

29725

YILDIZ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

ÇOK KABİNLİ ASANSÖR SİSTEMİNİN PC İLE OTOMASYONU

ELEKTRONİK MÜHENDİSLİĞİ
YÜKSEK LİSANS TEZİ
MÜH. UĞUR KESEN

TEZ YÖNETİCİSİ
Prof. Dr. SEZGİN ALSAN

İSTANBUL 1993

T.C. YÜKSEKÖĞRETİM KURULU
DOKÜMANTASYON MERKEZİ

ÖNSÖZ

Asansör sistemlerinin bilgisayar ile kontrolü Türkiye'de çok yeni bir konudur. Şimdiye kadar yalnızca teorik çalışmalara konu olmuştur. Bu çalışmada asansör sistemi incelenerek, değişkenler tespit edilmiş ve gerekli matematiksel bağıntılar kullanılarak, asansör sisteminin simülasyonu yapılmıştır.

Bu çalışmada yardımlarından dolayı tez danışmanım Prof.Dr. Sezgin Alsan, Öğr.Gör. Kadir Erkan, Müh. Fikret Günar ve Y.Müh. Bahadır Bahadıroğlu'na teşekkür ederim.



İÇİNDEKİLER

Sayfa No

1. Asansör Sistemleri.....	1
1.1. Asansörlerin Sınıflandırılması.....	1
1.2. Bir Hızlı Asansörler.....	1
1.3. İki Hızlı Asansörler.....	1
1.4. Otomatik Kumandalı Asansörler.....	1
1.4.1. Bir Kumandalı Asansörler.....	2
1.4.2. Toplama Kumandalı Asansörler.....	2
1.4.3. Çift Toplama Kumandalı Asansörler.....	3
2. Bina veya Yapı Asansörleri.....	4
2.1. Asansör Trafik Dizaynı.....	4
2.1.1. Trafik Analizi.....	5
2.1.2. Trafik Analizinin Uygulama Alanları.....	6
2.2. Bir Asansörle veya Asansörler Grubu ile Muhtelif Özelliklerin Taşıma Kapasitesi Üzerine Etkisi.....	6
2.2.1. Kabinle İlgili Faktörler.....	7
2.2.2. Makine Dairesiyle İlgili Faktörler.....	7
2.3. Tesis Edilmesi Gereken Asansör Sayısının Belirlenmesi.....	8
2.4. Asansörlerin Yapı İçindeki Yeri.....	10
3. Elektrik Kumandalı Asansörlerin Elemanları...	12
3.1. Her Katta Bulunan Elemanlar.....	12
3.2. Sistemin Kabindeki Elemanları.....	13
3.3. Servis Elemanları.....	14
3.4. Sistemin Elemanları.....	14
4. Türk Standartlarında Asansör İşletim Prensipleri.....	16
4.1. Elektrikli İnsan Asansörleriyle İlgili Hususlar.....	16
4.1.1. Röle ve Kontaklarla İlgili Hususlar.....	19
4.1.2. Sıcaklık Konusuna İlişkin Hususlar.....	20

4.1.3. Kablolar Konusuna İlişkin Kurallar.....	20
4.1.4. Kablolar Konusuna İlişkin Kurallar.....	20
4.1.5. Elektrik Tesisatına Konusuna İlişkin Kurallar.....	20
4.2. Elektrikli Yük Asansörleriyle İlgili Hususlar.....	21
5.Asansör Kumandasında Son Çalışmalar.....	23
5.1. PLC ile Asansör Kumandası.....	23
5.2. Asansör Kumandası ve Malzemelerde Türk Firmalarının Yenilikleri.....	24
5.2.1. Tüp Magnetik.....	24
5.2.2. Elektronik Kat Selektörü.....	24
5.2.3. Elektronik Kumanda Panosu.....	25
5.2.4. Hazır Tesisat Kullanılarak Üretilen Asansörlerin Avantajları.....	25
5.2.5. Standart Asansör Tesisatı.....	26
5.3. Gelişmiş Asansör Kontrol Sistemleri.....	26
5.3.1. Asansör İdaresi.....	27
5.3.2. Kullanım Faydaları.....	28
5.3.3. Performans.....	29
5.3.4. Bakımın Gözetimi.....	29
5.3.5. Grup Binaların Gözetimi.....	30
5.3.6. Data Tutmak/Nakil/Belge Tutmak.....	31
5.3.7. Data Analizi/Süzme/Sunuş.....	31
5.3.8. Gelişmiş Asansör Sisteminin Sonuçları....	32
6. Bilgisayar ve Asansör Sisteminin Haberleşmesi.....	34
6.1. Bugünkü Asansör Sistemi.....	34
6.2. Bilgisayar Sistemindeki İletişim.....	35
7. Programın İncelenmesi.....	37
7.1. Programda Yararlanılan Değişkenlerin Değerlendirilmesi.....	37
7.1.1. Integer Değişkenler.....	37
7.1.2. Word Değişkenler.....	38

7.1.3. Pointer Değişkenler.....	38
7.1.4. String Değişkenler.....	39
7.1.5. Char Değişkenler.....	39
7.1.6. Type Değişkenleri.....	40
7.1.7. Dizi Değişkenleri.....	40
7.1.8. Boolean Değişkenler.....	41
7.2. Programdaki Procedure'lerin Anlatımı.....	43
7.2.1. Procedure Kişisay1.....	43
7.2.2. Procedure Kişisay2.....	43
7.2.3. Procedure Katsay1.....	43
7.2.4. Procedure Katsay2.....	44
7.2.5. Procedure Tamyuk.....	44
7.2.6. Procedure Asırıy.....	44
7.2.7. Procedure Tusoku.....	44
7.2.8. Procedure Foncız.....	44
7.2.9. Procedure Kapıcız.....	45
7.2.10. Procedure Kapıcız1.....	45
7.2.11. Procedure Kapıcız2.....	45
7.2.12. Procedure Kapıcız3.....	45
7.2.13. Procedure Kapıcız4.....	45
7.2.14. Procedure Kabıncız.....	45
7.2.15. Procedure Basla.....	46
7.2.16. Procedure Reset.....	46
7.2.17. Procedure Kapıdurum.....	49
7.2.18. Procedure Kabink_1.....	50
7.2.19. Procedure Kabink_2.....	51
7.2.20. Procedure Katdur_1.....	51
7.2.21. Procedure Katdur_2.....	51
7.2.22. Procedure Yukarı_1.....	51
7.2.23. Procedure Yukarı_2.....	52
7.2.24. Procedure Asağı_1.....	52
7.2.25. Procedure Asağı_2.....	52
7.2.26. Procedure Hareket.....	52
7.2.27. Procedure Dur_1.....	53
7.2.28. Procedure Dur_2.....	53
7.2.29. Procedure Asakont_1.....	53
7.2.30. Procedure Yukakont_1.....	55

7.2.31. Procedure Yukarıg_1.....	57
7.2.32. Procedure Yukarıg_2.....	59
7.2.33. Procedure Renkdol.....	59
7.2.34. Procedure Kapiac.....	60
7.2.35. Procedure Aksam.....	60
7.2.36. Procedure Sabah.....	61
8. Sonuç.....	62



OZET

Günümüz Türkiye'sinde asansör sistemlerinin Bilgisayar ile kontrolü yeni bir konudur. Şu ana kadar yalnızca teorik çalışmalara konu olmuştur.

Bu tez çalışmasında, kontrolü yapılan asansör sistemi, iki kabinli, 5 katlı ve toplamalı asansör sistemidir. Bu sistemde kullanılan değişkenler asansör sisteminin elemanları incelenerek ortaya çıkarılacaktır. Bu sistemin bir özelliğide binanın, iş binası, konut veya otel olması durumuna göre yapılacak sistem değişikliklerine açık olmasıdır. Bu sistem değişikliklerinin, bilgisayar ile kontrolü yapılacaktır. Herhangi bir özel çalışma durumu, bir alt procedure ile programa eklenebilecektir. Programa istenildiğinde özel durumlar, yazılım değişiklikleriyle, sistemde bir değişiklik gerekmeksizin uygulanabilecektir. Röleli devrelerde ise özel değişiklikler gerektiğinde, sistemin kumandasının tamamen değişmesi gerekecektir.

Asansör sisteminin elemanları incelenecek ve özelliklerine göre programdaki değişkenler oluşturulacaktır. Sistemin işletim prensipleri doğrultusunda programın matematiksel bağıntıları yapılacaktır. Ayrıca binanın özel çalışma prensiplerini karşılayacak alt procedure'ler yapılacaktır. Son yıllarda asansör sistemleri konusunda Türkiye'de yapılan son çalışmalar ve gelişmiş asansör kontrol sistemleri incelenecektir.

SUMMARY

The computerized lift management system is a new subject in today's Turkey. Up to this time, it was only the concern of theoretical works.

In this project work, the controlled lift system is a two car, five located and selective one. The variables used in this system is going to be set out by examining the system elements. One of the characteristics of this system is to allow the system changes due to the building used in(whether it is a hotel, a business building or an apartment). These system changes are going to be controlled by the computer. Any special working condition can be added to the programme by a sub-procedures. Some special conditions can be applied by software modification without any change, in the system. However, if any changes needed in relay networks, all management of the system must be changed.

Lift system elements are going to be examined according to their features, and the variables of the programme are going to be set. Through the operating system principles, mathematical relations of the programme are going to be done. Moreover sub-procedures for the special working principles of the building are going to be produced. The recent works on the lift systems in Turkey and improved lift management systems are going to be studied.

1.ASANSÖR SİSTEMLERİ

1.1. Asansörlerin Sınıflandırılması

Binalardaki kullanılma ve büyüklük farklılıkları, bir çok asansör çeşitinin gelişmesine yol açar. Kabinin kumanda edilmiş şekline ve kabin hızına göre asansörler sınıflandırılabilirler. Kabin hızına göre asansörler iki kısma ayrılabilirler.

1.2. Bir Hızlı Asansörler

Bu asansörlerde kabinin hareketi aynı hızdadır. Aynı hızla harekete başlar, aynı hızla yoluna devam eder ve aynı hızla durur. Bir hızlı asansörler 5-10 katlı yapılarda kullanılırlar. Genelde kullanılan asansör çeşididir.

1.3. İki Hızlı Asansörler

Binalarda kat sayısı arttıkça, ulaşım ve cevaplama zamanını kısaltmak için kabin hızını arttırmak gerekir. Hız arttıkça, kabinin kat hizasında durması güçleşir. Bu durum, asansörleri iki hızlı yapmakla giderilir. İki hızlı asansörlerde kabin düşük hızla harekete başlar, yüksek hızla hareket eder ve durmaya yakın kabin düşük hıza geçer ve düşük hızla durur.

1.4. Otomatik Kumandalı Asansörler

Otomatik kumandalı asansörlerde, kabin hareketi, bir butona basmakla sağlanır. Kabin gideceği katta bir sınır seviye anahtarının aracılığı ile durur. Bu tip kumandalı asansörler genelde yolcular tarafından kumanda edilirler. Otomatik kumandalı asansörlerde kapıyı açıp kapamak ve kabine kumanda vermek için çok nadir olarak bir kabin görevlisi (otellerde ve restaurant gibi hizmet verilen yerlerde) bulundurulur. Bu tip asansörler başlıca 3 grupta

toplanabilir.

1.4.1. Bir Kumandalı Asansörler

Bir kumandalı asansörün kabininde her kat için bir kat butonu vardır. Binanın her katında ise bir çağırma butonu bulunur. Kabin içindeki kat butonlarından birine basıldığında kabin hareket eder. Gideceği katta bir sınır anahatıyla otomatik olarak durur. Kendisine verilen bir kumandayı yerine getirinceye kadar asansör başka kumanda almaz. Bir katta kabine giren kişilerin gidecekleri katları birbirine söylemeleri gerekir. Durum aydınlanınca, kabinin bir hareket yönünde, ilk inecek kişi, gideceği kat butonuna basar. Kabin hareket eder ve butonu basılan katta durur. Butona basan kişi indikten sonra, kabinin hareket yönünde, ilk inecek ikinci kişi butona basar. Asansör bu şekilde çalışmaya devam eder. Böylece asansörü kullanan kişilerin kabin içinde gereksiz yere gidip gelmeleri önlenir. Asansör kullanılmadığında, katlarda bulunan çağırma butonlarıyla kabin istenilen kata çağırılabilir. Bir çağırma yapıldığında asansör yine başka bir kumanda almaz. Bir kumandalı asansörler genelde konut bulunan yapılarda kullanılır.

1.4.2. Toplama Kumandalı Asansörler

Toplama kumandalı asansörler bir yönde verilecek tüm kumandaları kaydederler. Sonra verilen kumandaları sırasıyla yerine getirirler. Toplama kumandalı asansörlerin kabinlerinde her kat için bir kat butonu vardır. Yapının her katında ise aşağı ve yukarı yönler için iki çağırma butonu bulunur. Bu tip asansörde kabine giren her kişi, gideceği katın butonuna basar. Kapılar kapandıktan sonra kabin hareket eder. Kabin, butonu basılan katlarda sırasıyla durarak kumandaları yerine getirir. Kabin bir yönde hareket ederken, o yöndeki çağırma butonuyla herhangi bir kattan çağırma yapılırsa, kabin o katı geçmemişse kumandayı alır. Kabin o katı geçmişse kumandayı almaz. Kabin, çağırma

yapılan kata geldiğinde durur. Çağırma yapan kişi kabine binerek asansörü kullanır. Kabin bir yönde hareket ederken diğer yöndeki çağırma butonuyla çağırma yapılırsa, asansör bu çağrılara cevap vermez.

Toplama kumandalı asansörler genellikle konut veya iş yeri olarak kullanılan çok büyük yapılarda kullanılırlar. Toplama kumandalı asansörler, daha çok kumandayı daha kısa bir zaman içerisinde yerine getirirler.

1.4.3. Çift Toplama Kumandalı Asansörler

Çift toplama kumandalı asansörler çok büyük ve çok kalabalık yapılarda kullanılırlar. Çift toplama kumandalı asansör sisteminde iki adet toplama kumandalı asansör beraber çalışır. Her toplama kumandalı asansörün kabininde her kat için bir kat butonu bulunur. Her katta ise aşağı ve yukarı yönler için birer adet çağırma butonu bulunur. Kabinlerde bulunan kat butonları yalnızca buldukları kabine kumanda ederler. Katlarda bulunan çağırma butonları ise her iki kabinde kumanda ederler. İki kabinde hareketsiz iken bir çağrı yapıldığında yakın olan kabin, ikiside aynı yakınlıkta ise öncelikli olan kabin, kumandayı alır. Çift toplama kumandalı asansörlerde ulaşım ve cevaplama zamanı çok kısadır. Çok sayıda kumanda kısa bir zaman içerisinde cevaplandırılmış olur.

2. BINA VEYA YAPI ASANSÖRLERİ

Modern tip büyük yapılarda, düşey doğrultulardaki ulaşimleri tam anlamıyla, yeterli bir şekilde gerçekleştirmek amacıyla asansör tesisleri yapılmaktadır. Bu görevin merdivenler aracılığıyla yerine getirilmesi kat sayısı arttıkça sağlanamaz.

Binalarda kullanılacak asansör sistemlerinin belirlenmesi için aşağıdaki durumların gözönünde bulundurulması gerekir.

- 1-Asansörlerin karakteristiklerinin ve sayısının tayin edilmesi,
- 2-Asansörlerin bina içindeki yerlerinin tesbiti,

Çok karmaşık olan bu probleme tam bir çözüm aranması söz konusu olmayacaktır. Fakat, araştırmanın nasıl yürütüleceği, ilgili elemanların nasıl inceleneceği kısaca açıklanacak; bu sayede, mimarlar kontruktörlerin tekliflerini değerlendirebilecek bir duruma gelecek, onlarla sıkı bir işbirliğine girmek suretiyle, en uygun çözüm yolunu, kesin olarak tespit edebileceklerdir.

Konut yapılarıyla ilişkili olarak, sıra ile aşağıdaki hususlar incelenecektir.

- 1-Sağlanması gereken trafik miktarının tahmini;
- 2-Bir asansörün muhtelif karakteristiklerinin taşıma kapasitesi üzerindeki etkisi;
- 3-Tesis edilmesi gereken asansör sayısının tayini;

2.1. Asansör Trafik Dizaynı

Bir asansör sisteminin belirlenmesi için binanın içerdiği nüfusun gereksinimlerine cevap vermesi 1920'lerden

beri arařtırmalara konu olmuřtur. Bu problemi özümleyecek yöntemler, kaba bir yaklařıklıkla hesaplanmaktaydı. 1970`-lerden itibaren ise, belirli asansör trafik dizaynlarının tetkik edilmesine olanak saęlayan digital bilgisayar simülasyon teknikleri geliřtirilmiřtir. Bu bilgisayar yazılımları, doęrulukları kanıtlanmış matematiksel yöntemlerden faydalanılmasını mümkün kılar.

2.1.1. Trafik Analizi

Trafik analizi ve hesaplaması, özellikle yoğun nüfuslu yüksek binalarda mimarların bina mimarisini daha iyi yapmalarını saęlayan asansör taşımacılıęında önemli bir konudur.

Trafik, maksimum sayıda insanı, minimum zamanda ve bina hacmini en az kullanarak asansör tesisatının kurulmasını saęlamaktır. Yolcular ne kalabalık bir kabinde olmak, ne asansör içinde beklemek, ne de uzun bir süre kabinde kalmak isterler. Bu istekleri karřılamak için optimum bir özüm bulmak trafik analizi yapılmasıyla mümkündür.

Asansör endüstrisi de bu konuda büyük ataklar yapmıřtır. Son otuz yılda, otomatik kapılar ve kontrol sistemleri konularında büyük geliřmeler kaydedilmiřtir.

Günümüz Türkiye`sinde büyük oteller, iř merkezleri ve hastaneler gibi yüksek nüfuslu binalar için trafik analizinin iyi bir şekilde incelenmiř olması büyük önem taşımaktadır. Aksi halde ileride telafisi mümkün olmayan, sıkıntılar gecikmeler her zaman görülebilir.

Asansör trafik dizaynı için bilgisayar kullanımı 1970`li yıllardan itibaren bařlamıřtır. Ancak bilgisayarların bu konuda kullanılmasından önce, bir trafik dizaynı için temel kavramların anlatılmasında yarar vardır.

Trafik dizaynı yapılırken öncelikle bina nüfusunun yaklaşık olarak belirlenmesi gerekir. Sonra kabinin bir çağrı için minimum ve maksimum gidiş ve geliş zamanlarının hesaplanması gerekir.

Bu sürelerin hesaplanmasından sonra bina nüfusuna göre yolcuların ulaşım ve bekleme zamanlarının yüzdeleri hesaplanır. Bu değerlerin TSE'nin şartnamesine uyacak şekilde asansör sayısının ve kumanda şeklinin hesaplanması gerekir.

2.1.2. Trafik Analizinin Uygulama Alanları

Asansör sistemlerinin trafik boyutlandırması, her zaman ortalamalar ve ihtimallerle ilgilidir. Özel durumlarda bina nüfusunun ortalamasını almanın uygun olmadığı ve bina nüfusunun dağınık ve farklı olacağı açıktır.

Asansör trafik dizaynı, verilen bina nüfusuna göre yapılacağından; sağlanan asansör tesisatı ihtiyacın altında kalmayacaktır. Yinede yakınmalar olabilecektir.

Özellikle oteller, hastaneler yüksek binaların rafik dizaynı karmaşıktır. Hastaneler değişken nüfuslu ve çeşitli tipte asansörlerin bulunduğu yerlerdir ve özel trafik dizaynı gerektirirler.

Bilgisayarla yapılan hesaplamalar trafiği daha iyi karşılayacak ve bazı problemleri oluşmadan önleyeceklerdir.

