4:04 4:04

Escola Universitària Politècnica
de Vilanova i la Geltrú

UNIVERSITAT POLITÈCNICA DE CATALUNYA

PROJECTE FI DE CARRERA

TÍTOL: Simulació d'una planta automatitzada mitjançant un microcontrolador

AUTOR: Ramon Gasull Anguera

TITULACIÓ: Enginyeria Tècnica Industrial:Especialitat Electrònica industrial

DIRECTOR: Francisco Javier Ruiz Vegas

DEPARTAMENT: Enginyeria de Sistemes, Automàtica i Informàtica industrial

DATA: 29/06/2007

4:04 4:04

TÍTOL: Simulació d'una planta automatitzada mitjançant un microcontrolador

COGNOMS: Gasull Anguera **NOM: Ramon**

TITULACIÓ: Enginyeria tècnica industrial

ESPECIALITAT: Electrònica Industrial **PLA: 95**

DIRECTOR: Francisco Javier Ruiz Vegas

DEPARTAMENT: Enginyeria de Sistemes, Automàtica i Informàtica Industrial

QUALIFICACIÓ DEL PFC

TRIBUNAL

PRESIDENT **SECRETARI** **VOCAL**

DATA DE LECTURA:

Aquest Projecte té en compte aspectes mediambientals: □ Sí □ No

PROJECTE FI DE CARRERA

RESUM (màxim 50 línies)

Aquest projecte pretén contribuir a millorar l'aprenentatge de programació d'autòmats programables presentant una alternativa a les dues vies que existeixen tradicionalment per testar un programa realitzat amb aquests dispositius.

La via que proposa aquest projecte consisteix en simular una planta automatitzada amb un microcontrolador. Aquesta planta té un cost molt inferior a una planta real o una maqueta, és fàcil de connectar a l'autòmat i no ens suposa cap risc connectar-la a l'autòmat encara que el programa de control no sigui el adequat.

Per aconseguir simular la planta, s'ha dissenyat i construït una placa on tindrem el microcontrolador, programat perquè simuli una planta, i una sèrie d'indicadors lluminosos que permetran a l'usuari conèixer en tot moment l'estat de la planta així com dades útils en la detecció d'errors del programa de control. A més la placa està adequada per permetre la connexió entre l'autòmat i la planta.

El projecte ha consistit en la construcció d'un prototip de la planta simulada, l'estudi de la lògica de programació d'aquesta planta i la implementació real d'un procés de connexió autòmat-planta en la qual s'han analitzat diferents situacions.

En la implementació s'ha utilitzat un model de planta que consisteix en un ascensor de tres plantes ja que ens dóna joc per realitzar una important varietat de probes, tal i com es pot observar amb els exercicis d'exemple.

Paraules clau (màxim 10

Autòmat	Microcontrolador	Planta	Automatització
Simulació	Ascensor	Programació	PIC16F876

ÍNDEX

1. INTRODUCCIÓ 5

1.1. OBJECTIUS 6

1.2. ESTRUCTURA 7

2. AUTÒMATS PROGRAMABLES I MICROCONTROLADORS ..9

2.1. AUTÒMATS PROGRAMABLES O PLC'S 9

2.1.1. Els orígens 9

2.1.2. Estructura 10

2.1.2.1. La font d'alimentació 10

2.1.2.2. La Unitat Central de Processament (CPU) 11

2.1.2.3. Mòdul d'entrades 11

2.1.2.4. Mòdul de sortides 12

2.1.2.5. Terminal de programació 12

2.1.3. Configuració i instal·lació 13

2.1.3.1. Configuració 13

2.1.3.2. Instal·lació 13

2.1.4. Software i llenguatges de programació 14

2.1.4.1. Software de programació 14

2.1.4.2. Llenguatges de programació 15

2.1.5. Funcionament de l'autòmat 18

2.2. MICROCONTROLADORS 19

2.2.1. Microcontrolador i microprocessador 20

2.2.2. Aplicacions dels microcontroladors 21

2.2.3. Recursos habituals en els microcontroladors 21

2.2.3.1. Arquitectura bàsica 22

2.2.3.1.1. Unitat Central de Processament 22

2.2.3.1.2. Memòria 23
2.2.3.1.3. Portes d'entrada i sortida 25
2.2.3.1.4. Rellotge principal 25
2.2.4. Recursos especials 26
2.2.4.1. Temporitzadors 26
2.2.4.2. Watchdog 26
2.2.4.3. Protecció davant fallada d'alimentació 27
2.2.4.4. Estat de repòs 27
2.2.4.5. Convertidor A/D 27
2.2.4.6. Convertidor D/A 27
2.2.4.7. Comparador analògic 27
2.2.4.8. PWM 28
2.2.4.9. Ports de E/S digitals 28
2.2.4.10. Ports de comunicació 28
2.2.5. Desenvolupament del software 29
2.3. AUTÒMAT O MICROCONTROLADOR? 30
3. SIMULACIÓ DE LA PLANTA AUTOMATITZADA 33
3.1. PLANTA SIMULADA: L'ASCENSOR 34
3.1.1. Característiques específiques de la planta 34
3.1.2. Algoritme del codi del microcontrolador 38
4. DISSENY DE LA PLACA 41
4.1. DIFERENTS PARTS DE LA PLACA 41
4.1.1. El microcontrolador 42
4.1.2. Regulador de tensió 44
4.1.3. Circuit oscil·lador 46
4.1.4. Entrades 47
4.1.4.1. Definició i funcionament bàsic 48

4.1.4.2. Circuit d'optoacobladors utilitzat 48
4.1.5. Sortides 50
4.1.5.1. Funcionament dels transistors PNP i NPN 50
4.1.5.2. El nostre circuit 52
4.1.6. Indicadors 54
4.2. IMATGES DE LA PLACA 56
4.2.1. Primer prototip 56
4.2.2. Prototip definitiu 57
5. IMPLEMENTACIÓ DE LA METODOLOGIA D'APRENENTATGE 59
5.1. EXEMPLE 1: FUNCIONAMENT SIMPLE, PROGRAMA ERRONI 59
1.1.1. Descripció de l'exercici 59
5.1.1. Entrades i sortides de l'autòmat 60
5.1.1.1. Entrades 60
5.1.1.2. Sortides 61
5.1.2. Diagrama de contactes 61
5.1.3. Errors detectats 65
5.2. EXEMPLE 1.1: CORRECCIÓ DE L'EXEMPLE 1 66
5.2.1. Descripció de l'exercici 66
5.2.2. Entrades i sortides de l'autòmat 66
5.2.3. Diagrama de contactes 66
5.3. EXEMPLE 2 72
5.3.1. Descripció de l'exercici 72
5.3.2. Entrades i sortides de l'autòmat 72
5.3.2.1. Entrades 73
5.3.2.2. Sortides 73
5.3.3. Diagrama de contactes 73
5.4. EXEMPLE 3 81
5.4.1. Descripció de l'exercici 81

5.4.2. Entrades i sortides de l'autòmat 81
5.4.2.1. Entrades 82
5.4.2.2. Sortides 82
5.4.3. GRAFCET's 82
5.4.3.1. GRAFCET principal 83
5.4.3.2. GRAFCET 1 84
5.4.3.3. GRAFCET 2 84
5.4.4. Diagrama de contactes 85
6. CONCLUSIONS 97
7. LÍNIES FUTURES DE TREBALL 99
8. BIBLIOGRAFIA I RECURSOS ELECTRÒNICS WEB 101
9. ANNEX I: CODI DEL MICROCONTROLADOR 103
10. ANNEX II: DIAGRAMES DE CONTACTES DELS EXERCICIS 107
11. ANNEX III: ESQUEMA ELÈCTRIC DE LA PLACA 127

1. Introducció

Aquest projecte representa una contribució a la millora de l'aprenentatge de la programació dels autòmats programables i suposa una alternativa a les dues vies tradicionals que existeixen per testar un programa realitzat amb aquest tipus de dispositiu. Aquestes dues vies són, per una part, la connexió de l'autòmat a una planta real (o bé una maqueta) i, per altra part, la simulació del comportament de la planta.

La primera d'aquestes vies, la utilització d'una planta real, és sens dubte la millor opció des del punt de vista de la fiabilitat ja que permet fer una comprovació total del programa en les mateixes circumstàncies en què el programa funcionarà. També des del punt de vista de l'aprenentatge, la planta real proporcionarà un coneixement més exacte del funcionament del programa involucrant tots els elements relacionats amb el sistema d'automatització (tipus de sensors i actuadors, col·locació i alimentació d'aquests, adaptació de senyals, etc). No obstant, l'ús de plantes reals no sempre és factible, ja sigui per la impossibilitat de transportar la planta al laboratori, per la complexitat que suposaria la connexió de la planta a l'autòmat només per testar els programes, o pel perill de fer la planta malbé per una errada en el programa. La utilització de maquetes que simulen plantes reals poden pal·liar, en part, aquests impediments, però quan es vol testar diferents programes dissenyats per diferents plantes (semàfors, ascensors, plantes embotelladores,...) s'hauria de disposar d'una maqueta per a cada tipus de planta, cosa que incrementa substancialment el cost d'aquesta tasca.

La segona via, i la que es fa servir majoritàriament en les assignatures d'introducció a la programació d'autòmats programables de les escoles d'enginyeria, és la simulació manual de la planta. Això consisteix en introduir directament mitjançant interruptors o polsadors les entrades de l'autòmat i comprovar les sortides observant els leds associats a aquestes. No hi ha dubte que aquest és el sistema més simple i menys costós de realitzar però suposa que tenim un coneixement total del comportament de la planta que no sempre és real. Normalment, sorgiran moltes situacions que no havíem tingut en compte a l'hora de fer el disseny del programa i a l'hora de fer la comprovació. Aquests errors podrien ser fatals en cas que es produïssin durant el funcionament de la planta, ja que en el cas de les indústries podrien suposar importants pèrdues materials i econòmiques.

En aquest treball es proposa una tercera via que es troba entremig de les dues vies esmentades. Es tracta d'una planta simulada amb un microcontrolador. Aquesta planta té un cost molt inferior a una planta real i a una maqueta, és més fàcil de connectar a l'autòmat, no té risc de danys per errors de programació i permet una

gran flexibilitat en el sentit que canviar de planta suposa només la modificació del programa del microcontrolador.

El projecte ha consistit en la construcció d'un prototipus de planta simulada, l'estudi de la lògica de programació en aquesta planta (no és el mateix un programa per controlar una planta que un programa que simuli el comportament de la planta) i la implementació real d'un procés de connexió autòmat-planta en la qual s'han analitzat diferents situacions. Concretament, en la implementació s'ha utilitzat com a model un ascensor de tres plantes. Com es veurà posteriorment, aquesta planta ens dóna joc per realitzar una important varietat de proves.

1.1. Objectius

Una part molt important d'aquest treball consisteix en la programació del microcontrolador per tal que simuli el funcionament d'una planta automatitzada. En aquest treball, la planta automatitzada sobre la que treballem és un ascensor preparat per recórrer tres plantes. Per aconseguir que el microcontrolador aproximi al màxim el seu funcionament al de la planta real, i així poder assolir aquest primer objectiu, s'ha de dissenyar el programa del microcontrolador únicament pensant en les possibles reaccions de la planta, indiferentment de les accions de control que se li apliquin per part del autòmat. Per poder aconseguir un resultat òptim és important tenir present el nombre màxim de situacions en què es pot trobar la planta i les possibles reaccions que aquesta pot tenir.

El segon punt a assolir fa referència a la connexió entre l'autòmat i el microcontrolador. Una vegada aconseguim simular la planta automatitzada el que voldrem serà poder controlar-la. Per aquest motiu, necessitem ser capaços de connectar-la amb l'autòmat. La comunicació entre l'autòmat i el microcontrolador es produirà connectant les sortides del primer amb les entrades del segon i les sortides del segon amb les entrades del primer. Aquest fet ens permetrà tenir un llaç tancat que ens garanteix un control adequat i una simulació en temps real. Per tal d'aconseguir aquesta comunicació òptima s'hauran d'adequar els senyals d'entrada i sortida del dos dispositius.

Per aconseguir tancar la comunicació autòmat-planta necessitem disposar d'un medi. És per aquest motiu que és imprescindible el disseny i la construcció d'una placa que contingui tots els elements necessaris per a la connexió. En la construcció de la placa, també hem de tenir en compte que aquesta ha de disposar d'elements que permetin a l'usuari conèixer l'estat de planta en tot moment.

Al tractar-se d'una eina per testar diferents programes, fet que pot suposar que l'usuari hagi de realitzar diferents probes, la nostra placa ha d'estar preparada per funcionar durant un període de temps important i és imprescindible que l'usuari no s'hagi d'esperar cada vegada que s'ha de reiniciar el microcontrolador per començar el procés de nou.

Finalment, hem de ser capaços de demostrar que la nostra placa és útil. És per això que s'han de mostrar diferents exemples de possibles programes de control per a la planta. Per aquests programes és interessant que utilitzem els diferents recursos que ens ofereix l'autòmat i el seu entorn de programació, així tindrem exemples d'exercicis de diferents graus de dificultat.

1.2. Estructura

Aquest document està dividit en set parts clarament diferenciades, a més tenim dos annexes que contenen informació complementària per poder acabar de comprendre el contingut d'aquest treball.

En aquesta primera part, es realitza una breu introducció per presentar el tema del projecte i la metodologia que s'ha seguit per realitzar-lo.

El segon capítol pretén introduir els dos elements principals d'aquest treball, l'autòmat i el microcontrolador, perquè posteriorment utilitzarem dades o expressions que hi van associades. La informació que detallem, bàsicament, descriu el funcionament i l'arquitectura d'ambdós elements, així com també fem referència a possibles aplicacions en els que els podem trobar.

El tercer capítol fa referència a la descripció de la planta automatitzada simulada. En aquest capítol detallem com ha de ser la connexió entre l'autòmat i el microcontrolador per tal que els dos elements es comuniquin i interactuïn correctament. També descrivim la planta que hem simulat amb el microcontrolador i que posteriorment serà la base de tots els exemples d'exercicis que hem realitzat per mostrar la utilitat i la eficiència d'aquest treball.

En el quart capítol es realitza una descripció de l'element físic que ens permet la connexió entre l'autòmat i el microcontrolador. Així hi trobem tota la informació referent al disseny placa, tal com esquemes elèctrics, adaptació dels senyals, elements de protecció i leds indicadors.

En el cinquè capítol descrivim la metodologia de treball de la planta mitjançant quatre exemples pràctics. Aquests quatre exemple pretenen tenir nivells de dificultat

diferents i a més utilitzen els diferents recursos de programació que ens ofereix l'autòmat utilitzat. L'objectiu d'aquest apartat consisteix en que siguem capaços de conèixer i entendre les diferències i millores que tenim entre treballar amb l'autòmat connectat a la planta o amb l'autòmat sol.

En el sisè capítol podem trobar les conclusions que hem obtingut en la realització d'aquest projecte.

Finalment, en el setè capítol introduïm algunes propostes per a possibles línies futures de treballa realitzar en una hipotètica continuació del treball.

A més tenim dos Annexos. En el primer trobem el llistat del codi de programació del microcontrolador i en el segon els llistats dels diagrames de contactes dels diferents exemples utilitzats per a programar l'autòmat.

2. Autòmats programables i microcontroladors

El sistema que utilitzem està format principalment per dos elements. Aquests elements són un autòmat i un microcontrolador, per la qual cosa a continuació els introduïm.

2.1. Autòmats programables o PLC's

Entenem per autòmat programable o PLC (Controlador lògic programable) tota màquina electrònica dissenyada per controlar en temps real i en un medi normalment industrial processos seqüencials.

La seva funció bàsica consisteix en reduir el treball de l'usuari a l'hora de realitzar el programa, és a dir, relacionar els senyals d'entrada que s'han de complir per activar cada sortida, ja que elements tradicionals com relés, temporitzadors, comptadors, etc. en aquests dispositius són interns. Habitualment s'utilitza en el control de tot tipus de plantes automatitzades, ja sigui en la indústria, en sistemes d'alarmes, pàrkings, entre d'altres.

2.1.1. Els orígens

Els primers autòmats programables van aparèixer als Estats Units a finals dels anys seixanta en les indústries del sector de l'automòbil i un parell d'any més tard van arribar a Europa. Així, la seva aparició coincideix amb l'era del microprocessador i la generació lògica cablejada modular.

L'autòmat va ser la primera màquina la qual el conjunt de les seves instruccions s'orienta cap als sistemes d'evolució seqüencial. La creixent difusió d'aplicacions electròniques, la disminució del preu dels components, el naixement i desenvolupament dels microprocessadors i, sobretot, miniaturització dels circuits de

memòria ens permeten presagiar una introducció dels autòmats en una gran gama de nous camps d'aplicació.

2.1.2. Estructura

Un autòmat programable és un sistema basat en un microprocessador, que té com a parts fonamentals la Unitat Central de Processament (CPU), la memòria i el sistema d'entrades i sortides (E/S).

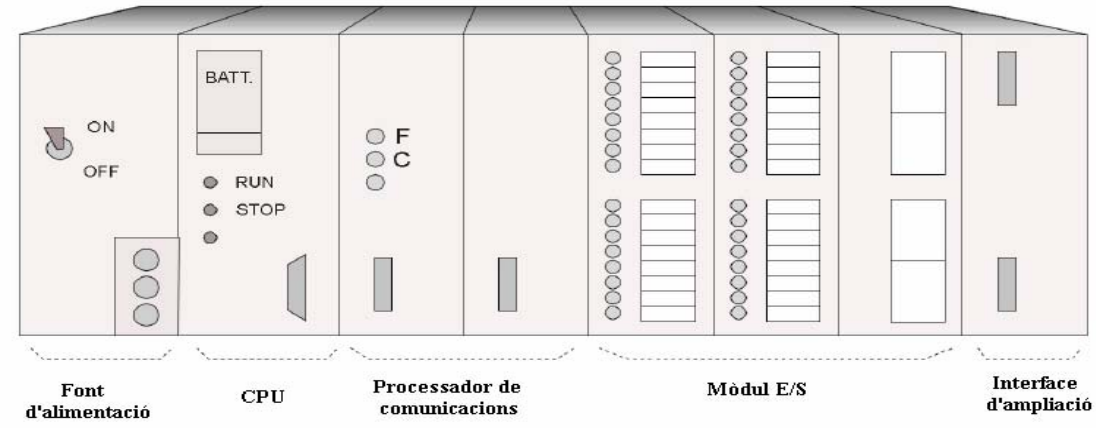


Figura 2.1 Representació de la vista frontal d'un autòmat

2.1.2.1. La font d'alimentació

La font d'alimentació de l'autòmat és l'encarregada de convertir la tensió que li arriba de la xarxa elèctrica, 220 volts de corrent alterna, a baixa tensió de corrent contínua, normalment a 24 volts. Sent aquesta última la tensió amb la que treballarem en els circuits que formen l'autòmat. La font d'alimentació també es capaç de subministrar energia a alguns sensors i actuadors (normalment lluminosos) de molt baixa potència.

2.1.2.2. La Unitat Central de Processament (CPU)

La Unitat Central de Processament és l'encarregada de realitzar el control intern i extern del autòmat, així com també la interpretació de les instruccions del programa. A partir de les instruccions que té guardades a la memòria i de les dades que rep procedents de les entrades, genera els senyals de les sortides. En la seva memòria hi trobem el programa destinat a controlar el procés.

La CPU conté les següents parts:

1) Unitat central o de procés

2) Temporitzadors o comptadors

3) Memòria

Així mateix la memòria es divideix en dos blocs:

1) ROM: Memòria de només lectura.

2) RAM: Memòria de lectura i escriptura.

A la memòria ROM s'hi emmagatzemen programes per al correcte funcionament de l'autòmat, com pot ser el programa de comprovació de posada en marxa i el programa d'exploració de la memòria RAM.

La memòria RAM es pot dividir en dues àrees:

1) Memòria de dades.

2) Memòria d'usuari.

En la memòria de dades s'hi emmagatzema la informació referent als estats de les entrades, de les sortides i de variables internes. Mentre que a la memòria d'usuari s'hi emmagatzema el programa amb el que treballarà l'autòmat.

2.1.2.3. Mòdul d'entrades

El mòdul d'entrades és el que recull la informació del procés controlat. Aquí és on tindrem connectats els captadors (interruptors, sensors, polsadors,...). Cada cert temps la informació referent a l'estat de les diferents entrades és enviada a la CPU perquè aquesta la pugui processar d'acord amb el programa.

Podem distingir dos tipus de captadors:

1) Captadors passius: Els captadors passius són els que canvien el seu estat lògic (activat o no activat) per mitjà d'una acció mecànica. Dins d'aquest grup hi trobem els interruptors, polsadors, finals de carrera, etc.

2) Captadors actius: Els captadors actius són dispositius electrònics que subministren una tensió a l'autòmat. Aquesta tensió acostuma a ser equivalent a una determinada magnitud física.

2.1.2.4. Mòdul de sortides

El mòdul de sortides és l'encarregat d'activar i desactivar els actuadors. La informació enviada per la CPU, una vegada ha estat processada, s'envia a la interfície de sortides perquè aquestes siguin activades i a la vegada els actuadors que estan connectats a elles.

Segons el tipus de procés a controlar per l'autòmat, podem utilitzar diferents mòduls de sortides, que diferenciem en tres tipus:

1) Relés: Són utilitzats en circuits de corrent contínua i corrent alterna. Estan basats en la commutació mecànica d'un contacte elèctric normalment obert.

2) Triac: Aquests s'utilitzen en circuits de corrent contínua i corrent alterna que necessiten accions de commutació molt ràpides.

3) Transistors a col·lector obert: Finalment, aquests normalment s'utilitzen en circuits que necessiten maniobres de connexió o desconnexió molt ràpides. La utilització d'aquet tipus de mòduls és restringit només circuits de corrent contínua.

2.1.2.5. Terminal de programació

El terminal o consola de programació de l'autòmat és el que ens permet comunicar l'usuari amb el sistema.

Les funcions bàsiques del terminal de programació són les següents:

1) Transferència i modificació dels programes.

2) Verificació de la programació.

3) Informació del funcionament dels processos.

Com a terminals de programació podem trobar les construïdes específicament per l'autòmat, o bé, un ordinador personal que suporti un software específicament dissenyat per resoldre problemes de programació i control.

2.1.3. Configuració i instal·lació

Uns passos molts importants en la posada a punt de l'autòmat són la configuració i la instal·lació d'aquest. És necessari realitzar aquestes dues tasques correctament per tal d'obtenir un bon funcionament de l'autòmat posteriorment.

2.1.3.1. Configuració

La configuració de l'autòmat és el procés en el qual es determina com i on es situen els diferents components del sistema de control. Aquesta configuració dependrà bàsicament de la tasca de control i del tipus de control que s'hagi decidit i contempla tant els elements de l'autòmat com els seus perifèrics.

Durant l'elaboració de l'algoritme de control s'han determinat les entrades i les sortides, tant les analògiques com les discretes, i aquestes s'han relacionat mitjançant diagrames o esquemes lògics. La quantitat i tipus d'entrades i sortides ens determina quins components són necessaris.

La millor manera de realitzar la configuració és confeccionant primer de tot un mapa de direccionament, en el qual mitjançant una representació de l'estructura de les entrades i les sortides s'indica quins components s'han d'ubicar junt a la unitat central i quins s'ubiquen en posicions remotes. Una vegada finalitzada la configuració, podem començar amb la instal·lació i la programació.

2.1.3.2. Instal·lació

Donades les característiques constructives i de disseny dels autòmats programables, la seva instal·lació és viable en pràcticament qualsevol ambient industrial, sempre que no es sobrepassin les especificacions donades pel fabricant. No obstant, existeixen algunes recomanacions interessants per assegurar un

correcte funcionament del sistema, que tenen a veure principalment amb les condicions de temperatura i humitat i la immunitat davant d'interferències elèctriques.

Generalment l'autòmat es muntarà en un armari de dimensions adequades per contenir sobradament els components de l'equip i la resta d'elements, com interruptors, fonts d'alimentació, circuits de protecció, conductes de cablejat,... És recomanable utilitzar armaris metàl·lics ja que minimitzen els efectes de la radiació electromagnètica generada pels equips de commutació instal·lats a prop. Per a la instal·lació es seguiran les normes i reglaments vigents habituals en qualsevol instal·lació elèctrica de control.

2.1.4. Software i llenguatges de programació

Per a la programació d'autòmats molts fabricants ofereixen paquets de software que facilita la tasca a l'usuari. La majoria de fabricants utilitzen software semblant per a la programació dels seus autòmats i tenim uns llenguatges de programació coneguts per la majoria de programadors per tal d'unificar criteris.

2.1.4.1. Software de programació

Els paquets de software per a la programació d'autòmats converteixen un ordinador personal en un equip de programació específic, aprofitant el seus potents recursos d'interfície amb altres sistemes, amb l'usuari i el baix preu del hardware a causa de l'estandardització i generalització de l'ús del ordinadors compatibles. Aquesta opció, PC i software, constitueix, juntament amb les consoles pràcticament la totalitat d'equips de programació utilitzats actualment pels programadors d'autòmats.

El requisits de hardware i software exigits per instal·lar sobre el PC un entorn de programació d'autòmats acostumen a presentar unes exigències que queden cobertes si l'equip ja està suportant l'entorn Windows.

El paquet de programació es completa amb la unitat externa de connexió que converteix i fa compatibles els senyals físics entre la sortida sèrie estàndard de PC (RS-232) i el port de connexió de la consola de l'autòmat, canal usualment utilitzat també per a la connexió amb el PC.

2.1.4.2. Llenguatges de programació

Quan van aparèixer els autòmats programables, els van fer amb la necessitat de substituir enormes quadres de maniobra construïts amb contactors i relés. Per tant, la comunicació home-màquina devia ser similar a la utilitzada fins aquell moment. El llenguatge utilitzat, havia de ser interpretat amb facilitat pels mateixos tècnics que anteriorment estaven en contacte amb la instal·lació. Aquests llenguatges han evolucionat de tal forma que alguns ja no tenen res a veure amb el típic panell elèctric de relès.

Els llenguatges més significatius són els següents:

1) **Llenguatge de contactes**: És el que té més semblances amb l'utilitzat per un electricista quan elabora quadres d'automatismes. Molts autòmats inclouen mòduls especials de software per poder programar gràficament d'aquesta forma.

Per programar un autòmat amb aquest llenguatge, a més d'estar familiaritzat amb les regles dels circuits de commutació, és necessari conèixer cada un dels elements que consta aquest llenguatge.

Els principals són els que descrivim a continuació:

a) Contacte NO(Normalment obert): S'activa quan hi ha un 1 lògic en l'element que representa, aquest element és una entrada, una variable interna o un bit del sistema.

b) Contacte NT(Normalment tancat): La seva funció és similar a la de l'element anterior, però en aquest cas s'activa quan hi ha un 0 lògic.

c) Bobina: Aquest element s'activa quan la combinació que hi ha a la seva entrada dóna com a resultat un 1 lògic i es desactiva quan la combinació que hi ha la seva entrada dóna com a resultat un 0 lògic. La seva activació equival a dir que té un 1 lògic i la seva desactivació a dir que té un lògic. Acostuma a representar elements de sortida.

d) Bobina SET: Una vegada s'activa no es pot desactivar si no és per la seva corresponent bobina de RESET. Serveix per memoritzar bits i usada juntament amb la bobina RESET ofereix una enorme potència en la programació.

e) Bobina RESET: Permet desactivar l'element anterior.

f) Temporitzadors: El temporitzador és un element que permet comptar el temps amb la finalitat d'activar bobines passat un cert temps des de l'activació.

g) Comptadors: Aquest és un element capaç de portar el còmput de les activacions de les seves entrades, pel que resulta adequat per memoritzar successos que no tinguin a veure amb el temps, però s'hagin de realitzar un determinat nombre de vegades.

2) Llenguatge per llista d'instruccions: En els autòmats de baixa gama és l'únic mode de programació. Consisteix en elaborar una llista d'instruccions que s'associen als símbols i la seva combinació en un circuit elèctric a contactes. També s'ha de dir que aquest tipus de llenguatge és, en alguns casos, la forma més ràpida de programació i inclús la més potent.

3) GRAFCET (Gràfic funcional d'etapes i transicions): Ha estat especialment dissenyat per resoldre problemes d'automatismes seqüencials.. Aquest llenguatge resulta enormement senzill d'interpretar per operaris sense coneixements d'automatismes elèctrics. Molts dels autòmats que existeixen en el mercat permeten la programació en GRAFCET, tant en mode gràfic com per llista d'instruccions. També es pot utilitzar per resoldre problemes d'automatització de forma teòrica i posteriorment convertir-lo a diagrama de contactes.

Per realitzar el programa corresponent a un cicle de treball en llenguatge GRAFCET, s'hauran de tenir en compte una sèrie de principis bàsics:

1) El procés es descompon en etapes que s'aniran activant una darrere l'altra.

2) A cada etapa se li associa una o varies accions que només hauran de ser efectives quan l'etapa estigui activa.

3) Una etapa s'activa quan es compleix la condició de transició.

4) El compliment d'una condició de transició implica l'activació de la següent etapa i la desactivació de l'etapa anterior (o etapes anteriors).

En un GRAFCET podem trobar-nos tres tipus de seqüències:

1) Lineals: En les seqüències lineals el cicle el composen una successió lineal d'etapes. El programa anirà activant cada una de les etapes i desactivant l'anterior conforme es vagin complint cada una de les condicions. Les accions es realitzaran en funció de l'etapa activa a la que estan associades. En la figura 2.2 podem observar un exemple de seqüència lineal.

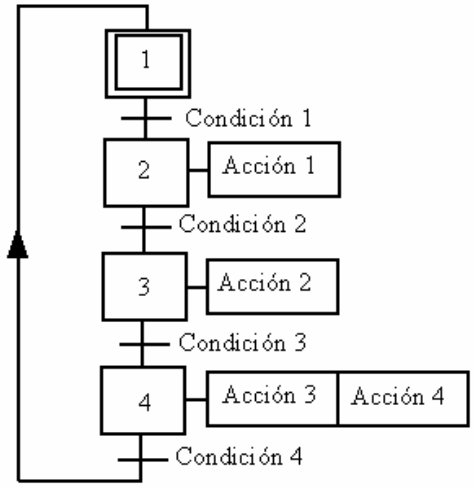


Figura 2.2 Exemple de GRAFCET de seqüència lineal

2) Amb direccionament: En un GRAFCET amb direccionament el cicle pot variar en funció de la condició que es compleixi. En la *figura 2.3* podem observar com, a partir de l'etapa inicial, es poden seguir tres camins diferents depenent de la condició que es compleixi, ara bé, solament una d'elles es pot complir mentre estigui activa l'etapa 1.

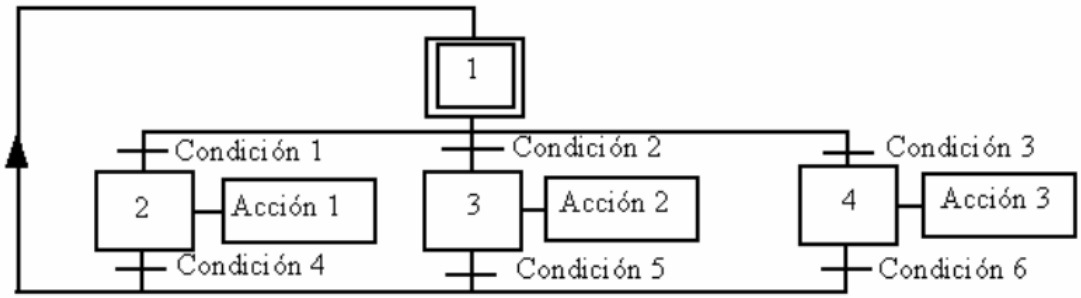


Figura 2.3 Exemple de GRAFCET de seqüència amb direccionament

3) Simultànies: En les seqüències simultànies, diferents camins poden estar funcionant al mateix temps per activació simultània de les etapes. En la *figura 2.4* observem com les etapes 2, 3 i 4 s'activen amb la mateixa condició.

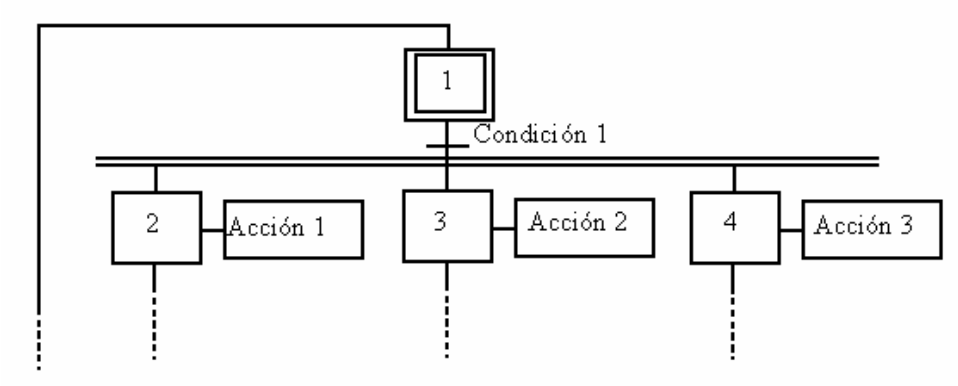


Figura 2.4 Exemple de GRAFCET de seqüència simultània

També podem classificar les accions segons el seu tipus. A continuació descrivim les més utilitzades:

1) Accions associades a vàries etapes: Una mateixa acció pot ser que estigui associada a més d'una etapa.

2) Accions condicionades: La execució d'aquest tipus d'accions es produeix quan a més de trobar-se activa l'etapa a la que està associada, s'ha de verificar una condició lògica suplementària.

3) Accions temporitzades o retardades: Aquest és un cas particular de les accions condicionades. El temps intervé com una condició lògica més.

4) Pla de funcions lògiques: Resulta especialment còmode d'utilitzar a tècnics habituats a treballar amb circuits de portes lògiques, ja que la simbologia utilitzada en els dos casos es equivalent.

