

Robot planar auxiliar de dos graus de llibertat per cel·la de braços robòtics

Autor:
Ferran Cortés i Celigüeta

Director:
Josep M^a Mirats i Tur



Agraïments

Vull agrair a totes les persones que han fet possible aquest Projecte Final de Carrera, especialment de l'Institut de Robòtica i Informàtica Industrial, i a l'Institut mateix.

Al director del projecte, Dr. Josep M^a Mirats Tur pel seu recolzament, ajuda i optimisme.

A Sergi Hernández per les llargues tardes davant l'ordinador ajudant-me en el desenvolupament i depuració del codi.

A Josep Camps, pel seu interès i per no dubtar ni un moment en oferir la seva ajuda tants cops com fes falta.

Al Dr. Ramón Bragós Bardia per les pautes i la confiança donades.

I en general, a tots aquells que, d'una forma o una altra, han ajudat a que el projecte hagi arribat a materialitzar-se: Joan Pérez, Patrick Grosch, Federico Thomas, Guillem Alenyà, i tots els demés.

A la meva família, es clar. En especial als meus pares, Fernando i Mercedes, els principals patidors, causants, artífexs i responsables de que avui, això, hagi esdevingut una realitat. I a la tieta M^a Rosa, pel seu exemple.

A l'Albert, per haver estat un gran company i amic en el camí conjunt que ens ha portat a aconseguir un mateix objectiu.

I a la Rosa, per la paciència, la comprensió i el sacrifici, especialment en les darreres setmanes de gestació... d'aquest projecte.

A ells, i a tots els altres, moltes gràcies.

*Si els teus problemes tenen solució, per què et preocupes?
Si els teus problemes no tenen solució, per què et preocupes?
(Confuci)*

SUMARI

1. Introducció	1
1.1. Marc en el que s'engloba el projecte	1
1.2. Descripció del problema	1
1.3. Motivacions del projecte	1
1.4. Objectius	2
2. Estat de l'art	3
2.1. Robots a la indústria	4
2.2. Robots cartesianes; posicionadors XY	5
3. Entorn de treball	7
3.1. Espai de treball	7
3.2. Braç robòtic Stäubli RX60	8
3.3. Dispositius accessoris	8
3.3.1. Sensor de força Schunk FTCL	8
3.3.2. Pincas Schunk MEG50-EC	9
3.3.3. Cortines òptiques Leuze	11
3.4. Font d'alimentació	12
3.5. Ordinador de control	13
4. Hardware del robot XY	15
4.1. Unitats lineals	15
4.1.1. Longitud de les unitats lineals	16
4.2. Motors	16
4.2.1. Càlcul de la velocitat nominal	17
4.2.2. Càlcul del parell nominal	17
4.2.3. Selecció dels motors	18
4.3. Encoders i finals de carrera	20
4.3.1. Encoders	20
4.3.2. Finals de carrera	21
4.4. Etapes de potència i controladors	21
4.4.1. Potència subministrada	22
4.4.2. Mètodes de control	22
4.4.3. Selecció dels controladors	22
4.5. Font d'alimentació	23
4.6. Interfície amb el Pc	24
5. Integració de sistemes	25
5.1. Armari de control	25
5.1.1. Línia de potència	26
5.1.2. Línia de connexions	26
5.1.3. Línia de senyals	27
5.1.4. Panel lateral; connexions externes	27
5.2. Sensor de força	28
5.3. Pincas	28

5.4. Barreres òptiques	29
5.5. Robot XY	29
6. Software	31
6.1. Arquitectura	31
6.1.1. Codi reutilitzat	32
6.1.2. <i>CLineal</i>	33
6.1.3. <i>CGeometria</i>	34
6.1.4. <i>CTrajectoria</i>	34
6.1.5. <i>CRobotXY</i>	35
6.1.6. Estructures	36
6.2. Estratègia de control	36
6.2.1. Discretització de la trajectòria	36
6.2.2. Concatenació de segments	37
6.2.3. Càlcul de la longitud del segment	38
6.2.4. Notificació de posició	41
6.2.5. Precàrrega de paràmetres	42
6.2.6. Sincronització dels motors	43
6.2.7. Càlcul dels paràmetres	44
6.3. Entorn visual	47
6.3.1. Funcions de baix nivell	48
6.3.2. Funcions d'alt nivell	49
7. Experiments i resultats	53
7.1. Trajectòria rectilínia única	53
7.2. Trajectòria rectilínia múltiple a igual velocitat	54
7.3. Trajectòria rectilínia múltiple a distintes velocitats	56
7.4. Anàlisi dels errors	57
8. Conclusions i treball futur	59
8.1. Conclusions	59
8.2. Treball futur	60
10. Bibliografia	61
11. Annexes	63
Annex I. Esquema de l'armari de control	63
Annex II. Posicionador XY; notes addicionals	75