2.2. Bir Asansörle veya Bir Asansörler Grubu ile Muhtelif Özelliklerin Taşıma Kapasitesi Üzerindeki Etkisi

Taşıma kapasitesi, asansör kabini içinde insanlar için ayrılan alana ve mekanik özelliklere, doğrudan doğruya bağlıdır. Bir asansör veya asansörler grubunun taşıma kapasitesinin arttırılması veya azaltılması konusunda diğer bazı faktörlerin de rolü olur.

2.2.1. Kabinle ilgili Faktörler

Kabin ile ilgili faktörler üç grupta toplanabilir.

a-)Kabinin boyut oranları: Kare biçimli bulunan kabinlere oranla büyük boyutu servis yüzüne paralel olan yani derinliği daha fazla olan kabinler, daha büyük kolaylıklar sağlar.

b-)Kabin kapısının boyutları: Kabin kapısı iki insanın yanyana geçmesine imkan verecek şekilde öngörülmüş bulunursa, kabinin dolma ve boşalma süreleri sadece tek bir insanın geçmesine imkan verebilen kapılara sahip olan kabinlere oranla daha kısacaktır.

c-)Kapıların tipi ve kumanda şekli: İki kanatlı sistemlerden birisi aracılığı ile ya da otomatik kumandalı bir sistem vasıtasıyla veya her iki sistemden aynı zamanda yararlanılması suretiyle, kabin içinde bulunan kimselerin daha çabuk bir şekilde dışarı çıkması sağlanırsa, kabin içine giriş ve kabinden çıkış sırasında asansörün durma süreleri azalacaktır.

2.2.2. Makina Dairesiyle ilgili faktörler

Bu faktörler :

1-Çalışma hızı;

2-Motorun kontrol şekli;

olmak üzere iki grup altında sınıflandırılabilir

Dikkat edilmesi gereken bir durum söz konusudur. Trafik akımının yoğun olduğu saatlerde, kabin, ancak gerekli zamanın, yaklaşık bir çeyreklik süresi zarfında tam hızla hareket halinde bulunur. Zamanın geri kalan kısmı, katlardaki duruşlara, insanların giriş ve çıkışına, kapıların açılış ve kapanışına ve duruşlar sırasında gerekli olan hızlanma ve yavaşlamalara harcanır.

Bu nedenle kabin hızının taşıma kapasitesi üzerindeki etkisi ilk bakışta sanıldığı kadar büyük olmaz. Tesisatın maliyeti hıza oranla daha süratle artar. Gerekli elektriksel güç hızla orantılıdır. Motorun kontrol şekli çalışma hızına uygun olmalıdır. Bir parça yüksek hız değerlerine erişilir erişilmez, bu hızın tamamından yararlanmak istenirse, değişken gerilimli sistemlerin uygulanması gerekir.

2.3. Tesis edilmesi gereken asansör sayısının belirlenmesi

Çeşitli yüksekliklere sahip olan ve içlerinde çeşitli sayıda insan barındıran yapılardaki ulaşımın en iyi şekilde sağlanması için gerekli asansör tipi ve sayısının tayini konusu ile ilgili bu kural ve bilgilerin esasları kısaca şöyle özetlenebilir.

Bu bilgilere göre, gerçekleştirilecek tesisatın tayininde;

- 1- En fazla 674 sınırına kadar erişen insan sayısı;
- 2- Üst sınır 25 olmak şartıyla, kat sayısı;
- 3- Muhtemel bekleme zamanı 90 veya 130 saniyeden kısa olmak şartıyla hizmetin kalitesi gibi hususlar dikkate alınmakta, hız ve faydalı yük karakteristikleri aşağıda belirtilen sınırlar içinde bulunan asansörlerden yararlanılmaktadır.

Bu sınırlar :

Hız için.....: 0,63 m/sn ile 1,50 m/sn;

Faydalı yük için...:300kg (4 kişi) ile 675kg (9 kişi) şeklinde belirlenmiştir.

1-Minimal veya asgari servis şartlarıyla ilgili kurallar:

a-) Asansör tesisatı, çıkış sırasında, beş dakika içerisinde, bina sakinlerinin %7,5 oranındaki bir kısmını taşıyabilecek bir yeteneğe sahip olmalıdır.

b-) Muhtemel maksimal bekleme zamanı en fazla 130 saniye

kadar olmalıdır.

c-) Asansör tesisatının hizmetinden yararlanan iki seviye arasındaki düşey mesafe üç kattan fazla olmamalı; yolcunun, merdiven aracılığı ile katedeceği düşey uzaklık ise, en fazla iki kat yüksekliğini geçmemelidir.

2- Standarta bağlı olarak bina tiplerine göre, binada oturan insan sayısı belirlenir. Bu insan sayısı ve kat sayısı göz önünde bulundurularak, kabin büyüklüğü, hızı ve diğer özellikleri belirlenir.

Büro olarak kullanılan yapılar:

Yapının tek bir işletme veya firma tarafından işgal edilmesi, yada yapıya ait bölümlerde, şirketler, ticarethaneler, iş büroları v.s... gibi birbirlerine yabancı çeşitli teşekküller bulunması durumuna göre, çözümlenmesi gereken problem çok değişik şekiller alacaktır.

Birinci halde, personelin iş saatleri dağılımı, geliş ve gidiş saatleri ve mevcudu belirli olduğu için, problemin bütün elemanları tayin edilebilir. Bundan dolayı, sorunun tesadüflere bağlı olan bir tarafı pek yoktur. Ama hemen sezinlenebileceği gibi, personelin hemen hemen tamamında, çalışma saatlerinin dağılımı, işe geliş ve işten dönüş saatleri aynı olursa, pek yoğun bir maksimal bir trafik akımı meydana gelecek, bu durum, pahalıya çıkan bir çözüm yolunun benimsenmesini gerektirecektir.

Faaliyetlerin çeşitliliği nedeniyle, ikinci halde durum böyle değildir. Ama yine aynı nedenle, problemin verilerinin ortaya konulması zorlaşır. Bu gibi hallerde, aşağıdaki şekilde hareket edilmesi uygun olacaktır:

Herşeyden önce, yapıdaki insan sayısı hesaplanacak; bu hesapta, bir ilk yaklaşıklıkla, 8m² lik bir alana 1 insanın tekabül ettiği kabul olunacaktır. Bu mevcudun bulunması için, zemin katın üzerinde yer alan bütün katlarda, işgal

edilen toplam alanın 8'e bölünmesi yeterlidir. Bu kadar sayıdaki insanın 30 ile 60 dakika arasında değişen zaman süresinde taşınmasına imkan veren bir çözüm yolunun uygun olduğu düşünülecektir. Amerika Birleşik Devletlerinde insanlar daha fazla talepkar görünmekte, bir asansör tesisatının birinci sınıf sayılması için, söz konusu taşınma işleminin, 30 ile 45 dakika arasında değişen bir zaman süresinde yapılması şart koşulmaktadır. Öte yandan, tesisatın uygun telakki edilmesi için, bekleme zamanlarının 60 saniyeyi aşmaması, çoğu halde 45 saniyenin altında kalması gerekir. Amerika Birleşik Devletlerinde, bu sürenin 20 ve hatta 15 saniyeye kadar indirildiği sık sık görülmektedir.

2.4. Asansörlerin Yapı İçindeki Yeri

İlk olarak, bir asansör tesisatını kendisine ait bir kuyu, bir boşluk içinde öngörülmesinin şart olduğunu, ancak, başka bir yeri bulunmayan eski binalarda, merdiven boşluğundan istifade edilebileceğini hatırlatılması uygun olacaktır.

Genellikle asansörlerin merdivenlere yakın olarak tesis edilmesinde ve eğer mümkünse, merdiven sahanlıklarından ayrı sahanlıklar öngörülmesinde yarar vardır. Böyle bir çözüm yolu benimsenirse, asansör bekleyen yolcuların, merdivende yürüyenlerin hareketini engelleme tehlikesi ortadan kalkar. Bazı saatlerde, asansör veya merdivendeki trafik akımının yoğun olması durumunda, bu yararlılık daha da artar. Bu nedenle, sahanlıkların ortak olması gibi bir zorunluluk söz konusu ise, sahanlıkların boyutları, hiç değilse, biraz önce anlatılan sakıncayı mümkün mertebe azaltacak ölçüde geniş bulunmalıdır.

Asansörlerin yapı planı içerisinde dağıtılmasının değil bir araya getirilmesinin yararı üzerindedir. Örneğin, büyük bir yapının giriş holünde, öngörülen dört asansörden oluşan bir grup, holün her iki tarafında

öngörülen ve aynı karakteristiklere sahip olan, ikişer asansörden oluşan, iki gruptan çok daha büyük bir verime sahiptir.

Yapı planı içinde minimal seviyede bir alanın işgal edilebilmesi için, yine asansörlerin bir araya toplanması gerekir. Böyle yapılırsa asansörlerin gözlenmesi ve bakımı kolaylaşır ve daha ucuza mal olur.



3. ELEKTRİK KUMANDALI ASANSÖRLERİN ELEMANLARI

3.1. Her Katta Bulunan Elemanlar

Yukarı Çağır Butonları: Her katta, yukarı çıkmak isteyen yolcuların isteğini belirtmek için kullandıkları butonlardır. En üst katta yukarı çıkma isteği olmayacağından yukarı çağır butonları sayısı, kat sayısından bir eksiktir. $Y_{\text{çbs}}=k_s-1$

Aşağı Çağır Butonları: Her katta, aşağı inmek isteyen yolcuların isteğini belirtmek için kullandıkları butonlardır. En alt katta aşağı inme isteği olmayacağından aşağı çağır butonları sayısı, kat sayısından bir eksiktir. $A_{\text{çbs}}=k_s-1$

Kapı Kilidi Bilgisi: Her kat kapısının açılıp kapanmasını sağlayan kilit sistemidir. Sistemin hareketine etki eden bir tahrik elemanıdır. Aynı zamanda sistemin tahrik ettiği bir elemandır. Kabin hangi kata geldiyse, o kata ait kabin çağırma veya gönderme bilgisini kontrol eder. Eğer bu bilgi varsa, sistem kapı aç bilgisini gönderir. Kabin aşağı veya yukarı hareket etmeden önce o kata ait kapının kapalı olması gereklidir. Böylece sistemin hareketine de etkisi olur. Kapı kilitlerinin en önemli özelliği, çift emniyetli olmasıdır. Fişkontak devresiyle ikinci emniyet sağlanır.

Kat Seviye Bilgisi: Kabinin hangi katta olduğunu bildiren kat seviye bilgisi her katta bulunur. Kat seviye bilgisi bir çok şekilde elde edilir. Bunlardan birincisi magnetik'tir. Magnetik bir alan oluşturulur. Kabin üzerinde bulunan bir kesici (bayrak) ile bu magnetik alan tam kat seviyesinde kesilir ve kat seviye bilgisi elde edilir. Bir röle yardımıyla her kat seviyesi, sisteme kat bilgisini gönderir. İkincisi ise kopilverk'tir. Kopilverk dönen bir disk üzerinde bulunan kontak ile kat seviye bilgisi oluşturulur. Bir diğer kat seviye bilgisi elde etmede, Gray kodda

bilginin gönderilmesidir. Burada Tüp magnetikler kullanılır. Bu konu ileride anlatılacaktır.

3.2. Sistemin Kabindeki Elemanları

Kat Butonları: Kabindeki yolcunun çıkmak istediği kat bilgisini bildirdiği butondur. Toplamalı (Sellektif) asansör sistemlerinde, kabinin hangi katlara uğrayacağını, yolcunun bilmesi için, basılan kat butonlarının ışıklı bir sistemle yanık kalması istenir.

Alarm Butonu: Sistemle hiç bir bağlantısı yoktur. Yalnızca bir zil sesi ile kabin dışına sesli ikaz verilmesini sağlayan butondur.

Kabin Dur Butonu: Kabin hareket halinde iken, herhangi bir seviyede, istenildiğinde kabinin durmasını sağlayan butondur.

Aşırı Yük Bilgisi: Kabinin maksimum taşıyabileceği yük miktarı belirlenir. Bu taşıyacağı yük miktarı, asansörün mekanik özelliklerine göre değişir. Taşıyacağı yük miktarına göre, kabin altında bulunan kontak konum değiştirir. Bu kontak, yaylı bir sistemle yapılmıştır. Eğer taşıma kapasitesini aşarsa, yaylı sistemin sıkışmasıyla, aşırı yük anahtarı, aşırı yük bilgisini oluşturur. Kabine fazla binen yolcu olduğunda, sesli veya ışık sinyaliyle uyarı yapılır.

Tam Yük Bilgisi: Tam yük bilgisi, aşırı yük bilgisi olduğunda tam yük bilgisi set olur. Kabin kat bilgisiyile yeni bir kata gelinceye kadar tam yük rölesi set olarak kalır.

Regülatör Şalter: Regülatör şalter, Kabini taşıyan halat sistemi tarafından çalışır. Kabin belirli bir hızın üzerine çıktığında fren sistemini kademeli olarak çalıştırır.

Paraşüt Kontaklı: Regülatör şalteri mekanizmasıyla çalışır. Regülatör şalterin etkisiz kaldığı konumda çalışır. Bu konum halat kopması durumunda geçerlidir.

3.3. Servis Elemanları

Servis Kumanda Butonu: Sistemin kumandasının, servis kumanda kutusuyla kontrolunu sağlar. Kat ve kabin butonlarını devre dışı bırakır.

Servis Kumanda Aşağı Butonu: Kabinin sistemden ayrı olarak aşağı doğru hareketini sağlar.

Servis Kumanda Yukarı Butonu: Kabinin sistemden ayrı olarak yukarı doğru hareketini sağlar.

3.4. Sistemin Elemanları

Kabin Aydınlatma: Kabinin meşgul olduğu süre boyunca kabin aydınlatmasının yapılmasıdır. Bu işlem sistem tarafından yapılır.

Katta Sinyali: Kabin hangi katta bulunuyorsa o katta bulunan katta lambasının yanması için sistemin gönderdiği bilgidir.

Aşağı İniyor Lambaları: Kabin aşağı hareket halinde iken tüm katlarda bulunan aşağı ok lambasının yanması için sistemin gönderdiği bilgidir.

Yukarı Çıkıyor Lambaları: Kabin yukarı hareket halinde iken tüm katlarda bulunan yukarı ok lambasının yanması için sistemin gönderdiği bilgidir.

Kat Göstergesi : Katlarda ve kabinde bulunan digital veya lambalı, kabinin kat bilgisini göstermek için sistemin gönderdiği bilgidir.

Çift Hız Sinyali: Kabinin katlar arasında daha hızlı hareket etmesini sağlamak için, yani ulaşım süresinin kısa zamanda olması için, asansör motorunun daha hızlı devirde döndürülmesidir. Bu sinyal yalnızca çift hızlı asansör sistemlerinde kullanılır.

Tek Hız Sinyali: Kabinin harekete başlarken veya hareketini tamamlarken, yolcunun hareketi daha az hissetmesini sağlamak için kullanılan motor hızıdır. Bu bilgi sistem tarafından gönderilir.

Aşağı Hareket Sinyali: Kabinin aşağı doğru hareket edeceğini bildiren bilgidir. Bu bilgi sistem tarafından gönderilir.

Yukarı Hareket Sinyali: Kabinin yukarı doğru hareket edeceğini bildiren bilgidir. Bu bilgi sistem tarafından gönderilir.

Zaman Gecikme Bilgisi: Sistemin katlardaki durma zamanı süresini tayin eden, zaman rölesinin, sisteme etkisidir.

4. TÜRK STANDARTLARINDA ASANSÖR İŞLETİM PRENSİPLERİ

4.1. Elektrikli İnsan Asansörleriyle İlgili Hususlar

Elektrikli insan asansörleriyle ilgili TS 863 sayılı Türk Standartlarında, insan asansörlerinde uygulanan manevra sistemlerine ilişkin olarak aşağıdaki tarifler verilmektedir.

-Çağırma ve gönderme düzenleri, asansörün, düğmenin basıldığı kata çağrılmasına veya ana durak da denilen zemin katına gönderilmesine yarayan sistemlerdir.

-Durdurma veya tehlike düğmesi, kabin içinde basılınca kabinin durmasını sağlayan düzenin anahtarıdır.

-Bekletme kat düzeni, kabini hareketini bitirdikten sonra, önceden tespit edilen kata otomatik olarak gönderen elektrik kesicisi veya kesiciler grubudur.

-Kat seçicisi, kabini, hareketi esnasında, önceden seçilmiş bulunan katta otomatik olarak durduran düzendir.

-Kumanda cihazı, ana kumanda elemanlarını içinde top-
layan araç veya araç topluluğudur.

-Kumanda düğmesi, kabinin içinde bulunan ve onun iste-
nilen kata çıkma veya inmesine kumanda eden elektrik kontak
elemanıdır.

-Kumanda elemanları, kabini istenilen yönde harekete
geçiren, hız veren ve durduran elemanlardır.

-Basit kumanda, kumanda düğmesine basarak kabini hare-
kete geçiren ve yalnız istenilen katta durduran düzendir.

-Dizi kumanda, Katlarda veya kabin içindeki düğmelere

basamak suretiyle kabini çıkıp inerken, önceden seçilen katlarda sırasıyla durdurmayı sağlayan otomatik kapı düzenidir.

-Karma kumanda, katlarda aşağı veya yukarı hareket düğmeleriyle yön seçebilen ve kabini hareket yönünde çağrılan katlarda durduran dizi kumanda sistemi olup her iki yöndeki çağrılar için kabini, ilk yöndeki hareketini tamamladıktan sonra diğer yönde harekete geçiren otomatik düzendir.

-Çift kumanda, kabinin hareketi asansörcü veya yolcu tarafından seçilen ve gerektiğinde binisler için kabini katta durdurmayan karma kumandalı otomatik düzendir.

-Seçici kumanda, birden çok asansörün bir arada bulunduğu yerlerde kat düğmelerinden birine basıldığı takdirde istenen hareket yönünde en yakın olan kabinin çağrılan katta durmasını sağlayan ve diğer kabinleri bu çağrı için serbest bırakan düzendir.

-Kumanda kablosu, kabin ile kumanda kutusu arasında elektriksel bağlantıyı sağlayan esnek bir kablodur.

-Röle, elektrik akımı devresini veya devrelerini otomatik olarak açıp kapayan düzendir.

-Yön göstergesi, kat ve kabinde, asansörün hangi yönde hareket ettiğini gösteren ışıklı düzendir.

İlgili Türk Standardı, kumanda şekillerini,

1-Basit kumanda;

2-Dizi kumanda;

3-Karma kumanda;

4-Çift kumanda;

5-Seçici kumanda;

olmak üzere beş grup altında sınıflandırmakta ve aşağıdaki

genel kurallara uyulmasını şart koşmaktadır.

1-Kumanda türlerine göre, kabin veya katta, durdurma, kumanda, çağırma-gönderme ve emniyet düzeninin düğmeleri ile yön göstergesi bulunmalıdır.

2-Kumanda düzenleri ve bağlandıkları yerler, yanmaz ve tutuşmaz yapılmalıdır.

3-Kumandaların içindeki dirençler havalandırılır durumda olmalıdır.

4-Kumandalara ait elektrik devreleri, şebekeden ayrı olarak sigorta ve benzeri malzeme ile ayrıca korunmuş durumda olmalıdır.

5-Kabin durduktan sonra, kumanda düzeni en az 3 saniye yeni bir kumandayı kabul etmemeli ve dış kumanda düzenine öncelik tanınmalıdır.

6-Asansörde, asansörü ilk ve son katta durdurmak için devrenin açılması doğrudan doğruya bir yayın çalışmasına veya başka bir elektrik devresinin tamamlanmasına bağlı olmalıdır.