2.1.5. Funcionament de l'autòmat

La majoria dels autòmats actuals es basen en el concepte de l'execució cíclica de les instruccions ubicades en la seva memòria.

El programa és una sèrie d'instruccions guardades a la memòria. Un cicle del procés consisteix inicialment en la consideració d'una sèrie d'entrades que

seguidament seran fixades per a tot cicle. Després, l'autòmat executa una instrucció darrere l'altra fins a finalitzar el programa i finalment es defineixen les ordres que s'han d'aplicar sobre les sortides. El cicle es reprodueix així indefinidament.

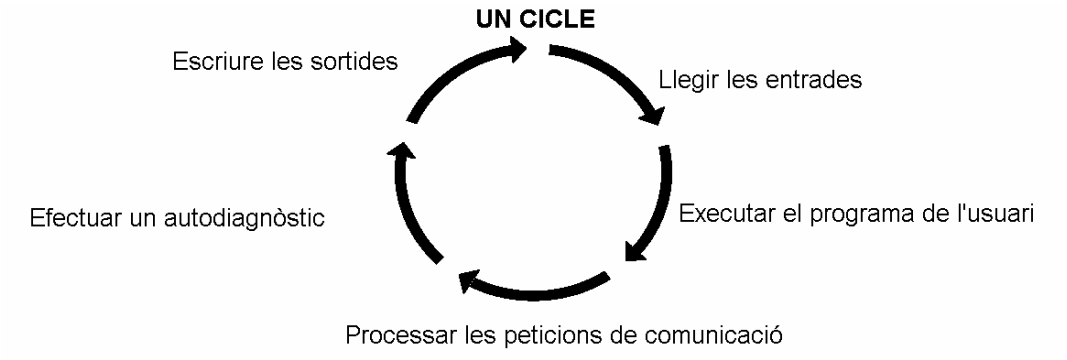


Figura 2.5 Representació del cicle de funcionament d'un autòmat..

2.2. Microcontroladors

Als voltants de l'any 1980, els fabricant de circuits integrats van iniciar la difusió d'un nou circuit per al control, medició i instrumentació al que van anomenar microcomputador en un sol xip, o de forma més exacta, microcontrolador.

Un microcontrolador és un circuit integrat que conté tota la estructura, arquitectura, d'un microcomputador, és a dir, la CPU, la RAM, la ROM i circuits d'entrada i sortida.

Alguns microcontrolador més especialitzats contenen també convertidors analògic-digitals, temporitzadors, comptadors i un sistema que permet la comunicació en sèrie i en paral·lel.

Es poden crear moltes aplicacions amb els microcontroladors. Aquestes aplicacions són il·limitades, alguns exemples d'aplicacions que podem dimensionar són: sistemes d'alarmes, jocs de llums, panells publicitaris, control de motors, control de màquines, control de temperatura, control de temps, adquisició de dades a través de sensors, etc.

2.2.1. Microcontrolador i microprocessador

A continuació pretenem diferenciar dos dispositius que sovint poden generar certa confusió com són el microcontrolador i el microprocessador.

El microprocessador és un circuit integrat que conté la Unitat Central de Processament (CPU), d'un computador. La CPU està formada per la Unitat de control, que interpreta les instruccions, i el bus de dades, que les executa.

Les potes d'un microprocessador treuen a l'exterior les línies dels seus busos de direccions, dades i control, per permetre la connexió amb la memòria i els mòduls E/S. Es diu que un microprocessador és un sistema obert perquè la seva configuració és variable d'acord amb l'aplicació a la que es vulgui destinar.

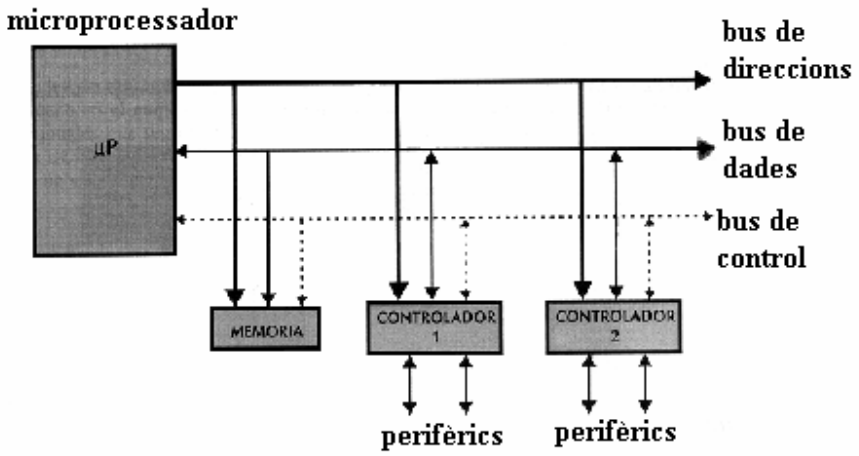


Figura 2.3 Esquema d'un sistema obert basat en un microprocessador.

Si únicament es disposés d'un microcontrolador, aquest hauria de tenir molt potenciats tots els seus recursos per poder adaptar-se a les exigències de les diferents aplicacions. A la pràctica, cada fabricant de microcontrolador ofereix un elevat nombre de models diferents. Així, se'ns permet seleccionar la capacitat de les memòries, el nombre de línies de E/S, la quantitat i potència dels elements auxiliars, la velocitat de funcionament, etc. Per tot això un aspecte destacat en el disseny és l'elecció del microcontrolador a utilitzar.

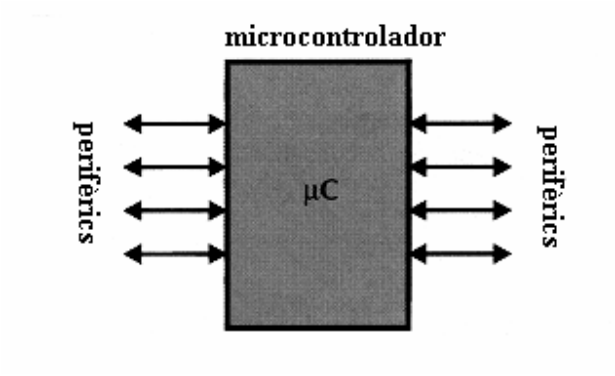


Figura 2.4 El microcontrolador és un sistema tancat. Totes les parts del computador estan en el seu interior i només surten a l'exterior les línies que governen els perifèrics.

2.2.2. Aplicacions dels microcontroladors

Cada vegada existeixen més productes que incorporen un microcontrolador amb la finalitat d'augmentar considerablement les seves prestacions, reduir les seves dimensions i cost, millorar la seva fiabilitat i reduir el seu consum.

Els microcontroladors estan sent utilitzats en multitud de sistemes presents en la nostra vida diària, com poden ser joguines, microones, neveres, televisors, impressores, mòdems, el sistema d'arrencament d'un cotxe, etc. També els podem trobar en altres aplicacions que no ens són tan familiars com la instrumentació electrònica, control de sistemes d'una nau espacial, etc.

Una aplicació típica és utilitzar diferents microcontroladors per controlar petites parts d'un sistema. Aquests microcontroladors es poden comunicar entre ells i amb un processador central, probablement més potent, per compartir la informació i coordinar les seves accions. De fet, això és el que succeeix habitualment en qualsevol PC.

2.2.3. Recursos habituals en els microcontroladors

Al tenir tots els microcontroladors integrats en un xip, la seva estructura fonamental i les seves característiques bàsiques són molt semblants. Tots han de contenir els blocs essencials processador, memòria de dades i d'instruccions, línies de E/S, oscil·lador de rellotge i mòduls controladors de perifèrics. Tot i així, cada

fabricant intenta emfatitzar els recursos més idonis per a les aplicacions a les que es destinen preferentment.

2.2.3.1. Arquitectura bàsica

Encara que inicialment tots els microcontroladors van adoptar la arquitectura clàssica de von Neumann, actualment s'imposa l'arquitectura Harvard. La arquitectura de von Neumann es caracteritza per tenir una sola memòria principal on s'emmagatzemen dades i instruccions de forma indiferent. A aquesta memòria s'hi accedeix a través d'un sistema de busos únic (direccions, dades i control).

L'arquitectura de Harvard disposa de dues memòries independents, una que conté solament instruccions i una altra que conté solament dades. Les dues disposen dels seus respectius sistemes de busos d'accés i és possible realitzar operacions d'accés (lectura o escriptura) simultàniament a les dues memòries. Els microcontroladors PIC responen a l'arquitectura Harvard.

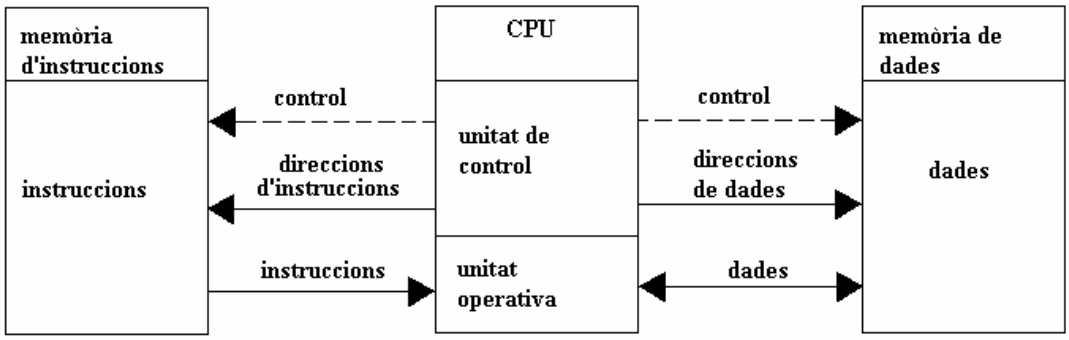


Figura 2.5 La estructura Harvard disposa de dues memòries independents per a dades i instruccions, permetent així accessos simultanis.

2.2.3.1.1. Unitat Central de Processament

És l'element més important del microcontrolador i determina les seves principals característiques, tant a nivell de hardware com de software. És l'encarregada de direccionar la memòria d'instruccions i rebre el codi de la instrucció en curs així com la cerca dels operants i l'emmagatzemament del resultat.

Existeixen tres orientacions en quant a arquitectura i funcionalitat dels processadors actuals:

1) CISC: Un gran nombre de processadors utilitzats en els microcontroladors estan basats en la filosofia CISC. Disposen de més de 80 instruccions màquina en el seu repertori, algunes de les quals són molt sofisticades i potents, requerint molts cicles per la seva execució. Un avantatge dels processadors CISC és que ofereixen al programador instruccions complexes que actuen com a macros.

2) RISC: Tant la indústria dels computadors comercials com la dels microcontroladors s'estan decantant cap a la filosofia RISC. En aquests processadors el repertori d'instruccions màquina és molt reduït i les instruccions són simples, i generalment, s'executen en un cicle. La senzillesa i rapidesa de les instruccions permeten optimitzar el hardware i el software del processador.

3) SISC: En els microcontroladors destinats a aplicacions molt concretes, el joc d'instruccions, a més de ser reduït, és específic, és a dir, les instruccions s'adapten les necessitats de l'aplicació prevista. Aquesta filosofia s'ha batejat amb el nom de SISC.

2.2.3.1.2. Memòria

En els microcontroladors la memòria d'instruccions i dades està integrada en el propi xip. Una part ha de ser no volàtil, de tipus ROM, i es destina a contenir el programa d'instruccions que governa l'aplicació. Una altra part de la memòria serà de tipus RAM, volàtil, i es destina a guardar les variables i les dades.

Hi ha dues peculiaritats que diferencien els microcontroladors dels computadors personals.

1) No existeixen sistemes d'emmagatzemament massiu com disc dur o disquets.

2) Com el microcontrolador es destina a una tasca en la memòria ROM, solament s'ha d'emmagatzemar un únic programa de treball.

La RAM en aquests dispositius és de poca capacitat ja que solament ha de contenir les variables i els canvis d'informació que es produeixin en el transcurs del programa. Per una altra part, com que solament existeix un programa actiu, no es requereix guardar una còpia del mateix en la RAM, s'executa directament des de la ROM.

Els usuaris de computadors personals estan habituats a manejar Megabytes de memòria, però els dissenyadors de microcontroladors treballen amb capacitat de ROM compreses entre 512 bytes i 8 kilobytes i de RAM compreses entre 20 i 512 bytes.

Segons el tipus de memòria ROM de que disposin els microcontroladors, l'aplicació i utilització dels mateixos és diferent. A continuació es descriuen els cinc tipus de memòria no volàtil que es poden trobar en els microcontroladors del mercat:

1) ROM amb màscara: És una memòria no volàtil, solament de lectura. El seu contingut es guarda durant la fabricació del xip. El elevat cost del disseny de la màscara solament fa aconsellable que s'utilitzin els microcontroladors amb aquest tipus de memòria quan es precisin quantitats superiors a varis milers d'unitats.

2) OTP: El microcontrolador conté una memòria no volàtil solament de lectura "programable una sola vegada per l'usuari. És l'usuari qui pot escriure el programa en el xip a través d'un senzill gravador controlat per un programa des d'un PC. La versió OTP és recomanable quan el cicle de disseny del producte és molt curt, o bé, en la construcció de prototips i sèries molt petites. Tant en aquest tipus de memòria com en la EPROM s'acostuma a utilitzar l'encriptació mitjançant fusibles per protegir el codi.

3) EPROM: Els microcontroladors que disposen de memòria EPROM es poden esborrar i gravar moltes vegades. La gravació es realitza, com en el cas de l'OTP, amb un gravador governat des d'un PC. Si, posteriorment, es desitja esborrar el contingut, disposen d'una finestra de cristall en la seva superfície per la que es sotmet a l'EPROM a raigs ultraviolats durant uns minuts. Les càpsules són de material ceràmic i són més cares que els microcontroladors amb memòria OTP que estan fets de material plàstic.

4) EEPROM: Es tracta de memòries solament de lectura, programables i esborrables elèctricament. Tant la programació com l'esborrat, es realitzen elèctricament des del propi gravador i sota el control programat d'un PC. Són molt còmodes i ràpides les operacions de gravat i esborrat. No disposen de finestra de cristall a la superfície. Els microcontroladors dotats de memòria de EEPROM, una vegada instal·lats en el circuit, es poden gravar i esborrar tantes vegades com es vulgui sense ser retirats del circuit. El nombre de vegades que es pot gravar i esborrar una memòria EEPROM és finit, per tant, no és recomanable una reprogramació contínua. Es va estenent en els fabricants la tendència d'incloure una petita zona de memòria EEPROM en les circuits programables per guardar i modificar còmodament una

sèrie de paràmetres que adeqüin el dispositiu a les condicions de l'entorn. Aquest tipus de memòria es relativament lent.

5) FLASH: Es tracta d'una memòria no volàtil, de baix consum, que es pot escriure i esborrar. Funciona com una ROM i una RAM, però consumeix menys i és més petita. A diferència de la ROM, és programable en el circuit i és més ràpida que la EEPROM. Les memòries EEPROM i FLASH són molt útils al permetre que els microcontroladors que les incorporen poden ser reprogramats en el circuit, és a dir, sense tenir que treure el circuit integrat de la targeta.

2.2.3.1.3. Portes d'entrada i sortida

La principal utilitat de les potes que posseeix la càpsula que conté el microcontrolador consisteix en suportar les línies de E/S que comuniquen la CPU amb els perifèrics exteriors.

Segons els controladors de perifèrics que posseeix cada model de microcontrolador, les línies de E/S es destinen a proporcionar el suport als senyals d'entrada, sortida i control.

2.2.3.1.4. Rellotge principal

Alguns microcontroladors disposen d'un circuit oscil·lador que genera una ona quadrada d'alta freqüència, que configura els impulsos del rellotge utilitzat en la sincronització de totes les operacions del sistema.

Generalment, el circuit de rellotge està incorporat en el microcontrolador i solament es necessiten alguns components exteriors per seleccionar i estabilitzar la freqüència de treball. Aquests components acostumen a consistir en un cristall de quars junt a elements passius.

Augmentar la freqüència del rellotge suposa disminuir el temps en que s'executen les instruccions, però suposa també un increment del consum d'energia.

2.2.4. Recursos especials

Cada fabricant ofereix nombroses versions d'una arquitectura bàsica de microcontrolador. En algunes amplia les capacitats de les memòries, en altres incorpora nous recursos, en altres redueix les prestacions al mínim per a aplicacions molt simples, etc. En cada cas s'ha d'utilitzar el microcontrolador que s'adapti més a l'aplicació per al que el volem utilitzar.

Entre els recursos especials més habituals trobem els temporitzadors, el watchdog, la protecció davant fallada d'alimentació, entre d'altres, que introduïm a continuació.

2.2.4.1. Temporitzadors

S'utilitzen per controlar períodes de temps i per comptar esdeveniments que succeeixen a l'exterior. Per a la mesura de temps es carrega un registre amb el valor adequat i a continuació aquest valor es va incrementant o decrementant al ritme dels impulsos del rellotge, o algun múltiple, fins que es desbordi i arribi a 0, moment en el que es produirà l'avís.

Quan es desitgin comptat esdeveniments que es materialitzin per canvis de nivell o flancs en alguna de les potes del microcontrolador, el registre s'anirà incrementant o decrementant al ritme dels impulsos.

2.2.4.2. Watchdog

Quan el computador personal es bloqueja per un error del software o una altra causa, es prem el botó per reinicialitzar el sistema. Un microcontrolador funciona sense control de un supervisor i de forma continuada les 24 hores del dia. El Watchdog consisteix en un temporitzador, que quan es desborda i passa per 0, provoca una reinicialització automàticament en el sistema.

S'ha de dissenyar el programa de treball que controla la tasca de manera que refresqui o inicialitzi al Watchdog abans de que provoqui la reinicialització. Si falla el programa o es bloqueja, no es refrescarà al Watchdog i, al completa la seva temporització provocarà la reinicialització.

2.2.4.3. Protecció davant fallada d'alimentació

Consisteix en un circuit que reinicialitza el microcontrolador quan el voltatge d'alimentació és inferior a un voltatge mínim. Mentre el voltatge d'alimentació sigui inferior a aquest voltatge mínim, el dispositiu es manté reinicialitzat, començant a funcionar normalment quan es sobrepassa aquest valor.

2.2.4.4. Estat de repòs

Són abundants les situacions reals de treball en que el microcontrolador ha d'esperar sense fer res a que es produeixi algun esdeveniment extern que el posi de nou en funcionament. Per estalviar energia, els microcontroladors disposen d'una instrucció especial que els posa en estat de repòs o de baix consum, en el qual els requeriment de potència són mínims. En aquest estat s'atura el rellotge principal i es congelen els seus circuits associats. Quan s'activa la interrupció ocasionada per l'esdeveniment esperat, el microcontrolador reprèn el seu treball.

2.2.4.5. Convertidor A/D

Els microcontroladors que incorporen un convertidor A/D poden processar senyals analògiques. Acostumen a disposar d'un multiplexor que permet aplicar a l'entrada del convertidor diverses senyals analògiques des de les potes del circuit integrat

2.2.4.6. Convertidor D/A

En aquest cas, es transformen les dades digitals obtingudes del processament del computador en el seu corresponent senyal analògic que es treu a l'exterior per una de les potes de la càpsula.

2.2.4.7. Comparador analògic

Alguns models de microcontroladors disposen internament d'un amplificador operacional que actua com a comparador entre un senyal fix de referència i una altra

variable que s'aplica per una de les potes de la càpsula. La sortida del comparador proporciona un nivell lògic 1 o 0 depenent de que un senyal sigui més gran o més petita que l'altre.

2.2.4.8. PWM

Són circuits que proporcionen a la seva sortida impulsos d'amplària variable, que s'ofereixen a l'exterior a través de les potes de l'encapsulat.

2.2.4.9. Ports de E/S digitals

Tots els microcontroladors destinen algunes de les seves potes a suportar línies de entrades/sortides digitals. En general, aquestes línies s'agrupen de vuit en vuit, formant ports.

Les línies digitals dels ports es poden configurar com entrada o com sortida, posant un 1 o un 0 en el bit corresponent del registre destinat a la seva configuració.

2.2.4.10. Ports de comunicació

Amb l'objectiu de dotar al microcontrolador amb la possibilitat de comunicar-se amb altres dispositius externs, alguns models disposen de recursos que permeten directament aquesta tasca.

Els que més destaquen són els següents:

1) UART: Adaptador de comunicació sèrie asíncrona.

2) USART: Adaptador de comunicació sèrie síncrona i asíncrona.

3) Porta paral·lela esclava per poder connectar el microcontrolador amb els bussos d'altres microprocessadors.

4) USB: Modern bus sèrie pels PC.

5) Bus I^2C: Interfície sèrie de fos fils desenvolupat per Philips

6) CAN: Permet l'adaptació amb xarxes de connexió multiplexades desenvolupat, conjuntament per Bosch i Intel, per al cablejat de dispositius en automòbils.

2.2.5. Desenvolupament del software

Un dels principals factors que es té en compte a l'hora de seleccionar un microcontrolador és el suport que disposa tant de software com de hardware. Un bon conjunt d'eines de desenvolupament pot ser decisiu per a l'elecció, ja que pot suposar una gran ajuda en el desenvolupament del projecte.

Les principals eines d'ajuda en el desenvolupament de sistemes basats en microcontroladors són els següents:

1) Ensamblador: És la principal eina de programació del microcontrolador que utilitza un llenguatge de baix nivell fent servir instruccions pròpies del microcontrolador. La programació en llenguatge ensamblador pot resultar complicada per un principiant, però permet desenvolupar programes molt eficients, ja que ofereix al programador el domini absolut del sistema. Els fabricants acostumen a proporcionar el programa ensamblador de forma gratuïta i en qualsevol cas sempre es pot trobar una versió gratuïta per als microcontroladors més populars.

2) Compiladors: La programació en un llenguatge d'alt nivell, com el C o el Basic, permet disminuir el temps de desenvolupament del producte. No obstant, si no es programa cuidadosament, el codi resultant pot ser molt més ineficient que el programat amb ensamblador. Les versions més potents acostumen a ser molt cares, encara que pels microcontroladors més populars es poden trobar versions limitades o fins i tot compiladors gratuïts.

3) Depuració: A causa de que els microcontroladors controlaran dispositius físics, els programadors necessiten eines que els permetin comprovar el bon funcionament del microcontrolador quan és connectat a la resta de circuits.

4) Simulador: Són capaços d'executar en un PC programes realitzats pel microcontrolador. Els simuladors permeten tenir un control absolut sobre l'execució del programa, sent ideals per a la depuració d'aquests. El seu gran inconvenient és que és difícil simular l'entrada i sortida de dades del microcontrolador. Tampoc es té en compte el possible soroll en les entrades, però al menys, permeten el pas físic de la implementació d'un mode més segur i menys costós.

5) Plaques d'avaluació: Es tracta de petits sistemes amb un microcontrolador ja muntat i que s'acostuma a connectar a un PC des del que es carreguen els programes que s'executen en el microcontrolador. Les plaques acostumen a incloure visualitzadors LCD, teclats, leds, fàcil accés als pins de E/S, etc. El sistema operatiu de la placa rep el nom de programa monitor. El programa monitor d'algunes plaques d'avaluació, a part de permetre carregar programes i dades en la memòria del microcontrolador, pot permetre en qualsevol moment realitzar l'execució pas a pas o modificar els valors emmagatzemats als registres o a la memòria.

6) Emuladors en circuit: Consisteix en un instrument que es col·loca entre el PC i el sòcol de la targeta del circuit imprès on s'allotjarà el microcontrolador definitiu. El programa és executat des del PC, però per la targeta d'aplicació és com si ho fes el microcontrolador que després anirà al sòcol. Presenta una pantalla amb tota la informació tal i com després succeirà quan es col·loqui la càpsula.

2.3. Autòmat o Microcontrolador?

Una vegada introduïts els dos elements principals del sistema podem observar que els dos elements contenen un arquitectura bàsica força semblant en quant a estructura. És per aquest motiu pel que en la taula que mostrem a continuació pretenem mostrar les principals diferències que trobem entre els dos elements i així determinar en quines ocasions serà més adequat utilitzar un element o bé un altre.

Microcontrolador	**Autòmat**
La majoria dels microcontroladors poden executar entre 1 milió i 5 milions d'instruccions per segon. És ideal per processaments molt ràpids en temps real.	Un autòmat programable executa entre 300 i 30000 instruccions per segon
S'acostuma a programar amb un llenguatge de baix nivell com ensamblador o C.	Es programa sempre amb un llenguatge d'alt nivell.

La utilització d'un microcontrolador requereix un període d'aprenentatge relativament llarg.	Una persona no experimentada pot començar a realitzar programes el primer dia. Pràcticament no requereix temps d'aprenentatge o coneixement previs.
És molt més flexible en el sentit de què totes les entrades/sortides són de propòsit general. Així obliga a l'usuari a desenvolupar la seva pròpia electrònica.	Cada entrada/sortida té una funció preestablerta. Tota l'electrònica ja està desenvolupada.
La seva electrònica és molt més simple, així el seu consum és inferior a 50 mA. Es poden aconseguir aplicacions concretes amb consums molt baixos de l'ordre dels 5 mA.	El seu consum i el seu pes són relativament alts, no sent adequats per a aplicacions portàtils o de llarga duració que s'hagin d'alimentar amb piles. El seu consum és superior als 500 mA i en un sistema de 64 entrades/sortides pot arribar als 3 A.
Per al mateix nombre d'entrades/sortides té un preu notablement inferior, però s'ha de tenir en compte que treballa amb tensions de 5 volts i corrent molt petites, de l'ordre de miliamperes.	El seu preu és superior, però les entrades/sortides poden acceptar directament tensions de 5, 12, 24, 110 i 220 volts.

3. Simulació de la planta automatitzada

Per realitzar la construcció de la planta automatitzada simulada hem utilitzat un microcontrolador PIC16F873 i per programar-lo hem utilitzat el software MPLAB IDE v7.51 de la casa MICROCHIP. Aquest software ens permet programar el microcontrolador en llenguatge C, fet que ens permet relacionar més còmodament la lògica de la planta amb la seva programació.

Hem de tenir en compte que el disseny de la planta ha de ser independent del control que hi volem aplicar. Per tant, a l'hora de programar el microcontrolador no s'ha de pensar en el funcionament esperat o òptim de la planta, sinó que es limiti a reaccionar obeint les ordres que li arribin del PLC. Serà el programador del control qui haurà de tenir en compte les característiques de la planta per realitzar el control. Per aconseguir aquest fet, s'han de tenir en compte totes les situacions possibles en què es pot trobar la planta i les seves possibles reaccions.

El control de la planta ve donat pel programa del PLC. Per a que aquest sigui efectiu i treballi en temps real s'ha de tancar el llaç del sistema. Així, aconseguim que el sistema estigui realimentat constantment, i per tant, que l'evolució de la planta simulada sigui la mateixa que obtindríem connectant el PLC a una planta real.

Per poder controlar la planta, connectem les sortides del PLC, que ens determinaran el control que volem aplicar sobre la planta, amb les entrades del microcontrolador, per a que se li apliqui el control. I per tancar el llaç del sistema connectem les sortides del microcontrolador amb les entrades del PLC.

En una planta real, les sortides del microcontrolador podrien ser diferents tipus de sensors. Al connectar-les amb les entrades del PLC el que aconseguim és conèixer en tot moment l'estat en que es troba la planta.

Hem de tenir en compte que al tractar-se d'una eina pensada per detectar possibles errors en el control de la planta, quan el programa de control no sigui el adequat la simulació ha de reaccionar erròniament i en el pitjor dels casos informar a l'usuari de que ha trencat alguna part de la planta.

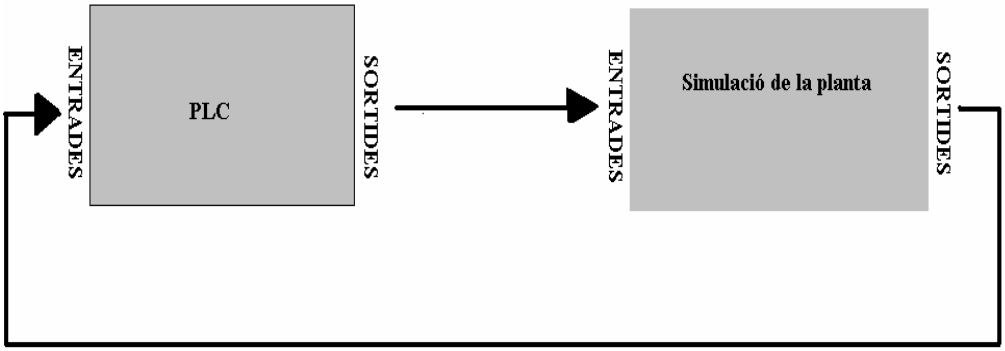


Figura 3.1 Esquema de la connexió entre PLC i planta simulada

3.1. Planta simulada: L'ascensor

Per tal d'il·lustrar el funcionament del simulador de plantes, hem triat com exemple un ascensor de tres plantes. Ens hem decidit per aquest cas perquè creiem que és una planta que té un funcionament conegut per tothom, ens ofereix diferents possibilitats a l'hora de programar el control que volem aplicar sobre ell i ens ofereix la possibilitat de poder provar l'efecte d'un control inadequat per part de l'usuari, com ara la possibilitat de cremar el motor en cas de no apliqui un control adequat sobre el sistema i per tant ens dóna una possibilitat d'error clara. El codi de programació complet del microcontrolador per aquesta planta es pot consultar a l'Annex I d'aquesta memòria.

3.1.1. Característiques específiques de la planta

Concretament, la planta que hem triat per simular-la amb el microcontrolador consisteix en un ascensor de tres plantes amb indicadors de nivell de planta i de porta oberta/tancada. És important fer conèixer les especificacions de la planta a l'usuari encarregat de programar el control, perquè per realitzar un control adequat s'han de tenir en compte les diferents variables i situacions en què es pot trobar la planta.

Tenim tres entrades de control sobre la nostra planta:

1) Ordre de pujar.

2) Ordre de baixar.

3) Ordre d'obrir la porta.

L'ordre de pujar és l'encarregada de dir-li a l'ascensor que pugi. En una planta real s'aplicaria sobre el motor encarregat de moure tot el sistema i li indicaria el sentit de gir adequat perquè l'ascensor anés en sentit ascendent.

L'ordre de baixar és l'oposada a l'anterior. És l'encarregada de dir-li a l'ascensor que baixi. En una planta real s'aplicaria sobre el motor encarregat de moure tot el sistema i li indicaria el sentit de gir adequat, oposat al del cas anterior, perquè l'ascensor anés en sentit descendent.

L'ordre d'obrir la porta és l'encarregada de dir-li a la porta de l'ascensor que s'obri. Quan no s'aplica aquest senyal la porta es mantindrà tancada.

Tenim cinc sortides que ens determinaran l'estat en què es troba la nostra planta:

1) 3 Sortides de posició.

2) Estat de porta oberta.

3) Estat d'error, motor cremat.

Les tres primeres sortides ens permetran conèixer en quina planta es troba l'ascensor en tot moment. Hi ha cinc possibilitats diferents. El nostre ascensor està preparat per circular a través de tres plantes, a més entre cada planta i tenim una zona que ens indica que l'ascensor es troba en una zona en què ja ha sortit de la planta anterior, però encara no ha arribat a la planta següent.

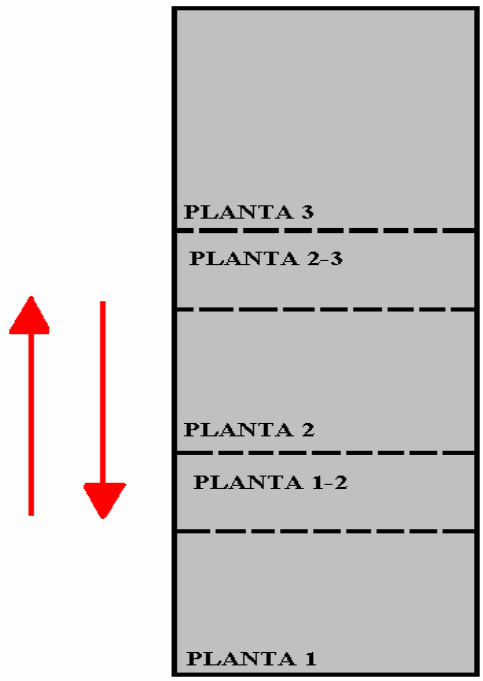


Figura 3.2 Esquema de les plantes per les que pot circular l'ascensor

Per conèixer l'estat de la planta en què es troba l'ascensor utilitzem el següent codi:

Planta / Sortida	Sortida 1	Sortida 2	Sortida 3
Planta 1	Activada	Desactivada	Desactivada
Planta 1-2	Activada	Activada	Desactivada
Planta 2	Desactivada	Activada	Desactivada
Planta 2-3	Desactivada	Activada	Activada
Planta 3	Desactivada	Desactivada	Activada

Quan l'ascensor es troba en una de les tres plantes i li arriba el senyal d'entrada de pujar o baixar immediatament es posa en moviment. Hem determinat que tardi un temps de tres segons des de que es posa en moviment fins que arriba a

la zona intermèdia. Aquest temps representa una característica de la planta i significa que l'indicador associat a que la cabina de l'ascensor està en una planta específica es pot veure durant aquest temps. Evidentment, aquest paràmetre es pot canviar fàcilment en el programa del microcontrolador.

Per tal de superar la zona intermèdia, ha d'arribar el senyal d'entrada de pujar o baixar i mantenir-se activada durant cinc segons. Si es dóna aquesta situació, llavors l'ascensor arribarà a la planta següent.

La sortida de porta oberta/tancada estarà activada quan la porta de l'ascensor estigui oberta. Aquesta sortida va directament relacionada amb el senyal d'entrada d'obrir la porta. Quan s'apliqui el senyal d'entrada d'obrir la porta, el senyal de sortida s'activarà i quan es deixi d'aplicar el senyal d'entrada, el senyal de sortida es desactivarà.