Índex de gràfiques i figures

2.1.	Tipus de robots segons la seva topologia	3
2.2.	Espais de treball segons la topologia del robot	4
2.3.	Robots cartesianes	4
2.4.	Robots cilíndrics i SCARA	5
2.5.	Robots articulats	5
2.6.	Unitats lineals comercials	6
3.1.	Espai de treball	7
3.2.	Braç Stäubli RX60	8
3.3.	Dimensions braç Stäubli RX60	8
3.4.	Sensor SCHUNK FTCL	9
3.5.	Diagrama de forces i parells	9
3.6.	Pinça SCHUNK MEG50E	10
3.7.	Controlador MEG C	10
3.8.	Tarja de entrades i sortides digitals NI-6220	10
3.9.	Tarja d'entrades i sortides analògiques NI-6722	10
3.10.	Optoacobladors Phoenix Contact	11
3.11.	Cortines òptiques Leuze	11
3.12.	Relé de seguretat Leuze MSI	12
3.13.	Font d'alimentació Puls 100W, 24V	13
4.1.	Unitats lineals en configuració XY	15
4.2.	Unitats lineals; mesures	16
4.3.	Motor Maxon RE	18
4.4.	Corbes del motorreductor	19
4.5.	Sensor de final de carrera Proxistor IPO002VSF	21
4.6.	Controlador de motors Faulhaber MCDC2805	22
4.7.	Font d'alimentació PULS 100W, 24V	23
4.8.	Cable conversor de USB a RS232	24
5.1.	Integració de sistemes, sistema complert	25
5.2.	Armari de control; línia de potència	26
5.3.	Armari de control; línia de connexions	26
5.4.	Armari de control; línia senyals	27
5.5.	Panel lateral; connexions	27
5.6.	Diagrama de connexió del sensor de força Schunk FTCL	28
5.7.	Diagrama de connexió de les pinces Schunk	28
5.8.	Diagrama de connexió de les barreres òptiques Leuze	29
5.9.	Diagrama de connexió del robot XY	29
6.1.	Diagrama de jerarquia de classes	31
6.2.	Diagrama d'herència de la classe <i>CLineal</i>	33
6.3.	Diagrama d'herència de la classe <i>CGeometria</i>	34
6.4.	Diagrama d'herència de la classe <i>CTrajectoria</i>	35
6.5.	Divisió en segments d'una trajectòria	36
6.6.	Representació gràfica de l'estratègia de segments independents	37
6.7.	Representació gràfica de l'estratègia predictiva	38

6.8.	Representació gràfica de l'estratègia de desacceleració nul·la ...	38
6.9.	Error de posició amb segments de longitud 10 mm	39
6.10.	Error de posició amb segments de longitud 50 mm	40
6.11.	Error de posició amb segments de longitud 100 mm	40
6.12.	Representació gràfica dels instants de notificació de posició	42
6.13.	Velocitats de les unitats lineals desincronitzades	44
6.14.	Velocitat del carro	44
6.15.	Velocitats de les unitats lineals sincronitzades	44
6.16.	Velocitat del carro	44
6.17.	Representació gràfica de diferents temps d'acceleració	45
6.18.	Error de posició per diferents temps d'acceleració	45
6.19.	Pantalla de control de posició i gràfica corresponent	48
6.20.	Pantalla de control de velocitat i gràfica corresponent	48
6.21.	Pantalla de control de trajectòries rectilínies	49
6.22.	Pantalla de control de trajectòries circulars	50
6.23.	Pantalla de control de trajectòries importades de fitxer	50
6.24.	Pantalla de control de trajectòries per dispositiu òptic	51
7.1.	Gràfica experimental de la trajectòria rectilínia	53
7.2.	Gràfica experimental de la Trajectòria rectilínia amb escala ampliada	53
7.3.	Error en l'eix X en la realització d'una trajectòria rectilínia	53
7.4.	Error en l'eix Y en la realització d'una trajectòria rectilínia	54
7.5.	Error de repetibilitat en l'eix X; velocitat positiva	54
7.6.	Error de repetibilitat en l'eix Y; velocitat negativa	55
7.7.	Error de repetibilitat en l'eix X; velocitat positiva	55
7.8.	Error de repetibilitat en l'eix Y; velocitat negativa	55
7.9.	Error de posició en funció dels temps	56
7.10.	Error de temps en funció de la posició	57

1. Introducció

1.1 Marc en el que s'engloba el projecte

El projecte desenvolupat i presentat en aquesta memòria s'ha realitzat en la seva totalitat a l'Institut de Robòtica i Informàtica Industrial (IRI), centre mixte participat per la Universitat Politècnica de Catalunya (UPC) i el Consejo Superior de Investigaciones Científicas (CSIC).

El seu desenvolupament s'inclou dins del Pla de Millora de les Infraestructures (PAI) del CSIC, l'objectiu principal del qual és dotar de recursos nous i millorar els recursos existents dels centres que hi pertanyen.

1.2 Descripció del problema

El Laboratori de Robòtica de l'IRI compta amb una cèl·lula d'assaig consistent en dos braços industrials Stäubli RX60 situats dins d'una estructura metàl·lica.

