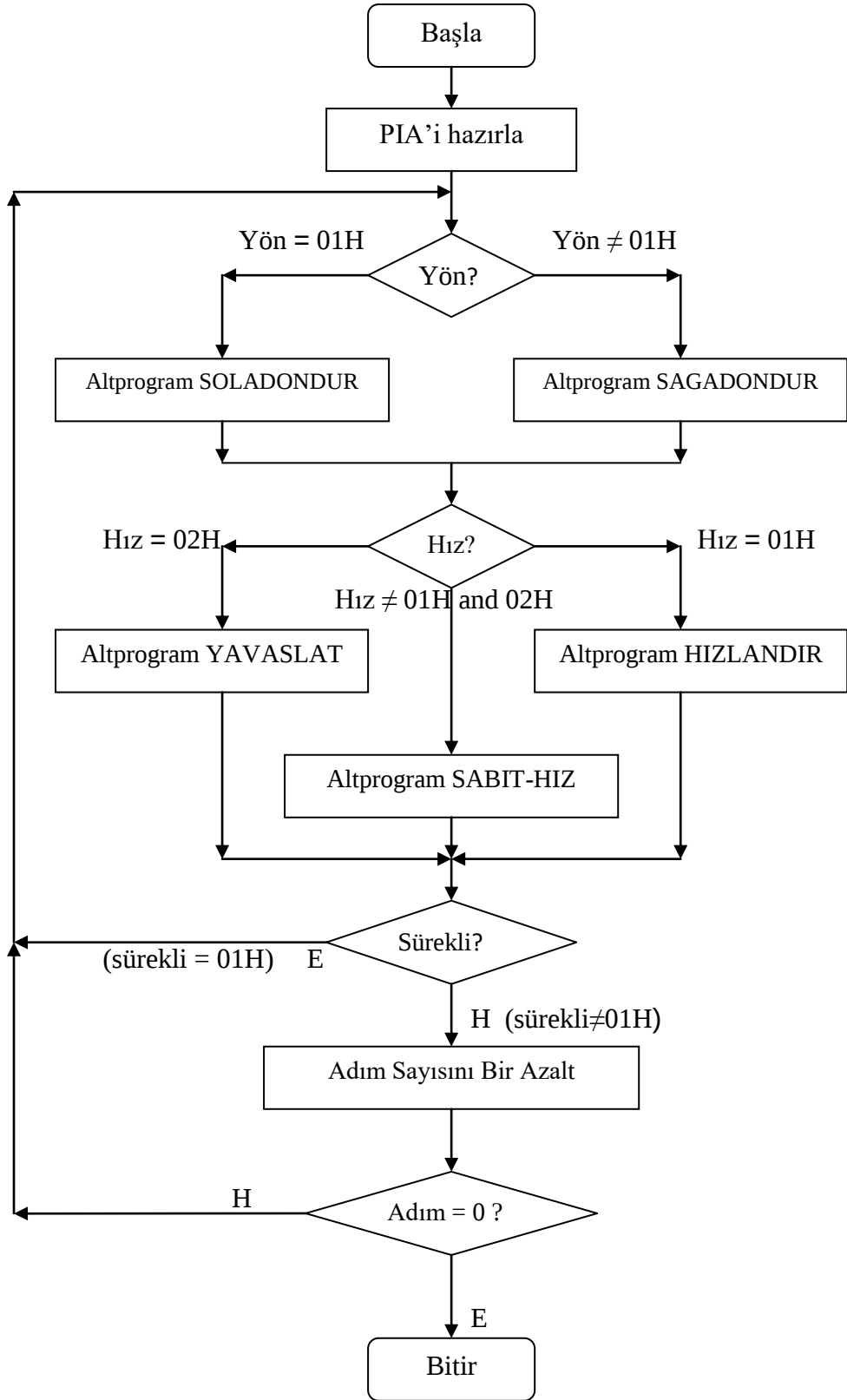
DENEYE HAZIRLIK

Kullanılacak adım motorunun 48 adımlık olduğunu ve motora uygulanacak bit dizisinin DS:0000H adresi üzerinden gönderildiğini kabul ederek:

- 1- Adım motorunu saat yönünde çeşitli hızlarda döndürebilen bir program yazınız.
- 2- Saat yönünde 90°, 240°, saatin ters yönünde 60° ve 150° döndüren bir program yazınız.
- 3- Motoru sürekli 90° sağa, 90° sola döndürecek programı yazınız.
- 4- Motoru belli bir hızla hızlandıran ve hızlanmış motoru belli bir hızla yavaşlatan bir program yazınız.

DENEYİN YAPILIŞI

Makine dilinde hazırladığınız programları deneyde sırasıyla makineye girerek çalıştırınız.



Şekil – 4. Şekil-3’deki adım motoru arayüzü için, dönme yönünü ve adım sayısını kontrol edecek altprogramın akış diyagramı.

ADIM MOTORU DENETİMİ YAPAN ÖRNEK PROGRAM:

NOT: Başlangıçta belirli değerlerin girilmesi gereken bellek yerleri:

[DS:0000H] = CC : motoru döndüren bit dizisinin ilk değeri.

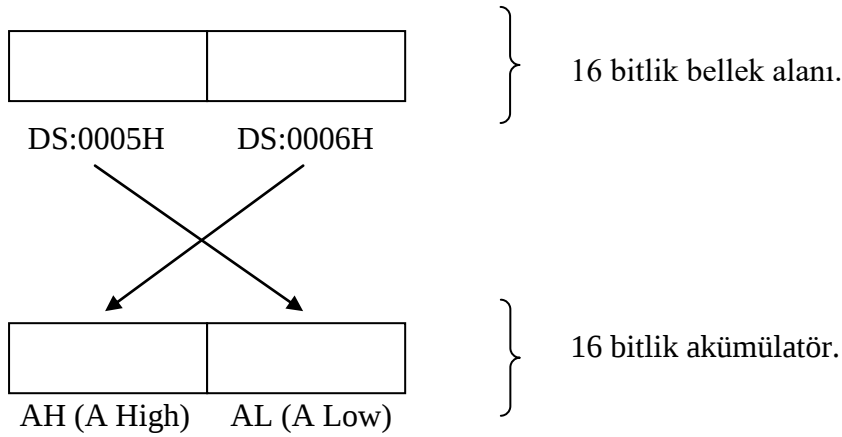
[DS:0001H] = Yön? : bellekte ki değer 01 ise saat yönünün tersine, değilse saat yönünde döner.

[DS:0002H] = Sürekli? : bellekte ki değer 01 ise sürekli, değilse adım sayısı kadar döner.

[DS:0003H] = Adım sayısı

[DS:0004H] = Hız? : bellekte ki değer 01 ise dönme hızlanır, 02 ise dönme yavaşlar, 01 ve 02 değerlerinden farklı ise sabit hızda döner.

[DS:0005H, DS:0006H] = Bu bellek alanlarındaki değerler akümülatöre atılır. Bellekten akümülatöre verinin nasıl taşındığı aşağıdaki şekilde gösterilmiştir. Bu alanlardaki değerler program tarafında girilmektedir.



Şekil – 5. Bellek alanındaki verinin akümülatöre taşınma biçimi.

Yukarıdaki şekil dikkate alındığında hız ayarı için akümülatörde tutulan bekleme süresi:

Hızlanma için : 01F4H değerinden 0032H değerine. (500 milisaniyeden 50 milisaniye).

Yavaşlama için: 01F4H değerinden 03E8H değerine. (500 milisaniyeden 1000 milisaniye).

Sabit hız için : 01F4H değeri olarak belirlenir. (500 milisaniye).

Not: Adım motorunun çalışma mantığını daha iyi anlamak için aşağıdaki videoları izleyebilirsiniz.