Finalment, el senyal de sortida d'error s'activarà sempre que s'hagi produït una situació en què el motor hagi estat forçat més del compte i representa que aquest s'haurà cremat. En aquest cas, el funcionament de la planta queda bloquejat en l'estat en el que es troba i s'ha de reinicialitzar el sistema per tal de poder tornar a fer funcionar la planta simulada. Aquesta situació vindrà produïda per un mal control del sistema, ja que es tracta d'una situació no desitjada per part de l'usuari.

Els diferents casos que poden fer arribar la planta a aquesta situació són els següents:

1) Ens trobem a la planta 3 i arriba senyal d'entrada de pujar.

2) Ens trobem a la planta 1 i arriba senyal d'entrada de baixar.

Ens trobem en qualsevol planta i arriben senyals d'entrada de pujar i de baixar al mateix temps.

Perquè s'activi el senyal d'error, s'ha de produir una de les situacions anteriorment citades durant un temps de cinc segons. Aquests cinc segons no corresponen al temps real que triga un motor en trobar-se en una situació en que és forçat i es crema, però al tractar-se d'una eina de simulació per a la detecció d'errors tampoc interessa que l'usuari hagi de perdre molt de temps cada vegada que posa en marxa la simulació.

S'ha de tenir en compte que els diferents valors de temps es poden modificar fàcilment en el codi de la simulació de la planta, ja que estan definits com constants. En algun cas pot ser que el valor del temps no s'adeqüi al valor de temps que tardaria la planta real, però igual com s'ha comentat anteriorment al tractar-se d'una simulació, no ens interessa que l'usuari perdi molt temps amb la simulació, simplement que s'adoni dels seus errors per poder anar millorant el programa de control.

3.1.2. Algoritme del codi del microcontrolador

A continuació mostrem l'algoritme, en trets generals, a partir del qual hem dissenyat el codi en llenguatge C per programar el microcontrolador perquè aquest simulés el funcionament de la planta automatitzada que hem triat, un ascensor.

```
Programa_Principal
{
      Configuració_Microcontrolador();
      Declaració_variables();
      Inicialització();

      Mentre (NO_ERROR) fer
      {
            Si (Senyal_obrir_porta) llavors
            {
                  Obrir_porta;
            }sinó{
                  Porta_tancada;
            }

            Si((Cabina_en_planta1)&(Ordre_pujar)&(NO_Ordre_baixar))llavors
            {
                  Comptadors_plantes=0;
                  Comptador_error=0;
                  Esperar;
                  Cabina_en_planta12;
            }

            Si((Cabina_en_planta1)&(Ordre_baixar)&(NO_Ordre_pujar))llavors
            {
                  Comptadors_plantes=0;
                  Comptador_error=Incrementar;
            }

            Si((Cabina_en_planta2)&(Ordre_pujar)&(NO_Ordre_baixar))llavors
            {
                  Comptadors_plantes=0;
                  Comptador_error=0;
                  Esperar;
                  Cabina_en_planta23;
            }

            Si((Cabina_en_planta2)&(Ordre_baixar)&(NO_Ordre_pujar))llavors
            {
                  Comptadors_plantes=0;
                  Comptador_error=0;
                  Esperar;
                  Cabina_en_planta12;
            }
```

```
Si((Cabina_en_planta3)&(Ordre_pujar)&(NO_Ordre_baixar))llavors
{
      Comptadors_plantes=0;
      Comptador_error=Incrementar;
}

Si((Cabina_en_planta3)&(Ordre_baixar)&(NO_Ordre_pujar))llavors
{
      Comptadors_plantes=0;
      Comptador_error=0;
      Esperar;
      Cabina_en_planta23;
}

Si((Cabina_en_planta12)&(Ordre_pujar)&(NO_Ordre_baixar))llavors
{
      Comptadors_planta_12=Incrementar;
      Altres_comptadors_planta=0;
      Comptador_error=0;
}

Si((Cabina_en_planta_12)&(Ordre_baixar)&(NO_Ordre_pujar))llavors
{
      Comptadors_planta_21=Incrementar;
      Altres_comptadors_planta=0;
      Comptador_error=0;
}

Si((Cabina_en_planta_23)&(Ordre_pujar)&(NO_Ordre_baixar))llavors
{
      Comptadors_planta_23=Incrementar;
      Altres_comptadors_planta=0;
      Comptador_error=0;
}

Si((Cabina_en_planta_23)&(Ordre_baixar)&(NO_Ordre_pujar))llavors
{
      Comptadors_planta_32=Incrementar;
      Altres_comptadors_planta=0;
      Comptador_error=0;
}

Si((NO_Ordre_pujar)&(NO_Ordre_baixar))llavors
{
      Comptadors_plantes=0;
}

Si(Comptador_error>Límit)llavors
{
      Error;
}

Si(Comptador_planta12>Temps_pas_de_planta)llavors
```

```
            {
                  Cabina_en_planta2;
            }

            Si(Comptador_planta12>Temps_pas_de_planta)llavors
            {
                  Cabina_en_planta2;
            }

            Si(Comptador_planta21>Temps_pas_de_planta)llavors
            {
                  Cabina_en_planta1;
            }

            Si(Comptador_planta23>Temps_pas_de_planta)llavors
            {
                  Cabina_en_planta3;
            }

            Si(Comptador_planta32>Temps_pas_de_planta)llavors
            {
                  Cabina_en_planta2;
            }

      }
}
```

4. Disseny de la placa

Una vegada tenim el microcontrolador programat, necessitem disposar d'un medi que permeti a l'usuari conèixer l'estat en què es troba la planta en tot moment. Per aquesta raó hem dissenyat una placa on connectarem el nostre microcontrolador i a través de leds indicadors l'usuari estarà en tot moment informat. Naturalment, aquesta placa està totalment adaptada per a poder ser connectada directament a un autòmat programable.

Les característiques de la placa creiem que són adequades per cobrir les necessitats que planteja el nostre problema. La seva funció bàsica consisteix què l'usuari, a través dels diferents indicadors, sigui capaç d'entendre tota la informació que li dóna, com si en comptes de tenir al davant aquesta placa hi tingués una maqueta o la planta real.

L'esquema elèctric complet de la placa el podem trobar a l'Annex III.

4.1. Diferents parts de la placa

Hem dividit l'esquema electrònic de la placa en sis zones ben diferenciades. D'aquestes sis zones, podem dir que el microcontrolador n'és l'eix central, ja que totes les altres zones estan connectades amb aquest. Cada zona té la seva funció dins de la placa, però totes depenen directament del funcionament del microcontrolador, i algunes també depenen del funcionament de l'autòmat. Les diferents zones de la placa són les següents: *1. Microcontrolador, 2. Regulador de tensió, 3. Circuit Oscil·lador, 4. Entrades, 5. Sortides, 7. Indicadors.*

La placa està dissenyada inicialment per un màxim de cinc sortides i tres entrades. Aquest límit és suficient per programar la planta d'exemple que il·lustrarà tot el projecte. De fet, hem ajustat les característiques de la placa a aquesta planta per reduir el cost, no obstant, el nombre d'entrades i sortides és fàcilment modificable ja que simplement haurem d'afegir els circuits corresponents per cada entrada o sortida que volem tenir de més i ajustar la configuració del microcontrolador correctament.

A continuació es detallen les característiques tècniques de cada zona, juntament amb la seva funció.

4.1.1. El microcontrolador

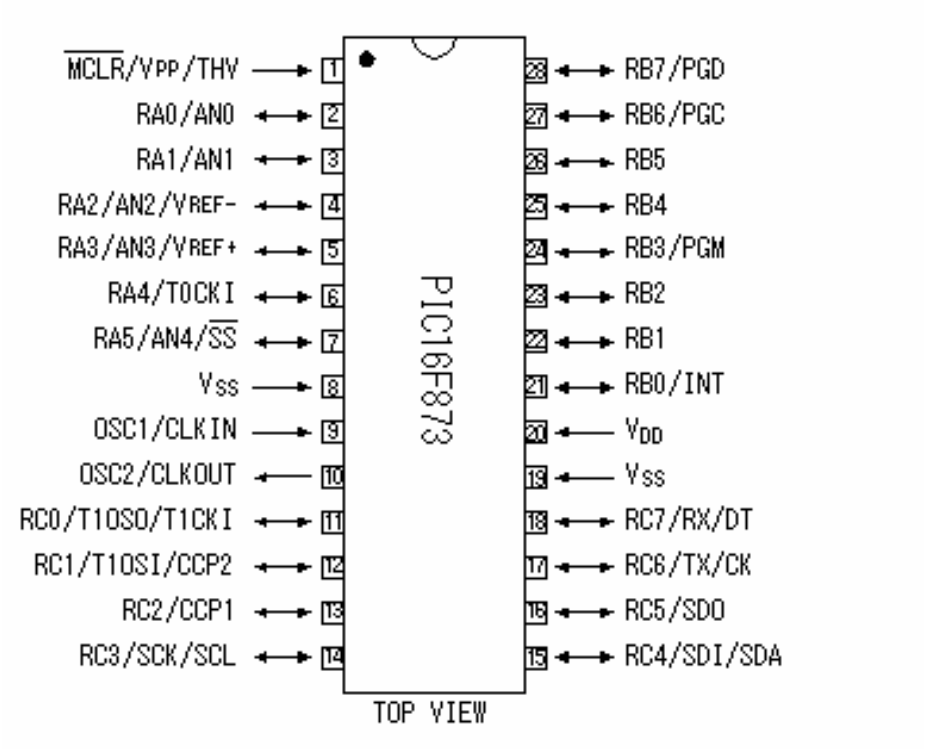


Figura 4.2 Esquema del microcontrolador

El microcontrolador que hem utilitzat per simular la nostra planta, tal i com hem indicat anteriorment és el PIC16F873 de la casa MICROCHIP. Aquest model consta de 28 pins.

A continuació detallem els pins que hem utilitzat i quina és la seva utilitat:

Pins 1 i 20: Aquests dos pins estan connectats a la sortida del regulador de tensió (veure *figura 4.3*) i són on tindrem la tensió necessària per alimentar el microcontrolador.

Pins 8 i 19: Aquests dos pins van connectats a terra. En l'esquema complet de la placa es pot observar que els connectem a la terra del regulador de tensió. Hem de tenir en compte que la presa de terra serà comuna per tots els diferents dispositius que trobem en els diferents circuits, a més de l'autòmat.

Pins del 2 al 7: En la nostra planta únicament utilitzem del pin 2 al pin 4, però tal i com hem comentat anteriorment seria factible utilitzar-los tots. Aquests pins estan configurats perquè siguin les entrades de la planta simulada. Així, és per aquests pins per on arribaran les ordres de control provinents de l'autòmat, un cop aquestes hagin estat adaptades pels optoacobladors.

Pins 9 i 10: En aquests dos pins hi tenim connectat el circuit oscil·lador, que detallem més endavant en la *figura 4.4.*

Pins del 11 al 18: En la nostra placa aquest pins no tenen cap funció, per aquest motiu no estan connectats en lloc. Per a altres plantes podrien tenir alguna utilitat, per exemple podrien funcionar com entrades o sortides.

Pins del 21 al 28: Finalment, d'aquest conjunt de pins en la nostra planta utilitzem del pin 21 al pin 25. Tal i com hem comentat en anteriors ocasions es podrien utilitzar tots si fos necessari, realitzant les modificacions pertinents. Aquests pins estan configurats perquè actuïn com a sortides de la planta simulada. Així, aquests pins seran els que enviaran la informació sobre l'estat de la planta a l'autòmat perquè aquest conegui l'estat que es troba la planta. Ara bé aquests pins no estan connectats directament a l'autòmat, ja que és necessari adequar el senyal i és per aquest motiu que estan connectats al circuit de transistors que s'explica posteriorment i que podem observar en la *figura 4.6*. També hem d'afegir que en aquests pins hi tindrem connectats els indicadors (*figura 4.7*) perquè l'usuari també pugui conèixer la informació que fa referència a l'estat que es troba la planta.

Aquest element, podríem dir, que és l'eix central de la placa, ja que la resta de parts del circuit n'hi estan connectades. La resta de zones que hem definit en la *figura 4.1* no influeixen en el funcionament de la planta, únicament són circuits que ens permeten que el microcontrolador funcioni correctament, o bé, ens adeqüen els diferents senyals per tal de poder establir la connexió entre els dos elements principals del sistema.

Per facilitar la comoditat dels usuaris i per a poder aprofitar més els recursos que ens ofereix la placa, no connectarem directament el microcontrolador a la placa. A la placa i tindrem un sòcol de 28 pins i així podrem treure i posar el microcontrolador quan a l'usuari li convingui. Aquest fet és important, ja que interessa no limitar la utilitat de la placa a una única planta. Així si volem comprovar el funcionament d'una altra planta, únicament haurem de treure el microcontrolador i programar-lo adequadament, per tot seguit tornar-lo a posar en el sòcol i per tant connectar-lo a la placa.

A més, creiem que és interessant treballar en sòcols amb els elements electrònics més delicats, ja que en cas de que sofrissin algun dany, puguin ser reemplaçats sense tenir la necessitat de dessoldar i soldar, amb totes les complicacions que això pot provocar.

4.1.2. Regulador de tensió

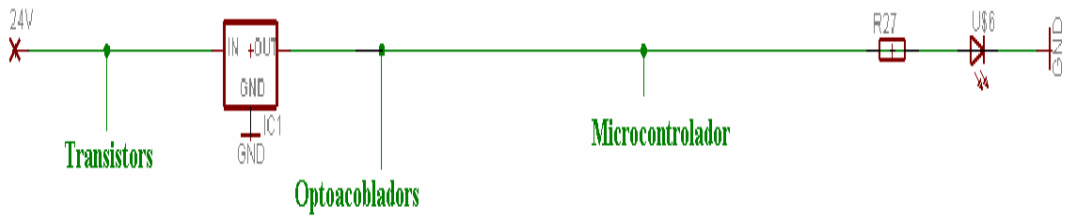


Figura 4.3 Esquema elèctric del circuit del regulador de tensió

Per posar en funcionament la placa aprofitem la tensió que hi ha a la font d'alimentació de l'autòmat. Els autòmats amb els que hem treballat, els SIEMENS CPU 214, tenen una font d'alimentació que ens dóna 24 volts de tensió contínua. Aquests 24 volts també els aprofitem en el circuit de transistors que comentem posteriorment en l'apartat referent al circuit de transistors.

Ara bé, per alimentar el microcontrolador necessitem una tensió inferior a 24 volts. Una tensió adequada per alimentar el microcontrolador són 5 volts. És per aquest motiu que utilitzem el regulador de tensió.

El regulador de tensió que hem utilitzat és el KA7805. La seva funció és simple, quan a la seva pota d'entrada i connectem una tensió contínua amb uns valors entre 7,5 i 24 volts, a la pota de sortida hi obtindrem una tensió constant aproximada de 5 volts, entre 4,75 i 5,25 volts. També, tenim en compte que el corrent màxim que pot manejar el regulador de tensió és d'1 amper, en el nostre cas és un valor suficient per poder tenir en funcionament les diferents zones de la placa. En la *figura 4.3.1* podem observar una representació del regulador de tensió.

1. Input 2. GND 3. Output

Figura 4.3.1 Regulador de tensió KA7805

A la pota 1 hi connectem els 24 volts de la font d'alimentació de l'autòmat, la pota 2 està connectada a terra, que és comuna per a tots els elements de la placa i a la pota 3 és on obtenim els 5 volts. En aquesta tercera pota hi connectarem els dos pins del microcontrolador, que hem comentat anteriorment, per a alimentar a aquest. A més, en aquesta pota també hi tenim connectat un led indicador, que ens permet conèixer en tot moment si la placa està alimentada i, per tant, en funcionament.

Hem utilitzat un led de color vermell, que s'encendrà quan la placa estigui alimentada. Per al correcte funcionament del led hem connectat una resistència de 470Ω en sèrie entre la sortida de la pota 3 del regulador de tensió i l'ànode del led. El càtode de led està connectat directament a terra.

La intensitat que ens circula a través del led s'obté a partir dels càlculs següents, i no ha de ser superior als 20mA, però ha de ser suficient perquè el led s'il·lumini:

$$V_{regulador} = V_{led} + R \cdot I$$
$$5 = 1.6 + 470 \cdot I$$
$$I = 7.23mA$$

En l'expressió anterior, tenim que $V_{regulador}$ és la tensió que obtenim a la pota de sortida del regulador de tensió, V_{led} és la tensió en borns del led (pels leds de color vermell és d'uns 1,6 volts), R és el valor de la resistència i finalment I és el valor del corrent que circularà a través d'aquesta branca.

Finalment també hem de comentar que connectem els 5 volts que obtenim a la sortida del regulador de tensió als optoacobladors. La seva funció l'expliquem en l'apartat 4.1.4.

4.1.3. Circuit oscil·lador

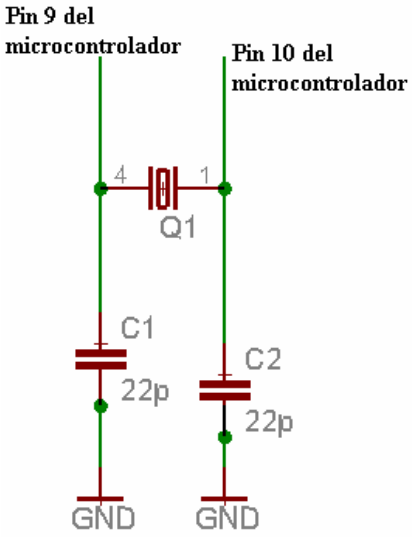


Figura 4.4 Esquema del circuit oscil·lador

Aquesta part de la placa és fonamental per al correcte funcionament de microcontrolador. El circuit oscil·lador és l'encarregat d'indicar al microcontrolador la velocitat a la que aquest ha de treballar.

El circuit oscil·lador genera polsos de rellotge. La freqüència màxima del rellotge, segons les característiques facilitades pel fabricant del microcontrolador, és de 20 MHz. En el nostre cas, en la programació del microcontrolador l'hem configurat perquè treballi a 4MHz, és per aquest motiu que utilitzem un cristall d'aquesta velocitat.

Hem de comentar que aquest circuit i els valors dels condensadors que hem utilitzat ens vénen facilitats pel fabricant. La condició bàsica perquè el circuit oscil·lador funcioni correctament consisteix en que el valor dels dos condensadors sigui el mateix. En el nostre cas, per aquesta velocitat, hem utilitzat dos condensadors de 22pF que estan connectats tal i com es pot observar en la *figura 4.4.*

4.1.4. Entrades

A continuació tenim l'esquema del circuit electrònic necessari per adequar els senyals provinents de l'autòmat perquè aquests puguin ser reconeguts pel microcontrolador.

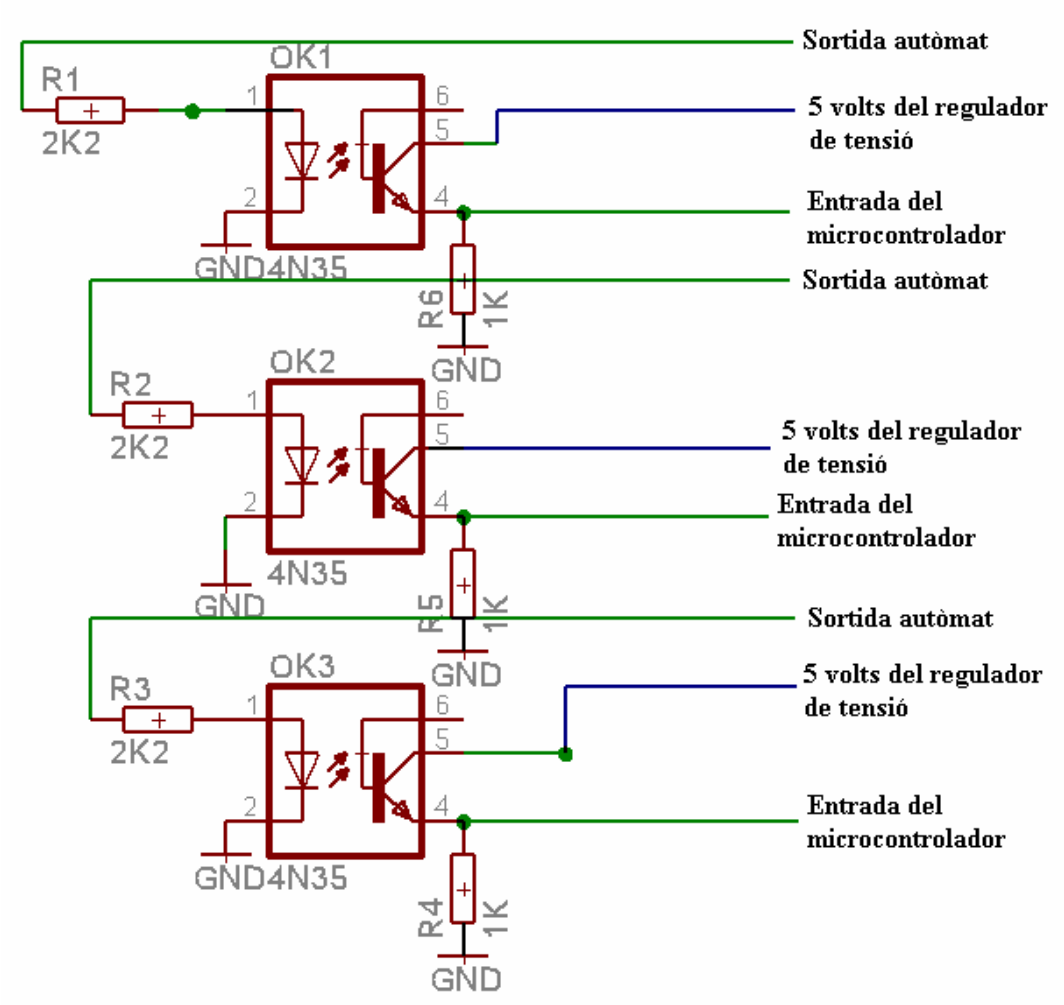


Figura 4.5 Esquema del circuit dels optoacobladors

4.1.4.1. Definició i funcionament bàsic

Els optoacobladors són conjunts integrats de components que permeten l'acoblament de senyals des d'un circuit a un altre per mitjà de llum visible o infraroja.

Una característica molt interessant d'aquests elements és que els dos circuits que estan en acoblament estan completament aïllats elèctricament. A més, actualment els optoacobladors tenen unes dimensions reduïdes, tenen un preu bastant econòmic i són plenament compatibles amb els circuits digitals.

Tots els optoacobladors contenen un dispositiu emissor de senyal lluminós, acostuma a ser un diode led, i un dispositiu receptor d'aquest senyal. Aquest dispositiu receptor es pot aconseguir de diferents formes, algunes de les més utilitzades són el fotodíode i el fototransistor.

El seu funcionament bàsic consisteix en que aplicant un senyal elèctric al diode led aquest la converteix en un senyal lluminós. Aquest senyal lluminós el receptor és capaç de convertir-lo en senyal elèctric el que genera un senyal de corrent de base, fet que provoca que el transistor entri en la zona de saturació i, per tant, actuï com un interruptor tancat entre el seu emissor i el seu col·lector. Així aconseguim tenir els 5 volts a l'emissor i, per tant, a l'entrada del microcontrolador.

4.1.4.2. Circuit d'optoacobladors utilitzat

Figura 4.6 Imatge d'un optoacoblador 4N26

Per al nostre circuit hem utilitzat un optoacoblador típic, com és el model 4N26. La principal funció que tenen els optoacobladors en la nostra placa consisteix en adequar el senyal provinent de l'autòmat (24 volts) perquè pugui ser connectada amb el microcontrolador (5 volts). Així doncs, quan arribi senyal de part de l'autòmat, a la sortida del optoacoblador haurem de tenir senyal perquè aquesta arribi al microcontrolador. Mentre que quan l'autòmat no envia senyal, a la sortida no n'hi haurem de tenir.

Tal i com hem comentat anteriorment, aquests dispositius ens permeten aïllar completament els dos circuits elèctrics. Aquest fet és molt important ja que ens garanteix una alta protecció per al microcontrolador.

Podem observar en la *figura 4.6* que els 4N26 disposen de sis potes. La pota 3 i la pota 6 les deixarem sense connectar, ja que per a la nostra aplicació no les necessitem.

A la pota 1 hi arribarà el senyal provinent de la sortida de l'autòmat. Quan l'autòmat enviï senyal, el diode led emetrà senyal lluminós. Perquè el diode led funcioni correctament hem connectat una resistència en sèrie de 2K2 entre la sortida de l'autòmat i l'ànode del diode led (pota 1).

A partir dels càlculs següents obtenim la intensitat que circula pel diode led, per aconseguir que emeti el senyal de llum.

$$V_{automàt} = V_{led} + I \cdot R$$
$$24 = 1.1 + I \cdot 2200$$
$$I = 10.4mA$$

On tenim que $V_{autòmat}$ són els 24 volts del senyal de les sortides de l'autòmat, V_{led} és la tensió en els borns del diode led, segons les dades facilitades pel fabricant d'uns 1,1 volts, R és el valor de la resistència i finalment I és el valor de la intensitat que circularà a través del diode led.

La pota 2 correspon al càtode del diode led i està connectada directament a terra.

A la quarta pota hi obtindrem el senyal que ha d'arribar al microcontrolador. Així doncs, aquestes potes estan connectades amb els pins del microcontrolador que estan configurats com entrades. A més, en aquesta pota també hi hem connectat una resistència d'1kΩ connectada a terra. La seva funció és únicament de protecció del microcontrolador, així quan tinguem 5 volts a la quarta pota del optoacoblador una part de la intensitat (5mA) circularà a través d'aquesta resistència i, per tant, no circularà tota la intensitat cap al microcontrolador, evitant així possibles valors superiors als màxims permesos en els pins d'aquest dispositiu i que aquest es danyi.

Finalment, a la pota 5 hi tenim connectats els 5 volts provinents del regulador de tensió. Aquests 5 volts són els que obtindrem a la sortida (pota 4) quan l'emissor de llum emeti suficient senyal lluminós.

4.1.5. Sortides

En la *figura 4.7* tenim l'esquema del circuit electrònic dissenyat per adequar els senyals provinents del microcontrolador perquè l'autòmat els pugui reconèixer.

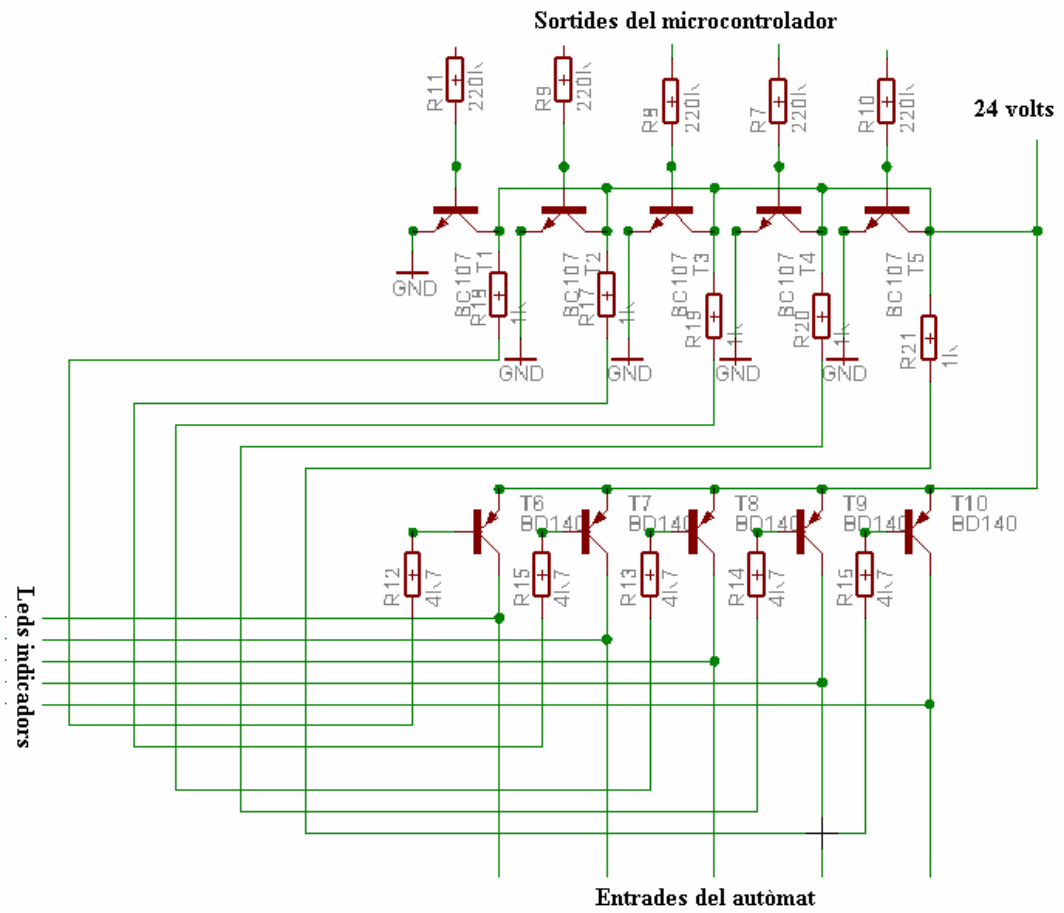


Figura 4.7 Esquema del circuit de transistors

4.1.5.1. Funcionament dels transistors PNP i NPN

Per entendre millor el funcionament del nostre circuit de transistors, a continuació farem una breu explicació sobre els transistors i les seves diferents regions de treball.

En el nostre circuit hem utilitzat transistors del tipus PNP i del tipus NPN, en la *figura 4.7.1* podem observar-ne el seu esquema.

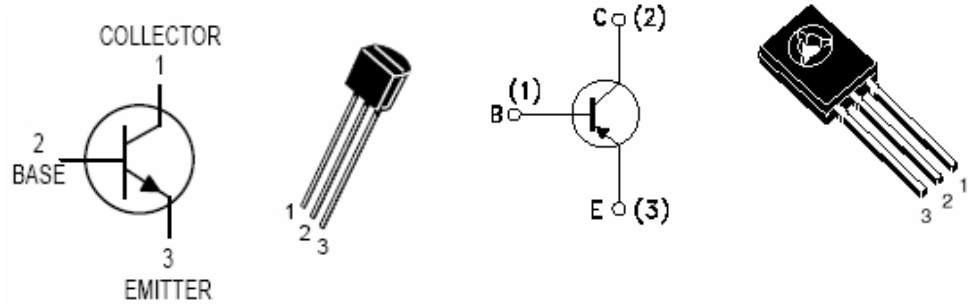


Figura 4.7.1 A l'esquerra esquema del transistor BC547 (NPN) i a la dreta del BD138 (PNP)

El transistor bipolar és un dispositiu electrònic d'estat sòlid consistent en dos unions PN molt properes entre si, que ens permet controlar el pas d'un corrent en funció d'un altre.

Un transistor bipolar està format per dos unions PN en un sol cristall semiconductor, separat per una regió estreta, així podem diferenciar-ne tres zones.

Emissor: Aquesta zona està fortament dopada i es comporta com un metall.

Base: Aquesta zona és molt estreta i ens separa l'emissor del col·lector.

Col·lector: És com la base, però de dimensions molt mes grans.

En aquests transistors podem diferenciar tres zones de funcionament diferents:

Zona de tall: Un transistor està en tall quan el corrent que circula pel col·lector i per l'emissor és igual a zero. En aquest cas la tensió entre el col·lector i l'emissor de l'emissor és el voltatge d'alimentació del circuit, com que no circula corrent no hi haurà caiguda de tensió. Normalment això succeeix quan el corrent de base és zero.

Zona de saturació: Un transistor està en saturació quan el corrent que circula pel col·lector i per l'emissor és igual a la corrent màxima del transistor. En aquest cas el corrent màxim depèn del voltatge d'alimentació del circuit i de les resistències connectades en el col·lector, en l'emissor o en els dos. Aquest cas normalment es presenta quan el corrent de base és suficientment gran com per induir un corrent de col·lector β vegades més gran.

$$I_c = I_B \cdot \beta$$

Zona activa: Finalment un transistor es troba en la zona activa quan no es troba ni en la zona de saturació ni en la zona de tall. En aquesta zona el corrent de col·lector depèn principalment del corrent de base, de β, que és el guany de corrent d'un amplificador i és una dada que dóna el fabricant i de les resistències que hi hagi connectades en el col·lector i l'emissor. Aquesta regió és la més important si es desitja utilitzar el transistor com un amplificador de corrent

4.1.5.2. El nostre circuit

En aquest cas, necessitem un circuit que, a partir dels senyals que tenim a les sortides del microcontrolador (5 volts), sigui capaç d'adequar el senyal perquè l'autòmat pugui treballar amb ella. La principal funció que ha de realitzar el circuit és obtenir una tensió de contínua suficient a la sortida quan tinguem 5 volts a l'entrada, i tenir un corrent suficient perquè l'autòmat reconegui els senyals correctament.

Per aconseguir aquests dos fets, hem dissenyat un circuit, per cada entrada de l'autòmat, format per dos transistors, un de tipus NPN i un de tipus PNP, i tres resistències.

El transistor NPN que hem utilitzat és el model BC547. Aquest transistor té la base connectada a una resistència de 220kΩ que està connectada en sèrie amb la sortida del microcontrolador. Així, quan tenim 5 volts a la sortida del microcontrolador circularà intensitat per la base del transistor, fet que farà que aquest entri a funcionar en la seva regió activa.