7-Alt ve üst katlarda veya tehlike anında imdat anahtarının veya diğer emniyet düzenin çalıştırılması halinde asansörü durdurmak için makina freninin uygulanması ve asansör motorunu besleyen güç kaynağının kesilmesi maksadıyla kullanılan kumanda sistemi herhangi bir elektrik devresinin tamamlanmasına bağlı olmamalıdır. Bu husus, dinamik frenlemeye veya hız kontroluna uygulanmaz.

8-Üç fazlı alternatif akım kaynağına bağlı asansörün kumanda düzeni, faz değişmesi veya bir fazın kesilmesi halinde, motoru devreden çıkarmalıdır.

9-Kumanda devreleri, bir topraklama hatası, açık devre

veyahut bir kondansatörün çalışmaması veya deşarjı sebebiyle emniyetsizlik yaratmayacak şekilde düzenlenmelidir.

10-Hareket yönünü deęiştirme kontakları, mekanik ve elektrikli olarak aralarında irtibatlandırılmış olmalıdır.

11-Gereken yerlerde rölelerin ve kontakların uygun bir sıra ile çalışmaları için elektrikli kilitleme sağlanmalıdır.

12-Kat seçici, şerit, tel v.s.... aracılığı ile harekete geçiriliyorsa, bunların kopması veya gevşemesi halinde asansör makinasını durdurmak için bir anahtar bulundurulmalıdır.

13-Normal şehir elektrik şebekesinde çalışan kumanda devreleri, faz ve nötr arasında bağlanmalıdır. Doğru akım sistemlerinde pozitif (+), 3 telli doğru akımlarda dış iletkenler ile negatif (-) kutup noktaları arasına bağlanır.

14-Kumanda devrelerinin düşük gerilimle beslenmesi halinde akım iki sargılı düşürücü transformatörden geçmeli veya bir redresör kullanıldığı hallerde, devrenin bir kutbu topraklanmalı ve diğer kutbu üzerine ölçme amacı için yapılacak düzen dışında tek kutuplu anahtar, sigorta v.s. gibi cihazlar konulmalıdır.

4.1.1.Röle ve Kontaklarla ilgili hususlar

1-Hareketli ve sabit kontaklar kolayca deęiştirilebilir durumda olmalıdır.

2-Asansörü durdurmak için devreyi açan kontakta başka bu kontakın arızalanması halinde çalışacak ikinci ve en az bağımsız bir diğer kontak bulunmalıdır.

4.1.2. Sıcaklık Konusuna İlişkin Hususlar

1-Asansör, en az iki saat süreyle tam yükte çalıştırıldığında kontaklarda ve kontakların temas ettiği malzemelerde sıcaklık etkisiyle herhangi bir arıza meydana gelmemelidir.

2-Yalıtkan madde özellikleri akım transformatörleriyle ilgili TS 620 standardına uygun olmalıdır.

4.1.3. Kablolar Konusuna İlişkin Kurallar

Bütün kumanda kabloları, TS 261'e uygun olmalıdır. Kablo telleri birbirine lehimlenmek ve klemenslerle bağlanmak suretiyle birleştirilmelidir. Birleşme yerlerinde tellerin dayanıklılığı bakımından bir zayıflama, esnekliği yönünden de bir aksaklık meydana gelmemelidir.

Kumandalara giriş ve çıkış kablolarının işaretlenmesi tavsiye olunur.

4.1.4. Bekletme Kat Düzeni ve Kat Seçicisi Konusuna İlişkin Kurallar

Bunlar önceden tespit olunan esaslara uygun bulunmalıdır.

4.1.5. Elektrik Tesisatı Konusuna İlişkin Kurallar

1-Elektrik tesisatında kullanılan malzemeler, asansör tesislerinde uygulanan malzemelerin özellikleri TSE'ye uygun olmalıdır.

2-Her asansörün makina devresinde, bu devreyi elektrik şebekesinden ayıran sigortalı bir ana kesici bulunmalıdır. Birden çok asansörün ana kesicisinin bir arada bulunması halinde, her kesicinin hangi asansöre ait olduğu kesicinin

üzerine belirtilmelidir.

3-Ana kesicinin açılması yani akımın kesilmesi halinde, asansörün frenleri harekete geçmelidir.

4-Kabin kablo kutusuna gelen kablolar, nereye ait oldukları belirtilecek şekilde işaretlenmelidir.

5-Kontaklar dışındaki metal kısımların tümü topraklanmış olmalıdır.

6-Kumanda devrelerin kabloları, diğer devrelere ait kablolardan boru v.s... yardımıyla ayrılmalıdır.

4.2. Elektrikli Yük Asansörleriyle ilgili Hususlar

Elektrik yük asansörleriyle ilgili TS 1108 sayılı Türk Standardında, yük asansörlerinde uygulanan manevra sistemlerine ilişkin olarak aşağıdaki tarifler verilmektedir.

-Basmalı kumanda, asansör kabınınin hareketini, kumanda düğmesine sürekli olarak basmak suretiyle sağlayan ve düğme serbest bırakılınca kabının hareketini durduran düzendir.

İlgili Türk Standardı, Kumanda şekillerini, hafif ve ağır yük asansörleri için,

1-Basit kumanda,

2-Dizi kumanda,

3-Çift kumanda,

olarak üç grup altında sınıflandırmakta;

Servis asansörleri için;

1-Basit kumanda,

2-Basınçlı kumanda,

öngörülmelidir.

İnsan Asansörleriyle ilgili TS 863 numaralı Türk Stan-

dardına göre, asansör tesislerinin konstrüksyonu ile ilgili genel kurallar:

1-Asansörler, genel olarak, kullanılma sırasında, içinde ve çevresindeki insanlara zarar vermeyecek ve tehli- ke yaratmayacak şekilde emniyetli yapılmış olmalıdır.

2-Asansörde bir kilit düzeni bulunmalıdır.

3-Asansör kısımlarının bakımları, temizlenmeleri ve parçalarının değiştirilmeleri kolaylıkla yapılabilmelidir.

4-Asansörlerin metal parçaları, dış etkilere karşı, yerine göre korozyona dayanıklı malzemedен yapılmış veya kaplanmış veyahut boyanmış olmalıdır.

5-Üzerinde toz ve pislik toplanabilecek yalıtkan par- çalar, bu gibi birikintilerin kimyasal etkileriyle yalıt- kanlığı bozulmayacak şekilde yapılmış bulunmalıdır.

6-Asansör, projede anılan anma gerilimi değerinden %10 kadar değişim olan gerilimlerde çalışabilmelidir.

7-Asansörlerin çeşitli parçalarının imalatıda Türk standartlarına uygun olmalıdır. Bunların dışında kullanıl- cak malzemeler de asansörün güvenlikle çalışmasını bozmaya- cak nitelikte olmalıdır.

8-En büyük yükün tayininde, insan ağırlığı 75kgf alınmalıdır.

9-Asansör parçalarının hesaplanmasında, emniyet kat- sayısı 5`ten aşağı seçilmemelidir.

5. ASANSÖR KUMANDASINDA SON ÇALIŞMALAR

5.1. PLC ile Asansör Kumandası

Marmara Üniversitesi'nde yapılan tez çalışmasında PLC'ler tanıtılmış ve bunların uygulamasına yönelik olarak asansörlerin kontrol sistemleri PLC'ye uygulanmıştır. PLC merkezi işlem ünitesinde Z80 mikroişlemci ünitesi bulunur. PLC'ler üretimin yapıldığı tozlu, kirli ve elektrikli gürültü gibi ağır şartlarda çalışacak şekilde dizayn edilmişlerdir. Bununla birlikte farklı bir programlama dili, arıza bulma ve bakım kolaylıklarının olması gibi özellikleri ile bilgisayarlardan farklıdır. Bilgisayarların arıza ve bakım servisi ile programlama dillerinin öğrenilmesi için özel bir eğitime gerek vardır. PLC programlama dili klasik kumanda devrelerine uygunluk sağlayabilecek şekildedir. Programlama klasik kumanda bilen birisi tarafından kolayca yapılabilir.

PLC ile yapılan asansör kumandasında, yalnız bir asansör ve üç kat için çalışmada bulunulmuştur. Aynı asansör için üç programlama yapılmıştır. Birincisinde şu an ticarete kullanılan röleli devre aynı şekilde PLC programına aktarılmıştır. İkincisinde ise lojik kapılar kullanılarak kumanda yapılmıştır. Üçüncüsünde ise kendi tasarladıkları programı düzenlemişlerdir. Dikkat edilirse bu çalışmada kat sayısı azdır ve aynı zamanda asansörün çalışma konumları tek tek kontrol halindedir. Kat sayısının artması programın daha çok karmaşık hale gelmesini gerektirecektir. Eğer bir de grup asansörleri konularına yönelince PLC programlaması çok zor ve karmaşık olacaktır.

Ayrıca asansörün kullanıldığı binaya göre, yerine getirilmesi istenilen özel isteklerin sağlanması çok zor olacaktır. Örneğin iş bürolarının bulunduğu bina, oteller, güvenlik kontrolü ile girilen binalar...

Ayrıca PC ile yapılan asansör kumandasında, asansörün optimize hesaplarını kolayca yapması ve cevaplama süresini minimuma indirerek, performansı arttırması gibi kolaylıkları vardır.

5.2. Asansör Kumandası ve Malzemelerde Türk Firmalarının Yenilikleri

5.2.1. Tüp Magnetik

Elektronik Kumanda Panolarına Uygunluk
Tozlu, Nemli ve Gazlı Her Ortamda Güvenilir
Dış Darbelerden ve Titreşimden Etkilenmeme
Çevredeki Metal Aksamdan Etkilenmeme
Kaliteli İthal Kontaktlı
Uzun Algılama Mesafesi (40mm %10)
Mıknatıs Kullanımında Tasarruf
İki Farklı Montaj Kullanım Şekli
Uzun Ömür
1 Yıl Garanti ve Firma Güvencesi

Farklı Montaj şekilleri

- 1- Mıknatıs Kutbu ve hareket yönüne göre çalışma
- 2- Mıknatıs yönüne göre çalışma

5.2.2. Elektronik Kat Selektörü

Teknik Özellikleri:

- Her çeşit kumanda panosuna tam uygunluk
- Yüksek güvenilirlik ve performans
- Elektriksel gürültülerden etkilenmeme
- Voltaj değişimlerinden etkilenmeme
- Elektrik kesilmelerinde kat konumunu koruma
- Montaj kolaylığı
- Uzun ömür

5.2.3. Elektronik Kumanda Panosu

Teknik Özellikleri:

- Kat selektörü : Tüp Manyetikli ve Gray kod çözümlemeli
- Kumanda gerilim : 48 VDC
- Sinyal Gerilimi : 12 VDC
- Pompa ve Fren Gerilimi : 180 VDC
- Kat Gösterge Çıkışları : Normal katta ve digital

Avantajları:

- TS 863 şartlarına uygunluk
- Montaj malzemesi ve zaman tasarrufu
- Montaj ve ayar kolaylığı
- Yüksek güvenilirlik ve performans
- Elektriksel gürültüden etkilenmeme
- Voltaj değişimlerinden etkilenmeme
- Elektrik kesilmelerinde kat konumunu koruma

Üretici Firma Garantileri:

- Ücretsiz teknik destek
- Ücretsiz teknik personel eğitimi
- 10 yıl yedek malzeme temin garantisi
- Garanti dahilinde ve dışında arızalı kartların 24 saat içinde onarımı veya değiştirme garantisi
- 1 yıl garanti ve Firma Güvencesi

5.2.4. Hazır Tesisat Kullanılarak Üretilen Asansörlerin Avantajları

1-NYAF Kablo kullanılması nedeniyle kablo kırılmaları ortadan kaldırılmıştır.

2-İthal bağlantı elemanları (soketler) kullanılarak yapılan

otomatik bağlantılar ve fabrikada yapılan kaset bağlantılarında kablo uçlarının lehimlenmesi nedeniyle zamanla meydana gelebilecek oksitlenme ve temassızlık problemleri ortadan kaldırılmıştır.

3-Fişkontak ve kilit arızalarını ihbar eden sisteminin sayesinde asansörün bu elemanların arızalarından dolayı uzun süre servis dışında kalması önlenmiştir.

4-İllerdeki yıllarda gerek görüldüğü hallerde yeni kumanda sistemlerine (Örnek; tekdüğme toplamalı, çift düğme toplamalı) adapte edilmesi veya ilaveler yapılması son derece kolay ve ekonomiktir.

5.2.5 Standart Asansör Tesisatı

Avantajları :

- Standartlaşma getirdiği için kaliteyi yükseltir.
- Kalifiye eleman sıkıntısını azaltır.
- Zamandan ve işçilikten büyük tasarruf sağlar.
- Montaj ve sonrasında tesisat sorunlarını ortadan kaldırır.

5.3. Gelişmiş Asansör Kontrol Sistemleri

Asansör sistemlerinden otomatik olarak data elde etmek iki işlemi gerektirir. Bunlardan birincisi uzak bir asansör tesisatından merkezi kumanda merkezine datanın iletilmesidir. İkinci işlem ise datadan elde edilen kumanda bilgisinin tanımını içerir.

Bu bölümde ikinci işlem ele alınacak ve kumanda sisteminin pratikte gerçekleştirilmesi üzerine fikirler ortaya konulacaktır.

Avrupa'da düşen fiyatlar sayesinde, küçük bilgisayar-

lar sistemlerinin bilgisinde artış, pek çok asansörünün idaresinde bilgisayar kullanılması yaygınlaştırılmıştır. Bu bölümde asansör idaresi ve onun faydaları açıklanacaktır. Performans analizi, bakım, tamir denetimi, birleşmiş bina idaresi gibi pratik kondirasyonları sırası geldiğinde özetlenecektir.

5.3.1. Asansör idaresi

Bir veya daha fazla binanın içindeki asansör gruplarının çalıştırılması bir fonksiyonlar hiyerarşisi olarak düşünülebilir. Bu hiyerarşinin en altında en temel fonksiyon vardır. Katlar arasında yolcuların çağrısına göre gidip gelecek bir tek asansörü hareket ettirmek. Bundan sonraki basamakta grup kontrolü vardır. Farklı katlardaki yolcu çağrılarını, daha önceden planlanmış (algoritma olarak bilinen) kurallara göre, bir grup asansör arasında paylaştırılır.

Asansör idaresi uzun vadeli bir işlemdir. Bununla, bir kaç grup içindeki bütün asansörler optimize edilirler ve en yüksek performansta işletilir. Bu yüzden, asansör kumandası hiyerarşinin en üst basamağıdır ve asansörlerin işleyişlerinin, buldukları binaya tam uygun olmasını sağlar. Asansör kumandası olmazsa, binanın sakinleri kendi hareketlerini asansörlere uydurmak zorunda kalırlar.

Asansör kumandası iki bölüme ayrılabilir.

Bölüm 1: Data tutma/hareket/bilgi alma

Bölüm 2: Data analizi/süzme (filtering)/ne olduğunu gösterme

Bilgisayarlarla ilgili esrarengiz hiçbirşey yoktur. Yukarıdaki iki bölümde insanlar tarafından yapılabilir. Fakat modern, güçlü bilgisayar teknolojisini kullanmak çok daha etkili ve yeterlidir.

Pek çok asansör üreticisi bölüm 1'i gerçekleştirmek için uzak iletişim sistemleri geliştirmişlerdir. (Bölüm 1 asansör kumandasında en temel ihtiyaçları içerir.) Fakat bu üreticilerin hiç biri bölüm 2'yi gerçekten hedef almamışlardır. Muhtemelen asansör şirketleri bu sistemleri rutin işletme işlerini desteklemek için bir araç gibi görmüşler ve bölüm 2'nin fonksiyonlarını geliştirmeye gerek görmemişler. Bölüm 2 asansör sahibinin ve kullanıcıların ihtiyaçlarını karşılamak üzere düzenlenmiştir. Aşağıda bunlar belirtilmiştir.

- 1-Merkezi bir noktaya gelen datanın analizi
- 2-Bütün asansörlerin durumlarını açıkca ve kısaca gösterme
- 3-Çağrılara cevap olarak yolculara sağlanacak servisin değerlendirilmesi ve asansör elverişliliği
- 4-Farklı eşit asansörler veya zaman periodları arasındaki ilişkileri gösteren yanlış hareketlerin (kusurlu işlemlerin) statistik analizi (tarihsel perspektif)
- 5-Güvenilmez parçaları yerleştirmek için ilişkileri kullanma
- 6-Gerekli dataya uygun olarak tamir işletme çalışmaları

5.3.2. Kullanım Faydaları

İdeal olarak bir asansör kumanda sistemi, bir bina ilk kullanıma açıldığında ve asansörlerin garanti süresi işleyebilir olmalıdır.

- 1-Düzenleme özelliklerine göre asansör servisini ölçmek
- 2-Üretici ve tüketici açısından tatmin edici bir asansör performansı oluşturmak
- 3-Asansörün önemli parçaları için bakım programları oluşturmak
- 4-Bakım ve programlanmamış çağrılara karşı hazırlıklı olmak
- 5-Tarihsel kayıtların, analizlerin kayıtlarını tutmak ve böylece tam bir performans sağlama

5.3.3. Performans

5.3.3.1. Performansın Ölçümü

Pek çok asansör sistem performansını ölçme metodları ileri sürülmüştür. Gerekli olan şey asansörlerin ölçülerini yolcu çağrılarını için gerekli zamanı ve benzerlerini kapsayan bir karakterizasyondur. Böyle performanslı tek bir asansör henüz ortaya çıkmamıştır. Bununla birlikte, bütün bu faktörler birbirinden ayrı olarak ele alınabilir ve daha sonra birleştirilebilir.

5.3.3.2. Performansın Analizi

Varolan bazı sistemler, kat çağrılarındaki cevaplama süresini yolcunun bekleme zamanıyla ölçerler. Bununla birlikte, bu datadaki bilginin sağlıklılığı sık sık kaybolur. Cevaplama süresinin yaygın değerleri ihtiyaç duyarlılığının etkisi kadar önemlidir.

5.3.3.3. Performansın Belirlenmesi

Bilginin belirlenmesi kısa ve açık olmalıdır. Hatalar ve önemli doğrultular aydınlatılmalıdır. Durumların tabloları az değerlidir. Çünkü bunlar açık olmayan, belirsiz konumlardır. Asansörlerin konumuna göre farklı çalışma sistemleri düşünülebilir. Örneğin Cartesion planları, Kiviat diagramları, kat çağrılarının cevaplama zamanlarının yüzdeleri dağılımları.

5.3.4. Bakımın Gözetimi

Bakımın bilgisayarlaşmış gözetimi bir program paketi ile iyileştirilebilir. Bu program her ayrıntıyı standart bir dizi operasyondan geçirdikten sonra sonucu kaydeder.

- 1-Arıza tesbitinde bakım görevlilerini uyarmak için otomatik ikaz cihazlarını aktive eder.
- 2-Kendi kendine üretilmiş hata teşhisini kaydetmek
- 3-Ikaz için monitör sorumlusu görevliler
- 4-Cevap elde edilene kadar alternatif bakım görevlilerini ikaz etmek.
- 5-Hata teşhisini görev kağıt raporlarını ve çareyi kabul etmek.
- 6-Rapor çelişkileriyle datayı karşılaştırmak.
- 7-Cevaplama zamanı ve hata tamir zamanının kayıtlarını tutmak

5.3.5. GRUP Binaların İdaresi

Microprocessor kontrollü yüksek performanslı asansör sistemleri en yeterli servisi verme kapasitesine sahiptir. Eğer kullanılacak nüfus ve yolcu sayısına göre uygun şekilde ayarlanırsa, asansör kumanda sistemleri, idarecilerin aşağıdakiler için data girmelerini kolaylaştırır.