Per altra banda, existeixen al mercat robots o manipuladors XY, que permeten el posicionament d'un element mòbil a una coordenada XY del pla de treball.

La combinació d'aquests dos dispositius permetria multitud d'aplicacions, entre elles, el seguiment dels braços mitjançant el posicionament manual o automàtic d'una càmera de vídeo.

1.3 Motivacions del projecte

Es tracta de dotar als braços industrials de noves funcionalitats, a la vegada que millorar el sistema existent.

Per això, es dissenyarà i construirà un robot pel posicionament d'un carro mòbil dins del pla XY, situat per sobre de la cota més elevada assolida pels braços.

Per tal d'integrar tots els elements d'alimentació i control, i simplificar les connexions, serà necessari construir un armari de control. Dins d'aquest armari de control es connectarà el robot XY, amb els seus sensors i actuadors, així com els sensors i actuadors accessoris dels braços robòtics.

1.4 Objectius

Així doncs, els objectius d'aquest projecte seran:

1. Dissenyar i construir un robot cartesià, constituït per elements de tipus:

- mecànics, consistents en unitats de desplaçament que permetran el moviment d'una plataforma o carro en els eixos X i Y.
- electrònics: motors per realitzar els moviments, controladors i fonts d'alimentació i potència.
- de control, per tal de realitzar trajectòries i moviments programats, així com el seguiment de diferents dispositius òptics i/o mecànics.

2. Dissenyar i construir l'armari de control en el qual centralitzar tots els elements elèctrics i electrònics, actius o passius, que formen part del sistema.

El sistema complert consisteix no només en el robot XY, si no que consta també d'altres elements accessoris utilitzats en els braços robòtics, com sensors de força, pinces o barreres òptiques, i que s'han d'integrar dintre del mateix armari.

3. Desenvolupar un software específic per la comprovació i l'ús del robot XY, dividir aquesta aplicació de control en dues parts principals. Per un costat, amb una sèrie de funcions bàsiques del robot:

- posicionament del carro en una consigna X i Y.
- assoliment d'una velocitat donada.

Per altra banda, unes aplicacions mes complexes:

- Posicionament del carro mòbil mitjançant la consigna enviada des del PC de control, mitjançant el ratolí o dispositiu similar.
- Importació de coordenades des d'un fitxer, per la realització de moviments enllaçats.

Donat que aquest software serà d'ús comú del laboratori, la seva interfície haurà de ser senzilla i intuïtiva. S'ha d'optar, doncs, per una aplicació en entorn gràfic, malgrat les classes de control de robot es dissenyin de forma que puguin ser utilitzades per aplicacions posteriors.

D'aquesta manera, es dona a l'usuari final la possibilitat de l'ús dels controladors del robot sense necessitat de fer servir cap entorn gràfic pel seu control.

2. Estat de l'art

Entre moltes altres definicions, es podria dir que un robot és aquell mecanisme que, per realitzar els seus moviments, es serveix de motors elèctrics o altres tipus d'actuadors, i que coneix la seva posició en tot moment mitjançant transductors de posicionament. Permet anar a qualsevol punt de l'espai dins del seu camp de treball i utilitza diferents velocitats dins del mateix desplaçament.

El robots que en aquesta memòria ens ocupen són aquells que estan principalment dissenyats per l'ús a la indústria. I dintre d'aquest mercat trobem multitud de classificacions possibles segons la seva funcionalitat, el seu tamany, el seu tipus d'intel·ligència o el seu propòsit. Encara que, òbviament, totes aquestes característiques es troben lligades entre sí.

Si ens fixem en la seva topologia, podem trobar robots de moltes varietats.

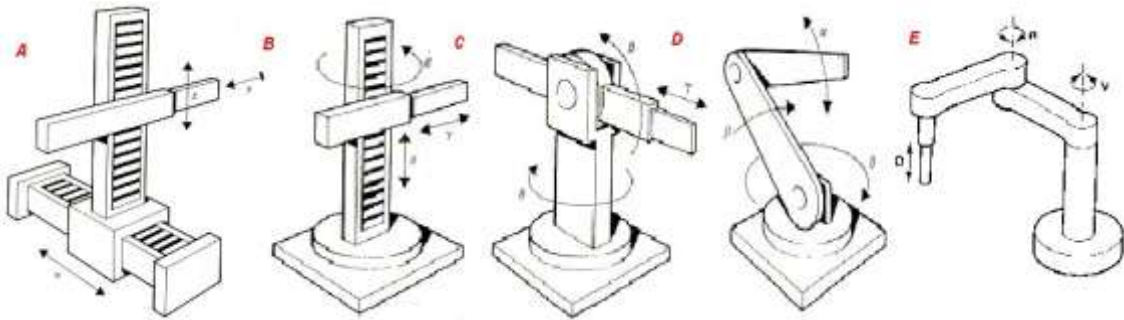


Figura 2.1. Tipus de robots segons la seva topologia

A títol enumeratiu, però no limitatiu, tindriem:

- A. Cartesians: estan constituïts per tres articulacions prismàtiques amb els eixos disposats segons un sistema de coordenades cartesianes. Utilitzen els 3 eixos cartesianes (x , y , z) com a camí pel seu desplaçament.
- B. Cilíndrics: braç constituït per, al menys, una articulació rotativa i una articulació prismàtica, i on els eixos estan disposats segons un sistema de coordenades cilíndriques.
- C. Esfèrics o polars: els seus eixos formen un sistema de coordenades polars.
- D. Articulats: el seu braç té articulacions rotatives. No necessiten els camins definits pels eixos cartesianes per accedir a qualsevol punt de l'espai de treball.
- E. SCARA (Selectively Compliant Arm for Robotic Assembly): robot amb dues articulacions rotatives amb eixos paral·lels, per proporcionar conformitat en un pla donat.

Òbviament, el seu espai de treball dependrà de la seva configuració:

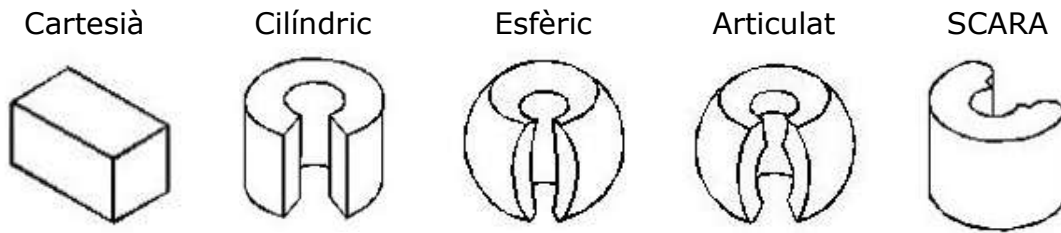


Figura 2.2. Espai de treball segons la topologia del robot

Degut, doncs, a tots aquest tipus existents de robots, serà l'aplicació final la que definirà el tipus de robot adequat. En el nostre cas, i com veurem en apartats posteriors, s'ha decidit optar per la construcció d'un robot de coordenades cartesianes, donat que la principal funció d'aquest ha de ser el posicionament de diversos dispositius dins d'un mateix pla.

2.1 Robots a la indústria

A l'àmbit industrial i domèstic, un robot es fa servir en multitud d'aplicacions, i amb dimensions molt diferents.

Així, veiem robots cartesianes en els plotter DXY 110 de Roland Digital Group; també en màquines fresadores per la fabricació de circuits electrònics com la Protomat del fabricant LPKF, paletitzadors OTAI per línies d'envasar, o en les gegantines grues pont de Jaso Industrial.



LPKF Protomat



Paletitzador OTAI



Grua pont Jaso

Figura 2.3. Robots cartesianes

També els robots cilíndrics i els SCARA, molt semblants entre ells en la seva topologia i en el seu espai de treball, es comercialitzen per aplicacions industrials. Malgrat que, per les seves característiques, no acostumen a assolir les dimensions gegantines que podem veure en els robots cartesianes. Tant Seiko com Factronics o Staubli són diferents fabricants d'aquest tipus de robots.



Seiko RT33



Staubli RS40



Factronics

Figura 2.4. Robots cilíndrics i SCARA

Però els actualment més abundants a la indústria, per la seva funcionalitat, són els robots articulats. Els podem trobar tant en línies de muntatge de cotxes, com en línies d'envasar o paletitzant caixes de llibres. En Fanuc o Staübli tenim dos dels mes importants fabricants a nivell mundial.



Otai KR



Staübli RX60



Fanuc LR

Figura 2.5. Robots articulats

2.2 Robots cartesianes; posicionadors XY

Donades les característiques del sistema a desenvolupar, que veurem posteriorment, s'ha optat per un robot cartesià que treballarà en dos dels eixos de coordenades (x,y).

Actualment, existeixen a nivell comercial multitud de dispositius amb els que el robot cartesià o posicionador XY que es pretén construir guarda força similitud.

Usualment, i donades les diferents necessitats dels usuaris, el fabricant de posicionadors XY es limita a fabricar una ampla gama d'unitats de moviment lineal. Aquestes es poden combinar de diferents maneres per obtenir un robot cartesià de característiques variables.

Així, marques comercials com Hepco, Feyc-rodamientos, Max, MK Hispania o Sinerges Item ofereixen aquest tipus de producte dins del seu catàleg. A més, la majoria d'elles ofereixen també la possibilitat de construir el robot sencer en les seves instal·lacions.