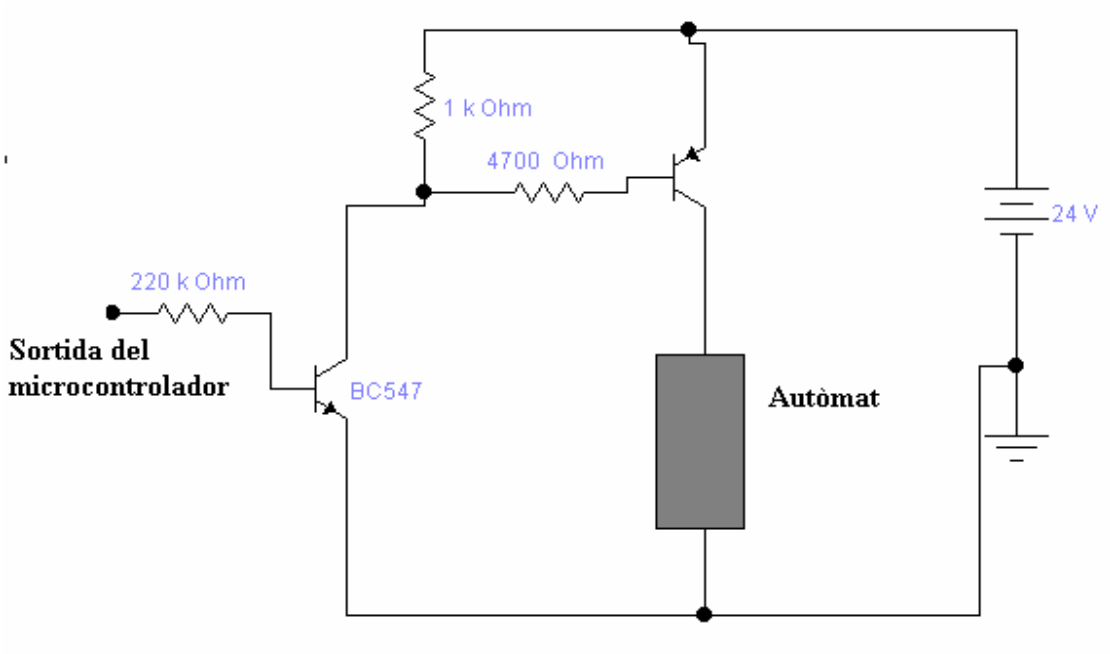


Figura 4.7.2 Esquema elèctric del circuit de transistors per una sola sortida del microcontrolador

La intensitat de base tindrà un valor d'uns 19.5μA, que obtenim dels càlculs que detallem a continuació:

$$V_{microcontrolador} = I \cdot R - V_{BE\gamma}$$
$$5 = I \cdot 220 \cdot 10^3 + 0.7$$
$$I = 19.5 \mu A$$

On $V_{microcontrolador}$ és la tensió que tenim a les sortides del microcontrolador, $V_{BE\gamma}$ és la tensió entre la base i l'emissor del transistor quan aquest treballa en al regió activa, aquest valor ve donat pel fabricant, R és el valor de la resistència i, finalment, I és el valor del corrent de base.

Així, quan tenim el transistor treballant en la regió activa provoquem que circuli corrent per l'emissor i, conseqüentment, pel col·lector del transistor. Aquest fet provoca que el transistor PNP treballi en la seva regió de saturació i, per tant, el

transistor entre el seu emissor i el seu col·lector actua com un interruptor tancat i ens permet obtenir 24 volts al col·lector, que anirà connectat a l'entrada de l'autòmat i als leds indicadors. Per al transistor PNP utilitzem el model BD138, ens interessa treballar amb un transistor de potència perquè en aquesta part del circuit treballem amb tensions i corrents elevades.

La resistència de 1kΩ l'utilitzem perquè ens interessa que una part important de la intensitat de base del transistor PNP circuli a través d'ella i no a través del transistor NPN.

El valor de la resistència de 4700Ω va relacionat amb el valor d'impedància d'entrada del autòmat. Ens interessa que la intensitat de base del transistor PNP tingui un valor adequat perquè la intensitat de col·lector sigui el suficientment gran i l'autòmat la reconegui.

4.1.6. Indicadors

Finalment tenim el circuit on tenim els leds indicadors. Aquests leds són els encarregats de donar la informació a l'usuari. Estaran il·luminats quan la sortida del microcontrolador estigui activada.

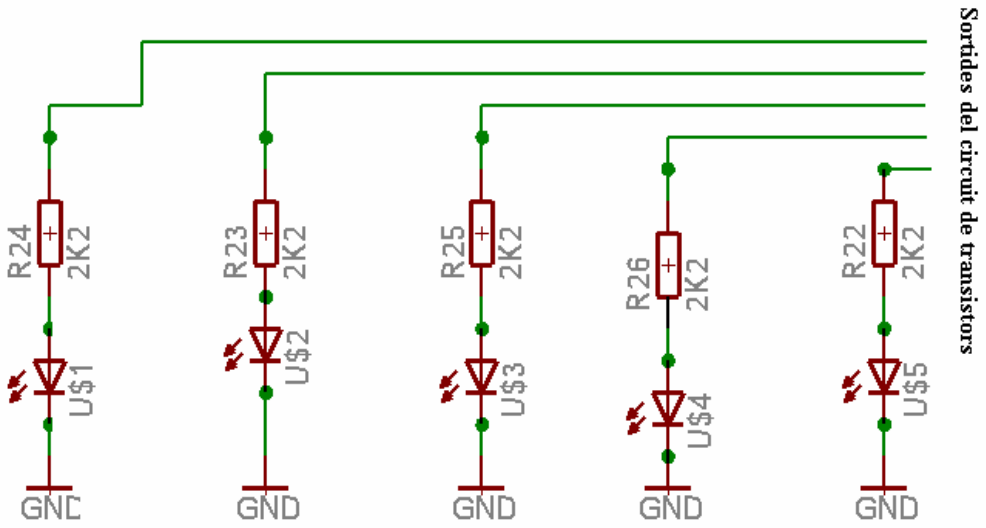


Figura 4.8 Esquema del circuit dels leds indicadors

Com es pot observar en la *figura 4.8* aquest circuit està connectat a la sortida del circuit de transistors i consisteix en connectar-hi una resistència de 2200Ω i un led vermell en sèrie en cada sortida.

El càlcul del valor de la resistència es detalla a les operacions següents:

$$V_{transistor} = V_{led} + I \cdot R$$
$$24 = 1.6 + I \cdot 2200$$
$$I = 10.18mA$$

On $V_{transistor}$ és el valor que tenim a la sortida del circuit de transistors, V_{led} és el valor de la tensió que tenim en els borns del led vermell, uns 1.6 volts per als leds vermells, R és el valor de la resistència i, finalment, I és el valor de la intensitat que circula a través del led i, que com podem observar, és inferior a la màxima i serà suficient per il·luminar el led.

4.2. Imatges de la placa

A continuació mostrem algunes imatges de la placa que hem construït i que hem utilitzat per dur a terme l'apartat d'implementació de la metodologia d'aprenentatge.

4.2.1. Primer prototip

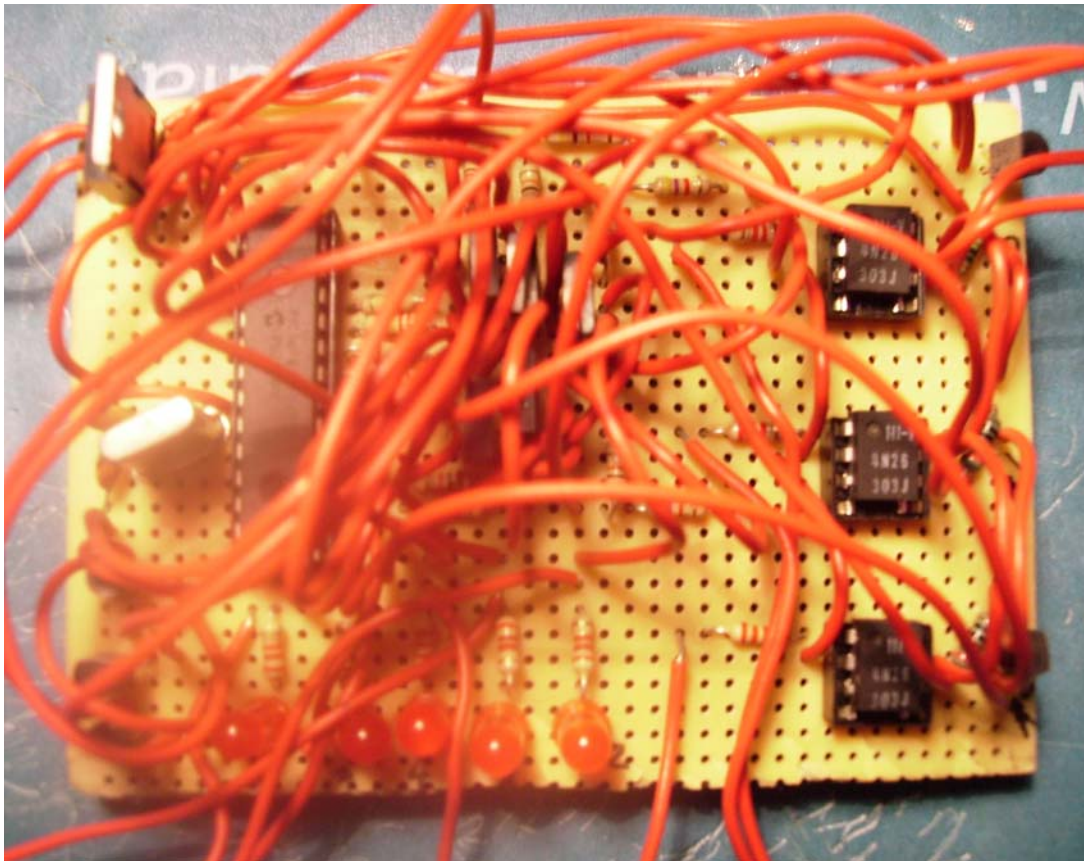


Figura 4.9 Imatge del primer prototip

A la *figura 4.9* podem observar el resultat del primer prototip que vam obtenir de la placa. Tal i com podem observar i fruit de la improvisació, degut a alguns canvis d'última hora, podem observar cert enrenou de cables. Tot i això podem identificar els elements que formen la placa.

A la part superior esquerra observem el regulador de tensió i a sota d'aquest el microcontrolador i el cristall de quars. A la part inferior hi trobem els leds, que ens serviran d'indicadors per conèixer l'estat en què es troba la planta.

A la part central de la imatge, tot i que costa apreciar-ho, hi trobem els circuit de transistors, i a la dreta del tot, hi trobem els optoacobladors.

4.2.2. Prototip definitiu

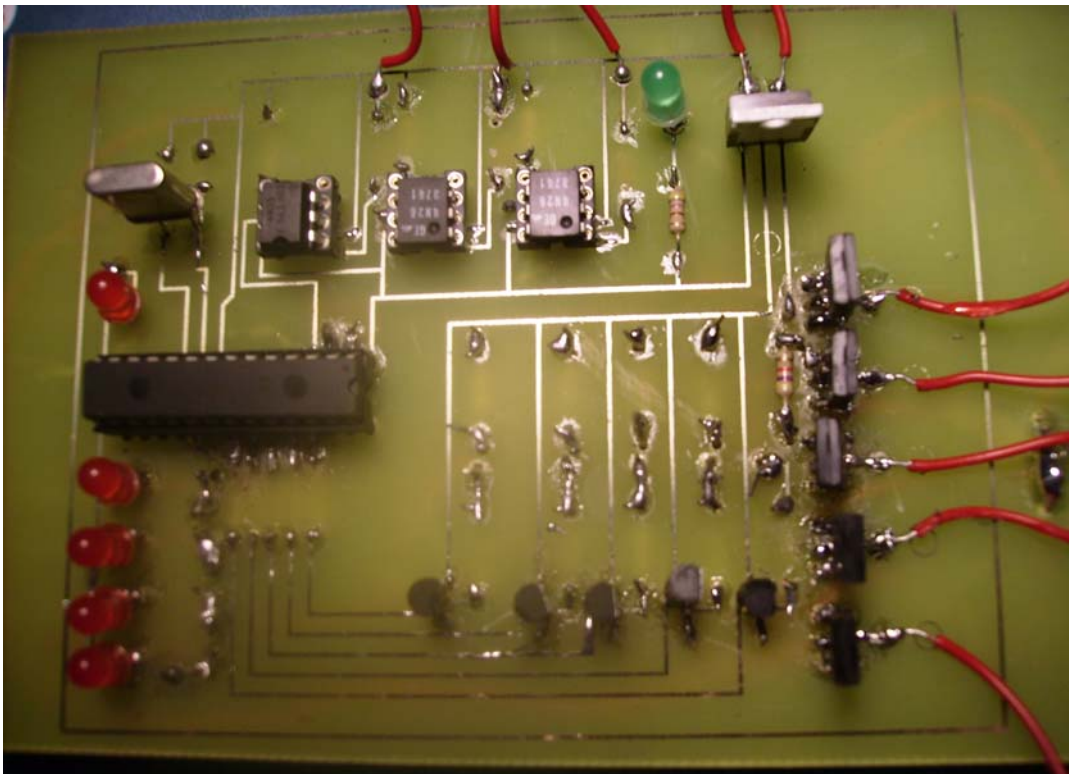


Figura 4.10 Imatge del prototip final

A la figura 4.10 podem observar el resultat obtingut com a prototip final. En aquesta imatge es poden diferenciar clarament les diferents zones de la placa que descrivim amb detall en l'apartat 4.1.

Així a la part central superior tenim els optoacobladors i per tant les entrades de la placa, a la seva dreta tenim el regulador de tensió i a l'esquerra el circuit oscil·lador.

A la part dreta de la imatge hi tenim el circuit de transistors i per tant les sortides de la placa, mentre que a la part esquerra hi podem observar el microcontrolador així com també els cinc leds indicadors, que estan associats a les sortides de la placa.

5. Implementació de la metodologia d'aprenentatge

Per poder demostrar l'eficiència i la utilitat de la placa, a continuació mostrarem les experiències que hem viscut amb una sèrie d'exemples de programació de l'autòmat que utilitzem per controlar la planta. Aquests exemples podrien ser perfectament exercicis de classe plantejats pels professors de l'assignatura de Laboratori d'Automatització Industrial als estudiants.

En aquests exemples s'intenta exposar diferents exercicis, començant per un primer amb unes funcions bàsiques i anant pujant el grau de dificultat en els posteriors. També hem volgut posar un exemple en el qual es pot apreciar l'avantatge de la planta per localitzar errors en el programa realitzat.

S'ha de tenir en compte, que hi pot haver diferents possibilitats a l'hora de programar l'autòmat, per tant les solucions que es trobaran a continuació no són úniques, això si, pretenen ser el més clares i adequades possibles d'acord amb el problema que es planteja en cada situació.

Per a realitzar els exercicis que plantegem a continuació creiem que en els tres primers casos no és necessari la utilització de diagrames GRAFCET. Únicament en l'últim exemple, a causa del seu grau de complexitat, hem cregut adient utilitzar els diagrames GRAFCET per dur a terme l'exercici.

5.1. Exemple 1: Funcionament simple, programa erroni

1.1.1. Descripció de l'exercici

Primer de tot, s'ha de comentar que en aquest primer exemple volem mostrar alguns dels errors que es poden cometre a l'hora de programar l'autòmat i que gràcies a la planta simulada podem detectar ràpidament.

La cabina de l'ascensor disposa de tres polsador que prem l'usuari per indicar a quina planta es vol dirigir. A més, a la part exterior de cada planta i trobem un

polsador per cridar l'ascensor des de la planta en la que ens trobem. Quan l'ascensor es troba aturat en una planta, la porta es manté oberta.

En el cas de que es premi algun dels sis polsadors, la porta es tancarà. En el cas de que l'ascensor hagi estat cridat des d'una planta diferent de la que es troba en aquell moment, l'ascensor haurà de pujar o baixar fins aturar-se a la planta des de que se l'ha cridat o s'ha indicat amb els polsadors de dins de la cabina de l'ascensor. Una vegada l'ascensor hagi arribat a la planta, automàticament s'aturarà i obrirà la porta.

L'ascensor romandrà aturat a la última planta fins on s'ha desplaçat fins que es premi un nou polsador.

5.1.1. Entrades i sortides de l'autòmat

A continuació detallem les diferents entrades i sortides de l'autòmat que hem utilitzat, i a quin element de la planta corresponen per tal de poder fer entendre millor el diagrama de contactes que mostrarem més endavant.

5.1.1.1. Entrades

A continuació detallem a quin element està associada cada entrada.

I1.0 Primer bit indicador de la planta en que es troba l'ascensor

I1.1 Segon bit indicador de la planta en que es troba l'ascensor

I1.2 Tercer bit indicador de la planta en que es troba l'ascensor

I0.0 Polsador interior de la cabina associat a la primera planta

I0.1 Polsador interior de la cabina associat a la segona planta

I0.2 Polsador interior de la cabina associat a la tercera planta

I0.3 Polsador exterior per cridar l'ascensor a la primera planta

I0.4 Polsador exterior per cridar l'ascensor a la segona planta

I0.5 Polsador exterior per cridar l'ascensor a la tercera planta

5.1.1.2. Sortides

A continuació detallem a quin element està associada cada sortida.

Q0.0 Senyal per fer pujar l'ascensor

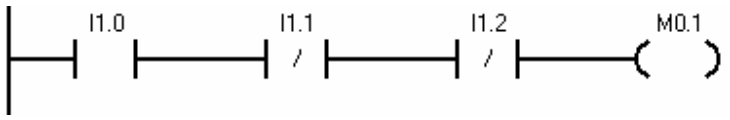
Q0.1 Senyal per fer baixar l'ascensor

Q0.2 Senyal per obrir la porta de l'ascensor

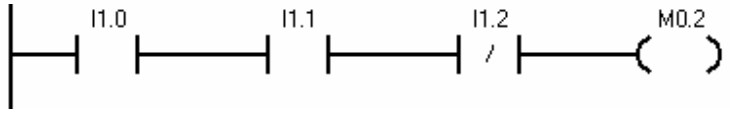
5.1.2. Diagrama de contactes

Els cinc primers diagrames de contactes corresponen a les marques que ens indicaran en quina planta es troba l'ascensor en cada moment.

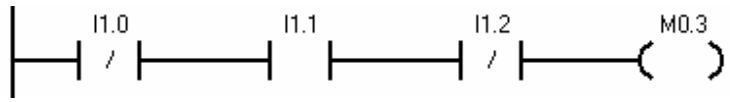
Network 1



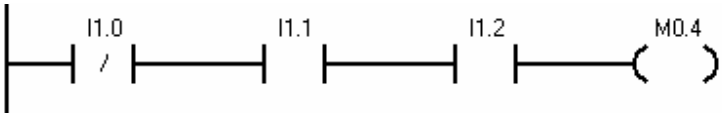
Network 2



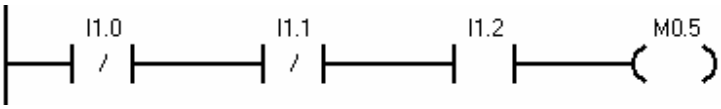
Network 3



Network 4

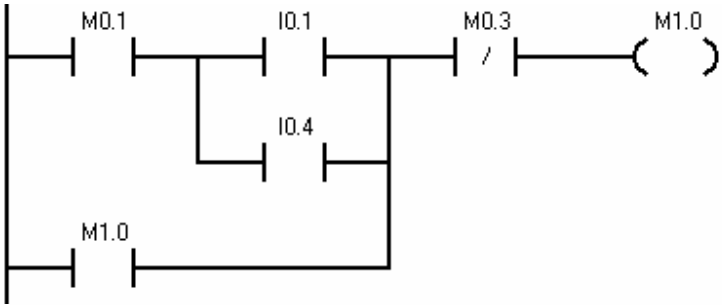


Network 5

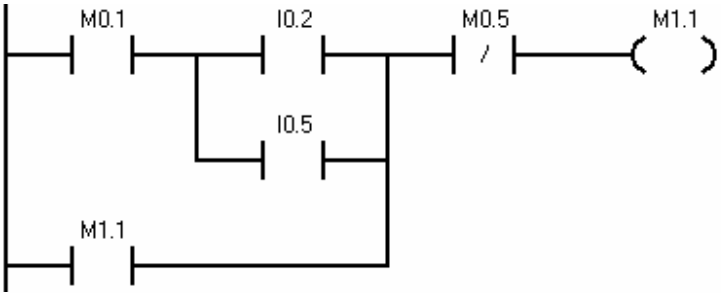


A continuació trobem les situacions en què l'ascensor es troba en una de les tres plantes i rep una ordre per desplaçar-se a una altra. Així, la cabina es posarà en moviments fins a arribar-hi.

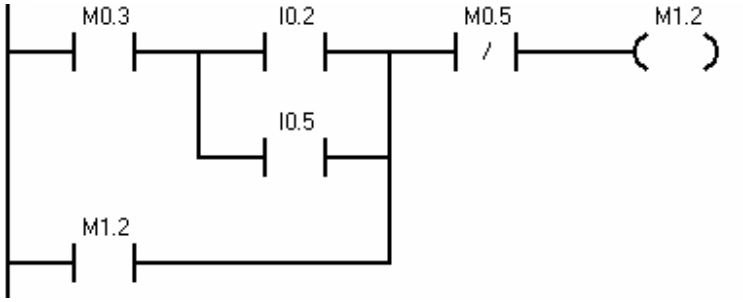
Network 6



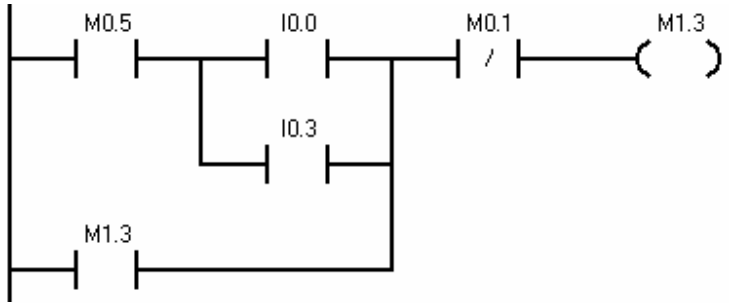
Network 7



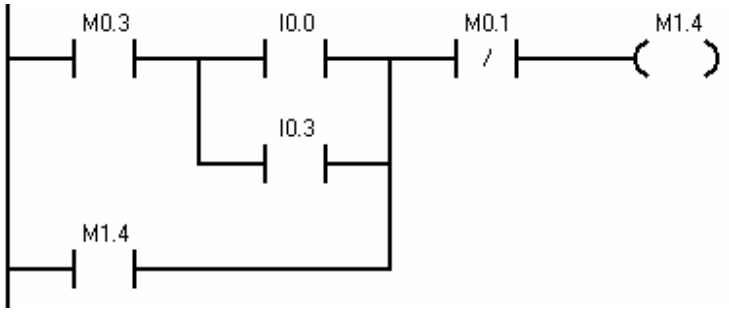
Network 8

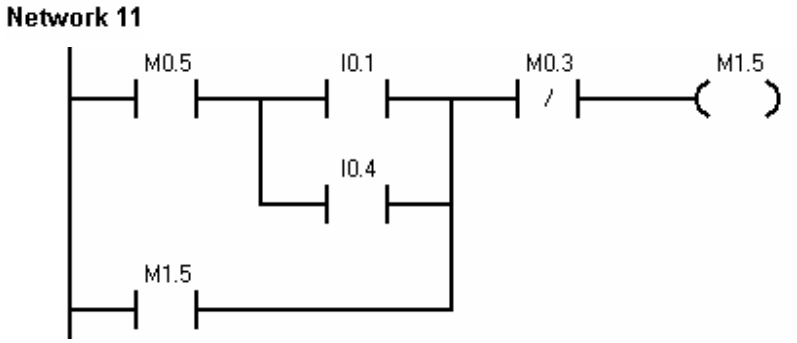


Network 9

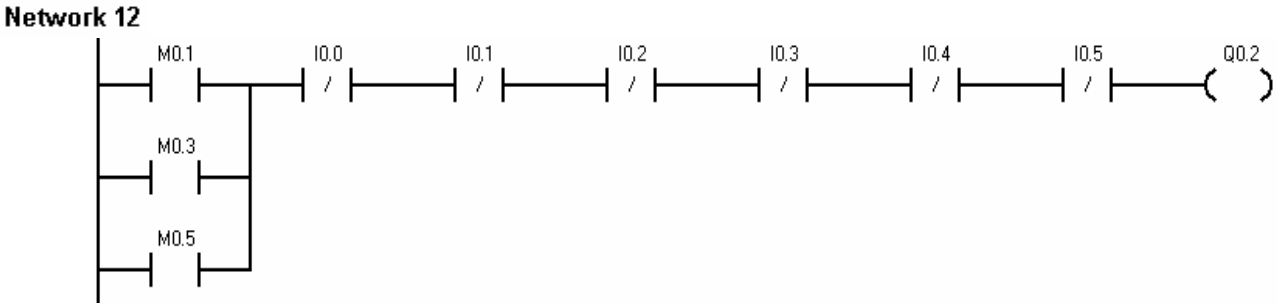


Network 10



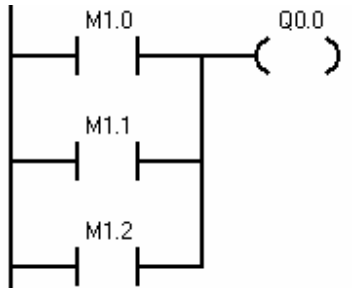


Tot seguit, necessitem controlar la porta. Així quan l'ascensor es trobi a la primera (M0.1), segona (M0.2) o tercera planta (M0.3) i no hi hagi cap polsador premut, la porta està oberta (Q0.2)

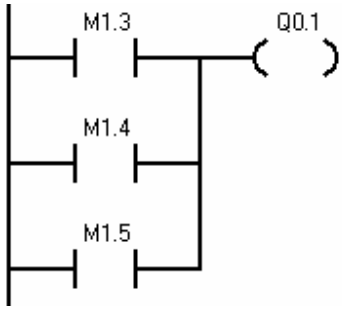


A continuació tenim el diagrama que ens mostra quan s'activarà els senyals de pujar (Q0.0) i baixar (Q0.1). Cada una d'aquestes accions va associada a unes de les marques anteriors.

Network 13



Network 14



5.1.3. Errors detectats

En aquest disseny del control de la planta simple hem pogut localitzar tres errors gràcies als indicadors de la planta.

El primer error que trobem fa referència al control de la porta de l'ascensor. Amb el disseny actual la porta s'obrirà sempre que es trobi en una de les tres plantes i no hi hagi cap polsador premut, però això també inclou que en el moment en què l'ascensor estigui en moviment, desplaçant-se d'una planta a una altra, no hi hagi cap polsador premut i passi per una d'aquestes plantes la porta s'obrirà.

El segon error es produeix en el fet de que no hem tingut compte què succeeix si es premen dos polsadors de dues plantes diferents a la vegada. Imaginem-nos que l'ascensor es troba a la segona planta i rep l'ordre de que ha d'anar a la primera planta i a la tercera al mateix temps. En aquest cas l'ascensor rebrà el senyal de pujar i baixar al mateix temps i conseqüentment es trencarà el motor, fet que podrem observar gràcies al led corresponent en la nostra placa.

El tercer error es produeix quan l'ascensor està en funcionament i rep una ordre diferent a la que està realitzant. Aquest cas només succeirà quan l'ascensor es trobi de pas per la segona planta i es premi un polsador referent a una planta diferent a la que es dirigeix. En aquest cas, succeirà el mateix que el que hem comentat anteriorment, l'ascensor rebrà la senyal de pujar i baixar al mateix temps i conseqüentment es trencarà el motor.

5.2. Exemple 1.1: Correcció de l'exemple 1

5.2.1. Descripció de l'exercici

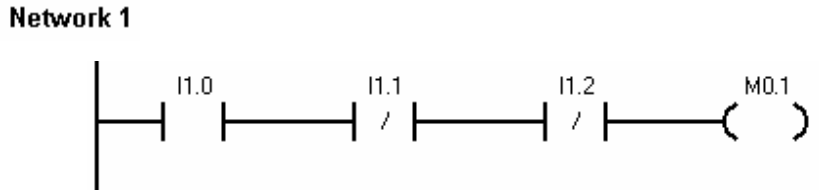
Aquest exercici pretén corregir els errors comesos en l'exemple 1. Per tant, les característiques de l'exercici són iguals a les de l'exemple 1. Únicament afegim que en el cas de que es cridi l'ascensor des de la mateixa planta a la que es troba la cabina o bé es premi el polsador del interior corresponent a la planta on es troba l'ascensor, aquest tancarà les portes per tornar a obrir-les transcorregut 1 segon

5.2.2. Entrades i sortides de l'autòmat

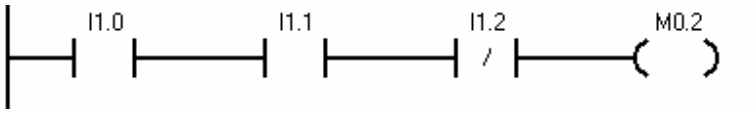
Les entrades i sortides que utilitzem per aquest exemple, són les mateixes que hem descrit en l'exemple 1.

5.2.3. Diagrama de contactes

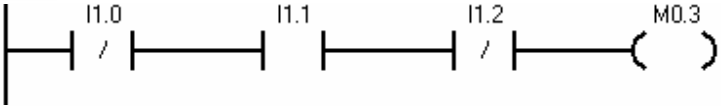
Les cinc primeres parts del diagrama de contactes d'aquest exemple, no sofriran cap canvi respecte a les que hem utilitzat en l'exemple 1, ja que no resulten problemàtiques en cap moment



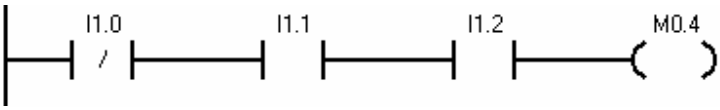
Network 2



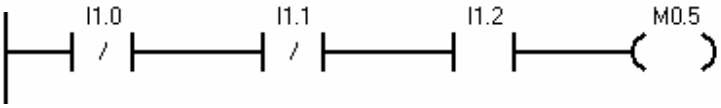
Network 3



Network 4

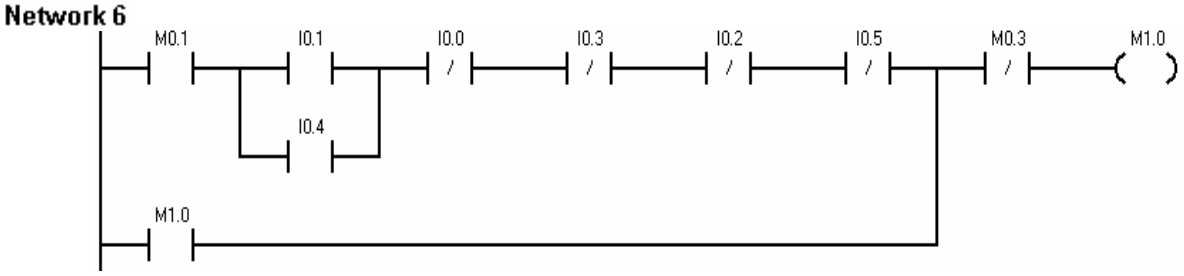


Network 5

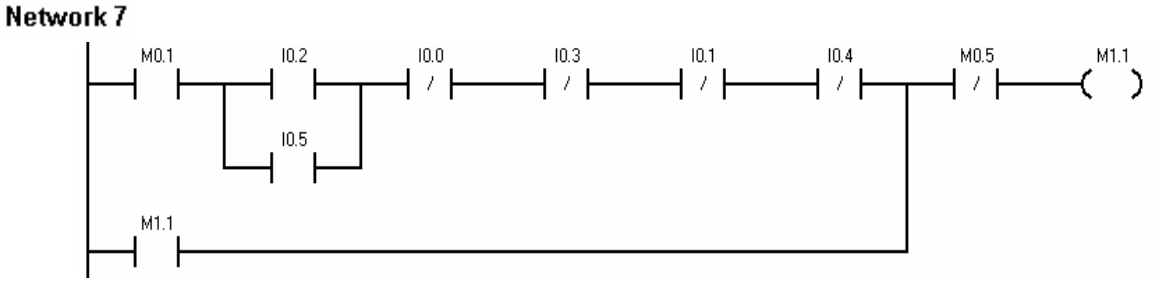


Les següents parts del diagrama són les que sofreixen més canvis. Per una part tenim les diferents situacions en que l'ascensor es troba a les plantes 1 o 3. En aquests casos a més de que es premi el polsador per indicar la planta a la que volem que es dirigeixi l'ascensor afegim que en aquell precís moment no estigui premut cap altre polsador. Així evitem que l'ascensor rebi dos ordres a la vegada. En els casos en que l'ascensor es troba a la segona planta també hem d'afegir la condició de que en el moment en que es troba en aquesta planta, l'ascensor no estigui ni pujant (Q0.0) ni baixant (Q0.1), és a dir, no estigui en moviment. Així evitem que es produeixi el tercer error.