- 1-Tatiller
- 2-Çalışma saatleri
- 3-Kantin saatleri
- 4-Konferans çizelgeleri
- 5-Geçiş güvenlik kontrolü

Aynı zamanda öncelikle servis katlarının tayini ve düzenlenmesinde, otomatik toplayıcı datanın girilmesinde kolaylaştırır. Asansör kumandası, ısıtma sistemi gibi binanın diğer servislerinin idaresi içinde tamamlayıcı bir servistir. (Teknik olarak daha karmaşık olmasına rağmen) ve genellikle aynı personel tarafından çalıştırılır.

Sonuçta ortaya çıkan asansör sistemi hala birkaç bilgisayarın kullanımını gerektirir. Fakat bunlar iletişim kuran cihazlarla uyumlu bir network'e bağlanabilir. Buda kullanıcıya tek bir sistem gibi görünür.

5.3.6. Data Tutmak/Nakil/Belge Tutmak

Bu konularla ilgili kısa açıklamalar aşağıdadır.

5.3.6.1. Data Tutmak

Asansör kontrollerine bilgisayar bağlantısı için yaygın bir standart gereklidir. Bu standartın output için donanım ve bilgi içeriğine sahip olması gerekir.

5.3.6.2. Data Nakli

Asansör sistemlerine özgü bir Network, kablo bağlantısı ekonomik kullanımı sağlayacaktır.

5.3.6.3. Data Belgelerini Tutmak (Arşiv)

Bütün datanın sağlam bir kopyası ayrı bir yerde tutulmalıdır. Değerli bilgiyi dışarıdan gelecek kaybolma, yangın sel gibi etmenlerden korumak için, aynı zamanda, bu sağlamlaştırılmış data, yeni bilgileri depolamak için atılmış eski bilgileride saklayacak, bir arşiv olarak kullanılabilir.

5.3.7. Data Analizi/Süzme/Sunuş

Asansör kumanda merkezi bilgisayarının kurulması basit olmalıdır. Hareket seçenekleri yeni kullanıcılara yardım edecek şekilde açıklanmış olmalı ve yabancı olmayan kullanıcılar içinde kolay ve hızlı işlemelidir. Sistemlerin kontrolü, çeşitli özellikler için uygun olmalı ve kullanıcının programın farklı bölümlerini çalıştırmak için depolama disklerini değiştirmesine gerek olmamalıdır. Basılacak anahtarların yeni kombinasyonlarını içermelidir. Bazı yanlış işlemler görüldüğünde, anlaşılamayan bir mesajı ortaya koymak için aniden normal işlemi bozabilmelidir. Hepsinin üstünde asansör kumanda sisteminin sistem parçaları tamamen

güvenilir olmalıdır ve yeterli servis, parça ve tamir organizasyonuyla desteklenmelidir.

5.3.7.1. Data Analizi

Asansör idarecisi birkaç binadan birden sorumludur ve bu yüzden de asansör kumanda sistemi gireni ve depolanmış datayı çabuk ve kolay karar verecek şekilde analiz eder. Böylece bilgisayar için datayı ortaya koymak yeterli değildir. Aynı zamanda kullanıcının doğru kararlar vermesi için imkanları analiz eder. Bu yüzden bilgisayar bu şekilde programlanmalı ve asansör idarecisi insanın düşünme metodlarını taklit etmelidir. Analiz biraz önce bahsedilen ilgi alanlarını içine almalıdır. Örneğin performans, güvenilirlik ve bakım.

5.3.7.2. Data Süzme(Filtering)

Potansiyel olarak tamamen yararlı asansör kontrolöründen büyük miktarda data elde etmek mümkündür. Fakat bunun için, diğer bilgi grubundan belli olayları izole etmek mümkün olmalıdır. Böylece kullanıcının belli bir ilgi alanına yönelmesini sağlayacak bir çeşit otomatik süzme işlemine ihtiyaç vardır. Örneğin katlara ulaşmak için belirlenmiş süre aşıldığında katları belirlemek.

5.3.7.3. Data Sunumu

Insan gözü bölümleri ayırmaya adapte olmuştur. Bu yüzden datanın renkli grafik sunuşları büyük miktardaki dataların en ideal sunuş şeklidir.

5.3.8. Gelişmiş Asansör Sisteminin Sonuçları

Asansör kumandasının sahiplerine ve kullanıcılarına ve asansörün bakımındaki görevlilere faydaları yıllardan beri

tartıŒılmaktadır. Donanım teknolojisi etkili idare için kullanılabilir. Yazılım daha da geliŒtirilebilir ve geliŒtirilmektedir.



6. BİLGİSAYAR VE ASANSÖR SİSTEMİNİN HABERLEŞMESİ

6.1. Bugünkü Asansör Sistemi

Türkiye'de asansör konusundaki gelişmeler şimdilik piyasadaki firmalar tarafından yapılmakta ve ticaret hayatını ilgilendirdiği için bir sır gibi saklanmaktadır. Zaten bu geliştirme çalışmalarını yapan firmaların sayısı % 1-2 civarındadır. Asansör firmalarının kendi aralarında montör ve imalatçı olarak ikiye ayrılmıştır. Bazı imalatçı firmalar montörlükte yapmakta veya malzeme teminini sağlayarak küçük firmalara kendisi yaptırmaktadır. İmalatçı olan firmalar ise genelde asansörün mekanik olan bölümünü üretirler. Sistemin elektrik kumandasını ise genelde birbirlerinden kopyalama suretiyle yapan firmalar çoğunluktadır. Bunuda daha çok para kazanmak için yapılan ticaret gereği olduğunu savunurlar ve bugün röleli devrelerden kurtulmayarak iş hayatına devam etmektedirler. Bugün elektronik pano kullanımı ve imalatı %1-2 civarındadır. Bunlarda Türkiye piyasasında belirli bir yeri bulunan firmalardır. Yazımızda Asansör firmalarının son çalışmalarından örnekler vermekteyiz. Ancak projelerin çalışması ve içerikleri açısından bir açıklama yapılmamıştır. Çünkü az önce açıkladığımız taklitçi firmalardan dolayı çalışmalar gizlilik içerisinde yapılmaktadır.

Bugünkü asansör sistemlerinde bir çok farklı yöntem kullanılmaktadır. Bunlardan birincisi röleli devreler, diğeri ise Elektronik devrelerdir. Bunların bilgileri elde edilimleri ise çok farklıdır. Röleli devrelerde direk olarak kat ve kabin bilgileri 12, 24 volt'luk gerilimlerle rölelere aktarılmakta ve devre tasarlanmaktadır. Elektronik devrelerde ise, kat bilgileri gray kodlama kullanılarak bilgi dijital olarak elektronik panoya aktarılmaktadır. Röleli kumandalarda ise kat bilgileri magnetik veya kopil-verk kullanılarak elde edilmektedir. Bilgiler pleksibil Kablo ile iletilmektedir. Sistem toplamalı ve çift hızlı

olmasına göre kablo adedi artmaktadır. Kat sayısına göre kablo uzunluğu ve yine kablo sayısı artmaktadır. Şu an kabin bilgisini ileten kablo sistemi kabinle birlikte yukarı aşağı hareket etmekte ve zamanla kablonun bilgi iletimi zayıflamakta arıza olasılığı artmaktadır. Burada her buton veya kat bilgisine göre kablo adedi bir artmaktadır. Gray kod çözümlenmeli sistem kullanıldığında kat bilgisi için 16 kata kadar dört adet kablo kullanılması yeterlidir. Çift hızlı sistemlerde ise, kat seviyelerinde bulunan çift hız seviyesi iki adet kablo kullanılarak temin edilebilir. Aynı zamanda bir kablo daha kullanılarak kat hassas seviyesi ayarlanarak, kabinin kat seviyesinde durması sağlanır ve kat seviyesindeki kabinin kaymaları önlenmiş olur.

6.2. Bilgisayar Sistemindeki İletişim

Bilgisayar sisteminin donanımında çeşitli durumlar düşünülebilir. En başta bilginin seri veya paralel olarak iletilmesi gelir. Ayrıca kullanılan sistem elemanlarından bilginin bilgisayarın anlayacağı şekilde nasıl elde edileceği, veya sistem elemanlarında ne gibi değişiklikler olabileceği bu bölümde anlatılacaktır.

Asansör sisteminde paralel haberleşmenin kullanılmasının kablo sayısına fazla bir etkisi yoktur. Kullanılan sistemdeki gray kod çözümlenme ile kat bilgileri, hız bilgisi ve hassas seviye kontrolü yedi adet kablo kullanılarak 16 kata kadar iletişimi sağlamaktadır. Ancak kabinin gelen kabin kat butonlarının bilgilerinde fazla bir değişikliğe neden olmamaktadır. Kabin kat bilgilerinin yine tek birer kablo üzerinden iletimi yapılmaktadır.

Sistemde seri haberleşmenin kullanılması ise çok büyük bir tasarrufu oluşturacak, ancak bilginin sağlıklı olarak iletilmesi, donanım'da büyük zorluklara neden olacaktır. Çünkü her katta bulunan kat bilgisinin tek bir kablo üzerinden iletilmesi, donanım olarak her katta bir devre

ihtiyacını oluşturacak ve mevcut olan bilginin üzerine o kattaki bilginin kodlanması zor olacaktır. Ancak kabin buton bilgilerinin seri olarak iletilmesi kablo sayısında büyük azalmalar neden olacak ancak bu bilginin, kabinle hareket eden bir kablo ile iletilmesi çok büyük zorluklara neden olacaktır. Çünkü hareketli kablonun kırılma ve bilginin zayıflaması gibi sorunları ortaya çıkaracaktır.

Yapılması en uygun ve basit çözüm ise kat seviye ve hız bilgilerinin yine gray kod çözümü ile elde edilmesidir. Kabin için ise NYAF kablolarla iletimin sağlandıktan sonra ise bilginin Analog-Digital Converter ile digital bilgiye dönüştürülmesi gerekir. Böylece sistemin bilgi girişi sağlanmış olacaktır.

Sistemin çıkış işlemleri ise motor kontrolü ve frenleme tertibatıdır. Bu ise bilgisayar çıkışındaki digital bilgi ile motor kontrol devresinin oluşturulmasıyla sağlanacaktır. Bir diğer çıkış işlemi ise katlarda bulunan digital kat göstergeleri ile ışıklı göstergelerin kontrolüdür. Bu tamamen digital olarak çözümlenebilir.

7. PROGRAMIN İNCELENMESİ

7.1. Programda Yararlanılan Değişkenlerin Değerlendirilmesi

7.1.1. Integer Değişkenler

-kkbg1: Programın grafik dizaynı bölümünde kullanılan değişkendir. Sistemde kullanılan kkb1 değişkeninin değişimi ile orantılı olarak değişir. 21 ile 381 sınırları arasında değer alır.

-kkbg2: kkb1 değişkeninin özelliklerini taşır. Sistemde kullanılan kkb2 değişkeninin değişimi ile orantılı olarak değişir.

-errorcode: Grafik ekranda çalışırken bir hata mesajı oluşursa, hata mesajının kod numarasını alır ve hatanın ekrana yazılmasını sağlar. 15 tane hata kodu vardır.

-graphmode: Grafik ekrana geçerken ekranın çalışılacak olan grafik kodunun değerini alır.

-graphdriver: Grafik ekrana geçmeden önce çalışılan ekranın türünün kodunun değerini alır.

-t,k: Program içerisinde for to do döngüleri içerisinde kullanılır.

-kbek1: Programın simulasyon bölümünde, basılan birinci kabin butonları arasında en yakın mesafedeki, kabin butonunu bulmak için kullanılan değişkendir.

-kbek2: kbek1 değişkeninin özelliklerini taşır. İkinci kabinin elemanıdır.

-kis1: Birinci kabin içerisinde bulunan yolcu sayısını saklayan değişkendir.

-kis2: İkinci kabin içerisinde bulunan yolcu sayısını saklayan değişkendir.

-kkb1: Birinci kabinin normal ekrandaki kat bilgisini saklayan değişkendir.

-kkb2: İkinci kabinin normal ekrandaki kat bilgisini saklayan değişkendir.

-i,j: Program içerisinde for to do döngüleri içerisinde kullanılır.

-n1: Programda, birinci kabin kat seviyesinde iken kapı aç komutu olduğunda, kapının açılıp kapanıncaya kadar yapılan program döngü sayısını sayan değişkendir.

-n2: n1 değişkeninin özelliklerini taşır. İkinci kabin kapısı için geçerlidir.

7.1.2. Word Değişkenler

-size: Programın simülasyon kısmından grafik ekrana geçtikten sonra çizimler yapılır. Bazı çizimleri hafızaya almak için size değişkeni kullanılır.

-getmaxcolor: Grafik ekranda çalışırken belirlenen renklerin, saklanmasını sağlayan değişkendir.

-pausetime: Programın çalışması sırasında zaman gecikmelerini sağlayan değişkendir.

7.1.3. Pointer Değişkenler

-k1: Grafik ekran çalışmasında, açılan kapı durumunun birinci konumunu, hafızada tutan değişkendir.

-k2: Grafik ekran çalışmasında, açılan kapı durumunun

ikinci konumunu, hafızada tutan değişkendir.

-k3: Grafik ekran çalışmasında, açılan kapı durumunun üçüncü konumunu, hafızada tutan değişkendir.

-k4: Grafik ekran çalışmasında, açılan kapı durumunun dördüncü konumunu, hafızada tutan değişkendir.

-t1: Programın grafik bölümünde, yukarı kat tuşlarının çizilmesinden sonra görüntüyü hafızada saklayan değişkendir.

-y: Programın grafik bölümünde, aşağı kat tuşlarının çizilmesinden sonra görüntüyü hafızada saklayan değişkendir.

-u: Programın grafik bölümünde, kabinin şeklini çizdikten sonra görüntüyü hafızada saklayan değişkendir.

-y1: Grafik ekranda kabin içerisindeki yukarı hareket okunun çizimini hafızada saklayan değişkendir.

-y2: Grafik ekranda kabin içerisindeki aşağı hareket okunun çizimini hafızada saklayan değişkendir.

7.1.4. String Değişkenler

-s: Programın grafik bölümünde, kişi sayısını out-textxy komutuyla ekrana yazdırmaya yarayan değişkendir.

7.1.5. Char Değişkenler

-ch: Programın simülasyon bölümünde, tuş okuma işlemi sırasında kullanılan değişkendir.

7.1.6. Type Değişkenleri

-xycoord: Programın grafik bölümünde dizi olarak şekil çiziminde kullanılan kayıt değişkenidir.

-p: Grafik bölümünde, dizi olarak şekil çizilirken, belirtilen koordinata çizerek veya move olarak taşınmasını sağlayan byte değişkendir.

-kt: Programın simülasyon bölümünde, tüm katlar için, beş elemanlı boolean bir dizidir.

-ky: Programın simülasyon bölümünde, yukarı işaretler için, kullanılan dört elemanlı boolean bir dizidir.

-ka: Programın simülasyon bölümünde, aşağı işaretler için, kullanılan dört elemanlı boolean bir dizidir.

7.1.7. Dizi Değişkenleri

-kb1: Simülasyon bölümünde, ikinci kabin butonlarının dizi halinde değerlerini koruyan boolean bir değişkendir.

-kb2: Simülasyon bölümünde, birinci kabin butonlarının dizi halinde değerlerini koruyan boolean bir değişkendir.

-yi: Simülasyon bölümünde yukarı işaret kat butonlarının boolean değerlerini saklayan dört elemanlı değişkendir.

-ai: Simülasyon bölümünde, aşağı işaret kat butonlarının boolean değerlerini saklayan dört elemanlı değişkendir.

vis: Programın Simülasyon bölümünde basılan yukarı kat butonlarının, yalnızca bir kabin sisteminin hareketine verilmesini sağlayan, dört elemanlı boolean bir değişkendir.

ais: Programın Simülasyon bölümünde basılan aşağı kat butonlarının, yalnızca bir kabin sisteminin hareketine verilmesini sağlayan, dört elemanlı boolean bir değişkendir.

7.1.8. Boolean Değişkenler

-mdur1: Programın simülasyon kısmında, birinci kabin içerisindeki stop düğmesinin değerini alan değişkendir. True olduğu zaman birinci kabinin durmasını sağlar.

-mdur2: Programın simülasyon kısmında, ikinci kabin içerisindeki stop düğmesinin değerini alan değişkendir. True olduğu zaman ikinci kabinin durmasını sağlar.

-kkapi1: Birinci kabinin kapısının, açık olup olmadığını bildiren değişkendir.

-kkapi2: İkinci kabinin kapısının, açık olup olmadığını bildiren değişkendir.

-kkapig1: Birinci kabinin kapısının grafik bölümde, açılıp kapanmasını sağlayan değişkendir.

-kkapig2: İkinci kabinin kapısının, grafik bölümde, açılıp kapanmasını sağlayan değişkendir.

-kin1: Değişkenin tuşuna basıldığı zaman birinci kabinin kişi sayısını bir azaltır. Yalnızca kabin kapısı açık olduğu zaman işlem yapılır.

-kin2: Değişkenin tuşuna basıldığı zaman ikinci kabinin kişi sayısını bir azaltır. Yalnızca kabin kapısı açık olduğu zaman işlem yapılır.

-kbin1: Değişkenin tuşuna basıldığı zaman birinci kabinin kişi sayısını bir arttırır. Yalnızca kabin kapısı

açık olduğu zaman işlem yapılır.

-kbin2: Değişkenin tuşuna basıldığı zaman ikinci kabinin kişi sayısını bir arttırır. Yalnızca kabin kapısı açık olduğu zaman işlem yapılır.

-tyi1: Birinci kabinin yolcu sayısının tam yükte olup olmadığını belirten değişkendir.

-tyi2: İkinci kabinin yolcu sayısının tam yükte olup olmadığını belirten değişkendir.

-ayi1: Birinci kabinin, kabindeki yolcu sayısının kabin kapasitesini aştığı zaman değişkenin değeri true olur ve sesli ikaz yaparak kapının açılmasını sağlar.

-ayi2: İkinci kabinin, kabindeki yolcu sayısının kabin kapasitesini aştığı zaman değişkenin değeri true olur ve sesli ikaz yaparak kapının açılmasını sağlar.

-kbdur1: Birinci kabinin durmasını sağlayan değişkendir. Tuşa basıldığı zaman tüm kat bilgilerini silerek resetlenmesini sağlar.

-kbdur2: İkinci kabinin durmasını sağlayan değişkendir. Tuşa basıldığı zaman tüm kat bilgilerini silerek resetlenmesini sağlar.

-myh1: Programın simülasyon bölümünde birinci kabinin hareketini sağlayan, motorun yukarı doğru hareketini gerektiren değişkendir.

-myh2: Programın simülasyon bölümünde ikinci kabinin hareketini sağlayan, motorun yukarı doğru hareketini gerektiren değişkendir.

-mah1: Programın simülasyon bölümünde birinci kabinin

hareketini sađlayan, motorun ařađı dođru hareketini gerektiren deđiřkendir.

-mah2: Programın simülasyon bölümünde ikinci kabinin hareketini sađlayan, motorun ařađı dođru hareketini gerektiren deđiřkendir.

7.2. Programdaki Procedure`lerin Anlatımı

7.2.1. Procedure Kiřisayil

Procedure`de birinci kabinin, kiři sayısı 0 ile 9 arasında deđerler alır. Birinci kabinin kiři sayısı 9`dan küçük ve kbin1 tuřuna basılmıř ise kiři sayısını bir artırır ve yeni kiři sayısını birinci kabinin displayine yazar. Eđer kin1 tuřuna basılmıřsa birinci kabindeki kiři sayısını bir azaltarak displye yazar.

7.2.2. Procedure Kiřisayi2

Kiřisayil`deki tüm özellikleri tařır. Birinci kabinde yapılan işlemleri ikinci kabin için tekrarlar.