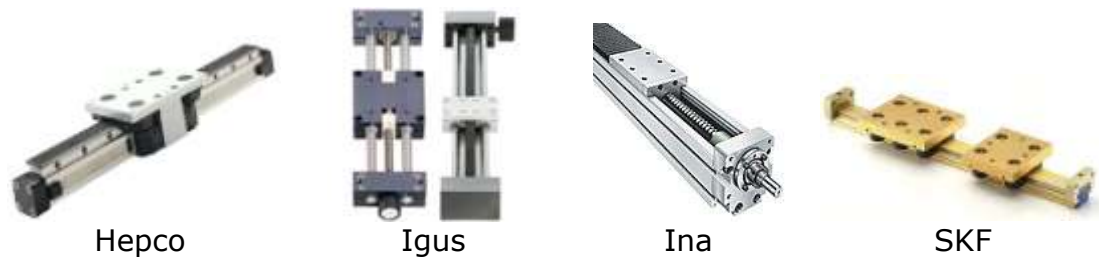


Figura 2.6. Unitats lineals comercials

Com a característica addicional, cal remarcar que aquest dispositiu necessitarà d'una unitat de control que actuï sobre les dues unitats lineals, la programació de la qual pot ser de molt diversa complexitat. Igual que passa en la construcció de les unitats de moviment lineal, algunes de les companyies fabricants la inclouen també dins del seu catàleg. Entre elles es troben Sinerges Item o MK Hispania. Aquestes unitats de control acostumen a estar preparades per la realització de diferents moviments, prèviament programats.

3. Entorn de treball

L'entorn de treball on s'ha desenvolupat el projecte ve definit per l'espai físic a ocupar, donat per l'estructura de perfil·leria d'alumini dins de la qual es troben del braços Stäubli, i el PC de control.

3.1 Espai de treball

Consisteix en una estructura de perfils d'alumini, de tres dimensions i forma rectangular, dintre de la qual se situen dos braços robòtics Stäubli RX60.

El posicionador XY ha de situar-se a la part superior d'aquesta estructura.



Figura 3.1. Espai de treball

Les mesures en planta de l'estructura són 2120 mm de amplada i 1620 mm de profunditat; aquestes dues dimensions defineixen les longituds de les unitats lineals que conformen el robot manipulador. Entre la base a on es troben instal·lats els braços i la part superior tenim una alçada de 1280 mm.

3.2 Braç robòtic Stäubli RX60

El braç robòtic Stäubli RX60 és un robot industrial multifuncional i reprogramable. Es capaç de moure matèries, peces, eines o dispositius especials, segons trajectòries variables, programades per realitzar tasques diverses.



Figura 3.2. Braç Stäubli RX60

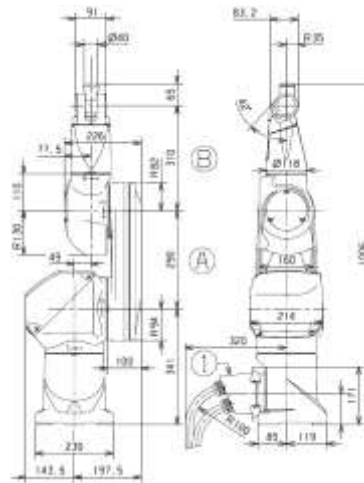


Figura 3.3. Dimensions braç Stäubli RX60

El punt més elevat assolible pels braços és de 1006 mm, quan el braç es trobi en posició completament vertical, més la longitud del sensor o sensors instal·lats al seu extrem.

En quant a velocitats cartesianes, les fulles de característiques ens donen els següents valors:

- Mode manual: 250 mm/s
- Màxima velocitat cartesiana: 1.8 m/s

Altres característiques tècniques, com la càrrega nominal o la seva cinemàtica, no apliquen al nostre projecte.

3.3 Dispositius accessoris.

Cada un dels braços robòtics consta a més de diferents sensors i actuadors, que es pretenen connectar al mateix armari de control que el posicionador XY, amb la finalitat de centralitzar elements elèctrics i electrònics.

3.3.1 Sensor de força Schunk FTCL50

Sensor que es situa a l'extrem del braç, sobre el qual s'instal·len altres dispositius, i dedicat a la detecció de forces efectuades sobre aquests darrers.



Figura 3.4. Sensor SCHUNK FTCL

Les seves característiques més rellevants són:

- Mesura de forces i parells (N i Nm) amb sis graus de llibertat
- Mesura de la desviació (mm i graus) en els sis eixos

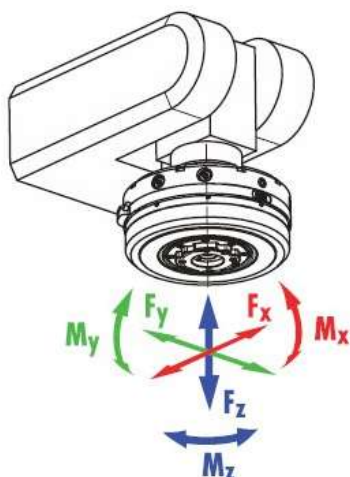


Figura 3.5. Diagrama de forces i parells

	Unitats	FTCL 50
Rang de la desviació		
Traslació X, Y, Z	mm	± 1,0
Rotació	°	± 1,0
Rang dels parells i forces		
Fx, Fy, Fz	N	± 300
Mx, My	Nm	7
Mz	Nm	15
Rang de moviment		
Traslació X, Y, Z	mm	± 1,4
Rotació	°	± 1,4
Masa		
	kg	0,96

I les seves especificacions són:

- Tensió d'alimentació: 10 – 26 Vdc
- Consum: 1.8 W (corrent de 75 mA alimentat a 24 Vdc)
- Comunicació: RS232, 1200 bps a 115200 bps
- Velocitat d'actualització de la mesura: 1 ms

3.3.2 Pinces Schunk MEG50-EC

Accessori pels braços robòtics que s'instal·la sobre el sensor de força indicat en l'apartat anterior.