<https://www.youtube.com/watch?v=MHdz3c6KLrg>

<https://www.youtube.com/watch?v=t-3VnLadIbc>

ANA PROGRAM

ADRES	OPCODE		MNEMONIC
1. 0100			INCLUDE PATCALLS.INC
2. 0100			ORG 0100H
3. 0100	B0 FF	BASLA:	MOV AL, 0FFH
4. 0102	E6 88		OUT UPORT1CTL, AL
5. 0104	B0 F4		MOV AL, 0F4H
6. 0106	3E A2 05 00		MOV DS:0005H, AL
7. 010A	B0 01		MOV AL, 01H
8. 010C	3E A2 06 00		MOV DS:0006H, AL
9. 0110	3E A0 01 00	TEKRAR:	MOV AL, DS:0001H
10. 0114	3C 01		CMP AL, 01H
11. 0116	74 12		JZ SOLADON
12. 0118	3E A0 00 00	SAGADON:	MOV AL, DS:0000H
13. 011C	E6 90		OUT UPORT1, AL
14. 011E	D0 C0		ROL AL, 01H
15. 0120	3E A2 00 00		MOV DS:0000H, AL
16. 0124	E8 6A 00		CALL BEKLE
17. 0127	E9 0F 00		JMP HIZAYAR
18. 012A	3E A0 00 00	SOLADON:	MOV AL, DS:0000H
19. 012E	E6 90		OUT UPORT1,AL
20. 0130	D0 C8		ROR AL, 01H
21. 0132	3E A2 00 00		MOV DS:0000H, AL
22. 0136	E8 58 00		CALL BEKLE
23. 0139	3E A0 04 00	HIZAYAR:	MOV AL, DS:0004H
24. 013D	3C 01		CMP AL, 01H
25. 013F	74 1E		JZ HIZLAN
26. 0141	3E A0 04 00		MOV AL, DS:0004H
27. 0145	3C 02		CMP AL, 02H
28. 0147	74 03		JZ YAVASLA
29. 0149	E9 26 00	SABITHIZ:	JMP ADIMKONTROL
30. 014C	3E A1 05 00	YAVASLA:	MOV AX, DS:0005H
31. 0150	3D E8 03		CMP AX, 03E8H

32. 0153	74 1D		JZ ADIMKONTROL
33. 0155	05 0A 00		ADD AX, 000AH
34. 0158	3E A3 05 00		MOV DS:0005H, AX
35. 015C	E9 13 00		JMP ADIMKONTROL
36. 015F	3E A1 05 00	HIZLAN:	MOV AX, DS:0005H
37. 0163	3D 32 00		CMP AX, 0032H
38. 0166	74 0A		JZ ADIMKONTROL
39. 0168	2D 0A 00		SUB AX, 000AH
40. 016B	3E A3 05 00		MOV DS:0005H, AX
41. 016F	E9 00 00		JMP ADIMKONTROL
42. 0172	3E A0 02 00	ADIMKONTROL:	MOV AL, DS:0002H
43. 0176	3C 01		CMP AL, 01H
44. 0178	74 96		JZ TEKRAR
45. 017A	3E A0 03 00		MOV AL, DS:0003H
46. 017E	3C 00		CMP AL, 00H
47. 0180	74 19		JZ BITIR
48. 0182	75 00		JNZ ADIMAZALT
49. 0184	3E A0 03 00	ADIMAZALT:	MOV AL, DS:0003H
50. 0188	FE C8		DEC AL
51. 018A	3E A2 03 00		MOV DS:0003H, AL
52. 018E	E9 7F FF		JMP TEKRAR
53. 0191	B4 10	BEKLE:	MOV AH, WTNMS
54. 0193	3E 8B 1E 05		MOV BX, DS:0005H
55. 0198	CD 28		INT 028H
56. 019A	C3		RET
57. 019B	BB 00 00	BITIR:	MOV BX,0000H ;Returns to PAT
58. 019E	B4 04		MOV AH,04H
59. 01A0	CD 28		INT 028H