7.2.3. Procedure Katsayil

Programın bu bölümünde kabkat1 boolean deđerkeni kontrol edilerek birinci kabinin kat bilgisini, birinci kabin içerisindeki kat displayine yazılmasını sađlar. Burada kabkat1 deđerkeninin kontrol edilmesinin amacı; Birinci kabin kat seviyesine geldiđi takdirde kat bilgisinin displaye bir defa yazılmasını sađlar. Böylece kabin kat seviyesinde dururken displayde yanıp sönmelere neden olmaz. Kat seviyeleri kkbgl deđerkeni kontrol edilerek case of end; döngüsü kullanılarak yapılmıřtır.

7.2.4. Procedure Katsayi2

İkinci kabinin kat sayısının yazılmasını sağlayan değişkendir. katsayi1 procedure'nün işleyiş mantığıyla çalışır.

7.2.5. Pocedure Tamyuk

Bu procedure'de her iki kabininde tam yük kontrolü yapılmaktadır. Procedure'ün her işleme girmesinden önce tyi1 ve tyi2 false yapılır. Daha sonra kişi sayısı kontrol edilerek tyi1 veya tyi2 true yapılır. Programın ilerdeki bölümlerinde de tyi1 ve tyi2'nin konumuna bakılmaktadır.

7.2.6. Procedure Asırlıy

Programın bu bölümünde her iki kabinin de aşırı yük kontrolü yapılmaktadır. Öncelikle ayi1 ve ayi2 false yapılır. Daha sonra kişi sayısı kontrol edilerek kabinlerin aşırı yük değişkenleri true yapılır.

7.2.7. Procedure Tusoku

Bu procedure'de tuş okuma işlemi yapılmaktadır. Bir tuşa basılmış ise tuşun klavye kodunu karşılaştırarak case of end; döngüsü içerisinde basılan tuşun koduna göre tuşun boolean değişkenini true yapmaktadır. Programın işleyişi sırasında değişkenler konumlarına göre tekrar false olurlar.

7.2.8. Procedure Fonciz

Programın bu bölümünde, grafik ekranın çerçevesi çizilmektedir. Const'da belirtilen fon dizisine göre for to do döngüsü içerisinde lineto ve moveto komutlarıyla ekranın çerçevesi çizilir.

7.2.9. Procedure Kapıciz

Bu procedure`de const`da belirtilen kapı dizisinin şekli for to do döngüsü içerisinde moveto ve lineto komutları kullanılarak kapı çizimi yapılır.

7.2.10. Procedure Kapıciz1

Bu procedure kapı açılma şeklinin birinci konumunun çizilmesini sağlar. Const`da belirtilen kapıl dizisi yardımıyla yine for to do döngüsü içerisinde moveto ve lineto komutlarıyla açık kapının birinci konumu çizilmektedir.

7.2.11. Procedure Kapıciz2

Bu procedure kapı açılma şeklinin ikinci konumunun çizilmesini sağlar. kapıciz1 procedure'nün mantığı ile çalışır.

7.2.12. Procedure Kapıciz3

Bu procedure kapı açılma şeklinin üçüncü konumunun çizilmesini sağlar. kapıciz1 procedure'nün mantığı ile çalışır.

7.2.13. Procedure Kapıciz4

Bu procedure kapı açılma şeklinin dördüncü konumunun çizilmesini sağlar. kapıciz1 procedure'nün mantığı ile çalışır.

7.2.14. Procedure Kabinciz

Programın bu bölümünde birinci kabinin çizimi yapılmaktadır. Öncelikle for to do döngüsü içerisinde çerçeve kısımları çiziliyor. İkinci for to do döngüsü içerisinde kabin kat butonları çiziliyor. Daha sonra da outtextxy

komutlarıyla kabin kat butonlarının yazıları yazılmaktadır.

7.2.15. Procedure Başla

Bu procedure Grafik ekranın başlangıç çizimlerini yapmaktadır. Önce fonciz alt procedure`üyle ekranın çerçevesi ve iç fonu çizilmektedir. Sonra kapıciz alt procedure`üyle kapı çizdirilmektedir. Sonra kabinciz alt procedure`üyle kabinin komple çizimi yapılarak hafızaya alınmakta ve daha sonra ikinci kabinin çizimi olarak kopyalanmaktadır. Sonra az önce çizilen kapı hafızaya alınmaktadır. Bu hafızaya alınan kapı çizimiyle for to do döngüsü içerisinde tüm normal kat kapıları konuluyor. Sonra tüm kat seviyelerinde bulunan kabin çağrı butonları çizimi yapılmaktadır. Burada da tek çizim yapıldıktan sonra katların kabin çağrı butonlarının çizim işlemi tamamlanıyor. Sonra kişisayısını ve kat seviyesini gösteren displaylerin arka fonunun çizimi yapılmaktadır. Sonra kabinin hareket ettiğini gösteren kırmızı desenli kapı resmi çizilerek hafızaya alınmaktadır. Daha sonra programın başlangıç konumu olan birinci kat seviyelerinin çizimleri yapılıyor. Daha ilerdeki bölümlerde kapının nasıl hareket ettirileceği anlatılacaktır. Sonra kapının açılması görüntüsünü oluşturan kapıciz1, kapıciz2, kapıciz3 ve kapıciz4 alt procedure`leriyle açılan kapı çizimleri yapılarak hafızaya alınıyor. En son olarakta ekranda boş olarak gözükken sağ ve sol üst bölümlerin desenleri çizilmektedir.

7.2.16. Procedure Reset

Bu pcedure`de programın simülasyon bölümünün resetleme işlemleri yapılmaktadır. Yani programın başlangıcında değişkenlerin hangi değerleri alacağı belirtilmektedir. Hangi değişkenlerin ne değerler aldığı aşağıda açıklanmaktadır.

n1, n2: Kabinlerin kata geldiğinde kapıların açılma

süresini sağlayan değişkenin başlangıçta sıfır olması gerekir. Çünkü başlangıçta kapı kapalı durumdadır.

kkb1, kkb2: Programın simülasyon bölümünde ekran 25 satır olarak tasarlanmıştır. Zemin kat seviyesi 23'tür. Kabinler başlangıçta zemin katta bulunduğu için kkb1, kkb2 değişkenleri 23 değerini alırlar.

kbek1, kbek2: Basılan kabin butonlarının, en yakın mesafedeki buton ile kabin seviyeleri arasındaki farkı, belirten değişkendir. Başlangıçta herhangi butona basıldığında kabin seviyesiyle basılan kat butonunun seviyesi arasındaki maksimum fark 20'dir. Başlangıçta bu değerler değişkenlere verilir.

t: Yukarıda belirtilen, kabin seviyesiyle basılan kat butonu arasındaki farkı geçici olarak hafızada tutan değişkendir. Ve 20 değerini alır.

kkbg1, kkb2: Programın simülasyon bölümünde kabin seviyelerini belirten kkb1 ve kkb2 değişkenlerine grafik ekranın dizaynında, kkb1 ve kkb2 değişkenleri karşılık gelmektedir. Burada zemin kat seviyesi 381'dir. Bunun için kkb1 ve kkb2 değişkenlerine başlangıçta 381 değeri verilir.

kabkat1, kabkat2: Kabinlerin kat seviyesini gösteren displaylerde kat seviyesinin devamlı olarak basılması ekranda bir flaşöre (yanıp sönmelere) neden olur. Yalnızca kat değişimi olmasında, kat seviyesinin yazılması için kabkat1 ve kabkat2 boolean değişkenleri kullanılır ve bu değişkenlerin başlangıçta false olması gerekir.

kis1, kis2: Başlangıçta kabinlerde bulunan kişi sayısı sıfır olduğundan yani kabinde kimse bulunmadığı için kis1 ve kis2 değişkenlerine sıfır değeri verilir.

kkap1, kkap2: Programın başlangıç bölümünde kabin kapıları kapalı olduğu için kkap1 ve kkap2 değişkenlerine true değeri verilir.

kkap1g1, kkap1g2: Yukarıda belirtilen kkap1 ve kkap2 değişkenleri programın simülasyon bölümünde geçerlidir. Aynı zamanda kapıların açılıp kapanıncaya kadar kkap1 ve kkap2 değişkenlerinin false olarak kalmasını sağlayan kkap1g1 ve kkap1g2 değişkenleri başlangıçta true olması gerekir.

myh1, myh2: Programın simülasyon bölümünde, kullanılan bu değişkenler, programın başlangıcında kabinler hareketsiz olduğu için myh1 ve myh2 değişkenleri false değerini alır.

mah1, mah2: Programın simülasyon bölümünde, kullanılan bu değişkenler, programın başlangıcında kabinler hareketsiz olduğu için mah1 ve mah2 değişkenleri false değerini alır.

kb1[I], kb2[I]: Programın bu bölümünde bir for to do döngüsü içerisinde birinci ve ikinci kabinlerin tüm kat butonlarının değerlerinin false olması gerekir. Başlangıçta herhangi bir bilgi girişi olmadıkça kabinin hareketsiz olarak kalması istenir.

a[I], y[I]: Aşağı ve yukarı kabin çağrı butonlarının başlangıçta false olması gerekmektedir. Başlangıçta kabinler hareketsiz durumdadır.

ais[I], yis[I]: Aşağı veya yukarı çağrı yapıldığı zaman bilginin yalnızca bir kabine verilmesini sağlayan değişkenlerdir. Herhangi bir kattan kabin çağrısı bulunduğu anda sistemin cevaplama için aşağı ve yukarı işaret sinyallerinin, yani ais ve yis'nin de false olması gerekir.

7.2.17. Procedure Kapidurum

Programın bu bölümünde, simülasyon ve grafik iç içe-
 dir. Öncelikle bu procedure`de kabin kat seviyesine gelme-
 den işlem yapılmaz. Kabin kat seviyesine geldiğinde, simü-
 lasyon ve grafik kat seviyeleri, birbirine eşit ise işleti-
 me geçer. Bu ilk konumda kkapıl false`dur. Simülasyon kıs-
 mında, kat seviyesine gelindiğinde kapı aç komutunu verir.
 Bu sırada n1 değişkeninde sıfır olması gerekir. Böylece
 bir anlamda çift emniyetli olmasında sağlanmış olur. n1
 sıfır ve kkapıl false ise kkapıl`deki bilgi kkapıgı`e akta-
 rılır. Yani kkapıgı`de false olur ve programın her devrinde
 n1 değişkeninin değeri bir arttırılır. Case of end; döngüsü
 içerisinde n1 değişkeninin değerine göre kapı konumunu be-
 lirler. n1 10 olduğunda kapı birinci konumunu alır. n1 20
 olduğunda kapı ikinci konumunu alır. n1 30 olduğunda kapı
 üçüncü konumunu alır. n1 40 olduğunda kapı dördüncü konu-
 mumu alır. Bu sırada kkapıl ve kkapıgı boolean değişkenleri
 false değerindedir. n1 değişkeninin değeri 100 oluncaya
 kadar kapı dördüncü konumunda kalır. n1 100 olduğunda case
 of end; döngüsünde n1 değeri 41 olarak değiştirilir ve
 kkapıgı true yapılır. Böylece sistemin işletiminde if then
 komutunda else komutu işleme girer. Else komutunun işle-
 minde ise n1 değişkeni bir azaltılarak bu seferde kapı
 konumları tersten olarak işleme girer. Böylece kapı kapa-
 nıyor görüntüsü oluşturulur. kkapıgı true ve n1 sıfır oldu-
 ğunda kkapıl değişkenine true değeri atanır. Böylece kapı-
 nın açılıp kapanma işlemi görüntü olarak sağlanmış olur. Bu
 işlem sırasında kişi sayısını belirten alt procedure çalış-
 tırılır. Böylece yalnızca kapı açık iken yolcu girişi ve
 çıkışı sağlanmış olur. Eğer kapının açık kalma süresinin
 uzatılması istenirse case of end; döngüsü içerisinde 100
 olarak alınan değer arttırılarak kapının daha fazla açık
 kalması sağlanabilir. Bu işletimin aynısı ikinci kabinde
 de yapılır. Burada ise döngü sayısını n2 gibi başka bir
 değişkenle yapmak gerekir. Kapı konumları, aynı sırada
 değişmediğinden yani aynı zamanda açılıp kapanmadığından,

farklı deęişkenler kullanmak gerekir. İkinci kapının açık olduęu zamanda `kişisay2` alt procedure`ü çalıştırılırlar.

7.2.18. Procedure `Kabink_1`

Öncelikle bu procedure`e başlarken, birinci kabin kat bilgisiyle, basılan en yakın birinci kabin kat butonu arasındaki farkı, gösteren `kbek1` deęişkenine 20 deęeri verilmelidir. Çünkü birinci kabin kat seviyesiyle basılan kat butonu seviyesi arasındaki maksimum deęer 20`dir. Burada birinci kabin kat seviyesiyle basılan kat butonlarının seviyesi arasındaki farkı geçici olarak hafızada tutan `t` deęişkenine de 20 deęeri verilir.

Bir `for to do` döngüsü içerisinde basılan kat butonunun olup olmadığı kontrol edilir. Basılan kat butonu varsa birinci kabin kat seviyesi ile basılan kat seviyesi arasındaki fark, `kbek1` deęişkenine aktarılır. Birden fazla basılan kat butonu varsa, `t` deęişkeni yardımıyla yakın olan kat butonu seviyesi ile birinci kabin kat seviyesi arasındaki fark, `kbek1` deęişkenine aktarılır. Burada `t` deęişkenini yazarken farkın mutlak deęeri alınmıştır. Çünkü eşitlikte negatif deęer çıkabilir. Mutlak deęer alınarak negatif çıkma olasılıęıda düşünölmüş olur.

Daha sonra $j = (kbb1 + kbek1 + 2) \text{ div } 5$ eşitlięiyle basılan kat butonunun hangi kat olduęu hesaplanır. `myh1` false ve `j` sıfır ile altı arasında bulunuyorsa ve `kb1[6-j]` true ise `mahl` true olur. Yukarıdaki eşitliklerin alınmasının nedeni araştırılırsa; `j` deęişkeninin 0 ile 6 arasında alınmasının nedeni; kabin basılan kat butonundan daha alt seviyede ise `j` deęişkeninin deęeri, basılan deęere göre daha alt seviyedeki kat butonu olarak algılar bunu önlemek için `kb1[6-j]` olarak, kat butonunu kontrol eder. `kb1[6-j]` false olduęundan `mahl` true olmayacaktır. Sonra `myh1` true olabilmesi için öncelikle `kbb1``in `kbek1``den daha büyük olması gerekir. Birinci kabin kat seviyesi arttıkça satır numarası azala-

caktır. Daha sonra j belirtilen formülle deęerini alır. Burada dikkat edilirse kbek1'in önüne eksi iřareti konulmuřtur. Yukarı hareket için basılan kat butonunun birinci kabin seviyesinden daha küçük olması gerekir. Yine burada mah1'inde false ve kb1[6-j]'ninde true olması gerekir.

7.2.19. Procedure Kabink_2

Bu procedure kabink_1 procedure'ünde birinci kabin için yapılan iřlemlerin ikinci kabin için yapılmasını sağlar.

7.2.20. Procedure Katdur_1

Programın bu procedure'ünde birinci kabinin hangi durumlarında kkap1'in false olacaęının belirtilmesidir. Burada case of end; döngüsü ile birinci kabinin kat seviyesi kontrol edilmektedir. Eęer kabin herhangi bir kat seviyesine gelmiřse bulunduęu katın kabin butonuna basılmıř veya tam yükte deęilken ve motor ařaęı hareket ediyor ve o katın ařaęı butonuna basılmıř veya yine tam yükte iken motor yukarı hareket ediyor ve o katın yukarı iřaret butonuna basılmıř ise kkap1 boolean deęiřkenini false yapar.

7.2.21. Procedure Katdur_2

Bu procedure katdur_1 procedure'ünde birinci kabin için yapılan iřlemlerin ikinci kabin için yapılmasını sağlar.

7.2.22. Procedure Yukarı_1

Birinci kabin yukarı doęru hareket ediyorsa bu alt procedure çalışacaktır. Birinci kabinin yukarı doęru hareket etmesi demek kkb1'nin bir azalması demektir. Kabin seviyesi artarken satır numarası bir azalır. Bunun için bu alt procedure'de öncelikle kkb1 bir azaltılır. Sonra case of end; döngüsü ile kkb1'nin deęerine göre iřlemler yapı-

ılır. Herhangi bir kata gelindiğinde katdur_1 alt procedure'ü çalıştırılır. Aynı zamanda o kata ait kabin buton bilgisi ve aşağı işaret ve yukarı işaret sinyalleri false yapılır.

7.2.23. Procedure Yukarı_2

Bu procedure yukarı_1 procedure'ünde birinci kabin için yapılan işlemlerin ikinci kabin için yapılmasını sağlar.

7.2.24. Procedure Aşağı_1

Birinci kabin aşağı doğru hareket ediyorsa bu alt procedure çalışacaktır. Birinci kabinin aşağı doğru hareket etmesi demek kkb1'nin bir artması demektir. Kabin seviyesi azalırken satır numarası bir artar. Bunun için bu alt procedure'de öncelikle kkb1 bir arttırılır. Sonra case of end; döngüsü ile kkb1'nin değerine göre işlemler yapılır. Herhangi bir kata gelindiğinde katdur_1 alt procedure'ü çalıştırılır. Aynı zamanda o kata ait kabin buton bilgisi ve aşağı işaret ve yukarı işaret sinyalleri false yapılır.

7.2.25. Procedure Aşağı_2

Bu procedure aşağı_1 procedure'ünde birinci kabin için yapılan işlemlerin ikinci kabin için yapılmasını sağlar.

7.2.26. Procedure Hareket

Bu procedure'de sistemin durumuna göre kabinlerin hangi yönde hareket edeceği belirtiliyor. Birinci kabinin yukarı doğru hareket etmesi için yalnızca myh1 değişkeninin true olması yeterli değildir. Aynı zamanda sistemin emniyet şartlarınınında yerine gelmesi gerekir. Öncelikle birinci kabinin aşırı yükte olup olmadığı söz konusudur. Birinci kabin motor yukarı hareket true, ve birinci kabinin kapısı

kapalı ve aşırı yükte değilse yukarı_1 alt procedure'ü çalışır. İkinci kabin motor yukarı hareket true ve ikinci kabinin kapısı kapalı ve aşırı yükte değilse yukarı_2 alt procedure'ü çalışır. Birinci kabin motor aşağı hareket true ve birinci kabinin kapısı kapalı ve aşırı yükte değilse aşağı_1 alt procedure'ü çalışır. İkinci kabin motor aşağı hareket true ve ikinci kabinin kapısı kapalı ve aşırı yükte değilse aşağı_2 alt procedure'ü çalışır.

7.2.27. Procedure Dur_1

Birinci kabinin durumuna göre motor aşağı hareket veya motor yukarı hareketin false yapılması tasarlanıyor. j değişkeni sıfırlandıktan sonra for to do döngüsü içerisinde, kabin seviyesinin üzerindeki kat butonları kontrol edilerek false olanların sayısı j değişkenine yüklenir. Birinci kabin seviyesinden yukarısında basılan bir kat butonu yok ise birinci kabinin, motor yukarı hareket değişkeni false yapılır. Yine j değişkeni sıfırlandıktan sonra for to do döngüsü içerisinde birinci kabin seviyesinin altındaki kat butonları kontrol edilerek false olanların sayısı j değişkenine yüklenir. Birinci kabin seviyesinden altında basılan bir kat butonu yok ise birinci kabinin motor aşağı hareket değişkeni false yapılır.

7.2.28. Procedure Dur_2

Bu procedure dur_1 procedure'ünde birinci kabin için yapılan işlemlerin ikinci kabin için yapılmasını sağlar.

7.2.29. Procedure Asakont_1

Bu procedure'de programın temel kuralları tasarlanmaktadır. Burada amaç, herhangi bir kattan aşağı çağrı işareti verildiğinde, hangi kabinin harekete geçeceğinin belirlenmesini sağlamaktadır. Öncelikle belirlenmesi gereken, hangi kabinin çalışıp çalışmadığının tespit edilmesidir. Bu

konumlar şu şekildedir. Ya kabinlerin ikisinde hareket halindedir veya kabinlerden birisi çalışır durumdadır veya iki kabinde hareketsiz konumdadır.