Figura 3.6. Pinça SCHUNK MEG50EC

El control s'efectua mitjançant el controlador MEG C del mateix fabricant Schunk, i que es disposarà dins de l'armari de control del sistema.

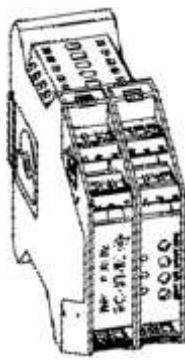


Figura 3.7. Controlador MEG C

Product designation ID NO.	MEG-C 306 005
Voltage supply	24 Vdc
Current input (max.)	1.5 A
Weight	0.3 kg
Protection class	IP30
Nominal temperature range	-10°C to 65°C
Adjustment options	Force adjustment Closing and opening time Stroke adjustment
Interfaces	Digital/analogue I/Os

Aquest controlador consta de diferents entrades i sortides analògiques i digitals per donar les corresponents ordres i rebre dades del dispositiu.

Aquests senyals seran tractats per targetes d'adquisició de dades. Donat el baix cost i els bons resultats que donen, s'han seleccionat dues targetes del fabricant National Instruments, amb les que obtenim les entrades i sortides necessàries per un correcte control de les dues pinces Schunk instal·lades.

- NI-6220



E/S Digitales: 24 DIO · 1 MHz
Entradas analógicas: 16 SE/8 DI · 250 kS/s · 16 bits
Contadores/Temporizadores: 2 · 32 bits · 80 MHz

- NI-6722



E/S Digitales: 8 DIO
Salidas analógicas: 8 · 800 kS/s · 13 bits
Contadores: 2 · 24 bits · 20 MHz

Donat que les entrades i sortides digitals de les targes de National Instruments tenen uns nivells de funcionament TTL de 5 Vdc, i el controlador de les pinces treballa amb senyals digitals de 24 Vdc, haurem d'instal·lar uns optoacobladors per convertir els dos nivells de tensió. Seleccionarem un model del fabricant Phoenix Contact.



Figura 3.10. Optoacobladors Phoenix Contact

Aquest tipus de dispositius són unidireccionals, per lo que en farem servir dos tipus diferents: conversió de 5 Vdc de entrada a 24 Vdc de sortida, i conversió de 24 Vdc de entrada a 5 Vdc de sortida.

D'aquesta forma podrem adaptar els senyals que circulen de les targes de National Instruments cap al controlador de les pinces i les que circulen del controlador a les targes, respectivament.

3.3.3 Cortines òptiques

S'han d'instal·lar unes barreres òptiques Leuze Solid-2, integrades a l'armari de control, que impedeixin el pas de les persones a l'interior de l'estructura quan al menys un dels braços robòtics estigui en funcionament automàtic.



Figura 3.11. Cortines òptiques Leuze

Aquestes barreres òptiques van controlades per un relé de seguretat MSI-SR2F, del mateix fabricant.



Figura 3.12. Relé de seguretat

Safety Relay	MSI-SR2/F
Max. safety category (EN954-1)	4
Stop Category (IEC-, EN60204-1)	0
Operation voltage	24 V AC/DC
Output contacts	2S/1Ö
Making and/or breaking capacity (according to EN 60947-5-1)	AC-15: 230V / 6A DC-13: 24V / 6A
Response time AOPD Type 4 with relay output, AOPD Type 2, safety switch [ms]	20
Reactivation time [ms]	70 manual start / 0,2 s automatic start
Simultaneity monitoring for 2 channels	approx. 20 ms
Operating temperature	-25° to +55°C
Dimensions (HxBxT) [mm]	99 x 22,5 x 111,5 mm

El consum total del conjunt és, segons el fabricant:

- Emisor: 24 V_{dc}; 140 mA
- Receptor: 24 V_{dc}; 45 mA
- Relé de seguretat: 24 V_{dc}; 40 mA

El que suma un total de 225 mA i que equival a una potència de 5.4 W.

3.4 Font d'alimentació

Tots els elements accessoris que conformen la cel·la de braços robòtics poden ser alimentats a 24 Vdc. El consum conjunt de tots ells serà igual a:

$$I_{total} = I_{sensor FTCL} + 2 \cdot I_{pincas} + I_{barreres \text{ òptiques}} \quad [3.1]$$

Fem constar que els corrents nominals dels controladors de les pincas i el relé de seguretat de les barreres òptiques ja contemplen el consum dels dispositius connectats a ells.