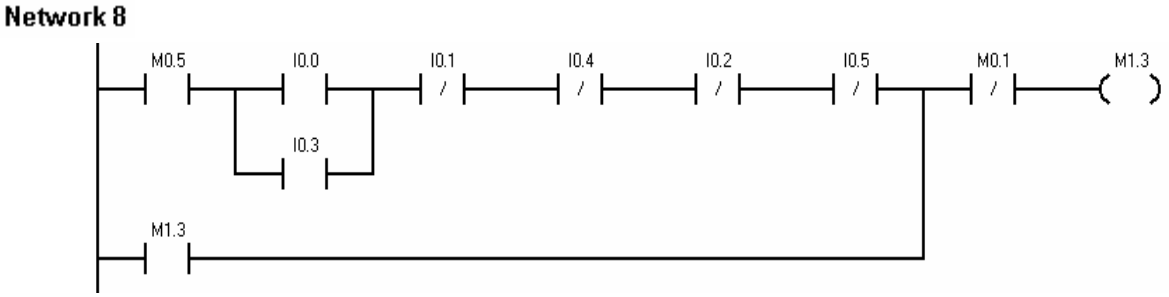
De la planta 1 a la planta 2



De la planta 1 a la planta 3

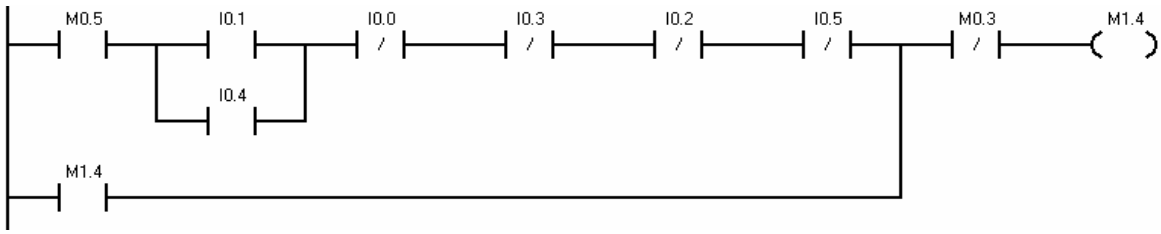


De la planta 3 a la planta 1



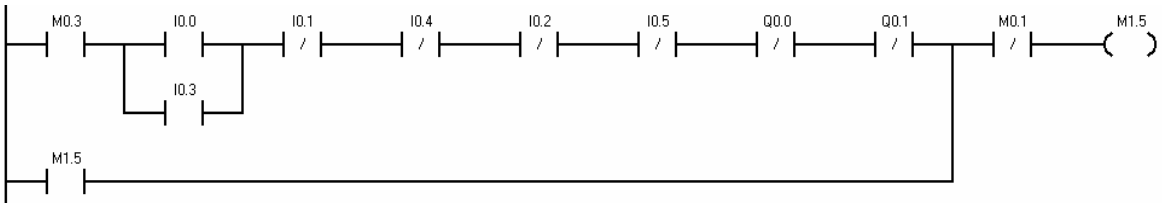
De la planta 3 a la planta 2

Network 9



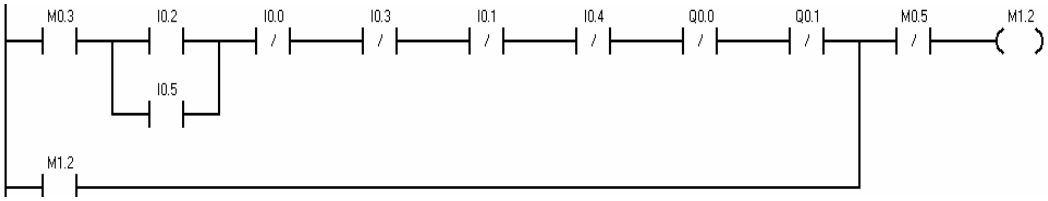
De la planta 2 a la planta 1

Network 10



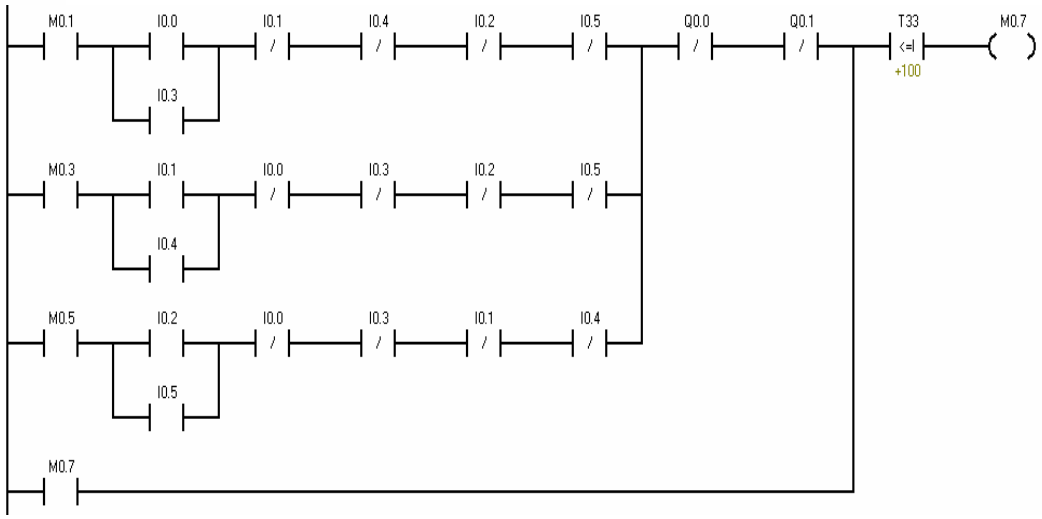
De la planta 2 a la planta 3

Network 11



En aquest diagrama de contactes afegim una nova marca (M0.7) que ens indica que l'ascensor està aturat i que s'ha premut un polsador que indica a l'ascensor que s'ha de desplaçar a la mateixa planta on es troba. Quan això succeeix es posa en funcionament el timer T33, que anirà comptant de forma ascendent. En el moment que el timer arribi a 1 segon, la marca M0.7 deixarà d'estar activa.

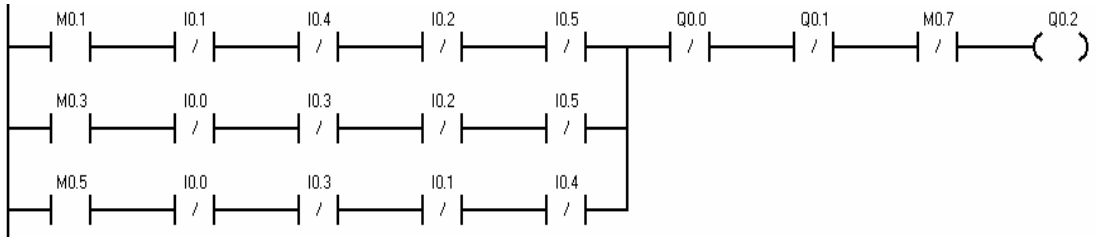
Network 13



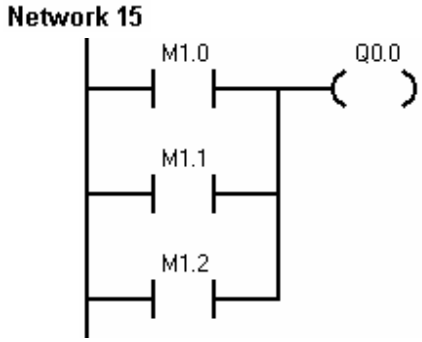
Network 14

Amb el següent segment controlem quan obrirem la porta (Q0.2). La porta estarà oberta sempre que l'ascensor es trobi aturat en la planta 1, 2 o 3 i també s'ha de complir que la marca M0.7 no es trobi activa. Així, tal i com hem descrit anteriorment, si algú prem un polsador que indica a l'ascensor que s'ha de desplaçar a la mateixa planta on es troba, el timer T33 es posarà a comptar i mantindrà activa M0.7 i per tant, les portes estaran tancades durant 1 segon. Una vegada ha passat aquest segon, si no hi ha cap instrucció nova les portes es tornaran a obrir.

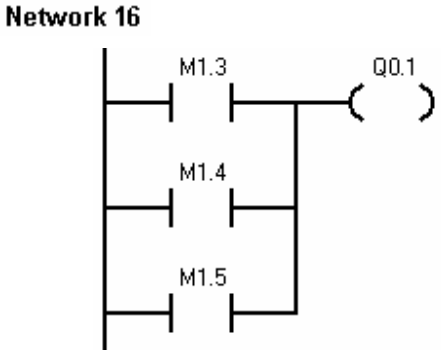
Network 12



Finalment tenim el diagrama que ens mostra quan s'activarà el senyal de pujar (Q0.0). Aquest senyal estarà actiu si la planta es troba en alguna de les situacions comentades anteriorment (etapes M1.0, M1.1 o M1.2).



Igual que en el cas del senyal que fa pujar l'ascensor, tenim el senyal que el fa baixar (Q0.1). En aquest cas el senyal estarà actiu si la planta es troba en M1.3, M1.4 o M1.5



5.3. Exemple 2

5.3.1. Descripció de l'exercici

La cabina de l'ascensor disposa de tres polsador que prem l'usuari per indicar a quina planta es vol dirigir. Tanmateix, l'ascensor disposa de tres polsadors exteriors per cridar l'ascensor

Quan l'ascensor està aturat, la porta es manté oberta.

En el cas que es premi algun dels sis polsadors, la porta es tancarà. En el cas de que l'ascensor hagi estat cridat des d'una planta diferent de la que es troba en aquell moment l'ascensor haurà de pujar o baixar fins aturar-se a la planta des de que se l'ha cridat o s'ha indicat amb els polsadors de dins de la cabina de l'ascensor. Un vegada l'ascensor hagi arribat a la planta, automàticament s'aturarà i obrirà la porta.

Hi haurà un retard de dos segons entre el moment en que es crida l'ascensor des d'un dels polsadors exteriors fins que aquest reacciona. D'aquesta forma, si durant aquest temps es polsa un dels polsadors interiors, aquesta última ordre tindrà prioritat. Aquesta característica serà útil perquè una ordre exterior no es pugui avançar a la intenció d'un usuari que es trobi dins de la cabina.

Quan l'ascensor es trobi en repòs en una planta, la porta es tancarà. Per tornar-la a obrir, simplement s'haurà de polsar el polsador exterior. La porta de l'ascensor s'obrirà automàticament quan arribi a la planta desitjada i es tancarà passats cinc segons.

5.3.2. Entrades i sortides de l'autòmat

A continuació detallem les diferents entrades i sortides de l'autòmat que hem utilitzat, i a quin element de la planta corresponen per tal de poder fer entendre millor el diagrama de contactes que mostrarem a continuació.

5.3.2.1. Entrades

A continuació detallem a quin element està associada cada entrada.

I1.0 Primer bit indicador de la planta en que es troba l'ascensor

I1.1 Segon bit indicador de la planta en que es troba l'ascensor

I1.2 Tercer bit indicador de la planta en que es troba l'ascensor

I1.3 Indicador de que la porta de l'ascensor es troba oberta

I0.0 Polsador interior de la cabina associat a la primera planta

I0.1 Polsador interior de la cabina associat a la segona planta

I0.2 Polsador interior de la cabina associat a la tercera planta

I0.3 Polsador exterior per cridar l'ascensor a la primera planta

I0.4 Polsador exterior per cridar l'ascensor a la segona planta

I0.5 Polsador exterior per cridar l'ascensor a la tercera planta

5.3.2.2. Sortides

A continuació detallem a quin element està associada cada sortida.

Q0.0 Senyal per fer pujar l'ascensor

Q0.1 Senyal per fer baixar l'ascensor

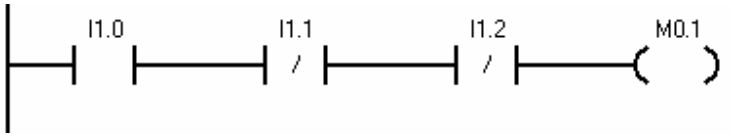
Q0.2 Senyal per obrir la porta de l'ascensor

5.3.3. Diagrama de contactes

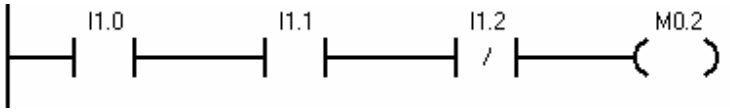
Les cinc primeres parts del diagrama de contactes d'aquest exemple, no sofriran cap canvi respecte a les que hem utilitzat en els exemples anteriors. Això succeeix perquè aquesta part és la que ens determina la forma com identifiquem la

planta en la que es troba l'ascensor. En tots els casos disposem del mateix nombre de plantes i les identifiquem de la mateixa manera.

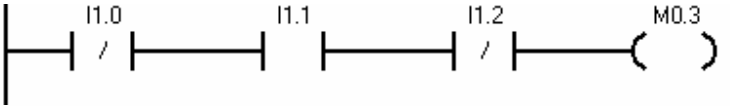
Network 1



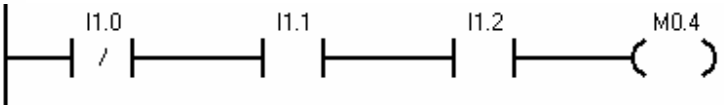
Network 2



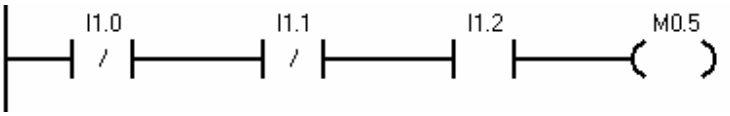
Network 3



Network 4

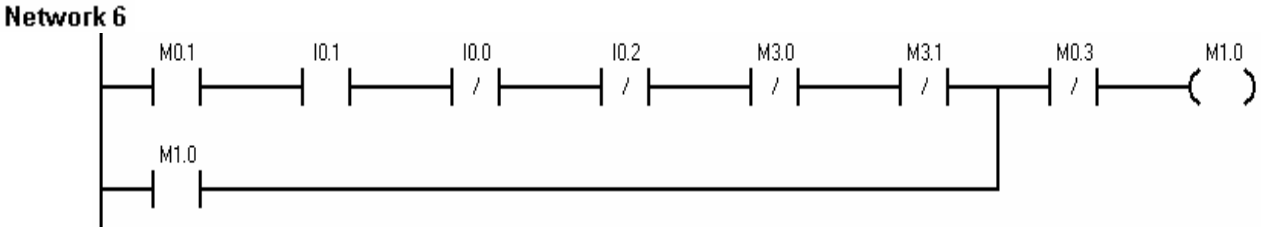


Network 5

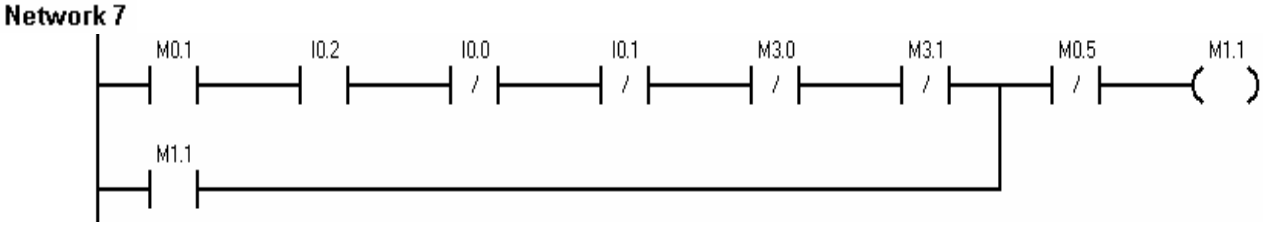


Per realitzar aquest exercici, el primer que fem és diferenciar quan l'ascensor executa una ordre enviada des d'un polsador exterior interior i un polsador exterior. A continuació tenim les ordres que arriben dels polsadors interiors. Així per cada cas activarem la etapa corresponent sempre que es premi el polsador corresponent i no estigui cap altre polsador interior premut o hi hagi cap etapa referent als polsador exteriors activa.

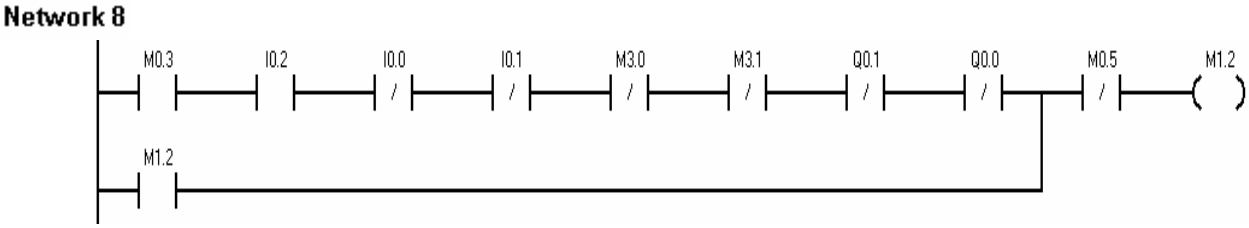
De la planta 1 a la planta 2



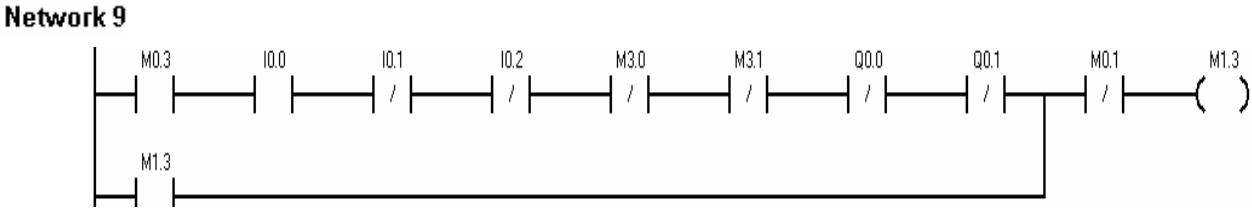
De la planta 1 a la planta 3



De la planta 2 a la planta 3

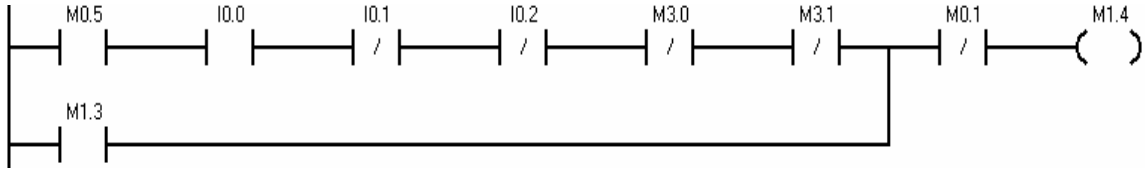


De la planta 2 a la planta 1



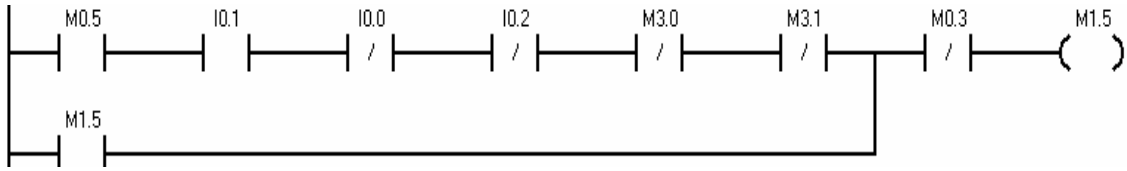
De la planta 3 a la planta 1

Network 10



De la planta 3 a la planta 2

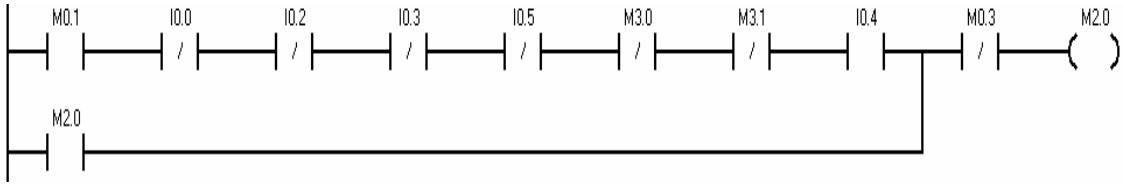
Network 11



A continuació tenim els segments que activaran les etapes perquè l'ascensor es desplaci de la mateixa manera que en els casos anteriors, però ara l'ascensor executarà les ordres que li arriben des dels polsadors exteriors.

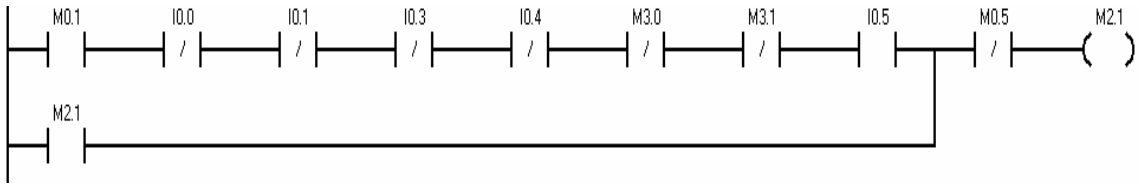
De la planta 1 a la planta 2

Network 12



De la planta 1 a la planta 3

Network 13

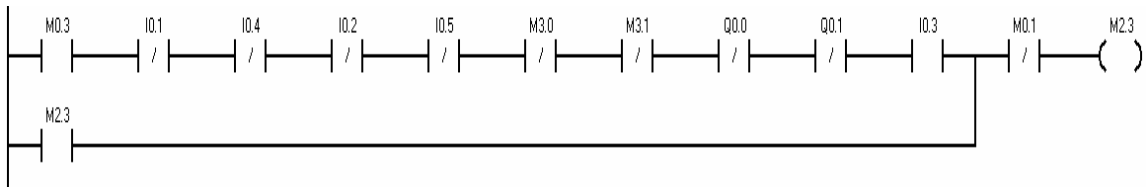


De la planta 2 a la planta 3

Network 14

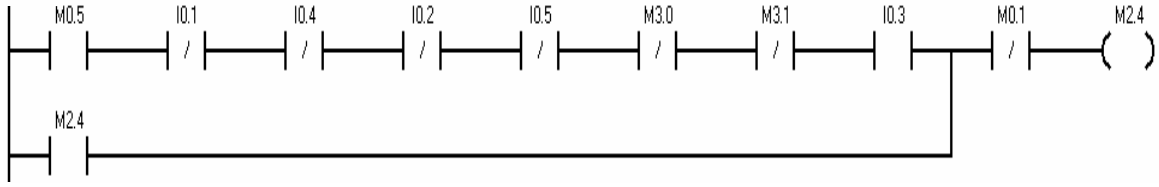
De la planta 2 a la planta 1

Network 15

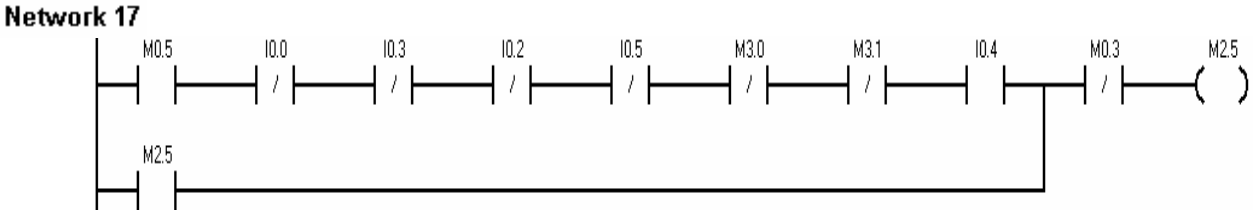


De la planta 3 a la planta 1

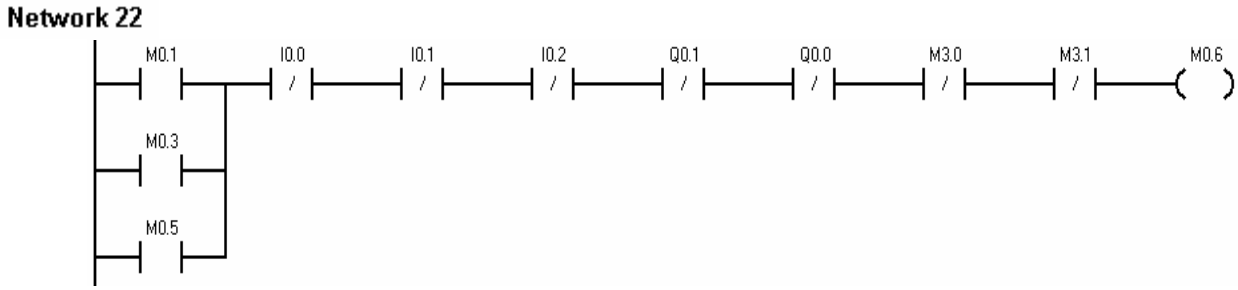
Network 16



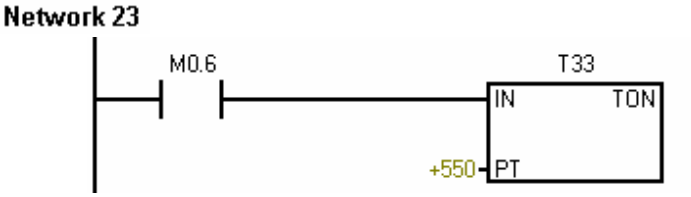
De la planta 3 a la planta 2



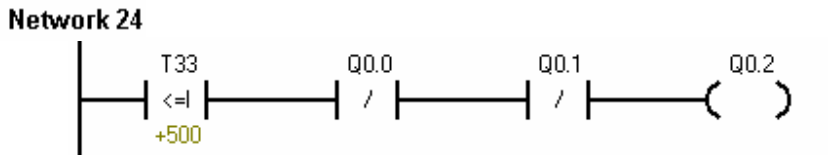
L'etapa M0.6 ens indicarà que l'ascensor es troba en repòs. Quan estigui activa es posarà en marxa el timer T33.



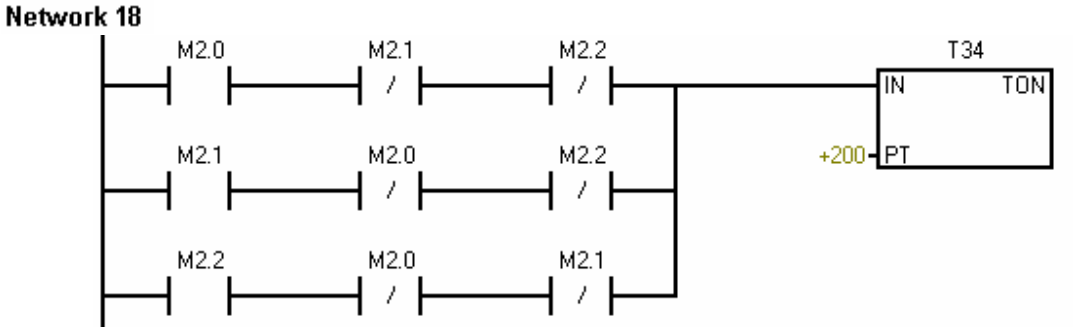
El timer t33 activarà l'utilitzem per determinar quan hauran passat els cinc segons.

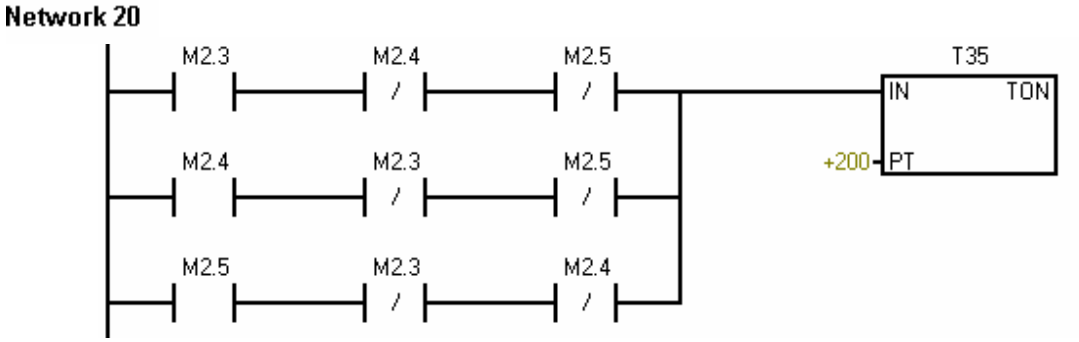


En el moment en que el timer T33 arribi als cinc segons, la porta es tancarà, Q0.2 deixarà d'estar activa.



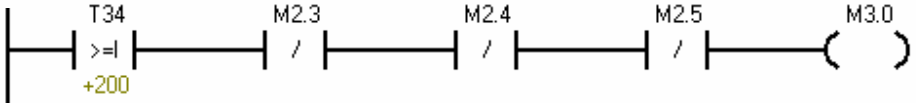
Per als retards dels polsadors exteriors utilitzarem dos timers més. El primer (T34) l'utilitzarem per les accions en que l'ascensor ha de pujar i el segon (T35) per a les accions en que l'ascensor ha de baixar.



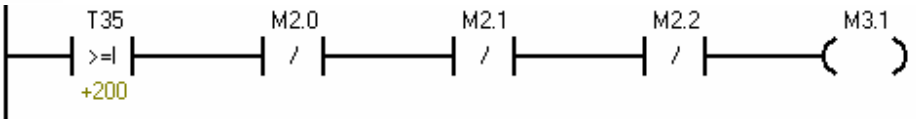


En el moment en que els timers arribin al seu valor límit, activaran el seu bit i conseqüentment activaran les etapes M3.0 i M3.1 que ens permetran activar les accions de pujar (Q0.0) i baixar (Q0.1) respectivament.

Network 19

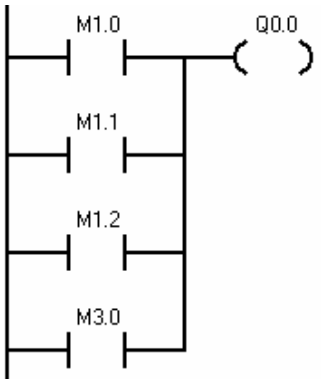


Network 21

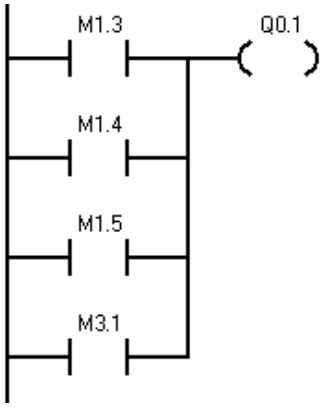


Les accions de pujar i baixar, s'activaran a causa dels polsadors interiors, o bé, a causa dels polsadors exteriors.

Network 25



Network 26



5.4. Exemple 3

5.4.1. Descripció de l'exercici

La cabina de l'ascensor disposa de tres polsadors que prem l'usuari per indicar a quina planta es vol dirigir.

Quan l'ascensor està aturat, la porta es manté oberta.

En el cas que es premi algun dels sis polsadors, la porta es tancarà. En el cas de que l'ascensor hagi estat cridat des d'una planta diferent de la que es troba en aquell moment l'ascensor haurà de pujar o baixar fins aturar-se a la planta des de que se l'ha cridat o s'ha indicat amb els polsadors de dins de la cabina de l'ascensor. Un vegada l'ascensor hagi arribat a la planta, automàticament s'aturarà i obrirà la porta.

En el cas de que l'ascensor estigui funcionant i es premi una altra ordre, el sistema memoritzarà aquesta ordre i la executarà quan hagi acabat l'ordre anterior.

Quan l'ascensor està aturat, la porta es manté oberta.

El temps que tenen els usuaris per sortir de la cabina des de que arriba l'ascensor fins que es torna a posar en funcionament, en el cas que tingui una altra ordre memoritzada, és de cinc segons.

5.4.2. Entrades i sortides de l'autòmat

A continuació detallem les diferents entrades i sortides de l'autòmat que hem utilitzat, i a quin element de la planta corresponen per tal de poder fer entendre millor el diagrama de contactes que mostrarem a continuació.

5.4.2.1. Entrades

A continuació detallem a quin element està associada cada entrada.

I1.0 Primer bit indicador de la planta en que es troba l'ascensor

I1.1 Segon bit indicador de la planta en que es troba l'ascensor

I1.2 Tercer bit indicador de la planta en que es troba l'ascensor

I0.0 Polsadors associats a la primera planta

I0.1 Polsadors associats a la segona planta

I0.2 Polsadors associats a la tercera planta

En aquest cas només utilitzem tres entrades per als sis polsadors, així el polsador interior i exterior de cada planta aniran connectats a la mateixa entrada

5.4.2.2. Sortides

A continuació detallem a quin element està associada cada sortida.

Q0.0 Senyal per fer pujar l'ascensor

Q0.1 Senyal per fer baixar l'ascensor

Q0.2 Senyal per obrir la porta de l'ascensor

5.4.3. GRAFCET's

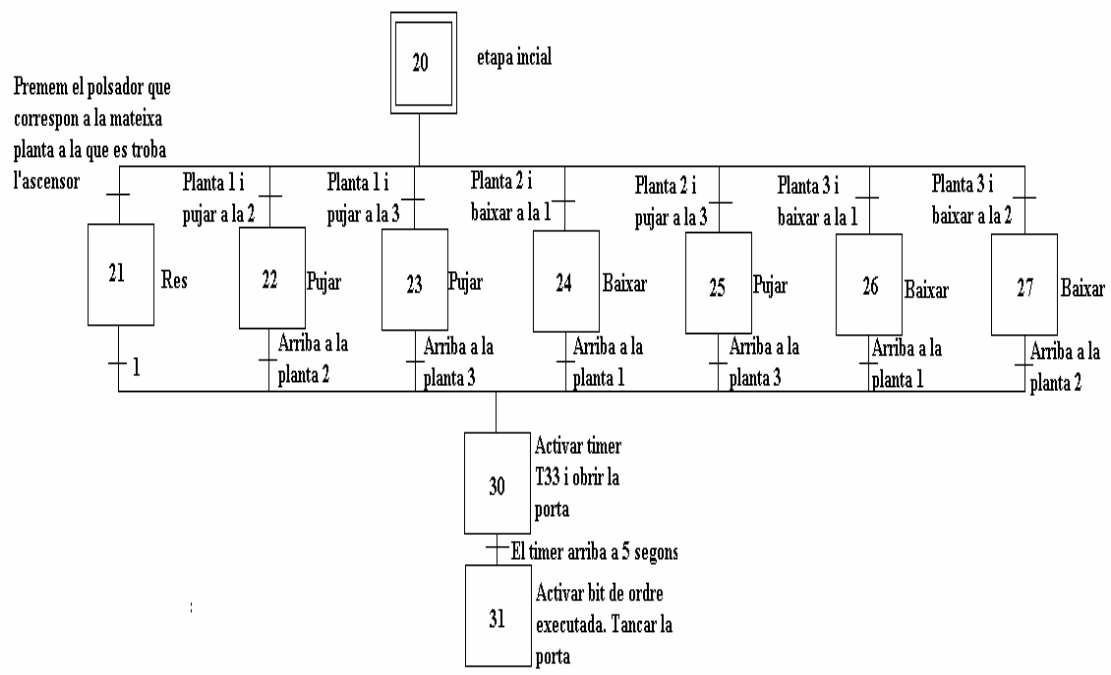
A diferència dels casos anteriors, la complexitat de l'exercici aconsella la realització d'un diagrama GRAFCET per dur a terme l'exercici. Els GRAFCETS que mostrem més endavant, són GRAFCETS de nivell 1.