İlk konum olarak iki kabininde hareketli olduğu düşünülürse, buna göre if then döngüleri gereğince, bu procedure'de hiç bir işlem yapmadan, procedure'den çıkar. Diğer procedure'lerde iki kabinden birisi çağrı yapılan kat seviyesinden geçerken, durur ve yolcuyla alır.

İkinci konum, iki kabinde hareketsiz olarak dururken, herhangi bir kattan bir çağrı işlemi yapılırsa; Çağrı yapılan kat seviyesiyle iki kabinin de kat seviyeleri arasındaki farklar hesaplanır. Birinci kabin kat seviyesi ile çağrı yapılan kat arasındaki fark, j değişkenine aktarılır. İkinci kabin kat seviyesi ile çağrı yapılan kat seviyesi arasındaki fark k değişkenine aktarılır. j k 'dan büyük veya eşitse ikinci kabin çağrıyı alır. Yinede görevi almadan önce tam yük kontrolü ve aşağı işaret sinyalinin önceden başka bir kabine verilmediği kontrol edilir. Birinci kabin daha yakın konumda ise birinci kabinin tam yük kontrolü ve aşağı işaret sinyali kontrol edildikten sonra birinci kabine görev verilir. Burada dikkat edilmesi gereken konu, aşağı çağrı bilgisinin kabine buton bilgisi olarak verilmesidir. Çağrıyı alan kabin, daha sonra uğradığı katlardan dolayı tam yüke ulaşsa bile çağrı yapılan kata uğramak zorundadır.

Üçüncü konum ise ikinci kabinin hareketli konumda ve birinci kabin hareketsiz olduğu konumda aşağı çağrı işlemi yapılırsa; bu konumda öncelikle kontrol edilmesi gereken ikinci kabinin çalışma şeklidir. Eğer ikinci kabinin kat seviyesi çağrı yapılan kattan daha yukarıda ve tam yük halinde değilse, görev ikinci kabine verilmiş olur. Burada dikkat edilecek konum ikinci kabine çağrı bilgisi kabin kat bilgisi olarak değil kattan geçerken kat çağrısını değerlendirerek durmasıdır. İkinci kabin görevi alacak durumda

değilse birinci kabin için konum kontrolüne devam edilir. Birinci kabin tam yük konumunda değil ve aşağı çağırma bilgisi başka bir kabine verilmemişse görev birinci kabine verilir.

Dördüncü konum ise birinci kabinin hareketli ve ikinci kabin hareketsiz olduğu konumda aşağı çağrı işlemi yapılırsa; bu konumda öncelikle kontrol edilmesi gereken birinci kabinin çalışma şeklidir. Birinci kabinin kat seviyesi çağrı yapılan kattan daha yukarıda ve tam yük halinde değilse görev birinci kabine verilmiş olur. Burada dikkati çeken özel bir durum vardır. Birinci kabine çağrı bilgisi kabin kat bilgisi olarak değil kattan geçerken kat çağrısını değerlendirerek durmasıdır. Birinci kabin görevi alacak durumda değilse, ikinci kabin için konum kontrolüne devam edilir. İkinci kabin tam yük konumunda değil ve aşağı çağırma bilgisi başka bir kabine verilmemişse görev ikinci kabine verilir.

Bu procedure'de dikkat edilmesi gereken nokta; Çağrı bilgisinin direkt olarak uygun olan kabine verilme işlemi, kabine bilginin kabin kat butonu bilgisi olarak verilmesidir. Programın bu bölümü dahada geliştirilebilir.

7.2.30. Procedure Yukakont_1

Bu procedure'de programın temel kuralları tasarlanmaktadır. Burada amaç herhangi bir kattan yukarı çağrı işareti verildiğinde hangi kabinin harekete geçeceğinin belirlenmesini sağlayan procedure'dür. Öncelikle belirlenmesi gereken hangi kabinin çalışıp çalışmadığının tespit edilmesidir. Bu konular şu şekildedir. Ya kabinlerin ikiside hareket halindedir veya kabinlerden birisi çalışır durumdadır veya da iki kabinde hareketsiz konumdadır.

İlk konum olarak iki kabinde hareketli olduğu düşünülürse buna göre if then döngüleri gereğince bu procedure'de

hiç bir işlem yapmadan procedure'den çıkar. Diğer procedure'lerde, iki kabinden birisi çağrı yapılan kat seviyesinden geçerken, durur ve yolcuyla alır.

İkinci konum iki kabinde hareketsiz olarak dururken herhangi bir kattan bir çağrı işlemi yapılırsa; Çağrı yapılan kat seviyesiyle iki kabinin de kat seviyeleri arasındaki farklar hesaplanır. Birinci kabin kat seviyesi ile çağrı yapılan kat arasındaki fark, j değişkenine aktarılır. İkinci kabin kat seviyesi ile çağrı yapılan kat seviyesi arasındaki fark, k değişkenine aktarılır. j , k 'dan büyük veya eşitse ikinci kabin çağrıyı alır. Yinede görevi almadan önce tam yük kontrolü ve yukarı işaret sinyalinin, önceden başka bir kabine verilmediği kontrol edilir. Birinci kabin daha yakın konumda ise birinci kabinin tam yük kontrolü ve yukarı işaret sinyali kontrol edildikten sonra birinci kabine görev verilir. Burada dikkat edilirse yukarı çağrı bilgisinin kabine buton bilgisi olarak verilmesidir. Görevi alan kabin daha sonra uğradığı katlardan dolayı tam yüke ulaşsa bile çağrı yapılan kata uğramak zorundadır.

Üçüncü konum ise ikinci kabinin hareketli ve birinci kabin hareketsiz olduğu konumda, yukarı çağrı işlemi yapılırsa; bu konumda öncelikle kontrol edilmesi gereken ikinci kabinin çalışma şeklidir. Eğer ikinci kabinin kat seviyesi, çağrı yapılan kattan daha aşağıda ve tam yük halinde değilse görev ikinci kabine verilmiş olur. Yalnız burada dikkat edilecek konum ikinci kabine çağrı bilgisi kabin kat bilgisi olarak değil kattan geçerken, kat çağrısını değerlendirerek durmasıdır. İkinci kabin görevi alacak durumda değilse birinci kabin için konum kontrolüne devam edilir. Birinci kabin tam yük konumunda değil ve yukarı çağırma bilgisi başka bir kabine verilmemişse görev birinci kabine verilir.

Dördüncü konum ise birinci kabinin hareketli ve ikinci kabin hareketsiz olduğu konumda yukarı çağrı işlemi yapılırsa; bu konumda öncelikle kontrol edilmesi gereken birin-

ci kabinin çalışma şeklidir. Birinci kabinin kat seviyesi çağrı yapılan kattan daha aşağıda ve tam yük halinde değilse görev birinci kabine verilmiş olur. Burada dikkat edilecek konum birinci kabine çağrı bilgisi kabin kat bilgisi olarak değil kattan geçerken, kat çağrısını değerlendirerek durmasıdır. Birinci kabin görevi alacak durumda değilse ikinci kabin için konum kontrolüne devam edilir. İkinci kabin tam yük konumunda değil ve yukarı çağırma bilgisi başka bir kabine verilmemişse görev ikinci kabine verilir.

Bu procedure'de dikkat edilmesi gereken nokta; Çağrı bilgisinin direkt olarak uygun olan kabine verilme işlemi, kabin çağrı bilgisinin, kabin kat butonu bilgisi olarak verilmesidir. Programın bu bölümü dahada geliştirilebilir.

7.2.31. Procedure Yukarıg_1

Bu procedure'de, simülasyon bölümünde, karar verilen yukarı veya aşağı olarak, kabin hareketlerinin ekranda görüntülenmesini sağlayan tasarımlar yapılmıştır. Grafik ekranda, kabin hareketini oluşturan grafik düzenlemesi putimage komutu kullanılarak yapılıyor. Yalnız aynı koordinatlara exor modunda ikinci bir putimage komutu yapılırsa çizilen şekli bu defa siler. Burada da bu mantık kullanılmıştır. Programın simülasyon bölümünde kkb1 değeri değiştirilmişse, yani kabin hareket ettirilmişse, önceden çizilmiş olan hareketli kapı görüntüsünün üzerine bir defa daha putimage komutu uygulanır ve önceki çizimi siler. kkb1 değeri arttırılıp veya azaltılarak putimage komutu tekrar kullanılır.

Bu procedure'de öncelikle kkb1 ve kkb1 değişkenleri arasında matematiksel bir bağıntı oluşturulmuştur. Simülasyon bölümünde kat seviyeleri arasındaki fark 5 satırdır. Grafik ekranda ise kat seviyeleri arasındaki fark 90 pixel-dir. Bu iki değişken arasındaki bağıntı $kkb1=(kkb1*18)-33$ şeklinde çıkarılmıştır. Yukarıdaki matematiksel bağıntıdan faydalanarak, iki değişken karşılıklı olarak kontrol edil-

mektedir. kkb1 deęişkeni birinci kabinin yukarı doğru hareketini gerektirecek şekilde deęişmişse, kkbgl değeri ile en son putimage yapılmış çizime, bir defa daha putimage komutu uygulanarak çizim silinir. Sonra kkbgl değeri 3 azaltılır. Ve tekrar putimage komutu uygulanır. Böylece grafik ekranda kapı 3 pixel yukarı hareket etmiş gibi görünür. Yalnız burada kkb1 deęişkeninin 1 değer deęişmesine karşılık kkbgl 18 pixel hareket edebilmesi için yukarıda belirtilen matematiksel eşitlik sağlanıncaya kadar repeat until döngüsü içerisinde çizim yapılır. Bu döngü içerisinde iken ikinci kabinin hareketide sözkonusudur. Bunun için birinci kabinin 18 pixellik hareketi boyunca ikinci kabininde hareket etmesi gerekir. Bunun için birinci kabinin çizimi sırasında ikinci kabinin yukarı veya aşağı doğru hareketinin olup olmadığının kontrolüde yapılıyor. Birinci kabin 3 pixel yukarı doğru hareket ettiği sırada ikinci kabinin yukarı doğru hareketi varsa kkbgl2 deęişkeni, 3 azaltılarak ve putimage komutundan yararlanılarak ikinci kabinin yukarı doğru hareketi sağlanır. Eğer aşağı doğru hareketi varsa kkbgl2 değeri 3 arttırılarak ve putimage komutu yardımıyla ikinci kabinin yukarı doğru hareketi gerçekleştirilir.

Birinci kabinin aşağı doğru hareketi varsa yine repeat until döngüsü içerisinde birinci kabinin 18 pixellik hareketi gerçekleştirilir. Bu hareketi gerçekleştirilmesi sırasında, ikinci kabininde aşağı veya yukarı hareketinin olup olmadığı kontrolü yapılmaktadır.

Burada dikkat edilmesi gereken nokta birinci kabinin 18 pixellik hareketi boyunca repeat until komutundan dolayı ana programa ulaşamamasıdır. Bu sırada ikinci kabinin kapı açılması durumu var ise n2 deęişkeninin arttırılması için bu repeat until döngüsü içerisinde üç tane kapıdurum procedure'ü kullanılmıştır. Dikkat edilmesi gereken bir noktada kkbgl'in ne kadar miktarda arttırılacağı veya azaltılacağıdır. Burada kkb1'in 1 deęişimine karşılık kkbgl'de 18 pixel deęişim olduğunu biliyoruz. Bu nedenle kkbgl

değişkenini 18'in bölenleri (1,2,3,6,9,18) kadar arttırılabilir veya azaltılabilir. Böylece kabinlerin hızında ayarlanabileceği görülür.

7.2.32. Procedure Yukarıg_2

Bu procedure yukarıg_1 procedure'ünde birinci kabin için yapılan işlemlerin ikinci kabin için yapılmasını sağlar.

7.2.33. Procedure Renkdol

Bu procedure'de ekranın renk tasarımları yapılmıştır. Bu işlemi yaparken setfillstyle ve floodfill komutları kullanılmıştır. Setfillstyle komutuyla hangi desen ve renkte doldurulacağı belirtilir. Floodfill komutuyla koordinat ve sınır rengi belirtilir.

mah1 değişkeni kontrol edilerek, birinci kabinin kat göstergesinin altındaki aşağı doğru ok işaretinin renk doldurulması yapılır. myh1 değişkeni kontrol edilerek, birinci kabinin kat göstergesinin üstündeki yukarı doğru ok işaretinin renk doldurulması yapılır.

mah2 değişkeni kontrol edilerek, ikinci kabinin kat göstergesinin altındaki aşağı doğru ok işaretinin renk doldurulması yapılır. myh2 değişkeni kontrol edilerek, ikinci kabinin kat göstergesinin üstündeki yukarı doğru ok işaretinin renk doldurulması yapılır.

kb1[I] dizisinin elemanları tek tek kontrol edilerek birinci kabin kat butonlarının renk doldurulması yapılır. Burada üçüncü bir konum söz konusudur. Eğer kabin kat seviyesinde bulunuyorsa üçüncü bir renkle doldurulur ve birinci kabinin hangi katta olduğu birinci kabin butonlarından anlaşılabilir.

kb2[I] dizisinin elemanları tek tek kontrol edilerek

ikinci kabin kat butonlarının renk doldurulması yapılır. kb1[I] dizisinin özelliklerini taşır.

yi[I] dizisinin elemanları kontrol edilerek yukarı çağrı yapılmış ise yukarı çağrı butonunun lambası, bir renkle doldurulur. Eğer yukarı çağrı butonuna basılmamışsa normal renkte kalır.

ai[I] dizisinin elemanları kontrol edilerek aşağı çağrı yapılmış ise aşağı çağrı butonunun lambası, bir renkle doldurulur. Eğer aşağı çağrı butonuna basılmamışsa normal renkte kalır.

7.2.34. Procedure Kapiac

Bu procedure'de kabin herhangi bir kat seviyesinde iken bulunduğu kattan, bir çağrı yapıldığında kabin kapısının açılması sağlanır. Aşağı çağrı ve yukarı çağrı kat butonlarının sayısı 4 olduğu için 1'den 4'e bir for to do döngüsü yapılır. Sonra birinci kabinin hareket etmediği kontrol edilir ve birinci kabin çağrının bulunduğu katta ise kkap1 false yapılır. Aynı işlem yukarı çağrı butonu içinde yapılır. Sonra ikinci kabinin hareket etmediği kontrol edilir ve ikinci kabin çağrının bulunduğu katta ise kkap2 false yapılır. Aynı işlem yukarı çağrı butonu içinde yapılır. Eğer iki kabinde aynı katta bulunuyorsa birinci kabin kapısının açılması için ikinci kabinin kontrolü sırasında birinci kabininde çağrı yapılan katta bulunup bulunmadığı kontrol edilir.

7.2.35. Procedure Aksam

Bu procedure'de, özel şartlar gözönünde bulundurularak tasarımlar yapılmıştır. Bir binada akşam üzeri, binaya dönen ev sahiplerinin belirli bir süre için devamlı olarak zemin-den üst katlara doğru bir nakil işlemi vardır. Binaya dönen yolcuların, zemin katta bekleme süresini, minimuma indirmek

için kabinlerin hareketleri sona erdiğinde yani kabin hareketi bittiğinde kabinin zemin kata gitmesi sağlanır. Böylece zemin katta sistemin cevaplama süresi azalacaktır.

7.2.36. Procedure Sabah

Bu procedure'de, özel şartlar gözönünde bulundurularak tasarımlar yapılmıştır. Bir binada sabahleyin, binadan ayrılan ev sahiplerinin belirli bir süre için devamlı olarak asansörün yukarıdan aşağı doğru kabinlerin hareketi söz konusudur. Binadan ayrılan yolcuların, katlarda bekleme süresini, minimuma indirmek için kabinlerin hareketleri sona erdiğinde yani kabin hareketi bittiğinde kabinin belirtilen üst katlara gönderilmesi gerekir. Böylece katlarda aşağı inmek için bekleyen yolcuların bekleme süresi optimize edilmiş olur.

8. SONUÇ

Asansör sistemlerinin Bilgisayar ile kontrolü Türkiye'de çok yeni bir konudur. Şu ana kadar yalnızca teorik çalışmalara konu olmuştur. Bunun nedeni şu an kullanılan röleli devrelerin, bilgisayarla kontrol sistemine göre daha ucuz olmasıdır.

Bu tez çalışmasında, kontrolü yapılan asansör sistemi iki kabinli, 5 katlı ve toplamalı asansör sistemidir. Bu sistemde kullanılan değişkenler asansör sisteminin elemanları incelenerek ortaya çıkarılmıştır. Bu sistemin bir özelliğide binanın, iş binası, konut veya otel olması durumuna göre yapılacak sistem değişikliklerine açık olmasıdır. Bu sistem değişikliklerinin, bilgisayar ile kontrolü yapılmaktadır. Herhangi bir özel çalışma durumu, bir alt procedure ile programa eklenebilmektedir. Yapılan çalışmayı donanım olarak uygulayabilecek bir ortam bulunmadığından sistemin sadece simülasyonu yapılmıştır. Input ve output işlemlerinin yapılması durumunda asansör sistemini kumanda edebilir. Programa istenildiğinde özel durumlar, software değişiklikleriyle, sistemde bir değişiklik gerekmeksizin uygulanabilir. Röleli devrelerde ise özel değişiklikler gerektiğinde, sistemin kumandasının tamamen değişmesi gerekecektir. Bilgisayar kontrolü yapılan sistemler genelde çok kabinli asansör sistemleridir. Çünkü çok kabinli asansör sistemlerinin işletim prensipleri çok karmaşıktır. Ama yapılan yazılımla bu karmaşıklığı çözümlenmektedir.

Yapılan simülasyon programının asansör sistemiyle haberleşmesi üzerine düşünceler açıklanmıştır. Uygulama yapılmamıştır. Kullanılan değişkenler, asansör sistemindeki değişkenlere uygun olduğu için input-output bağlantısı yapıldığında, simülasyon programı, sistemle uyum içerisinde çalışacaktır. Yazılımı asansör sistemine uygulayabilecek ortam hazırlanarak, programın sistemle haberleşmesi üzerine araştırmalar yapılabilir.

KAYNAKLAR

- 1- Birinci Ulusal Asansör Senpozyumu, Yıldız Üniversitesi Kocaeli Müh. Fak. 1992
- 2- Teknik Toplantılar "Asansörler : Tasarım, Uygulama ve Sorunlar", İ.T.Ü. Elektrik Elektronik Müh. Bölümü 17-Mayıs-1993
- 3- "Röleli Lojik Devre İşlemlerinin Mikro Bilgisayar ile gerçekleşmesi", Müh.Seyfi SARISEL, Yıldız Üniversitesi Yüksek Lisans Tezi 1986
- 4- Asansör Tesisleri, Georges TEXIER 1972
- 5- Asansör Projeleri ve Asansör Sistemi ile ilgili notlar, Asmelsan Asansörleri
- 6- Elevator, May/June 1990 İtalya
- 7- "Asansörlerin PLC ile kontrolü ", Tamer Esen , Marmara Üniversitesi Teknik Eğitim Fakültesi Elektrik Bölümü Bitirme Tezi 1992

OZGEÇMİŞ

29-01-1968 tarihinde Gaziantep'de doğdu. İlk ve Orta Öğrenimini Gaziantep'te yaptı. Gaziantep Lisesi'nden mezun olarak 1986 yılında Yıldız Üniversitesi Kocaeli Mühendislik Fakültesi Elektronik ve Haberleşme Mühendisliği Bölümüne girdi. 1990 yılında mezun olarak Yıldız Üniversitesi'nde Yüksek Lisans Eğitimine başladı.