Per tant, el corrent total i la potència requerida de tots els elements serà de:

$$I_{total} = 0.075 + 2 \cdot 1.5 + 0.225 = 3.3 \text{ A} \quad [3.2]$$

$$P_{total} = V_{cc} \cdot I_{total} = 24 \cdot 3.3 = 79.2 \text{ W} \quad [3.3]$$

Per tant, la font de 100 W del fabricant Puls és ideal per les nostres necessitats, tant per la potència que subministra com per la possibilitat de ser muntada sobre carril DIN.



Figura 3.13. Font d'alimentació PULS 100W, 24V

3.5 Ordinador de control

L'ordinador de control consisteix en un PC Dell Dimension 8300, amb un processador Pentium IV a 3'2 GHz, 1 Gb de memòria RAM i 120 Gb de disc dur. Té una tarja gràfica NVidia Geforce FX5200 i una tarja de xarxa Ethernet IntelPro/100VE.

El sistema operatiu instal·lat, i sobre el qual s'ha treballat, és un Ubuntu Linux 7.10.

Com a ports de comunicació, consta d'un port RS232, 6 ports USB i una tarja Firewire amb tres ports.

A més, compta amb dues targetes PCI de National Instruments. Per una banda, la NI6722, amb 8 sortides analògiques i 8 entrades/sortides digitals; per una altra, la NI6220, amb 16 entrades analògiques i 24 entrades/sortides digitals.

4. Hardware del robot XY

Els components que formen el sistema es poden dividir en tres parts clarament diferenciades:

- Sistema de moviment de l'element mòbil o carro, a base de tres unitats lineals i els elements complementaris, tal com motors, encoders i finals de carrera.
- Etapes de potència, alimentació i controladors del sistema, agrupats dins de l'armari de control.
- Control per PC.

4.1 Unitats lineals

Per construir el posicionador XY s'utilitzaran tres unitats lineals d'una longitud tal que s'ajustin a la part superior de l'estructura metàl·lica. S'ha escollit el model PDU2 de la firma Hepco pel seu preu competitiu.

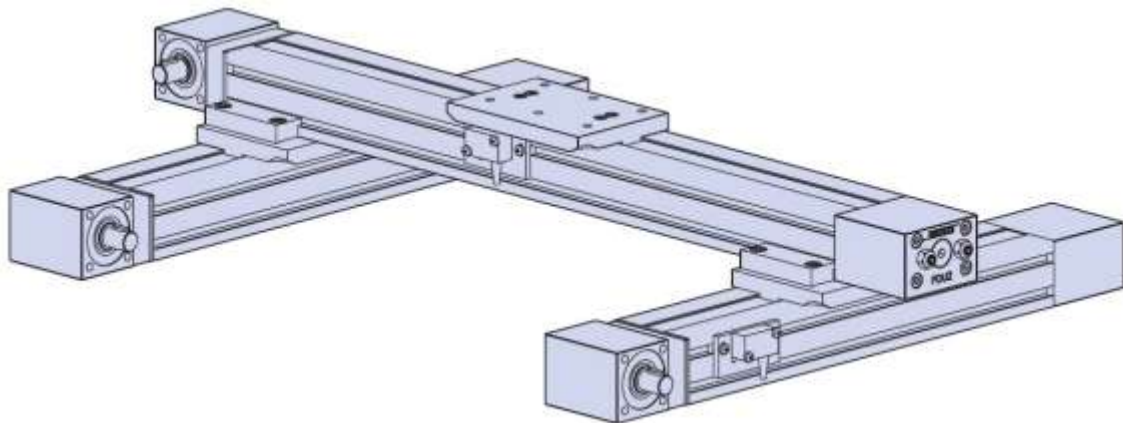


Figura 4.1. Unitats lineals en configuració XY.

Dues d'aquestes unitats lineals es situaran en paral·lel l'una de l'altra, i comptaran amb un sistema d'unió mecànica per tal de que el seu moviment sigui solidari. El seu muntatge serà de forma tal que el carro de cada una d'elles es situï a la part superior.

La tercera unitat lineal es muntarà sobre aquest dos carros, de forma que pugui desplaçar-se al llarg de l'eix Y.

El carro d'aquesta tercera unitat lineal estarà instal·lat en la part inferior, i el seu moviment de translació es farà al llarg de l'eix X.

4.1.1 Longitud

Donat que el fabricant permet seleccionar la longitud de les unitats lineals amb una resolució de 100 mm, agafarem les mesures immediatament superior a les imposades per l'estructura metàl·lica.