5.4.3.1. GRAFCET principal

El primer és el que anomenem GRAFCET principal. Quan l'ascensor es trobi en repòs, tindrà sempre activa la seva primera etapa (etapa 20). La funció d'aquest GRAFCET consisteix en executar una ordre de l'usuari emmagatzemada en una pila d'ordres de tipus FIFO. Aquesta pila està inicialment buida. Les ordres de l'usuari no s'executen directament sinó que s'emmagatzemen en la pila.

Quan hi ha una ordre a la pila es posa en marxa la cabina i es desplaça fins a la planta desitjada per l'usuari. Tot seguit obre la porta durant cinc segons i la torna a tancar. Per finalitzar activa una marca associada al fet que s'ha executat una ordre correctament.

Quan s'arribi a l'etapa 31, aquesta es manté activa fins que és desactivada per un altre GRAFCET que també s'encarrega de tornar a activar l'etapa 20.

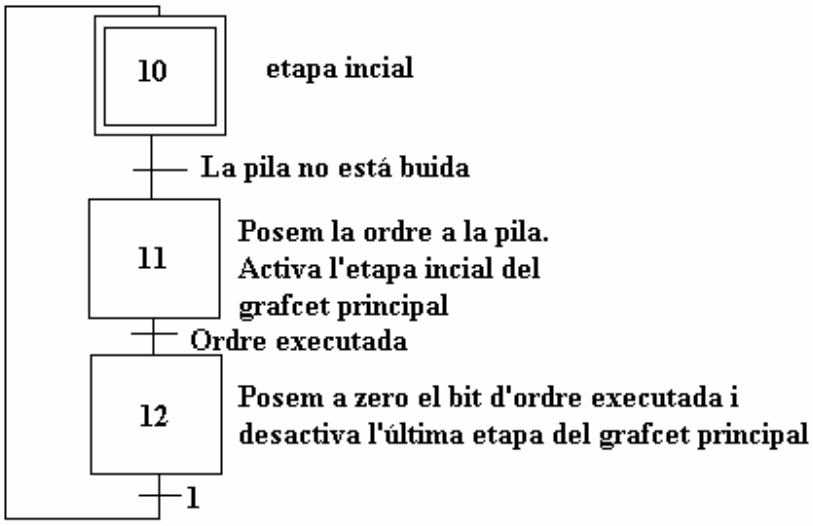


5.4.3.2. GRAFCET 1

La funció del GRAFCET 1 consisteix en determinar en quin moment la pila d'ordres està buida. Aquest GRAFCET té una etapa inicial que s'activa en el moment en que es comença a executar el programa.

Aquesta etapa inicial espera a que arribi una ordre, i per tant la pila d'ordres no estigui buida, per passar a la següent etapa. En la següent etapa es posa la ordre en la posició de la pila que li correspon i s'activa la primera etapa del GRAFCET principal. És en aquest moment quan es posarà en marxa el GRAFCET principal.

Per posar la ordre en la seva posició de la pila, utilitzem un bloc anomenat FIFO. Quan aquest finalitzi i gràcies a la marca associada al fet que s'ha executat una ordre correctament passarem a la següent etapa on desactivarem la última etapa del GRAFCET principal i desactivarem la marca de fi d'ordre executada, per tornar finalment a l'etapa inicial una altra vegada.



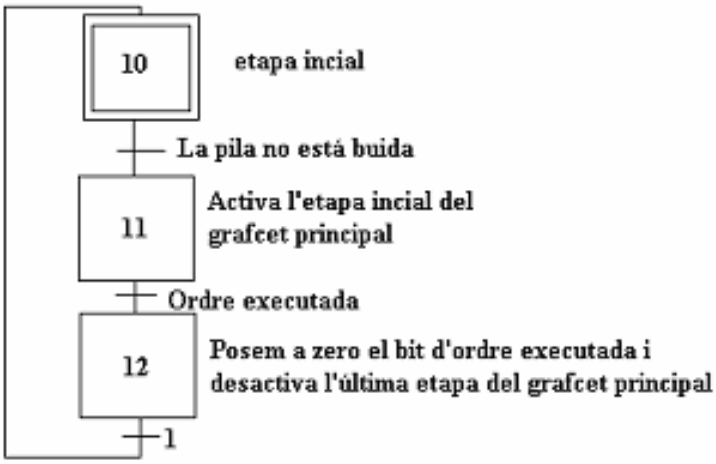
5.4.3.3. GRAFCET 2

Finalment la funció del tercer GRAFCET consisteix en emmagatzemar les ordres de l'usuari a la pila, sempre que aquesta ordre que ha arribat sigui diferent a la anterior i si s'ha deixat de prémer el polsador.

Aquest GRAFCET conté una etapa inicial que s'activa en el moment en que es comença a executar el programa. L'etapa inicial està esperant a que es premi algun dels polsadors. Quan això succeeix es posa en marxa un mecanisme que ens permet determinar si la última acció que s'ha rebut és diferent a la anterior.

El funcionament d'aquest mecanisme consisteix en treure la penúltima ordre rebuda de la pila per poder comparar el seu valor amb el de la última ordre. Això ho aconseguim amb un bloc anomenat LIFO. En el cas que siguin diferents, simplement s'introdueixen les dues ordres a la pila amb el seu ordre corresponent, mentre que si són iguals es torna a introduir a la pila l'ordre que s'ha tret.

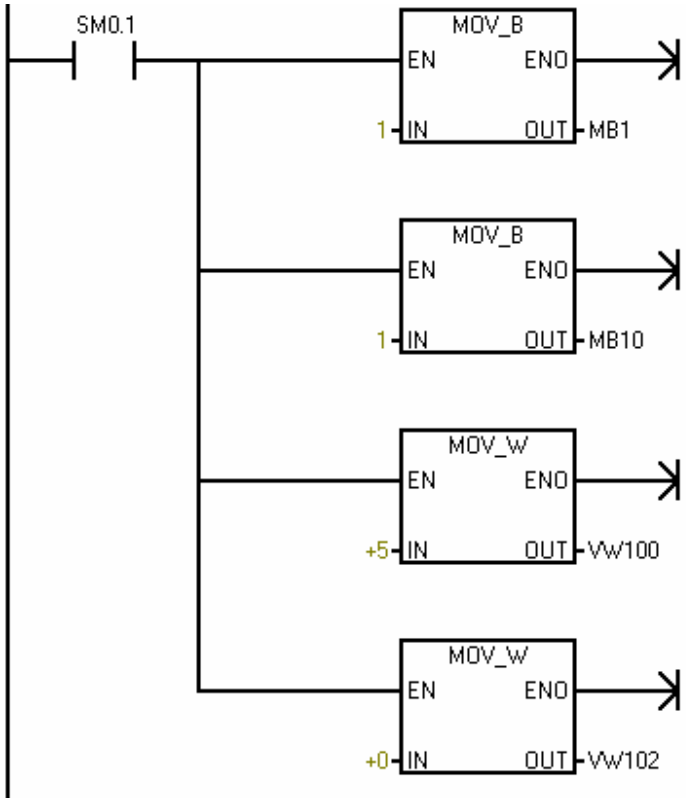
Una vegada finalitzat aquest procés, per passar a la següent etapa necessitem que s'hagi deixat de prémer el polsador. Per determinar aquest fet, utilitzem el mateix mecanisme i anem comparant els valors de les dues últimes ordres fins que deixen de ser iguals. Una vegada això succeeix tornem a l'etapa inicial d'aquest GRAFCET per poder tornar a començar el procés.



5.4.4. Diagrama de contactes

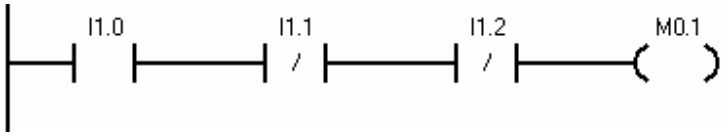
En aquest primer segment inicialitzem les primeres etapes dels diferents GRAFCET's i també definim el nombre màxim d'accions que serà capaç d'emmagatzemar la pila (5), així com la posició on posarà la primera instrucció. Com que inicialment estarà buida, li indiquem la posició 0.

Network 1

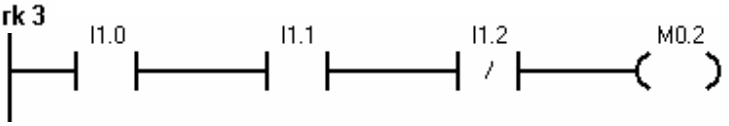


Els segments que tenim a continuació, com en els exemples anteriors, són les marques que fan referència a les diferents plantes en les que es pot trobar l'ascensor. Així quan la cabina es trobi a la primera planta s'activarà la marca M0.1 i així successivament fins a M0.5. Hem de tenir en compte que tenim cinc marques perquè també tenim en compte les plantes intermèdies.

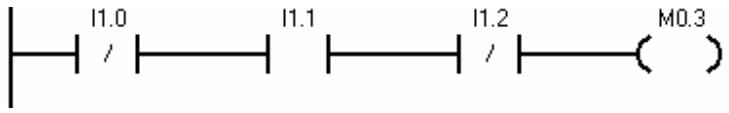
Network 2



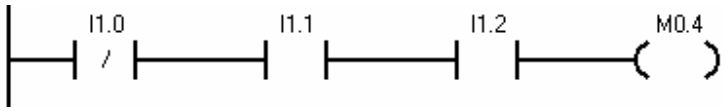
Network 3



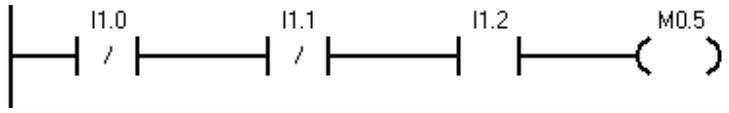
Network 4



Network 5



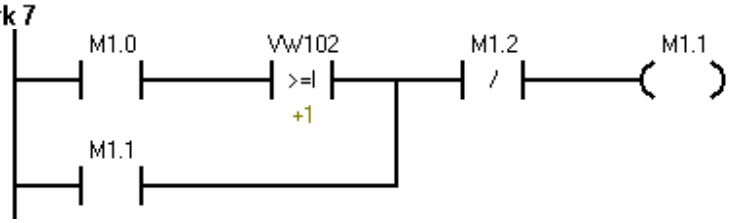
Network 6



A continuació trobem els segments que corresponen al GRAFCET 1:

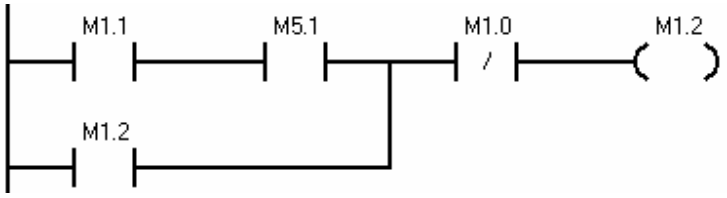
El primer segment que trobem ens descriu que quan ens trobem a l'etapa inicial (M1.0) no es passarà a l'etapa següent (M1.1) fins que no hi hagi alguna acció a la pila, és a dir, fins que no s'hagi premut algun polsador i per tant i hagi alguna acció de control pendent de realitzar.

Network 7

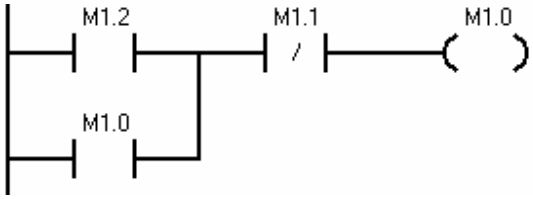


Una vegada estem a l'etapa M1.1, per poder passar a la següent etapa (M1.2) s'ha de complir que M5.1 estigui activa. M5.1 ens indica que la ordre ja ha estat executada. Com es pot observar M1.2 passa directament a l'etapa inicial (M1.0), ja que simplement s'utilitza perquè el GRAFCET tingui més de dos etapes i per tant el seu funcionament sigui correcte.

Network 8

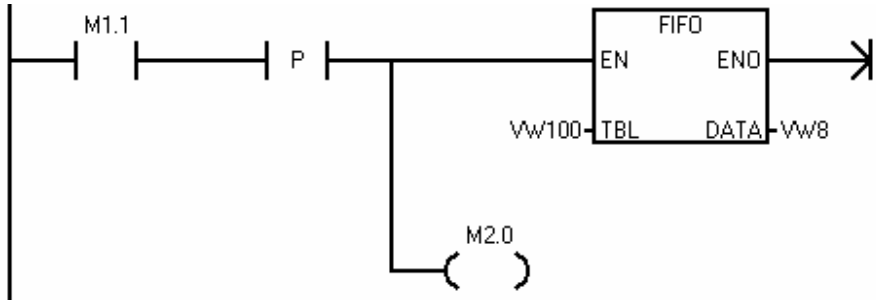


Network 9



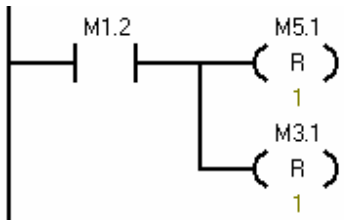
El segment que tenim a continuació, és el que ens permet buidar la pila d'ordres. Així, en el moment en que estigui activa l'etapa M1.1, agafarem la primera ordre que ha entrat a la pila (VW100) i la guardarem a VW8 per poder treballar amb ella. Aquesta acció es possible gràcies al bloc FIFO. En el moment que això succeeixi també activarem la primera etapa del GRAFCET principal, que a la fi, serà el que farà que l'ascensor es desplaci fins a la planta desitjada.

Network 10



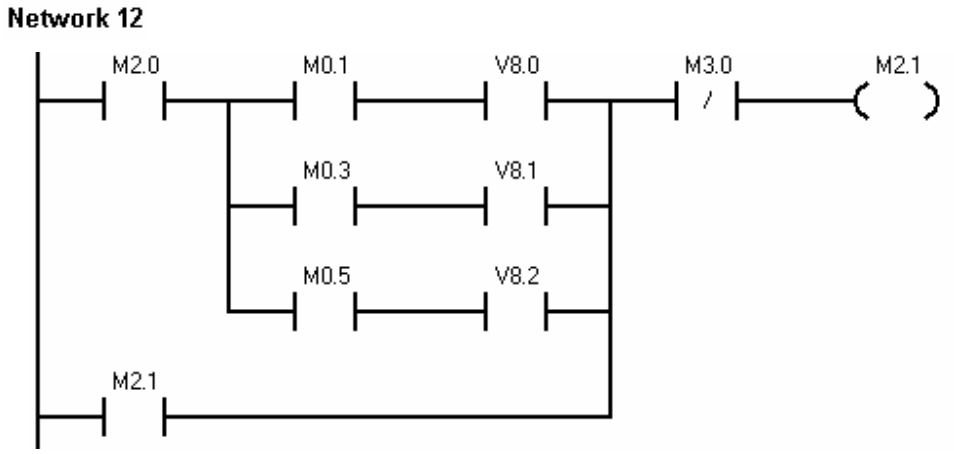
En quant al GRAFCET 1, finalment tenim que quan està activa l'etapa M1.2, reiniciem l'etapa que ens indica que l'acció ha estat executada (M5.1) i la última etapa del GRAFCET principal (M3.1), perquè el funcionament sigui el correcte ens accions posteriors.

Network 11



A continuació trobem els segments que fan referència al GRAFCET principal:

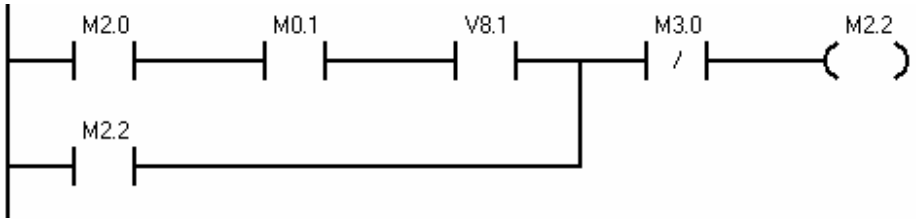
El primer segment que trobem, ens indica que quan ens trobem a l'etapa inicial (M2.0) i premem el polsador que fa referència a la mateixa planta a la que ens trobem en aquell moment, passarem a la següent etapa (M2.1). En aquest cas l'ascensor no es mourà.



Els sis segments que trobem a continuació tenen tots la mateixa estructura i corresponen a les sis possibles situacions en les que es pot trobar l'ascensor en quan a la planta en que es pot trobar i a la planta a la que s'haurà de dirigir.

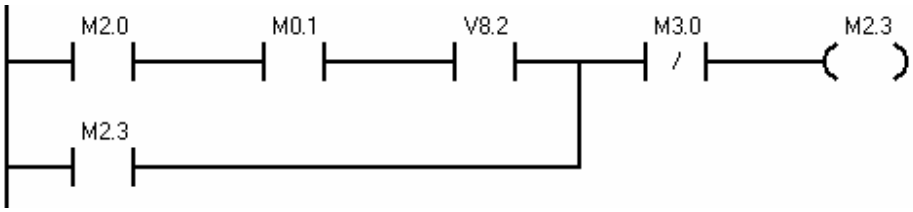
El primer segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la primera planta, si premem el polsador que fa referència a la segona planta, s'activarà l'etapa M2.2.

Network 13



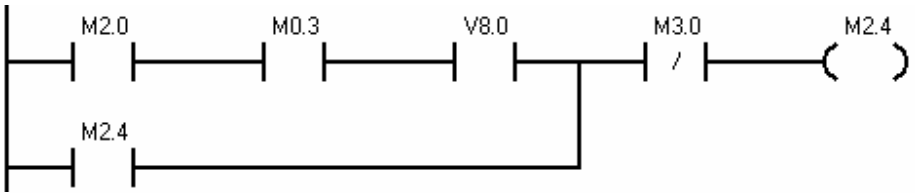
El segon segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la primera planta, si premem el polsador que fa referència a la tercera planta, s'activarà l'etapa M2.3

Network 14



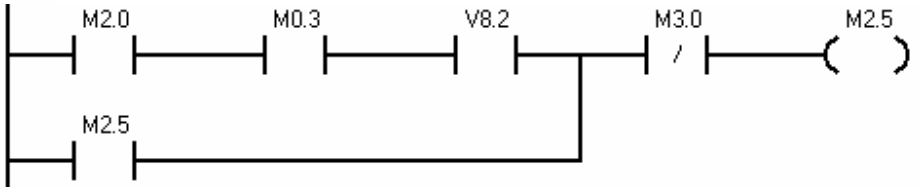
El tercer segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la segona planta, si premem el polsador que fa referència a la primera planta, s'activarà l'etapa M2.4.

Network 15



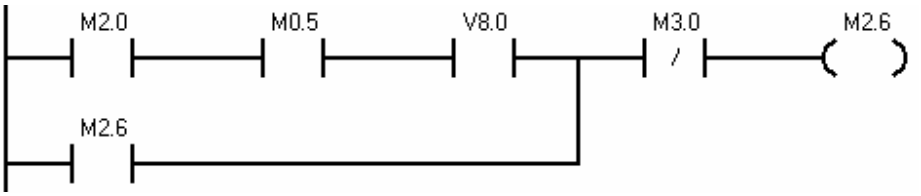
El quart segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la segona planta, si premem el polsador que fa referència a la tercera planta, s'activarà l'etapa M2.5

Network 16



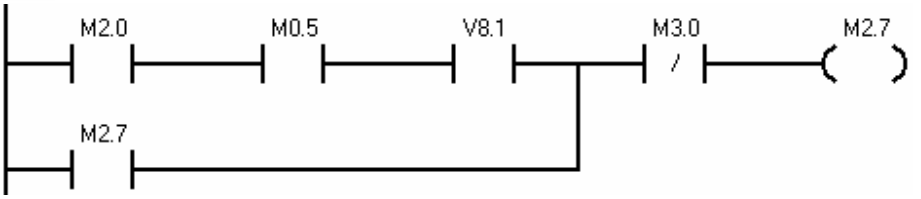
El cinquè segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la tercera planta, si premem el polsador que fa referència a la primera planta, s'activarà l'etapa M2.6.

Network 17



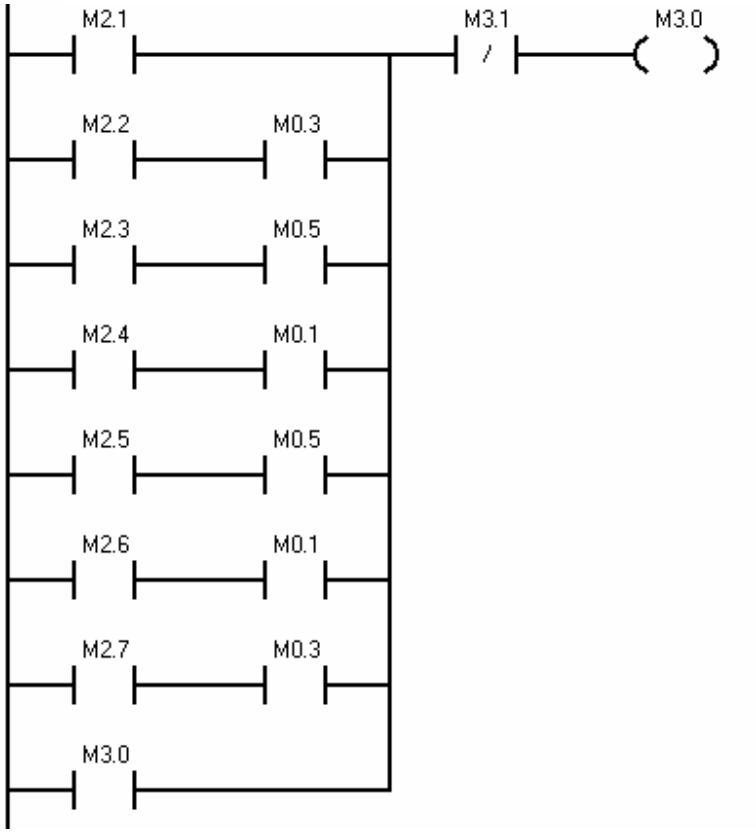
Finalment, el sisè segment ens indica que quan ens trobem a l'etapa inicial (M2.0) i l'ascensor es troba a la tercera planta, si premem el polsador que fa referència a la segona planta, s'activarà l'etapa M2.7

Network 18



El segment que tenim a continuació és el que ens indicarà quan la cabina arriba a la planta desitjada i per tant ha finalitzat el seu recorregut. Quan això succeeixi s'activarà l'etapa M3.0. Aquí podem trobar diferents casos, des del primer en que l'ascensor no s'haurà de moure, serà el cas en que premem el polsador que fa referència a la mateixa planta a la que ens trobem en aquell moment, fins a l'últim cas que serà el que hem descrit en l'etapa M2.7.

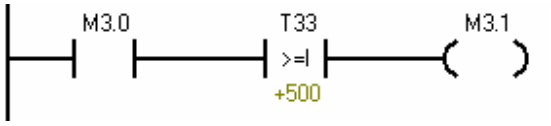
Network 19



Una vegada s'ha executat la ordre i passem a l'etapa M3.0, activarem el timer T33. Quan el valor d'aquest timer superi els cinc segons, s'activarà la última etapa del GRAFCET principal, l'etapa M3.1 Aquest temps de cinc segons l'utilitzem per tancar la porta quan hagin passat cinc segons des de que l'ascensor ha arribat a una planta, és el temps que tenen els usuaris per sortir de la cabina.

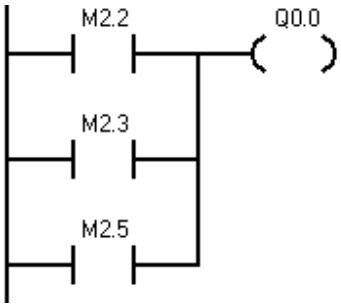
Network 20

Network 21

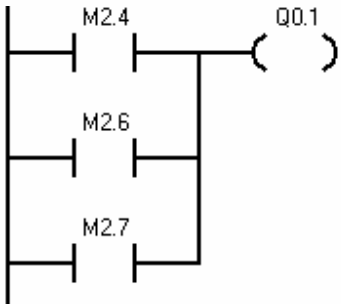


A continuació, tenim els dos segments que ens permeten activar les accions de pujar (Q0.0) i baixar (Q0.1). Aquestes accions estan associades directament a les etapes que hem descrit anteriorment i que fa referència a les possibles situacions en les que es pot trobar l'ascensor. Així quan estiguin actives M2.2, M2.3 o M2.5 s'activarà l'acció de pujar i quan estiguin actives M2.4, M2.6 o M2.7 s'activarà l'acció de baixar.

Network 22

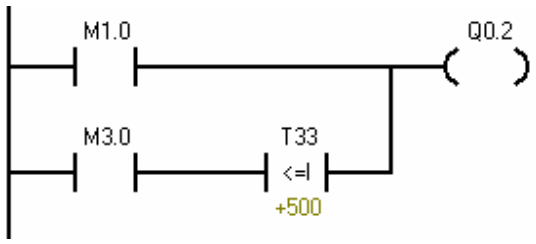


Network 23

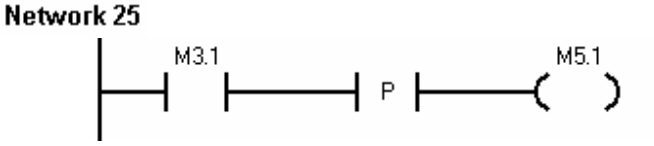


El següent segment és el que ens controla la porta la de l'ascensor. Sempre que estigui activa l'etapa M1.0, i per tant, l'ascensor estigui esperant una ordre, la porta estarà oberta. La porta s'haurà d'obrir quan l'ascensor arribi a la planta desitjada per l'usuari, en aquest cas, s'obrirà durant cinc segons perquè l'usuari pugui sortir de la cabina.

Network 24

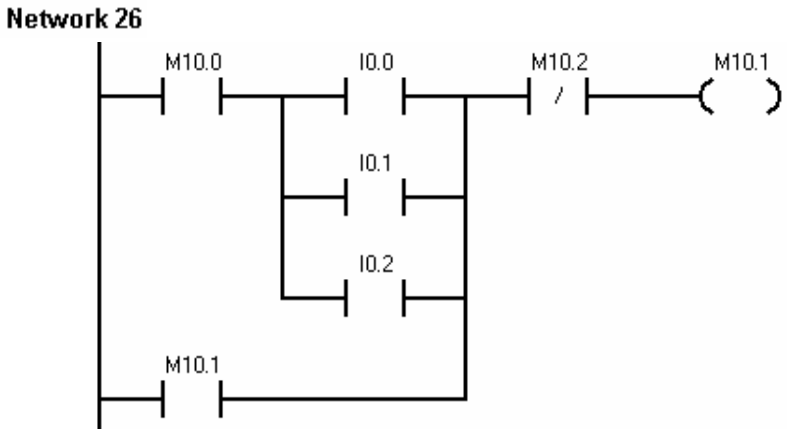


Aquest segment activa la marca 5.1. Aquesta marca indica que la ordre ha estat executada i finalitzada, és a dir, l'ascensor ha arribat a la planta desitjada i està a punt per dur a terme un altre recorregut.



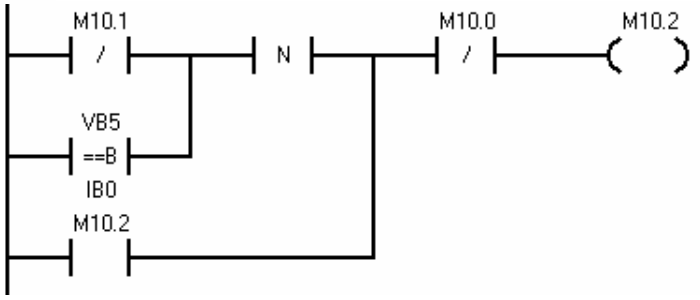
A continuació trobem els segments que fan referència al GRAFCET 2:

El primer segment, ens mostra que si ens trobem a l'etapa inicial (M10.0) i es prem qualsevol dels polsadors s'activa la següent etapa (M10.1).



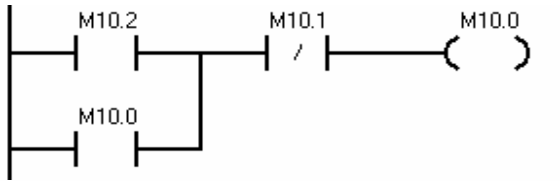
Una vegada ens trobem a l'etapa M10.1, per passar a la següent etapa (M10.2) s'ha de complir que s'hagi deixat de prémer el polsador o que la última ordre sigui diferent de la última memoritzada.

Network 27



En aquest cas, tornem a l'etapa inicial ja que l'etapa M10.2 simplement la utilitzem perquè el GRAFCET no tingui dues etapes i pugui funcionar correctament.

Network 28

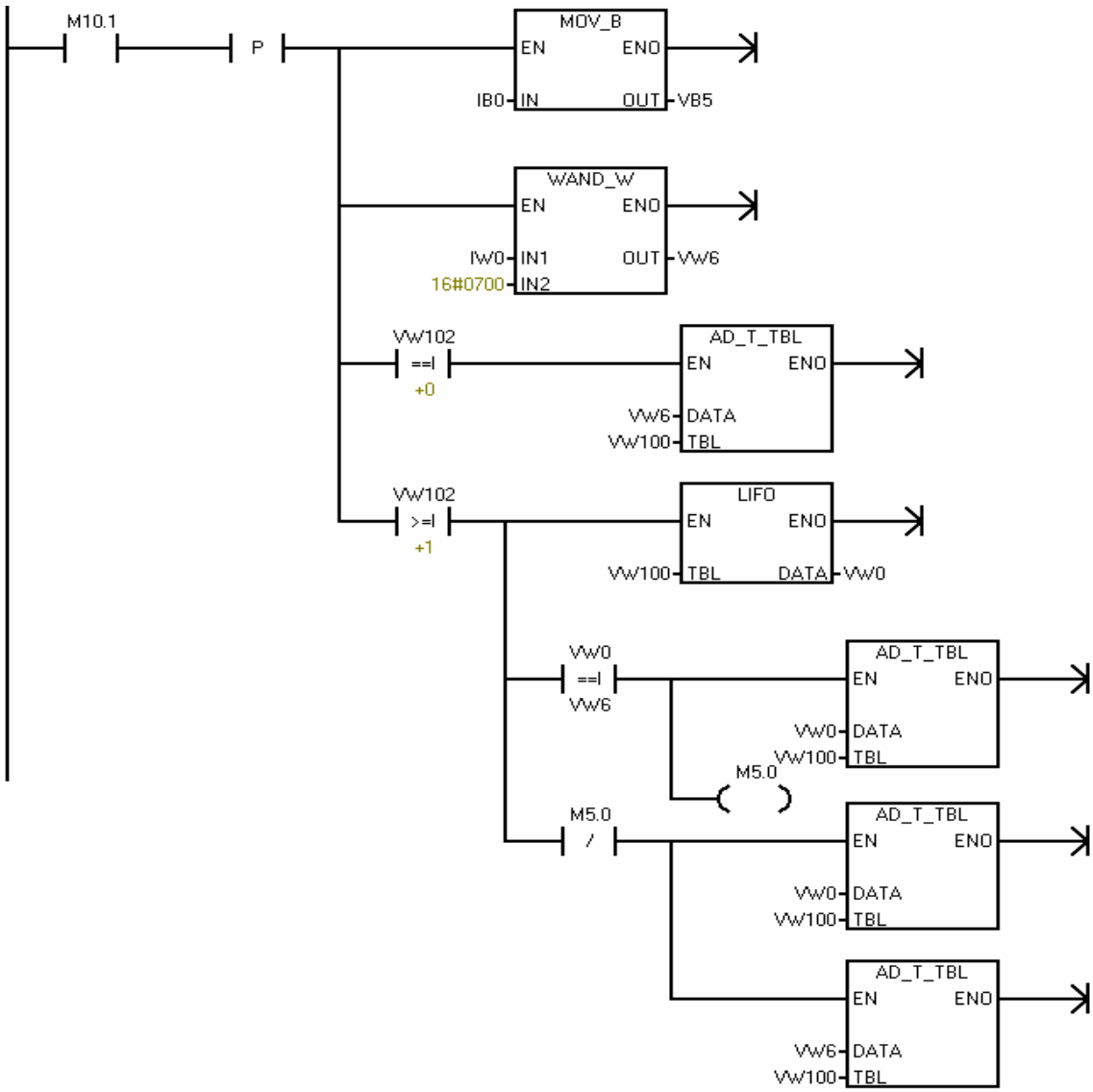


El següent segment és el que ens permet determinar si la última ordre que rep l'ascensor és diferent a la última que s'ha memoritzat. En el moment en que l'etapa M10.1 està activa, el primer que es fa és transferir el valor del polsador que s'ha premut (IBO) a VB5.

Amb el bloc WAND_W, realitzem una màscara per assegurar-nos d'agafar només el valor dels polsadors que ens interessen.

Tot seguit, comprovem si la pila està buida. En el cas que estigui buida, simplement afegim la nova ordre a la cua. En cas contrari, transferim l'últim valor introduït a la pila a VW0. Aquesta acció la podem aconseguir gràcies al bloc LIFO. Aquest procés l'utilitzem per comprovar si la última ordre és igual a la última ordre memoritzada. En el cas que les dues ordres siguin iguals, activem la marca M5.0 i tornem a posar a la pila la ordre que hem tret anteriorment i la nova, que és igual que la anterior, no la utilitzem. En el cas que les dues ordres siguin diferents, simplement posem les dues ordres a la pila en l'ordre correcte perquè es puguin executar correctament.

Network 29


M10.1
P
MOV_B
EN
ENO
IB0
IN
OUT
VB5
WAND_W
EN
ENO
IW0
IN1
OUT
VW6
16#0700
IN2
VW102
==I
+0
AD_T_TBL
EN
ENO
VW6
DATA
VW100
TBL
VW102
>=I
+1
LIFO
EN
ENO
VW100
TBL
DATA
VW0
VW0
==I
VW6
AD_T_TBL
EN
ENO
VW0
DATA
VW100
TBL
M5.0
M5.0
/
AD_T_TBL
EN
ENO
VW0
DATA
VW100
TBL
AD_T_TBL
EN
ENO
VW6
DATA
VW100
TBL

6. Conclusions

Al començament d'aquest projecte ens vam marcar un objectiu principal que consistia en introduir i intentar explotar un nova via en l'aprenentatge de la programació d'autòmats programables, principalment en l'àmbit de la docència. Aquesta nova via es troba entre les dues que s'utilitzen habitualment, és a dir, entre la utilització de la planta real i la simulació manual d'aquesta.