Yüksek Lisans Eğitimine devam ederken 1,5 yıl Bulut Mak.Asansör San. Tic. Ltd. Şirketinde çalıştı. 15-06-1992 tarihinde Marmara Üniversitesi Teknik Bilimler M.Y.Okulu'na Araştırma Görevlisi olarak girdi. Halen bu üniversitede görevine devam etmektedir.



EK-1

Uses

Crt, graph, dos;

type

xycoord=record

x,y:integer;p:byte;end;

kt=Array[1..5] of Boolean;

ky=Array[1..4] of Boolean;

ka=Array[2..5] of Boolean;

const

kapi:array[1..17] of

xycoord=((x:1;y:1;p:0),(x:110;y:1;p:1),(x:110;y:75;p:1),

(x:1;y:75;p:1),(x:1;y:1;p:1),(x:71;y:24;p:0),

(x:81;y:24;p:1),(x:81;y:55;p:1),(x:71;y:55;p:1),

(x:71;y:24;p:1),(x:30;y:24;p:0),

(x:40;y:24;p:1),(x:40;y:55;p:1),(x:30;y:55;p:1),

(x:30;y:24;p:1),(x:55;y:1;p:0),(x:55;y:75;p:1));

kabin:array[1..10] of

xycoord=((x:0;y:279;p:0),(x:190;y:279;p:1),

(x:190;y:479;p:1),(x:0;y:479;p:1),(x:0;y:279;p:1),

(x:185;y:284;p:0),(x:185;y:474;p:1),(x:161;y:474;p:1),

(x:161;y:284;p:1),(x:185;y:284;p:1));

fon:array[1..17] of

xycoord=((x:0;y:0;p:0),(x:639;y:0;p:1),(x:639;y:479;p:1),

(x:0;y:479;p:1),(x:0;y:0;p:1),(x:190;y:0;p:0),

(x:190;y:479;p:1),(x:195;y:479;p:0),(x:195;y:0;p:1),

(x:304;y:0;p:0),(x:304;y:479;p:1),(x:334;y:0;p:0),

(x:334;y:479;p:1),(x:443;y:0;p:0),(x:443;y:479;p:1),

(x:448;y:0;p:0),(x:448;y:479;p:1));

kapi1:array[1..19] of

xycoord=((x:1;y:1;p:0),(x:110;y:1;p:1),(x:110;y:75;p:1),

(x:1;y:75;p:1),(x:1;y:1;p:1),(x:86;y:24;p:0),

(x:96;y:24;p:1),(x:96;y:55;p:1),(x:86;y:55;p:1),

(x:86;y:24;p:1),(x:15;y:24;p:0),

(x:25;y:24;p:1),(x:25;y:55;p:1),(x:15;y:55;p:1),

(x:15;y:24;p:1),(x:40;y:1;p:0),(x:40;y:75;p:1),

(x:70;y:1;p:0),(x:70;y:75;p:1));

kapi2:array[1..19] of

xycoord=((x:1;y:1;p:0),(x:110;y:1;p:1),(x:110;y:75;p:1),

(x:1;y:75;p:1),(x:1;y:1;p:1),(x:101;y:24;p:0),

(x:110;y:24;p:1),(x:110;y:55;p:1),(x:101;y:55;p:1),

(x:101;y:24;p:1),(x:1;y:24;p:0),

(x:10;y:24;p:1),(x:10;y:55;p:1),(x:1;y:55;p:1),

(x:1;y:24;p:1),(x:25;y:1;p:0),(x:25;y:75;p:1),

(x:85;y:1;p:0),(x:85;y:75;p:1));

kapi3:array[1..9] of

xycoord=((x:1;y:1;p:0),(x:110;y:1;p:1),(x:110;y:75;p:1),

(x:1;y:75;p:1),(x:1;y:1;p:1),(x:10;y:1;p:0),

(x:10;y:75;p:1),(x:100;y:1;p:0),(x:100;y:75;p:1));

kapi4:array[1..5] of

xycoord=((x:1;y:1;p:0),(x:110;y:1;p:1),(x:110;y:75;p:1),

(x:1;y:75;p:1),(x:1;y:1;p:1));

```

x_offset=10;
y_offset=10;
Katb5=3;
Katb4=8;
Katb3=13;
Katb2=18;
Katb1=23;
Kisisay=6;
Kats=5;
var
kkg1,kkg2,errorcode,graphmode,graphdriver:integer;
size,getmaxcolor,pausetime:word;
k1,k2,k3,k4,t1,y,u,y1,y2:pointer;s:string[2];
ch:char;
kb1,kb2:kt;
yi,yis:ky;
ai,ais:ka;
kabkat1,kabkat2,mdur1,mdur2,kkapi1,kkapi2,kkapig1,kkapig2,k
bin1,kbin2,kin1,kin2:Boolean;
tyi1,tyi2,ayi1,ayi2,kbdur1,kbdur2,myh1,myh2,mah1,mah2:Boole
an;
t,k,kbek1,kbek2,kis1,kis2,kkb1,kkb2,i,j,m1,m2,n1,n2:integer
procedure kisisay1;
begin
if (kbin1=true) and (kis1<9) then
begin
kis1:=kis1+1;putimage(115,290,y^,normalput);
end;
if (kin1=true) and (kis1>0) then
begin
kis1:=kis1-1;putimage(115,290,y^,normalput);
end;
kbin1:=false;kin1:=false;
str(kis1,s);
outtextxy(125,300,s);
end;
procedure katsay1;
begin
if kabkat1=false then
begin
case kkg1 of
381 : begin
putimage(115,390,y^,normalput);
outtextxy(123,401,'Z');
end;
291 : begin
putimage(115,390,y^,normalput);
outtextxy(125,400,'1');
end;
201 : begin
putimage(115,390,y^,normalput);
outtextxy(125,400,'2');
end;
111 : begin

```

```

putimage(115,390,y^,normalput);
outtextxy(125,400,'3');
end;
21 : begin
putimage(115,390,y^,normalput);
outtextxy(125,400,'4');
end;
end;
if (kbb1*18-33)=kbbg1 then
kabkat1:=true;
end;
if (mah1=true) or (myh1=true) then
kabkat1:=false;
end;
procedure katsayi2;
begin
if kabkat2=false then
begin
case kbbg2 of
381 : begin
putimage(565,390,y^,normalput);
outtextxy(573,401,'Z');
end;
291 : begin
putimage(565,390,y^,normalput);
outtextxy(575,400,'1');
end;
201 : begin
putimage(565,390,y^,normalput);
outtextxy(575,400,'2');
end;
111 : begin
putimage(565,390,y^,normalput);
outtextxy(575,400,'3');
end;
21 : begin
putimage(565,390,y^,normalput);
outtextxy(575,400,'4');
end;
end;
if (kbb2*18-33)=kbbg2 then
kabkat2:=true;
end;
if (mah2=true) or (myh2=true) then
kabkat2:=false;
end;
procedure kisisayi2;
begin
if (kbin2=true) and (kis2<9) then
begin
kis2:=kis2+1;putimage(565,290,y^,normalput);
end;
if (kin2=true) and (kis2>0) then
begin

```

```

kis2:=kis2-1;putimage(565,290,y^,normalput);
end;
kbin2:=false;kin2:=false;
str(kis2,s);
outtextxy(575,300,s);
end;
procedure tamyuk; {tam yuk kontrolu yapiliyor.}
begin
tyi1:=false;
if kisl=kisisay then
tyi1:=true;
tyi2:=false;
if kis2=kisisay then
tyi2:=true;
end; {tam yuk procedure sonu}
procedure asiriy; {asiri yuk kontrolu yapiliyor}
begin
ayi1:=false;
if kisl>kisisay then
ayi1:=true;
ayi2:=false;
if kis2>kisisay then
ayi2:=true;
if (ayi1=true) and (kkapi1=true) then
begin
sound(220);delay(50);nosound;kkapi1:=false;
end;
if (ayi2=true) and (kkapi2=true) then
begin
sound(220);delay(50);nosound;kkapi2:=false;
end;
end; {asiriy procedure sonu}
procedure tusoku;{tus okuma islemi yapiliyor}
begin
for I:=1 to 5 do
begin
delay(1);
if keypressed then
begin
ch:=readkey;
if ch=#0 then ch:=readkey;
case ch of
#59:KB1[1]:=true;
#60:KB1[2]:=true;
#61:KB1[3]:=true;
#62:KB1[4]:=true;
#63:KB1[5]:=true;
#84:KB2[1]:=true;
#85:KB2[2]:=true;
#86:KB2[3]:=true;
#87:KB2[4]:=true;
#88:KB2[5]:=true;
#95:AI[2]:=true;
#96:AI[3]:=true;

```

```

#97:AI[4]:=true;
#98:AI[5]:=true;
#104:YI[1]:=true;
#105:YI[2]:=true;
#106:YI[3]:=true;
#107:YI[4]:=true;
`Z`,`z`:kbdur1:=true;
`X`,`x`:kbdur2:=true;
`M`,`m`:kkapi2:=false;
`N`,`n`:kkapi1:=false;
#43:kbin1:=true;
#45:kin1:=true;
#42:kbin2:=true;
#47:kin2:=true;
end;
end;
end;
end;{tusoku procedure sonu}
procedure fonciz;
begin
for i:=1 to 17 do
begin
if fon[i].p=0 then moveto(fon[i].x,fon[i].y)
else lineto(fon[i].x,fon[i].y);
end;
end;
procedure kapticiz;
begin
for i:=1 to 17 do

begin
if kapi[i].p=0 then moveto(kapi[i].x,kapi[i].y)
else lineto(kapi[i].x,kapi[i].y);
end;
end;
procedure kapticiz1;
begin
for i:=1 to 19 do
begin
if kapi1[i].p=0 then moveto(kapi1[i].x,kapi1[i].y)
else lineto(kapi1[i].x,kapi1[i].y);
end;
end;
procedure kapticiz2;
begin
for i:=1 to 19 do
begin
if kapi2[i].p=0 then moveto(kapi2[i].x,kapi2[i].y)
else lineto(kapi2[i].x,kapi2[i].y);
end;
end;
procedure kapticiz3;
begin
for i:=1 to 9 do

```

```

begin
if kapi3[i].p=0 then moveto(kapi3[i].x,kapi3[i].y)
else lineto(kapi3[i].x,kapi3[i].y);
end;
end;
procedure kapticiz4;
begin
for i:=1 to 5 do
begin
if kapi4[i].p=0 then moveto(kapi4[i].x,kapi4[i].y)
else lineto(kapi4[i].x,kapi4[i].y);
end;
end;
procedure kabinciz;
begin
for i:=1 to 10 do
begin
if kabin[i].p=0 then moveto(kabin[i].x,kabin[i].y)
else lineto(kabin[i].x,kabin[i].y);
end;
for i:=1 to 7 do
begin
circle(173,(275+(i*26)),10);
setfillstyle(12,10);
floodfill(173,(275+(i*26)),10);
end;
setfillstyle(1,5);
floodfill(173,290,10);
setfillstyle(1,4);
floodfill(153,290,10);
setcolor(10);
outtextxy(166,(299),'al');
outtextxy(170,(325),'4');
outtextxy(170,(352),'3');
outtextxy(170,(377),'2');
outtextxy(170,(403),'1');
outtextxy(170,(429),'Z');
outtextxy(166,(455),'st');
setcolor(10);
end;
procedure basla;
begin
setcolor(7);
fonciz;
setfillstyle(1,9);
floodfill(197,478,7);
setfillstyle(1,9);
floodfill(440,478,7);
setfillstyle(9,1);
floodfill(320,478,7);
setfillstyle(1,9);
setcolor(11);
kapticiz;
setcolor(10);

```

```

kabinciz;
size:=imagesize(0,279,190,479);           {kabin ciziliyor}
getmem(u,size);
getimage(0,279,190,479,u^);
Putimage(448,279,u^,normalput);
size:=imagesize(1,1,75,110); {normal kat kapilari
ciziliyor}
getmem(u,size);
getimage(1,1,110,75,u^);
for i:=0 to 4 do
begin
putimage(334,i*90+21,u^,normalput);
putimage(195,i*90+21,u^,normalput);
end;
moveto(50,168);lineto(52,168);           {ok ciziliyor}
lineto(52,160);lineto(54,160);
lineto(50,160);lineto(50,168);
circle(51,161,10);
size:=imagesize(40,150,62,172);
getmem(t1,size);
getimage(40,150,62,172,k1^);
putimage(40,150,t1^,xorput);
moveto(52,154);lineto(50,154);           {ok ciziliyor}
lineto(50,162);lineto(48,162);
lineto(51,167);lineto(54,162);
lineto(52,162);lineto(52,154);
circle(51,161,10);
size:=imagesize(40,150,62,172);
getmem(y,size);
getimage(40,150,62,172,y^);
putimage(40,150,y^,xorput);
for i:=1 to 4 do
begin
{kat tuslarin cemberi ciziliyor}
putimage(308,i*90+33,t1^,normalput);
Putimage(308,i*90-25,y^,normalput);
end;
moveto(1,100);
lineto(1,128);                           {sayi cercevesi ciziliyor}
lineto(1,100);
size:=imagesize(1,1,19,28);
getmem(y,size);
getimage(1,100,19,128,y^);
putimage(115,290,y^,normalput);
putimage(565,290,y^,normalput);
putimage(1,100,y^,xorput);
putimage(115,390,y^,normalput);
putimage(565,390,y^,normalput);
moveto(10,100);
lineto(19,112);
lineto(10,100);
size:=imagesize(1,1,20,13);
getmem(y1,size);
getimage(1,100,20,113,y1^);
putimage(1,100,y1^,xorput);

```

```

putimage(135,390,y1^,normalput);
putimage(585,390,y1^,normalput);
moveto(10,128);
lineto(19,116);
lineto(10,128);
size:=imagesize(1,1,20,14);
getmem(y2,size);
getimage(1,115,20,128,y2^);
putimage(1,115,y2^,xorput);
putimage(135,405,y2^,normalput);
putimage(585,405,y2^,normalput);
kapticiz1;          { hareketli kapi ciziliyor}
setfillstyle(4,red);
floodfill(2,2,red);
floodfill(109,2,red);
setfillstyle(4,9);
floodfill(55,2,red);
size:=imagesize(1,1,110,75);
getmem(k1,size);
getimage(1,1,110,75,k1^);
putimage(1,1,k1^,xorput);
kapticiz2;          { hareketli kapi ciziliyor}
setfillstyle(4,red);
floodfill(2,2,red);
floodfill(109,2,red);
setfillstyle(4,9);
floodfill(55,2,red);
size:=imagesize(1,1,110,75);
getmem(k2,size);
getimage(1,1,110,75,k2^);
putimage(1,1,k2^,xorput);
kapticiz3;          { hareketli kapi ciziliyor}
setfillstyle(4,red);
floodfill(2,2,red);
floodfill(109,2,red);
setfillstyle(4,9);
floodfill(55,2,red);
size:=imagesize(1,1,110,75);
getmem(k3,size);
getimage(1,1,110,75,k3^);
putimage(1,1,k3^,xorput);
kapticiz4;          { hareketli kapi ciziliyor}
setfillstyle(4,9);
floodfill(55,2,red);
size:=imagesize(1,1,110,75);
getmem(k4,size);
getimage(1,1,110,75,k4^);
putimage(1,1,k4^,xorput);
moveto(2,2);
lineto(189,2);
lineto(2,2);
setfillstyle(9,1);
floodfill(3,3,red);
moveto(449,2);

```

```

lineto(639,2);
lineto(449,2);
setfillstyle(9,1);
floodfill(450,3,red);
end;
procedure reset;{tum tus resetlemeleri yapiliyor}
begin
m1:=0;
m2:=0;
n1:=0;
n2:=0;
kkb2:=23;
kkb1:=23;
kbek1:=20;
kbek2:=20;
t:=20;
kkbg1:=381;
kkbg2:=381;kabkat1:=false;kabkat2:=false;
for I:=1 to 5 do
begin
kbl[I]:=false;
kbdur1:=false;
kis1:=0;
kis2:=0;
kb2[i]:=false;
kbdur2:=false;
kkapi1:=true;kkapig1:=true;
kkapi2:=true;kkapig2:=true;
ai[i]:=false;
yi[i]:=false;
ais[i]:=false;
yis[i]:=false;
myh1:=false;
myh2:=false;
mdur1:=false;
mdur2:=false;
mah1:=false;
mah2:=false;
end;
end;
{reset procedure sonu}
procedure kapidurum;
begin
if (kkb1=3) or (kkb1=8) or (kkb1=13) or (kkb1=18) or
(kkb1=23) then
begin
if (kkb1*18-33)=kkbg1 then
if kkapi1=false then
begin
kisisay1;
if (n1=0) and (kkapi1=false) then
kkapig1:=kkapi1;
if kkapig1=false then
begin
n1:=n1+1;

```

```

case n1 of
10: begin
putimage(195,kkbg1,u^,xorput);
putimage(195,kkbg1,k1^,xorput);
end;
20: begin
putimage(195,kkbg1,k1^,xorput);
putimage(195,kkbg1,k2^,xorput);
end;
30: begin
putimage(195,kkbg1,k2^,xorput);
putimage(195,kkbg1,k3^,xorput);
end;
40: begin
putimage(195,kkbg1,k3^,xorput);
putimage(195,kkbg1,k4^,xorput);
end;
100: begin
n1:=41;kkapig1:=true;
end;
end;
end
else
begin
n1:=n1-1;
case n1 of
10: begin
putimage(195,kkbg1,k^,xorput);
putimage(195,kkbg1,k2^,xorput);
end;
20: begin
putimage(195,kkbg1,k2^,xorput);
putimage(195,kkbg1,k1^,xorput);
end;
30: begin
putimage(195,kkbg1,k3^,xorput);
putimage(195,kkbg1,k2^,xorput);
end;
40: begin
putimage(195,kkbg1,k4^,xorput);
putimage(195,kkbg1,k3^,xorput);
end;
end;
end;
if (n1=0) and (kkapig1=true) then
kkapi1:=kkapig1;
end; end else kkapi1:=true;
if (kkb2=3) or (kkb2=8) or (kkb2=13) or (kkb2=18) or
(kkb2=23) then
begin
if (kkb2*18-33)=kkbg2 then
if kkapi2=false then
begin
kisisayi2;

```

```

if (n2=0) and (kkapi2=false) then
kkapig2:=kkapi2;
if kkapig2=false then
begin
n2:=n2+1;
case n2 of
10: begin
putimage(334,kkbg2,u^,xorput);
putimage(334,kkbg2,k1^,xorput);
end;
20: begin
putimage(334,kkbg2,k1^,xorput);
putimage(334,kkbg2,k2^,xorput);
end;
30: begin
putimage(334,kkbg2,k2^,xorput);
putimage(334,kkbg2,k3^,xorput);
end;
40: begin
putimage(334,kkbg2,k3^,xorput);
putimage(334,kkbg2,k4^,xorput);
end;
100: begin
n2:=41;kkapig2:=true;
end;
end;
end
else
begin
n2:=n2-1;
case n2 of
10: begin
putimage(334,kkbg2,k1^,xorput);
putimage(334,kkbg2,u^,xorput);
end;
20: begin
putimage(334,kkbg2,k2^,xorput);
putimage(334,kkbg2,k1^,xorput);
end;
30: begin
putimage(334,kkbg2,k3^,xorput);
putimage(334,kkbg2,k2^,xorput);
end;
40: begin
putimage(334,kkbg2,k4^,xorput);
putimage(334,kkbg2,k3^,xorput);
end;
end;
end;
if (n2=0) and (kkapig2=true) then
kkapi2:=kkapig2;
end; end else kkapi2:=true;
end;
procedure kabink_1;           {1.kabinin kontrolu yapiliyor}