La via que proposem consisteix en simular la planta automatitzada mitjançant un microcontrolador. Per introduir aquesta nova metodologia i demostrar-ne la seva utilitat hem utilitzat alguns exemples de diferents graus de dificultat.

Per arribar al resultat final hem seguit diferents fases que ens han plantejat diferents problemes en diferents àmbits de l'enginyeria.

La primera fase va consistir en la programació del microcontrolador. El principal problema que vam observar va consistir en el fet que fóssim capaços d'adaptar-nos a pensar en la lògica de la planta sense tenir en compte les possibles accions de control. La lògica per programar la planta és molt diferent a la que utilitzem quan dissenyem programes de control per un autòmat per controlar aquesta planta.

En aquest cas el grau de dificultat a l'hora de programar el microcontrolador vindrà donat pel grau de complexitat de la planta en sí i també pel nombre de situacions que siguem capaços de tenir en compte i, per tant, d'aproximar el màxim possible la simulació a la planta real.

La planta simulada que hem escollit, un ascensor, ens ofereix un ampli ventall de possibilitats a l'hora de programar l'autòmat. Tal i com hem vist en els exemples que plantegem es poden dissenyar programes de control amb unes característiques ben diferents.

En quant a la connexió entre la l'autòmat i la planta hem pogut comprovar que és del tot factible poder comunicar aquests dos elements amb una sèrie de components que formen circuits electrònics que no són excessivament complexos. A més, aquests circuits no contenen elements amb un preu elevat i per tant el preu total de la placa no és elevat.

La placa que hem dissenyat té unes característiques molt útils per a la funció perquè la volem utilitzar. La primera característica és que es tracta d'una planta que permet una connexió fàcil amb l'autòmat, també té unes dimensions reduïdes que ens permet poder moure la placa sense problemes i, per tant, utilitzar-la en diferents autòmats o diferents laboratoris.

El conjunt d'exercicis que plantegem ofereix exemples de diferents graus de complexitat així com també utilitza diferents elements a l'hora de programar l'autòmat. És important demostrar que la planta que hem escollit pot oferir vàries possibilitats de control ja que, a la fi, ens interessa que l'usuari treballi i aprengui a utilitzar al màxim les eines de què disposa en el software de programació.

Finalment hem de parlar de la implementació de la metodologia d'aprenentatge de la placa. Hem demostrat que és possible utilitzar la nova via d'aprenentatge que hem proposat.

Hem pogut comprovar que gràcies a la placa que hem dissenyat i construït podem detectar errors en els programes de control que sense aquesta placa no podríem detectar, o bé, ens seria més complicat aconseguir-ho.

Si únicament comprovem el funcionament del programa de control amb l'autòmat, es pot donar el cas que la planta evolucioni correctament, però s'hagi produït alguna circumstància que, en el cas de que apliquéssim aquest programa de control sobre la planta real, ens danyés la planta. Aquest error únicament el podríem detectar gràcies al indicador que tenim en la placa que simula planta automatitzada.

Així, els indicadors de la placa són de gran utilitat. Sobretot el que ens indica que s'ha produït un error, o bé, que la planta s'ha danyat, ja que l'usuari rep una informació que és capaç d'interpretar a l'instant sense cap mena de problema.

7. Línies futures de treball

Aquest treball, podríem dir que només és el primer pas en l'àmbit de la simulació de plantes automatitzades amb microcontrolador. Per aquest motiu es pot continuar aquesta tasca en el camí que proposem més endavant.

En aquest treball hem simulat una única planta automatitzada. Perquè els estudiants tinguessin un ventall més ampli de possibles exemples seria interessant introduir nous codis en el microcontrolador que simulessin altres plantes automatitzades.

Seria ideal que el microcontrolador emmagatzemés tants codis de plantes com la seva memòria ens permetés, així l'usuari podria escollir amb quina planta vol treballar en cada moment mitjançant una possible combinació de polsadors, interruptors o algun altre mecanisme.

S'ha de tenir en compte que la placa dissenyada en aquest treball està adaptada a les necessitats que ens demana la planta que hem simulat, en el nostre cas un ascensor. Per aquest motiu, hem de tenir en compte que les diferents plantes que simulem han de tenir unes característiques semblants, és a dir, que es puguin aprofitar els indicadors de la placa per a tots els casos.

El nombre d'entrades i sortides de la nostra placa es pot modificar sense inconvenients, ja que, tal i com hem comentat al llarg del treball, el circuit que utilitzem per cada entrada és el mateix i succeeix el mateix per a les sortides. Aquests circuits s'aprofitarien per a la nova placa.

En quant als indicadors, es podrien utilitzar altres indicadors que oferissin la informació de diferent manera a l'usuari.

Podria ser interessant que la placa disposés de una pantalla LCD, no seria necessari que tingués unes grans dimensions, on l'usuari pogués anar llegint la informació referent a l'estat de la planta. Possiblement seria més fàcil d'entendre o, si més no, no seria tan necessari com en el nostre cas explicar amb detall quina és la informació que ens dóna cada indicador.

Pels indicadors que ens informen que s'ha produït un error en la planta, es podrien utilitzar indicadors sonors, com una sirena o un timbre. En aquest cas l'usuari associaria ràpidament la informació que rep a un error més o menys greu en el

funcionament de la planta i, per tant, s'hauria de modificar el programa de control si vol tenir un control adequat sobre la planta.

8. BIBLIOGRAFIA I RECURSOS ELECTRÒNICS WEB

1) Michel, G., "*Autómatas programables industriales arquitectura y aplicaciones"*, Marcombo Boixareu Editores, 1990**.**

2) Mandado, E., "*Controladores lógicos y autómatas programables*", Marcombo, S.A./Boixareu Editores, 1990.

3) Boix, O., "*Automatització industrial amb GRAFCET*", Edicions UPC, 2001.

4) Stenerson, J., "*Fundamentals of programmable logic controllers, sensors, and communications*", Upper Saddle River, NJ : Prentice Hall, 1999

5) Hintz, Kenneth J., "*Microcontrollers architecture, implementation and programming*", New York McGraw-Hill, 1992.

6) Torres, V., "*Microcontroladores*", Escuela Politécnica Superior de Gandía. Universidad Politécnica de Valencia, 1999.

7) Angulo, J., "*Microcontroladores "PIC" diseño práctico de aplicaciones*", Madrid McGraw-Hill 1999-2000.

8) Del Barrio, R. "*How to program PICS with C-language*", projecte fi de carrera UPC. 2002.

9) *www.infoplc.net*

Pàgina amb informació general sobre els autòmats i aplicacions d'aquests.

10) *www.microchip.com*

Pàgina de la casa Microchip, fabricant del microcontrolador utilitzat.

11) *www.wikipedia.org*

Enciclopèdia virtual.

12) *www.siemens.es*

Pàgina de la casa Siemens, fabricant de l’autòmat utilitzat.

13) *www.alldatasheet.com*

Buscador de datasheets de diferents dispositius electrònics.

9. ANNEX I: CODI DEL MICROCONTROLADOR

```
#include <16f873.h>
#fuses XT,NOWDT,NOPROTECT,NOLVP,PUT,BROWNOUT
#use delay (clock=6000000)
#use fast_io(b)
#byte PORTB=6

//ENTRADES: A0->pujar; A1->baixa; A2->obrir la porta
//SORTIDES: B0;B1;B2->plantes; B3->porta oberta; B4>error;

#define tiempomax 5
#define contador 5

void main()
{
   long int tiempo=0;
   int c12=0;
   int c21=0;
   int c23=0;
   int c32=0;
   setup_timer_0(RTCC_INTERNAL|rtcc_div_256);
   disable_interrupts(global);
   set_tris_b(0x00);
   set_tris_a(0x07);
   PORTB=0x01;

   while(!input(PIN_B4))
   {
         delay_ms(333);

         if(input(PIN_A2))
         {
                  output_bit(PIN_B3,1);
         }else{
                  output_bit(PIN_B3,0);
               }

         if(((input(PIN_B0)&!input(PIN_B1)&!input(PIN_B2))&(input(PIN_A0))&(!input(PIN_A1))))
         {
                  tiempo=0;
                  c12=0;
                  c21=0;
                  delay_ms(3000);
```

```
        PORTB=0x03;
}

if((input(PIN_B0)&!input(PIN_B1)&!input(PIN_B2))&(input(PIN_A1))&(!input(PIN_A0)))
{
        c12=0;
        c21=0;
        tiempo++;
}

if((!input(PIN_B0)&input(PIN_B1)&!input(PIN_B2))&(input(PIN_A0))&(!input(PIN_A1)))
{
        tiempo=0;
        c12=0;
        c21=0;
        c23=0;
        c32=0;
        delay_ms(3000);
        PORTB=0x06;
}

if((!input(PIN_B0)&input(PIN_B1)&!input(PIN_B2))&(input(PIN_A1))&(!input(PIN_A0)))
{
        tiempo=0;
        c12=0;
        c21=0;
        c23=0;
        c32=0;
        delay_ms(3000);
        PORTB=0x03;
}

if((!input(PIN_B0)&!input(PIN_B1)&input(PIN_B2))&(input(PIN_A0))&(!input(PIN_A1)))
{
        c23=0;
        c32=0;
        tiempo++;
}

if((!input(PIN_B0)&!input(PIN_B1)&input(PIN_B2))&(input(PIN_A1))&(!input(PIN_A0)))
{
        tiempo=0;
        c23=0;
        c32=0;
        delay_ms(3000);
        PORTB=0x06;
}

if((input(PIN_B0)&input(PIN_B1)&!input(PIN_B2))&(input(PIN_A0))&(!input(PIN_A1)))
{
        tiempo=0;
        c21=0;
        c12++;
```

```
}

if((input(PIN_B0)&input(PIN_B1)&!input(PIN_B2))&(input(PIN_A1))&(!input(PIN_A0)))
{
	tiempo=0;
	c12=0;
	c21++;
}

if((!input(PIN_B0)&input(PIN_B1)&input(PIN_B2))&(input(PIN_A0))&(!input(PIN_A1)))
{
	tiempo=0;
	c32=0;
	c23++;
}

if((!input(PIN_B0)&input(PIN_B1)&input(PIN_B2))&(input(PIN_A1))&(!input(PIN_A0)))
{
	tiempo=0;
	c23=0;
	c32++;
}

if((!input(PIN_A0))&(!input(PIN_A1)))
{
	c12=0;
	c21=0;
	c23=0;
	c32=0;
}

if((input(PIN_A0))&(input(PIN_A1)))
{
	tiempo++;
}

if(tiempo>=tiempomax)
{
	output_bit(PIN_B4,1);
}

if(c12>=contador)
{
	PORTB=0x02;
}

if(c21>=contador)
{
	PORTB=0x01;
}

if(c23>=contador)
{
```

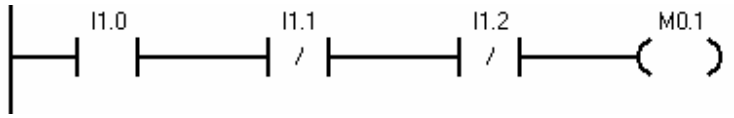
```
                PORTB=0x04;
        }

        if(c32>=contador)
        {
                PORTB=0x02;
        }
    }
}
```

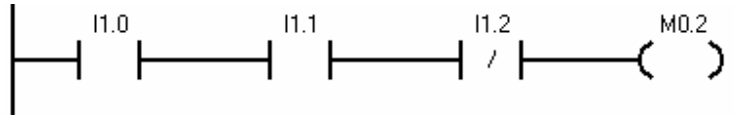
10. ANNEX II: DIAGRAMES DE CONTACTES DELS EXERCICIS

Diagrama de contactes de l'exemple 1

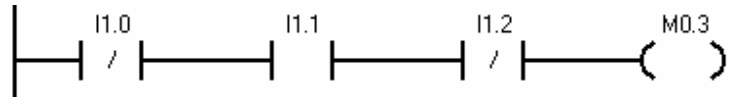
Network 1



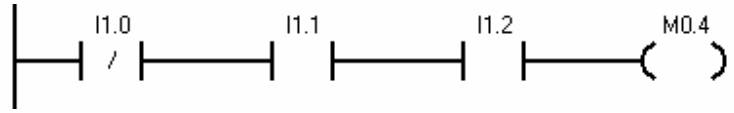
Network 2



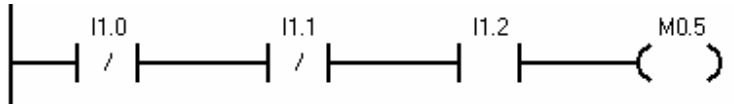
Network 3



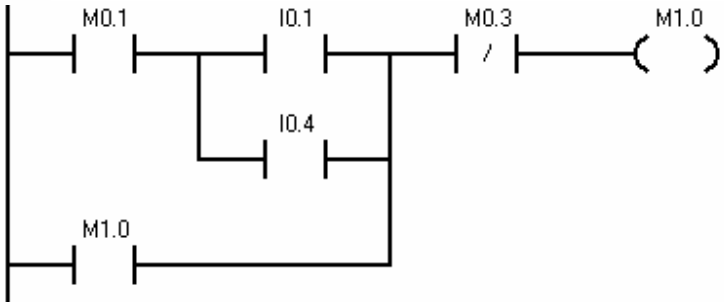
Network 4



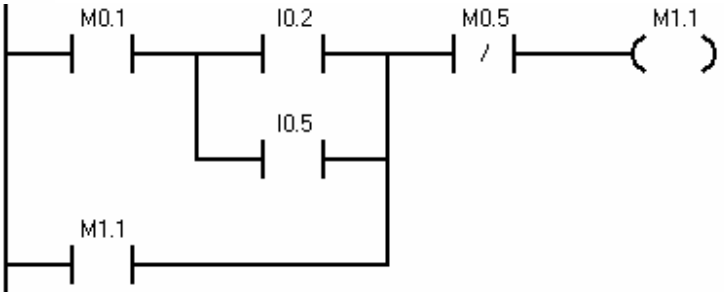
Network 5



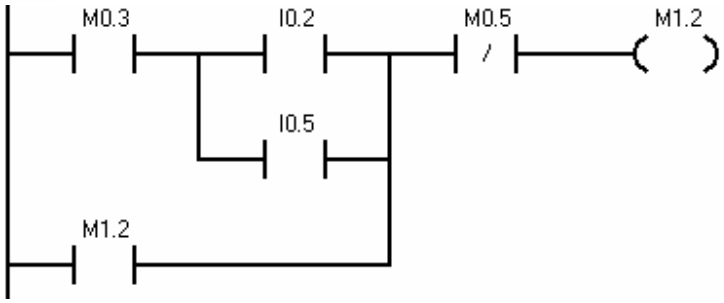
Network 6



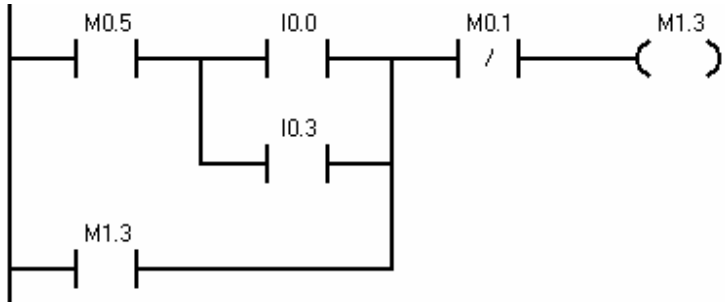
Network 7



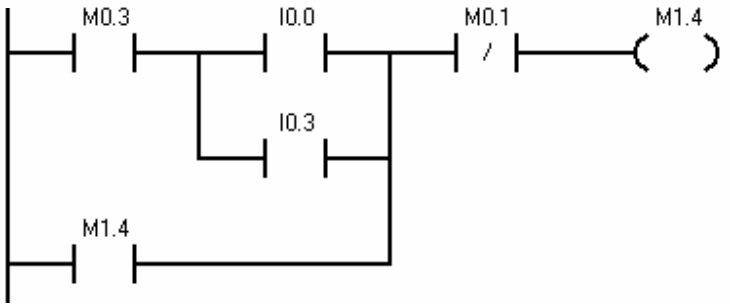
Network 8



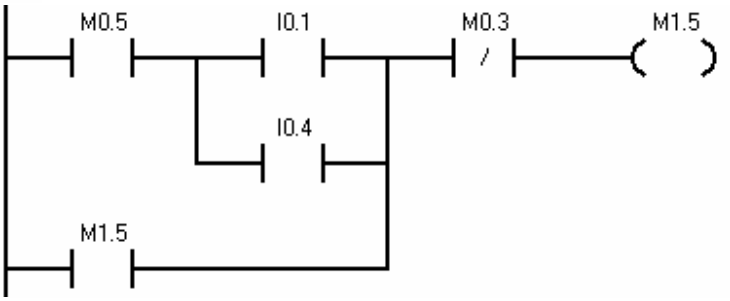
Network 9



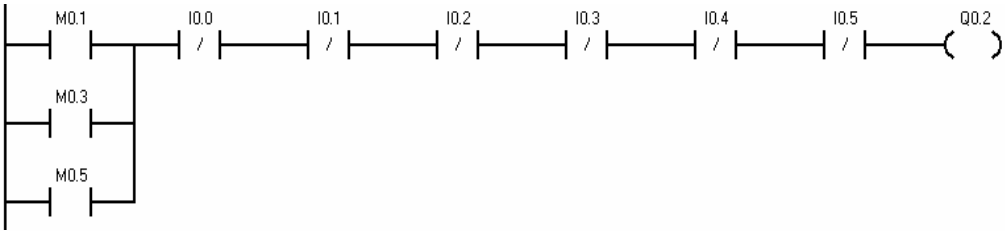
Network 10



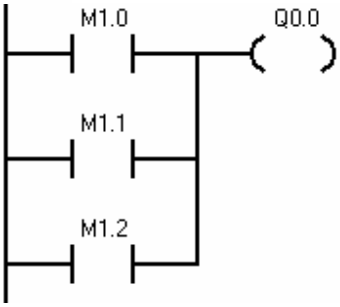
Network 11



Network 12



Network 13



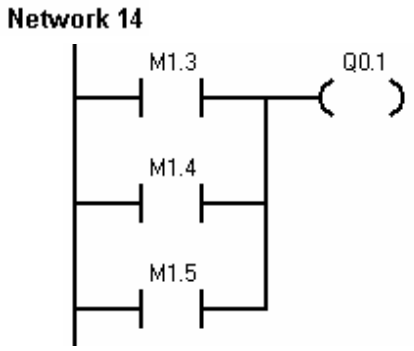
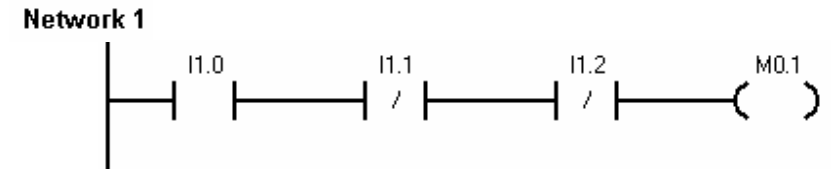
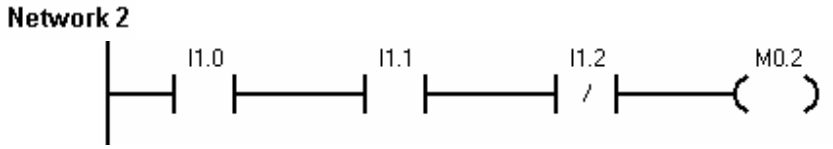
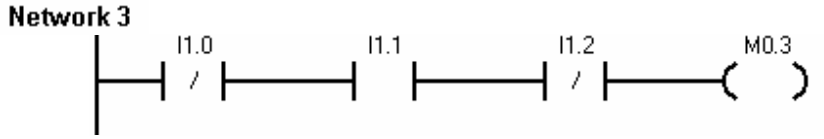
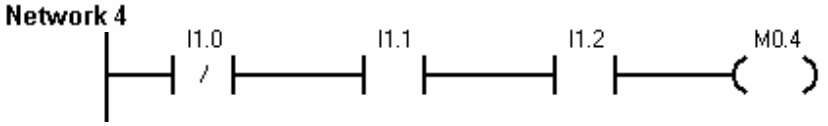


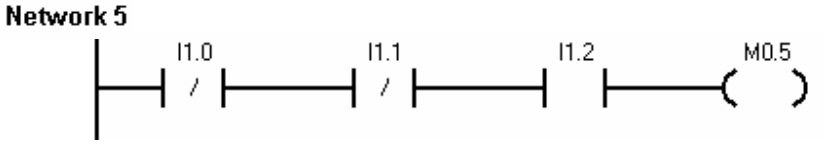
Diagrama de contactes de l'exemple 1.1



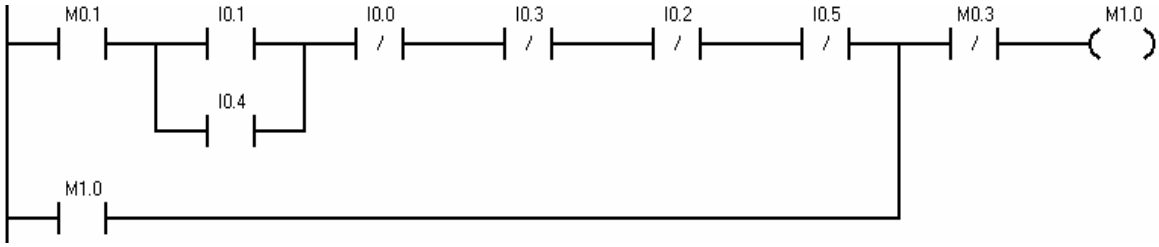




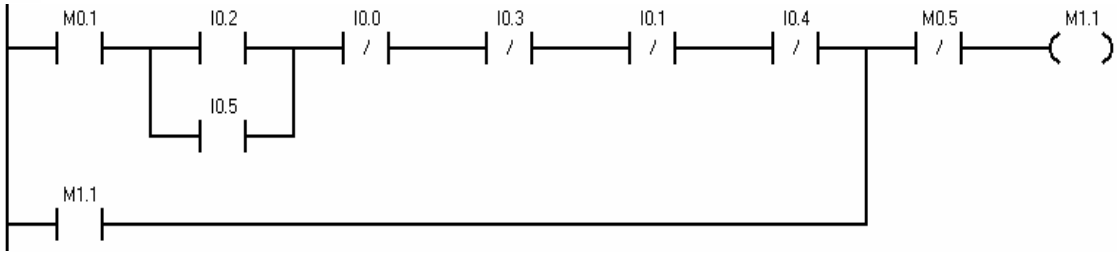




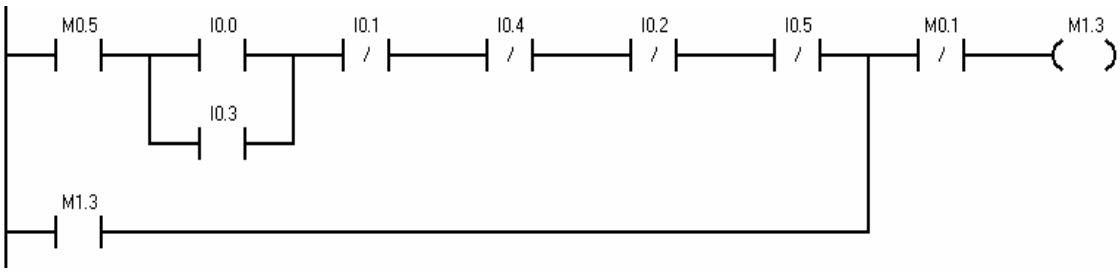
Network 6



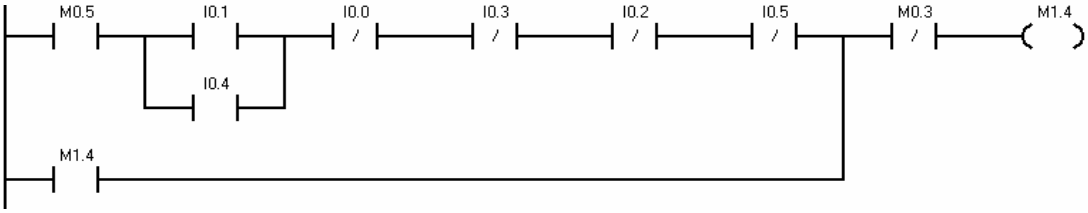
Network 7



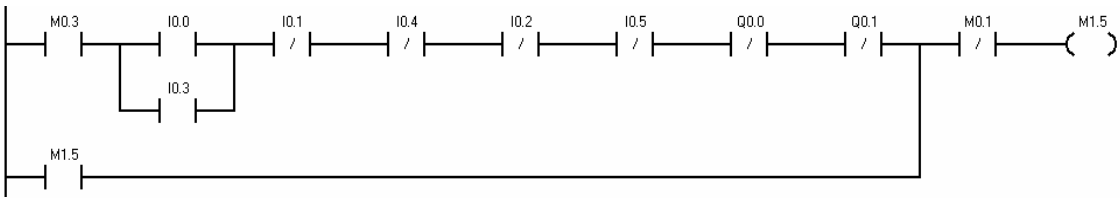
Network 8



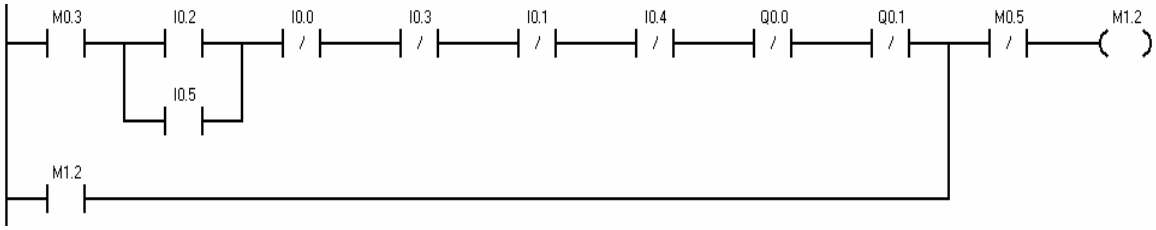
Network 9



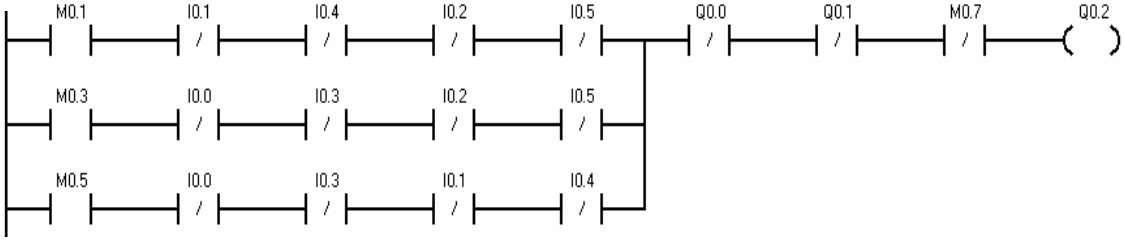
Network 10



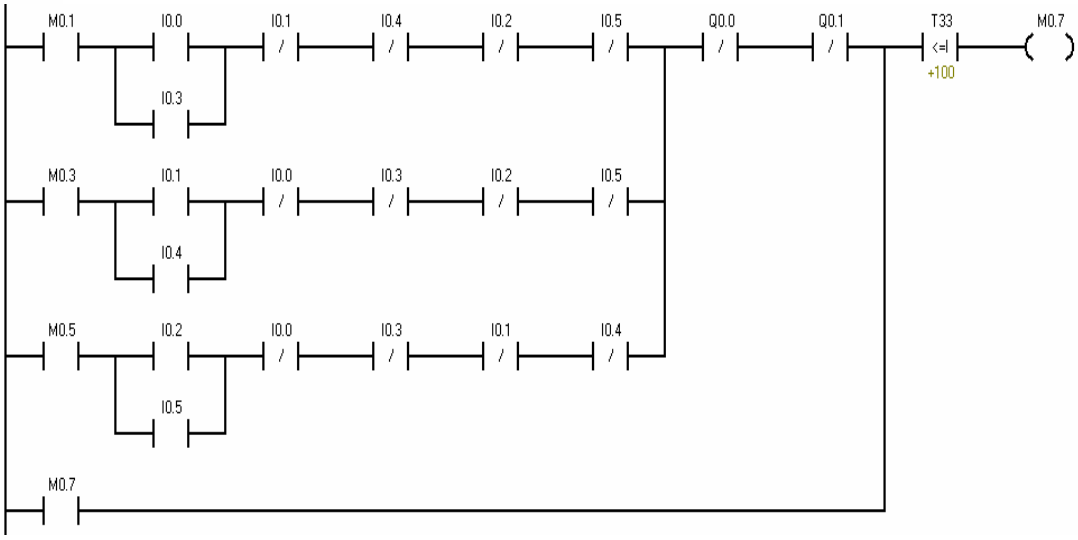
Network 11



Network 12

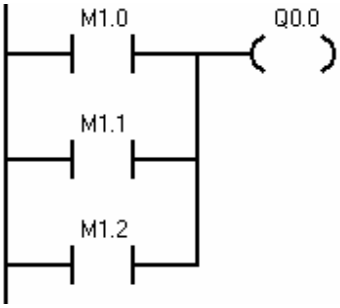


Network 13



Network 14

Network 15



Network 16

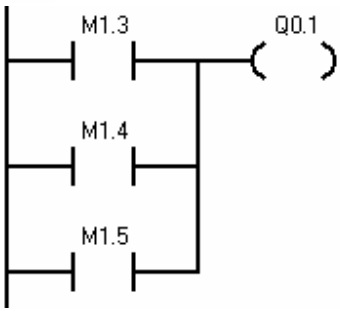
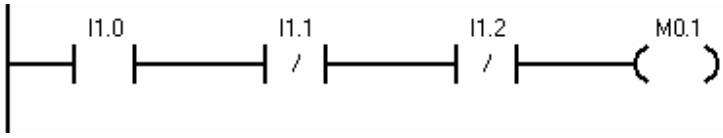
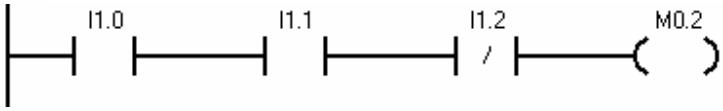