```

```

begin
kbe1:=20;
t:=20;
for i:=1 to 5 do
begin
if kb1[i] then
begin
t:=abs(kkb1-(5*(5-i)-2));
if (5-i)<>(kkb1+2) div 5 then
if kbe1>t then kbe1:=t;
end;
end;
j:=(kkb1+kbe1+2) div 5;
if j<6 then
if j>0 then
if myh1=false then
if kb1[5-j] then
begin
mah1:=true;
end;
if (kkb1-kbe1)>0 then
j:=(abs(kkb1-kbe1)+2) div 5;
if j>0 then
if j<5 then
if mah1=false then
if kb1[(5-j)] then
begin
myh1:=true;
end;
end;{kabink_1 procedure sonu}
procedure kabink_2;{2.kabinin kontrolu yapiliyor}
begin
kbe2:=20;
t:=20;
for i:=1 to 5 do
begin
if kb2[i] then
begin
t:=abs(kkb2-(5*(5-i)-2));
if (6-i)<>(kkb2+2) div 5 then
if kbe2>t then kbe2:=t;
end;
end;
j:=(kkb2+kbe2+2) div 5;
if j<6 then
if j>0 then
if myh2=false then
if kb2[6-j] then
begin
mah2:=true;
end;
if (kkb2-kbe2)>0 then
j:=(abs(kkb2-kbe2)+2) div 5;
if j<5 then

```

```

if j>0 then
if mah2=false then
if kb2[(6-j)] then
begin
myh2:=true;
end;
end;{kabink_2 procedure sonu}
procedure katdur_1;
begin
case kkb1 of
katb5: if (kb1[5]=true) or (((ai[5]=true) and (mah1=true))
and (tyi1=false)) then
kkapi1:=false;
katb4: if (kb1[4]=true) or (((ai[4]=true) and (mah1=true))
or
((yi[4]=true) and (myh1=true))) and (tyi1=false)) then
kkapi1:=false;
katb3: if (kb1[3]=true) or (((ai[3]=true) and (mah1=true))
or
((yi[3]=true) and (myh1=true))) and (tyi1=false)) then
kkapi1:=false;
katb2: if (kb1[2]=true) or (((ai[2]=true) and (mah1=true))
or
((yi[2]=true) and (myh1=true))) and (tyi1=false)) then
kkapi1:=false;
katb1: if (kb1[1]=true) or ((yi[1]=true) and (myh1=true))
and (tyi1=false)) then
kkapi1:=false;
end;
end;
procedure katdur_2;
begin
case kkb2 of
katb5: if (kb2[5]=true) or (((ai[5]=true) and (mah2=true))
and (tyi2=false)) then
kkapi2:=false;
katb4: if (kb2[4]=true) or (((ai[4]=true) and (mah2=true))
or
((yi[4]=true) and (myh2=true))) and (tyi2=false)) then
kkapi2:=false;
katb3: if (kb2[3]=true) or (((ai[3]=true) and (mah2=true))
or
((yi[3]=true) and (myh2=true))) and (tyi2=false)) then
kkapi2:=false;
katb2: if (kb2[2]=true) or (((ai[2]=true) and (mah2=true))
or
((yi[2]=true) and (myh2=true))) and (tyi2=false)) then
kkapi2:=false;
katb1: if (kb2[1]=true) or ((yi[1]=true) and ((myh2=true)
and (tyi2=false))) then
kkapi2:=false;
end;
end;
procedure yukari_1;{1.kabinin yukari hareket ciziliyor ve

```

```

resetleme yapiliyor}
begin
kkb1:=kkb1-1;
case kkb1 of
3: begin
katdur_1;kb1[5]:=false;ais[5]:=false;
end;
8: begin
katdur_1;kb1[4]:=false;ais[4]:=false;ysis[4]:=false;
end;
13: begin
katdur_1;kb1[3]:=false;ysis[3]:=false;ais[3]:=false;
end;
18: begin
katdur_1;kb1[2]:=false;ais[2]:=false;ysis[2]:=false;
end;
23: begin
katdur_1;kb1[1]:=false;ysis[1]:=false;
end;
end;
end;
                                {yukari_1 procedure sonu}
procedure yukari_2;{2.kabinin yukari hareket ciziliyor ve
resetleme yapiliyor}
begin
kkb2:=kkb2-1;
case kkb2 of
3: begin
katdur_2;kb2[5]:=false;ais[5]:=false;
end;
8: begin
katdur_2;kb2[4]:=false;ais[4]:=false;ysis[4]:=false;
end;
13: begin
katdur_2;kb2[3]:=false;ais[3]:=false;ysis[3]:=false;
end;
18: begin
katdur_2;kb2[2]:=false;ais[2]:=false;ysis[2]:=false;
end;
23: begin
katdur_2;kb2[1]:=false;ysis[1]:=false;
end;
end;
end;
                                {yukari_2 procedure sonu}
procedure asagi_1;{1.kabinin asagi hareket ciziliyor ve
resetleme yapiliyor}
begin
kkb1:=kkb1+1;
case kkb1 of
3: begin
katdur_1;kb1[5]:=false;ais[5]:=false;
end;
8: begin
katdur_1;kb1[4]:=false;ais[4]:=false;ysis[4]:=false;
end;

```

```

13: begin
katdur_1;kb1[3]:=false;ysis[3]:=false;ais[3]:=false;
end;
18: begin
katdur_1;kb1[2]:=false;ais[2]:=false;ysis[2]:=false;
end;
23: begin
katdur_1;kb1[1]:=false;ysis[1]:=false;
end;
end;{asagi_1 procedure sonu}
procedure asagi_2;{2.kabinin asagi hareket ciziliyor ve
restleme yapiliyor}
begin
kkb2:=kkb2+1;
case kkb2 of
3: begin
katdur_2;kb2[5]:=false;
end;
8: begin
katdur_2;kb2[4]:=false;
end;
13: begin
katdur_2;kb2[3]:=false;
end;
18: begin
katdur_2;kb2[2]:=false;
end;
23: begin
katdur_2;kb2[1]:=false;
end;
end;{asagi_2 procedure sonu}
procedure hareket;{motor durumuna gore hareket yapiliyor}
begin
if (myh1=true) and (kkapi1=true) and (ayi1=false) then
begin
yukari_1;
end;
if (myh2=true) and (kkapi2=true) and (ayi2=false) and
(mdur2=false) then
begin
yukari_2;
end;
if (mah2=true) and (kkapi2=true) and (ayi2=false) and
(mdur2=false) then
begin
asagi_2;
end;
if (mah1=true) and (kkapi1=true) and (ayi1=false) and
(mdur1=false) then
begin
asagi_1;
end;

```

```

end;{hareket procedure sonu}
procedure dur_1;{1.kabinin motor resetlemesi yapiliyor}
begin
j:=0;
for i:=(6-(kkb1+2) div 5) to 5 do
if kb1[i]=false then
begin
j:=j+1;
end;
if j=(kkb1+2) div 5 then
myh1:=false;
j:=0;
for i:=((kkb1+2) div 5)+1 to 5 do
if kb1[6-i]=false then
begin
j:=j+1;
end;
if j=5-((kkb1+2) div 5) then
mah1:=false;
if kkb1<3 then
myh1:=false;
if kkb1>22 then
mah1:=false;
end;{dur_1 procedure sonu}
procedure dur_2;{2.kabinin motor resetlemesi yapiliyor}
begin
j:=0;
for i:=(6-(kkb2+2) div 5) to 5 do
if kb2[i]=false then
begin
j:=j+1;
end;
if j=(kkb2+2) div 5 then
myh2:=false;
j:=0;
for i:=((kkb2+2) div 5)+1 to 5 do
if kb2[6-i]=false then
begin
j:=j+1;
end;
if j=5-((kkb2+2) div 5) then
mah2:=false;
if kkb2<3 then
myh2:=false;
if kkb2>22 then
mah2:=false;
end;{dur_2 procedure sonu}
procedure asakont_1;{asagi isaretin kontrolu yapiliyor}
begin
if (mah1=false) and (myh1=false) then
begin
if (mah2=false) and (myh2=false) then
begin
for i:=1 to 4 do

```

```

begin
if ai[6-i]=true then
begin
k:=abs(kkb2-(5*i-2));
j:=abs(kkb1-(5*i-2));
if j>=k then
begin
if tyi2=false then
if ais[6-i]=false then
begin
kb2[6-i]:=true;ais[6-i]:=true;
end;
end
else
if tyi1=false then
if ais[6-i]=false then
begin
kb1[6-i]:=true;ais[6-i]:=true;
end;
end;
end;
end;
for i:=1 to 4 do
begin
if not((tyi2=false) and ((mah2=true) and (kkb2<5*i-2)))then
if tyi1=false then
if ai[6-i]=true then
if tyi1=false then
if ais[6-i]=false then
begin
kb1[6-i]:=true;ais[6-i]:=true;
end;
end;
end
else
if (mah2=false) and (myh2=false) then
begin
for i:=1 to 4 do
begin
if not((tyi1=false) and ((mah1=true) and (kkb1<5*i-2)))
then
if tyi2=false then
if ai[6-i]=true then
if tyi2=false then
if ais[6-i]=false then
begin
kb2[6-i]:=true;ais[6-i]:=true;
end;
end;
end;
end;
end;{asakont_1 procedure sonu}
procedure yukakont_1;{yukari isaretin kontrolu yapiliyor}
begin
if ((mah1=false) and (myh1=false)) then

```

```

begin
if (mah2=false) and (myh2=false) then
begin
for i:=2 to 5 do
begin
if yi[6-i]=true then
begin
k:=abs(kkb2-(5*i-2));
j:=abs(kkb1-(5*i-2));
if tyi2=false then
if j>=k then
begin
if yis[6-i]=false then
begin
kb2[6-i]:=true;yis[6-i]:=true;
end;
end
else
if tyi1=false then
if yis[6-i]=false then
begin
kb1[6-i]:=true;yis[6-i]:=true;
end;
end;
end;
end;
if (mah2=true) or (myh2=true) then
for i:=2 to 5 do
begin
if not((tyi2=false) and ((myh2=true) and (kkb2>5*i-2)))
then
if yi[6-i]=true then
if tyi1=false then
if yis[6-i]=false then
begin
kb1[6-i]:=true;yis[6-i]:=true;
end;
end;
end
else
if (mah2=false) and (myh2=false) then
begin
begin
for i:=2 to 5 do
begin
if not((tyi1=false) and ((myh1=true) and (kkb1>5*i-2)))
then
if yi[6-i]=true then
if tyi2=false then
if yis[6-i]=false then
begin
kb2[6-i]:=true;yis[6-i]:=true;
end;
end;
end;
end;

```

```

end;
end;
end;
procedure yukarig_1;
begin
repeat
if (kkb1*18-33)<kkbg1 then
begin
kapidurum;kapidurum;kapidurum;
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1-3;
putimage(195,kkbg1,u^,xorput);
if (kkb2*18-33)<kkbg2 then
begin
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2-3;
putimage(334,kkbg2,u^,xorput);
end;
if (kkb2*18-33)>kkbg2 then
begin
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2+3;
putimage(334,kkbg2,u^,xorput);
end;
end;
if (kkb1*18-33)>kkbg1 then
begin
kapidurum;kapidurum;kapidurum;
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1+3;
putimage(195,kkbg1,u^,xorput);
if (kkb2*18-33)<kkbg2 then
begin
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2-3;
putimage(334,kkbg2,u^,xorput);
end;
if (kkb2*18-33)>kkbg2 then
begin
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2+3;
putimage(334,kkbg2,u^,xorput);
end;
end;
until (kkb1*18-33)=kkbg1
end;
procedure yukarig_2;
begin
repeat
if (kkb2*18-33)<kkbg2 then
begin
kapidurum;kapidurum;kapidurum;
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2-3;

```

```

putimage(334,kkbg2,u^,xorput);
if (kkb1*18-33)<kkbg1 then
begin
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1-3;
putimage(195,kkbg1,u^,xorput);
end;
if (kkb1*18-33)>kkbg1 then
begin
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1+3;
putimage(195,kkbg1,u^,xorput);
end;
end;
if (kkb2*18-33)>kkbg2 then
begin
kapidurum;kapidurum;kapidurum;
putimage(334,kkbg2,u^,xorput);
kkbg2:=kkbg2+3;
putimage(334,kkbg2,u^,xorput);
if (kkb1*18-33)<kkbg1 then
begin
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1-3;
putimage(195,kkbg1,u^,xorput);
end;
if (kkb1*18-33)>kkbg1 then
begin
putimage(195,kkbg1,u^,xorput);
kkbg1:=kkbg1+3;
putimage(195,kkbg1,u^,xorput);
end;
end;
until (kkb2*18-33)=kkbg2
end;
procedure renkdol;
begin
if mah1=true then
begin
setfillstyle(1,yellow);
floodfill(142,410,10);
end
else
begin
setfillstyle(1,9);
floodfill(142,410,10);
end;
if myh1=true then
begin
setfillstyle(1,yellow);
floodfill(142,400,10);
end
else
begin

```

```
setfillstyle(1,9);
floodfill(142,400,10);
end;
if myh2=true then
begin
setfillstyle(1,yellow);
floodfill(600,400,10);
end
else
begin
setfillstyle(1,9);
floodfill(600,400,10);
end;
if mah2=true then
begin
setfillstyle(1,yellow);
floodfill(592,410,10);
end
else
begin
setfillstyle(1,9);
floodfill(592,410,10);
end;
if kb1[1]=true then
begin
setfillstyle(1,12);
floodfill(164,431,10);
end
else
begin
if kkb1<>381 then
begin
setfillstyle(1,9);
floodfill(164,431,10);
end;
if kkb1=381 then
begin
setfillstyle(1,5);
floodfill(164,431,10);
end;
end;
if kb1[2]=true then
begin
setfillstyle(1,12);
floodfill(164,405,10);
end
else
begin
if kkb1<>291 then
begin
setfillstyle(1,9);
floodfill(164,405,10);
end;
if kkb1=291 then
```

```
begin
setfillstyle(1,5);
floodfill(164,405,10);
end;
end;
if kb1[3]=true then
begin
setfillstyle(1,12);
floodfill(164,379,10);
end
else
begin
if kkb1[3]<>201 then
begin
setfillstyle(1,9);
floodfill(164,379,10);
end;
if kkb1[3]=201 then
begin
setfillstyle(1,5);
floodfill(164,379,10);
end;
end;
if kb1[4]=true then
begin
setfillstyle(1,12);
floodfill(164,353,10);
end
else
begin
if kkb1[4]<>111 then
begin
setfillstyle(1,9);
floodfill(164,353,10);
end;
if kkb1[4]=111 then
begin
setfillstyle(1,5);
floodfill(164,353,10);
end;
end;
if kb1[5]=true then
begin
setfillstyle(1,12);
floodfill(164,327,10);
end
else
begin
if kkb1[5]<>21 then
begin
setfillstyle(1,9);
floodfill(164,327,10);
end;
if kkb1[5]=21 then
```

```
begin
setfillstyle(1,5);
floodfill(164,327,10);
end;
end;
if kb2[1]=true then
begin
setfillstyle(1,12);
floodfill(612,431,10);
end
else
begin
if kkb2<>381 then
begin
setfillstyle(1,9);
floodfill(612,431,10);
end;
if kkb2=381 then
begin
setfillstyle(1,5);
floodfill(612,431,10);
end;
end;
if kb2[2]=true then
begin
setfillstyle(1,12);
floodfill(612,405,10);
end
else
begin
if kkb2<>291 then
begin
setfillstyle(1,9);
floodfill(612,405,10);
end;
if kkb2=291 then
begin
setfillstyle(1,5);
floodfill(612,405,10);
end;
end;
if kb2[3]=true then
begin
setfillstyle(1,12);
floodfill(612,379,10);
end
else
begin
if kkb2<>201 then
begin
setfillstyle(1,9);
floodfill(612,379,10);
end;
if kkb2=201 then
```

```
begin
setfillstyle(1,5);
floodfill(612,379,10);
end;
end;
if kb2[4]=true then
begin
setfillstyle(1,12);
floodfill(612,353,10);
end
else
begin
if kkb2[4]=111 then
begin
setfillstyle(1,9);
floodfill(612,353,10);
end;
if kkb2=111 then
begin
setfillstyle(1,5);
floodfill(612,353,10);
end;
end;
if kb2[5]=true then
begin
setfillstyle(1,12);
floodfill(612,327,10);
end
else
begin
if kkb2[5]=21 then
begin
setfillstyle(1,9);
floodfill(612,327,10);
end;
if kkb2=21 then
begin
setfillstyle(1,5);
floodfill(612,327,10);
end;
end;
if yi[1]=true then
begin
setfillstyle(1,12);
floodfill(311,403,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,403,10);
end;
if yi[2]=true then
begin
setfillstyle(1,12);
```

```
floodfill(311,313,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,313,10);
end;
if yi[3]=true then
begin
setfillstyle(1,12);
floodfill(311,223,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,223,10);
end;
if yi[4]=true then
begin
setfillstyle(1,12);
floodfill(311,133,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,133,10);
end;
if ai[2]=true then
begin
setfillstyle(1,12);
floodfill(311,345,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,345,10);
end;
if ai[3]=true then
begin
setfillstyle(1,12);
floodfill(311,255,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,255,10);
end;
if ai[4]=true then
begin
setfillstyle(1,12);
floodfill(311,165,10);
end
else
begin
```

```

setfillstyle(1,9);
floodfill(311,165,10);
end;
if ai[5]=true then
begin
setfillstyle(1,12);
floodfill(311,75,10);
end
else
begin
setfillstyle(1,9);
floodfill(311,75,10);
end;
end;
procedure kapiac;
begin
for i:=1 to 4 do
begin
if (mah1=false) and (myh1=false) then
if ai[i+1]=true then
if kkb1=(5*(5-i)-2) then
kkapi1:=false;
if (mah1=false) and (myh1=false) then
if yi[i]=true then
if kkb1=(5*(6-i)-2) then
kkapi1:=false;
if (mah2=false) and (myh2=false) then
if ai[i+1]=true then
if kkb1<>(5*(6-i)-2) then
if kkb2=(5*(5-i)-2) then
kkapi2:=false;
if (mah2=false) and (myh2=false) then
if yi[i]=true then
if kkb1<>(5*(6-i)-2) then
if kkb2=(5*(6-i)-2) then
kkapi2:=false;
end;
end;
procedure sabah;
begin
if kkb1<>3 then
if kkapi1=true then
if (mah1=false) and (myh1=false) then
kb1[5]:=true;
if kkb2<>3 then
if kkapi2=true then
if (mah2=false) and (myh2=false) then
kb2[5]:=true;
end;
procedure aksam;
begin
if kkb1<>23 then
if kkapi1=true then
if (mah1=false) and (myh1=false) then

```

```

kb1[1]:=true;
if kkb2<>23 then
if kkapi2=true then
if (mah2=false) and (myh2=false) then
kb2[1]:=true;
end;
begin {ANA PROGRAM}
graphdriver:=detect;
initgraph(graphdriver,graphmode,'');
errorcode:=graphresult;
if errorcode<>grok then
begin
writeln('error=',grapherrormsg(errorcode));
end;
reset;basla;setcolor(yellow);settextstyle(triplexfont,horiz
dir,2);
settextjustify(center,center);outtextxy(55,300,'Kis
i sayisi');
outtextxy(505,300,'Kisi sayisi');outtextxy(505,400,'Kat
sayisi');
outtextxy(55,400,'Kat sayisi');kisisayi1;kisisayi2;
repeat
kapiac;
tusoku;
yukakont_1;
kabink_1;
kabink_2;
hareket;
yukarig_1;
yukarig_2;
renkdol;
tamyuk;
asiriy;
asakont_1;
kapidurum;
dur_1;
katsayi1;
dur_2;
katsayi2;
until kbdur1;
closegraph;
end.{ana program sonu}

```

**T.C. YÜKSEKÖĞRETİM KURULU
KURUMSAL YENİLEŞTİRME VE
KALİTE GÜVENCE MERKEZİ**