Diagrama de contactes de l'exemple 2

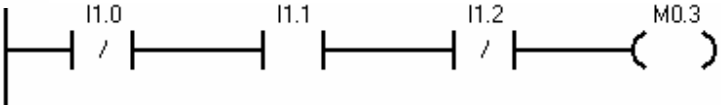
Network 1



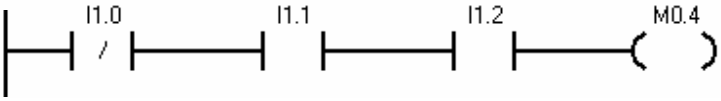
Network 2



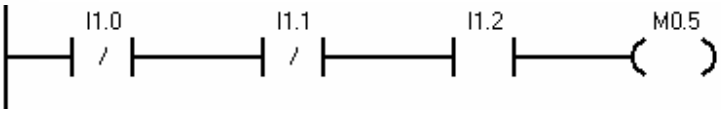
Network 3



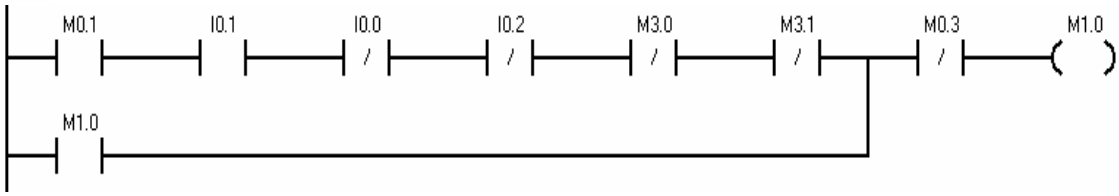
Network 4



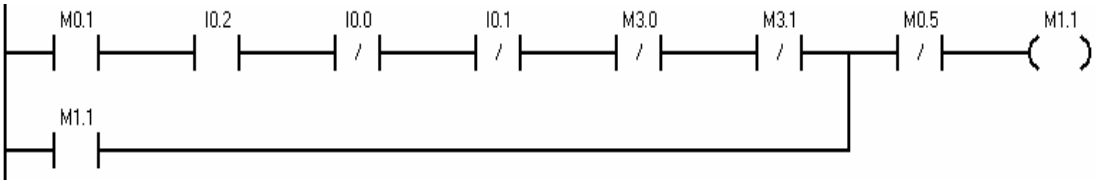
Network 5



Network 6

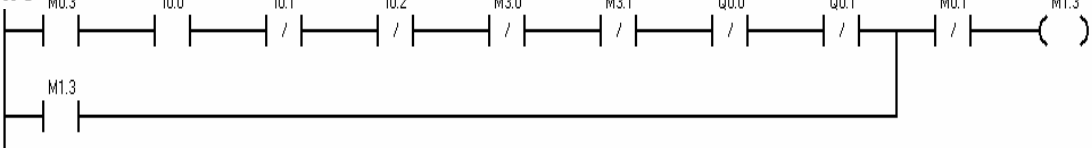


Network 7

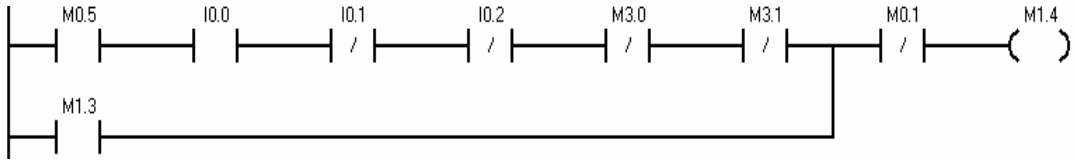


Network 8

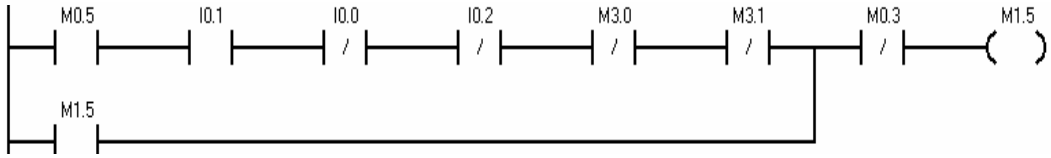
Network 9



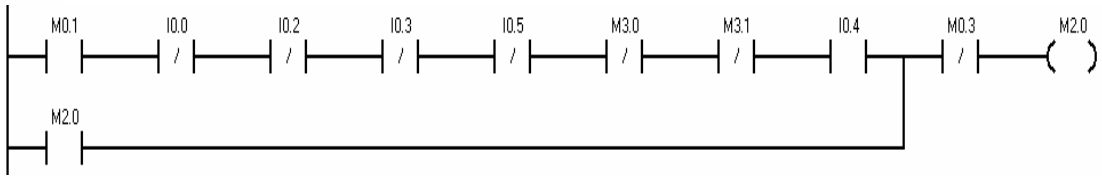
Network 10



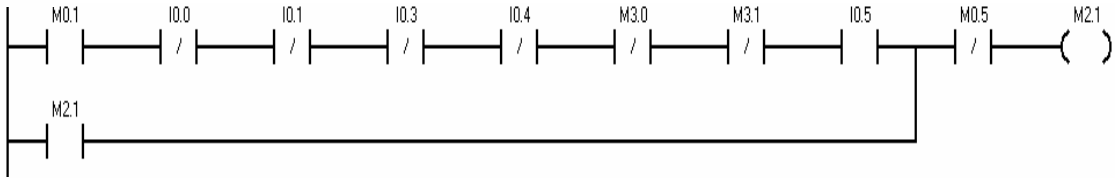
Network 11



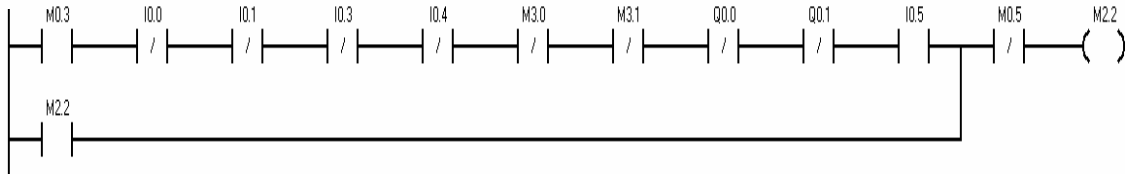
Network 12



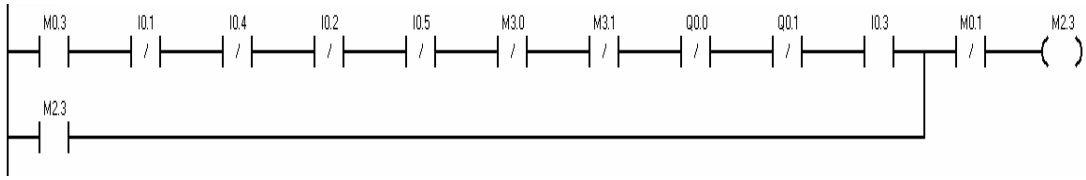
Network 13



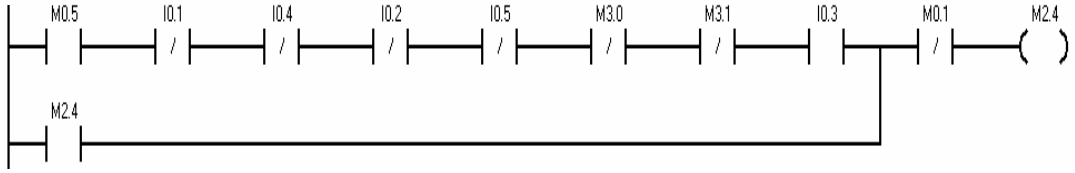
Network 14



Network 15

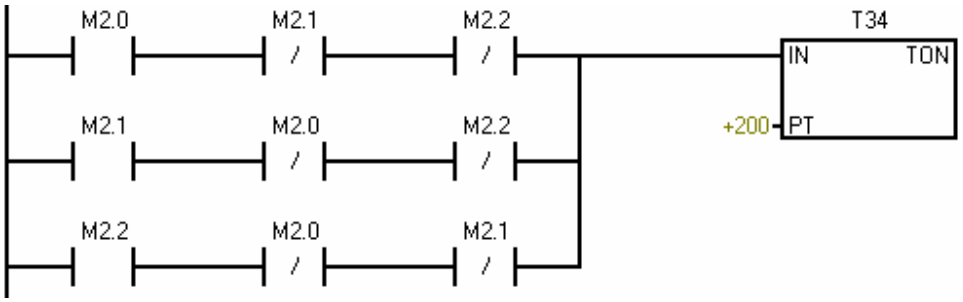


Network 16

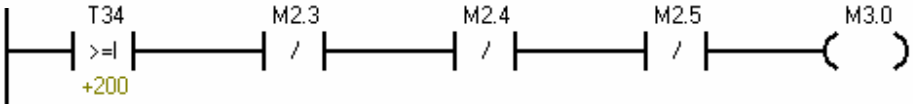


Network 17

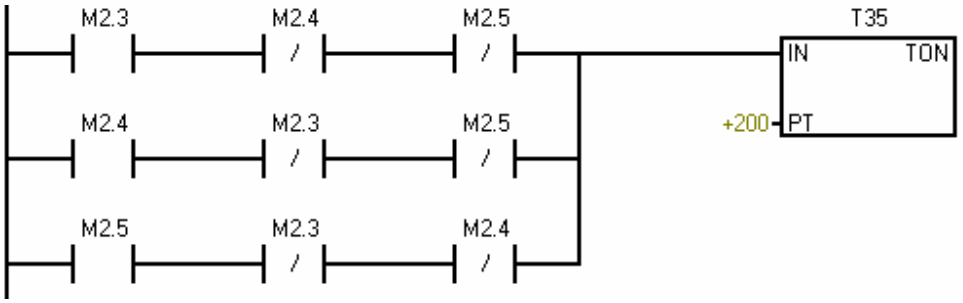
Network 18



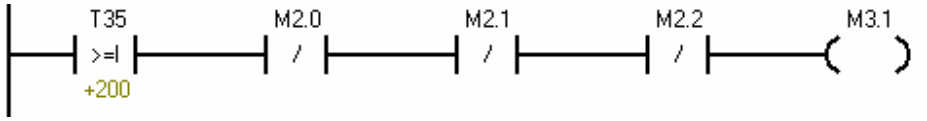
Network 19



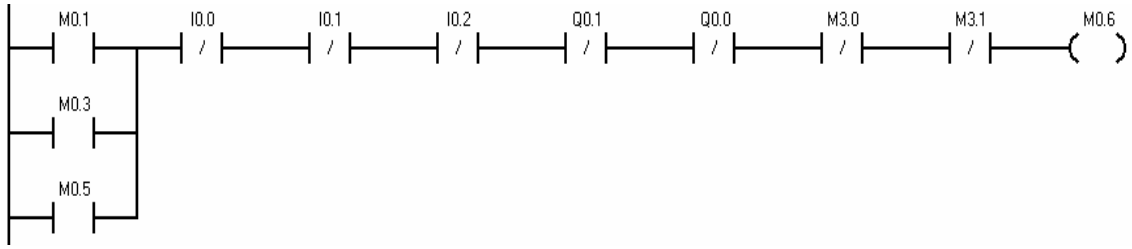
Network 20



Network 21

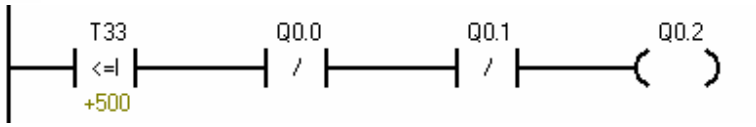


Network 22

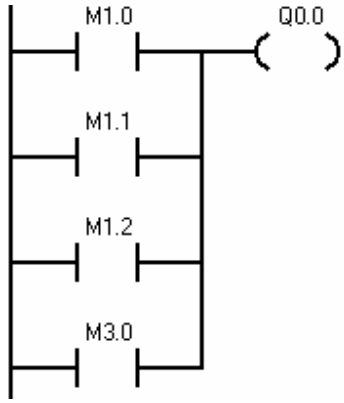


Network 23

Network 24



Network 25



Network 26

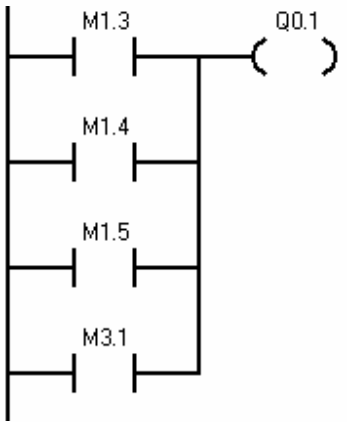
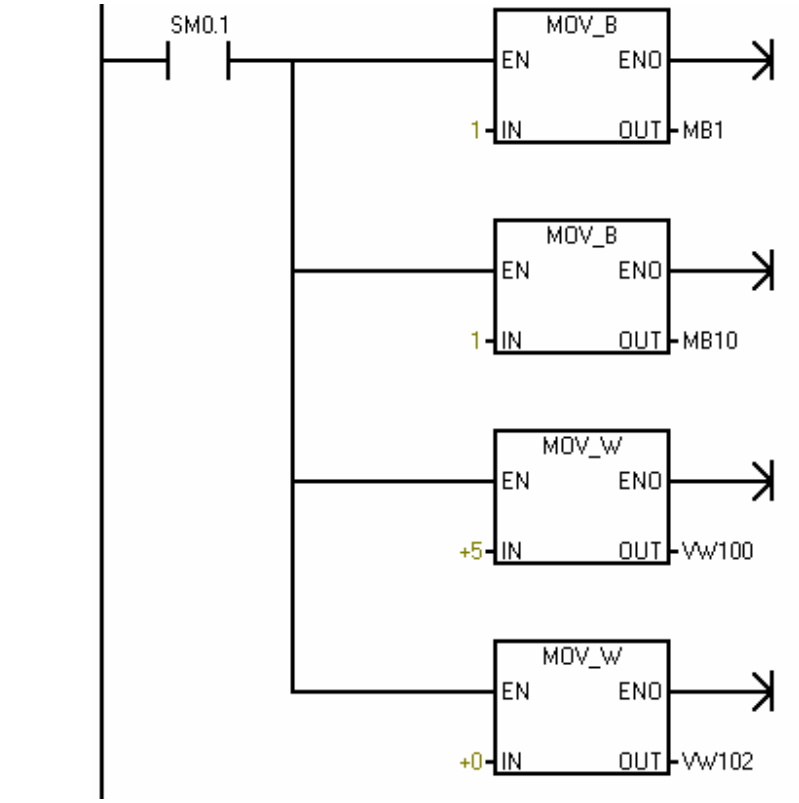
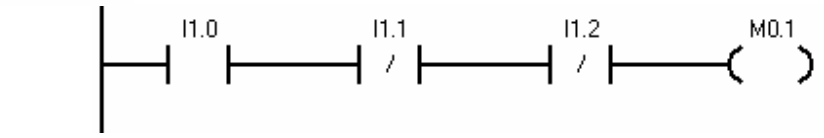


Diagrama de contactes de l'exemple 3

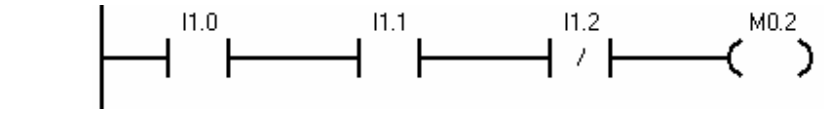
Network 1



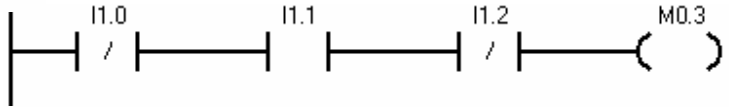
Network 2



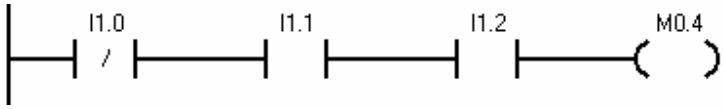
Network 3



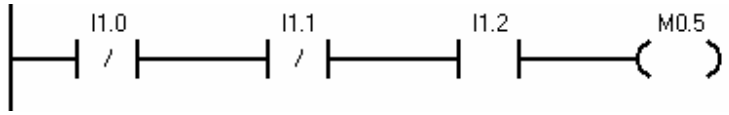
Network 4



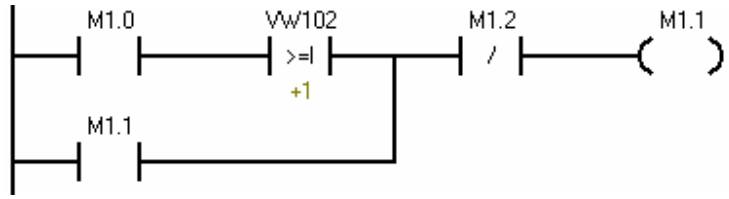
Network 5



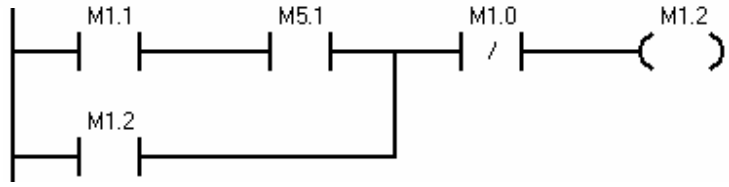
Network 6



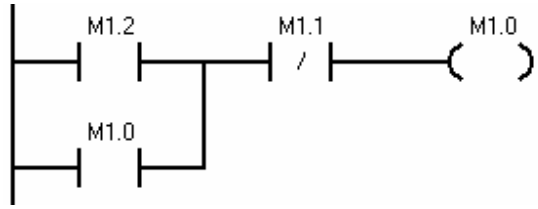
Network 7



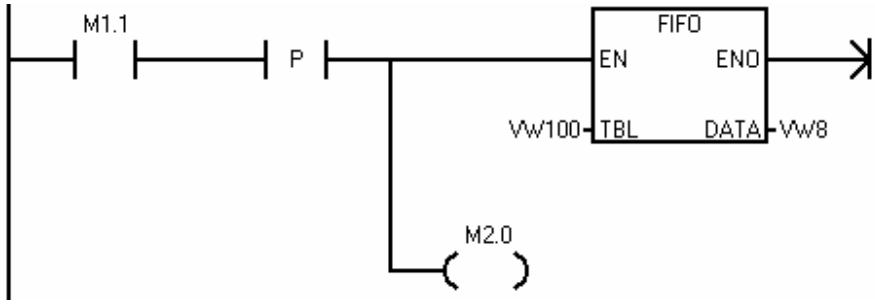
Network 8



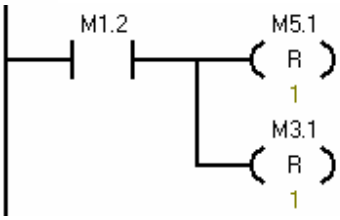
Network 9



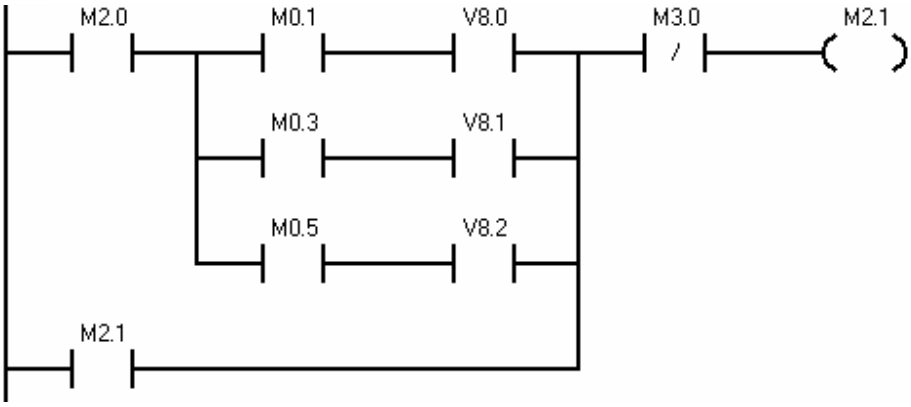
Network 10



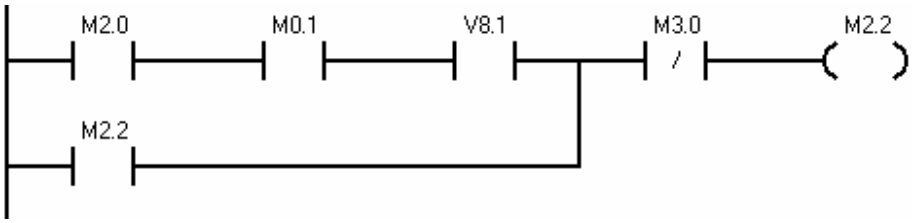
Network 11



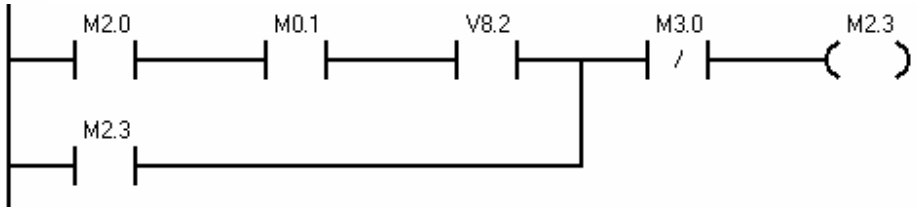
Network 12



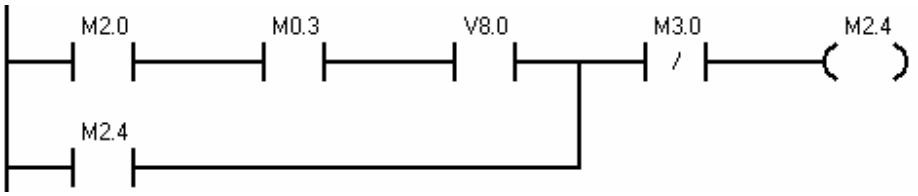
Network 13



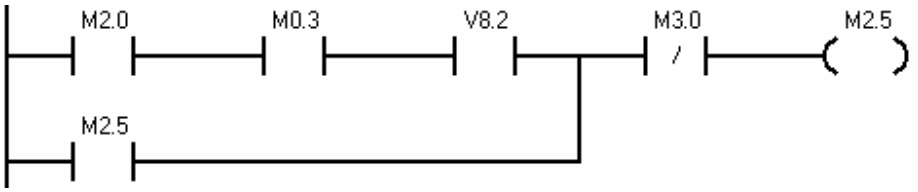
Network 14



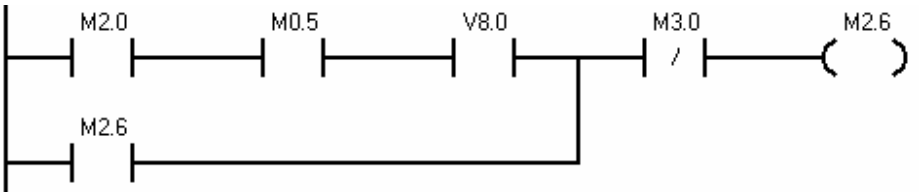
Network 15



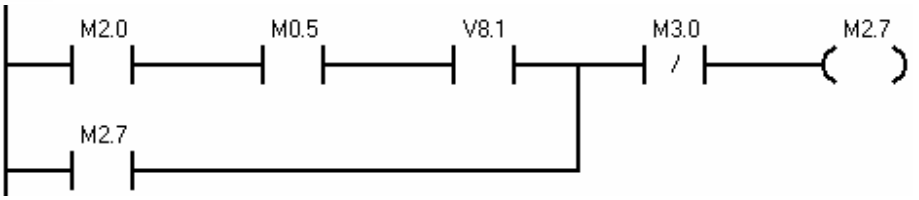
Network 16



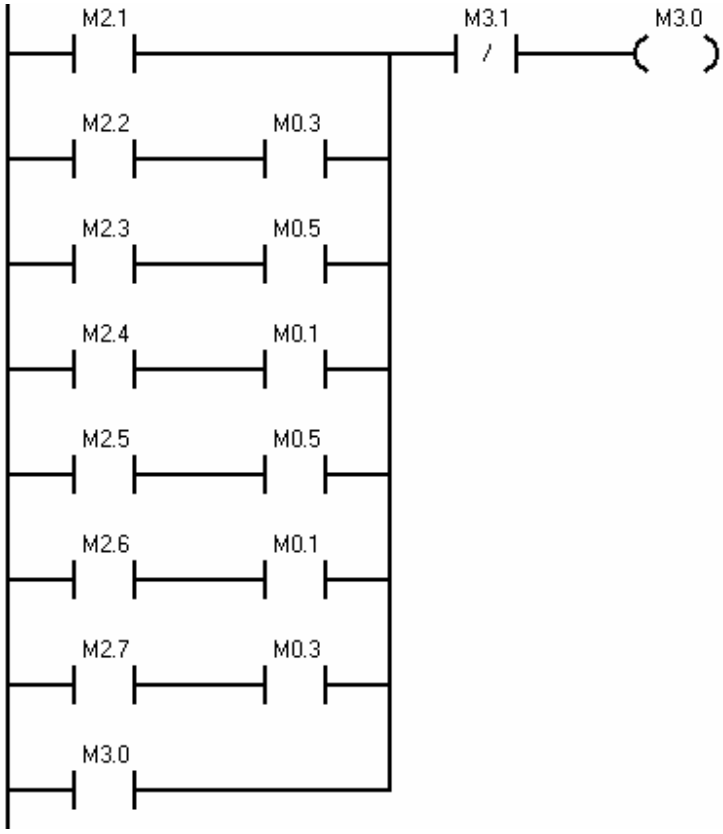
Network 17



Network 18

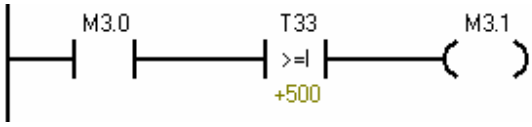


Network 19

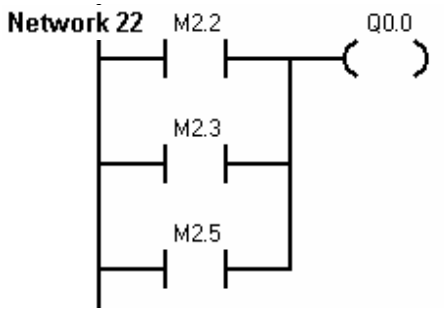


Network 20

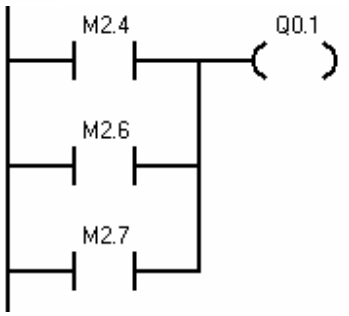
Network 21



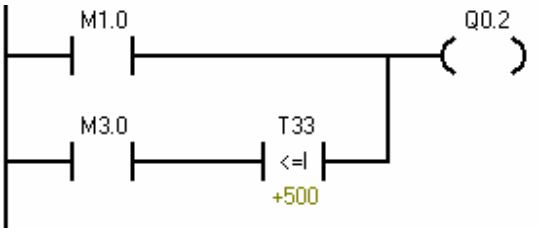
Network 22



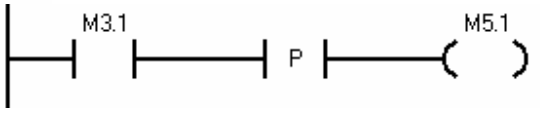
Network 23



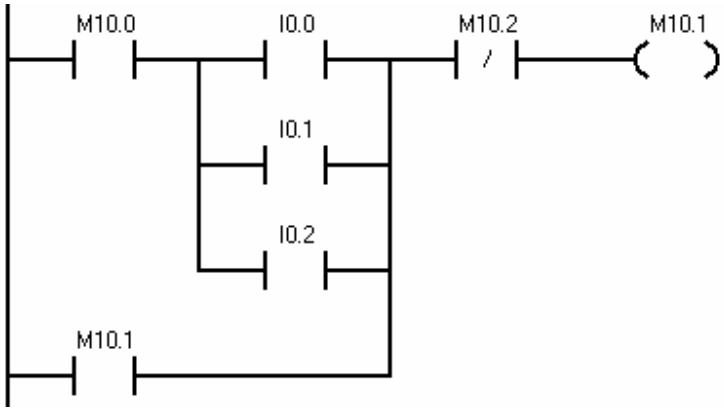
Network 24



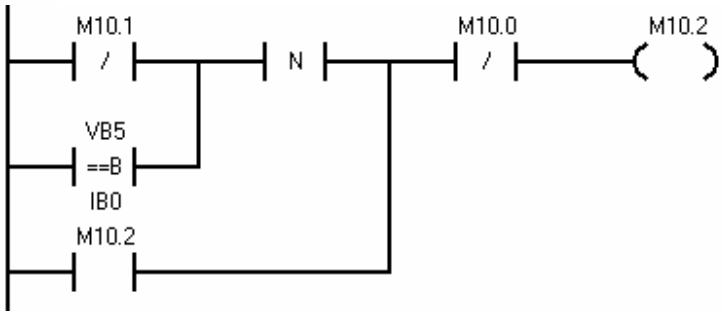
Network 25



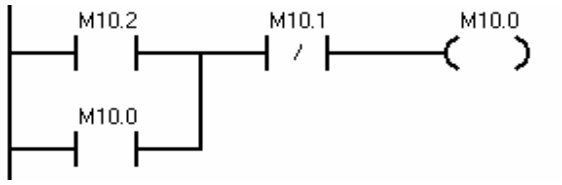
Network 26



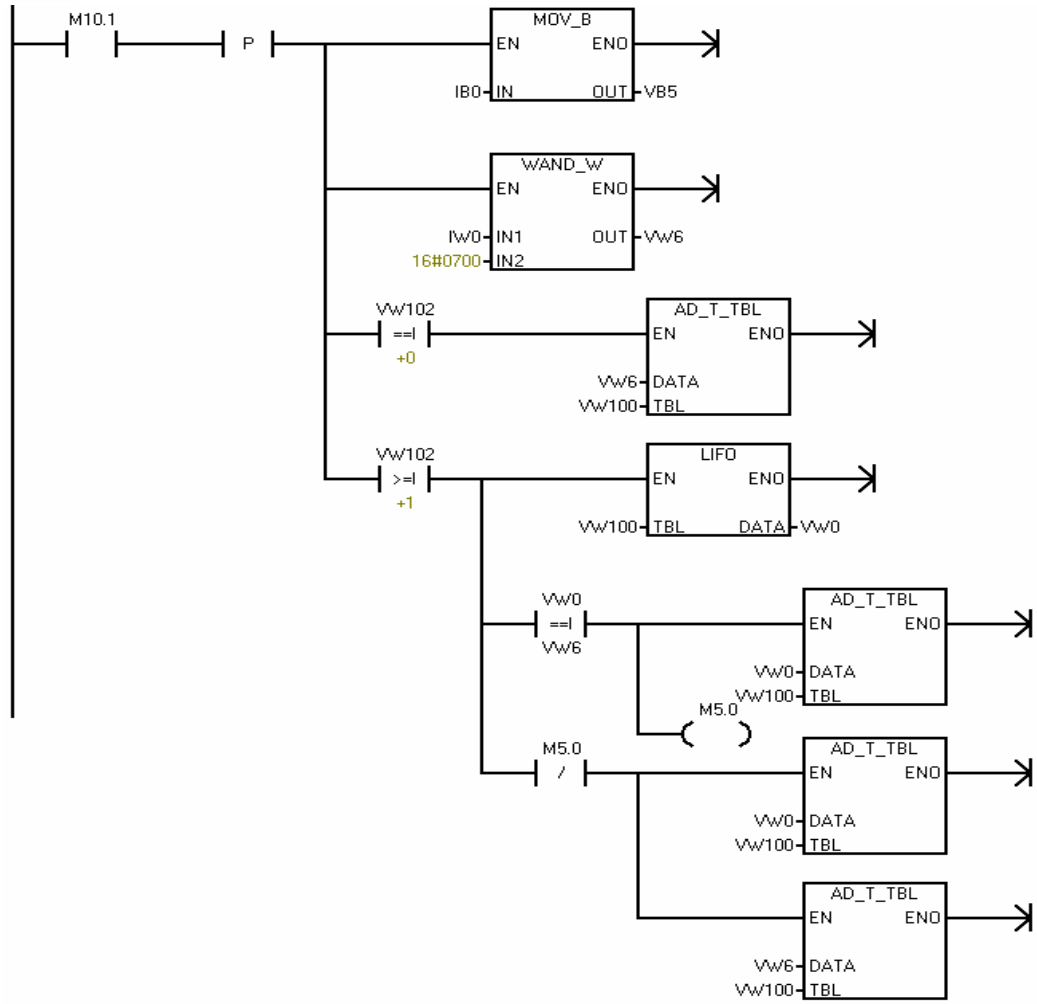
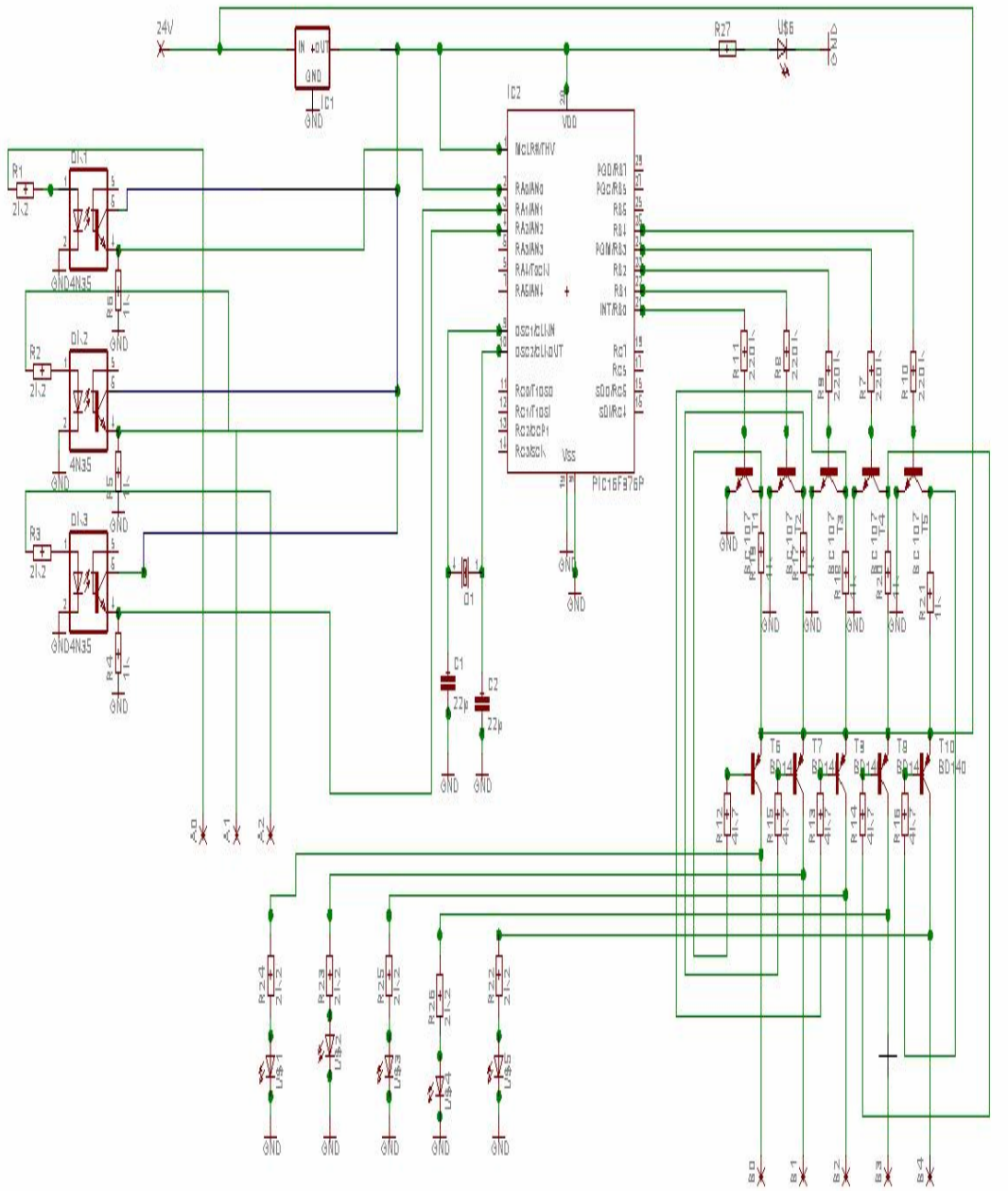
Network 27



Network 28



Network 29


M10.1
P
MOV_B
EN
ENO
IB0
IN
OUT
VB5
WAND_W
EN
ENO
IW0
IN1
OUT
VW6
16#0700
IN2
VW102
==I
+0
AD_T_TBL
EN
ENO
VW6
DATA
VW100
TBL
VW102
>=I
+1
LIFO
EN
ENO
VW100
TBL
DATA
VW0
VW0
==I
VW6
AD_T_TBL
EN
ENO
VW0
DATA
VW100
TBL
M5.0
M5.0
/
AD_T_TBL
EN
ENO
VW0
DATA
VW100
TBL
AD_T_TBL
EN
ENO
VW6
DATA
VW100
TBL

11. ANNEX III: ESQUEMA ELÈCTRIC DE LA PLACA