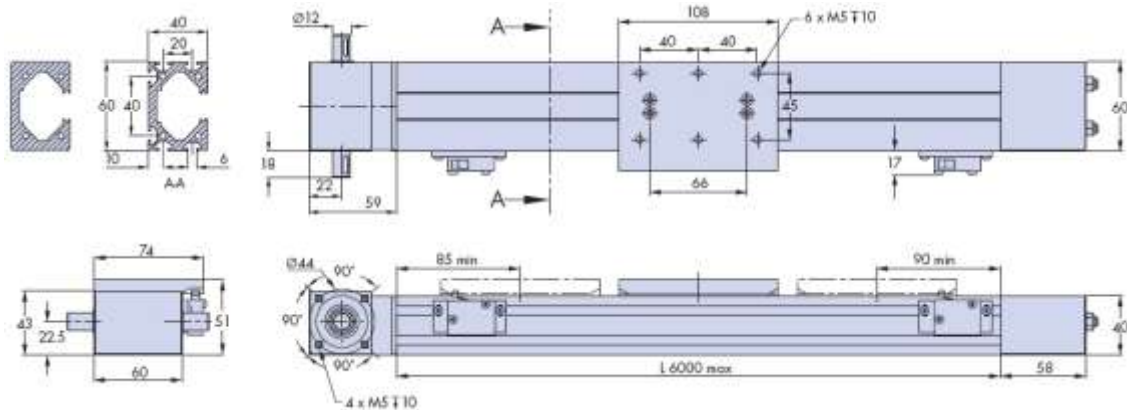


Figura 4.2. Unitats lineals; mesures

Les dues unitats que formen l'eix Y van muntades sobre la part més llarga de l'estructura, de 2120 mm. El model adient serà doncs el de 2200 mm.

L'eix X descansa sobre els carros de les altres dues unitats, que es troben situats a 1620 mm de distància mesurada des dels extrems exteriors. Seleccionarem el model de 1700 mm.

La barra d'unió entre les unitats lineals de l'eix Y, utilitzada per obtenir el moviment solidari d'ambdues, haurà de tenir una longitud igual a la separació entre l'interior d'aquestes, es a dir, 1500 mm.

4.2 Motors

Es necessitaran dos motors pel moviment del sistema, un per cada eix de desplaçament lineal.

Les característiques principals que els definiran seran:

- la velocitat de desplaçament en buit que es vol assolir en cada un dels eixos del posicionador XY, que es pretén que sigui de 0.5 m/s treballant sense càrrega.
- l'acceleració del sistema en cada un dels eixos es vol de 1 m/s²
- el parell nominal que han de tenir per arrastrar una càrrega donada. Aquesta càrrega es preveu d'un màxim de 5 kg.

4.2.1 Càlcul de la velocitat nominal

La velocitat nominal dels motors dependrà de la velocitat lineal desitjada als dos eixos i de la relació de desplaçament lineal per revolució de l'eix.

Segons requeriments del sistema i dades del fabricant, aquests seran:

- velocitat màxima en buit: $v = 0.5 \text{ m/s}$
- relació de desplaçament per revolució de l'eix: $n = 96 \text{ mm/rev.}$

Per tant, tindrem que:

$$\omega = 0.5 \frac{m}{seg} \cdot \frac{60 \text{ seg}}{1 \text{ min}} \cdot \frac{1 \text{ rev}}{0.096 \text{ m}} = 312.5 \text{ rpm} \quad [4.1]$$

4.2.2 Càlcul del parell nominal

Per calcular el parell nominal dels motors hem de tenir en compte que el pitjor dels casos el tindrem pel motor de les unitats lineals de l'eix Y, es a dir, les que es situen paral·lelament entre elles. Això és deu a que, a més del pes de la càrrega, haurà de desplaçar el pes de la unitat lineal que forma l'eix X.

$$\text{Pes Unitat Lineal} = 0'0022 \cdot l + 1.4 = 0.0022 \cdot 1700 + 1.4 = 5.14 \text{ kg} \quad [4.2]$$

$$\text{Pes Càrrega} = 5 \text{ kg} \quad [4.3]$$

$$\text{Pes total} = \text{Pes Càrrega} + \text{Pes Unitat Lineal} = 5.14 + 10 \cong 10 \text{ kg} \quad [4.4]$$

La força necessària pel moviment dependrà de l'acceleració desitjada i la massa, i serà igual a:

$$F = m \cdot a = 10 \cdot 1 = 10 \text{ N} \quad [4.5]$$

I a partir de les dades donades pel fabricant, en visita del comercial a les nostres instal·lacions, i la massa total a moure podem calcular la força necessària per superar el valor de la fricció:

$$\text{Fricció} = 0.08 \cdot m = 0.08 \cdot 10 = 8 \text{ N} \quad [4.6]$$

Ens faltará afegir valors propis de les unitats lineals i que no depenen de la càrrega. Degut a la falta de dades en la fulla d'especificacions, s'han aconseguit a través de consulta telefònica amb el departament tècnic del fabricant.

$$\text{Brake Overlays} = 8 \cdot \text{número d'unitats} = 8 \cdot 2 = 16 \text{ N} \quad [4.7]$$

$$\text{Unió mecànica entre unitats} = 10 \text{ N} \quad [4.8]$$

La força total ens sortirà de sumar els valors trobats a les equacions [4.5] a [4.8] i multiplicar per un coeficient de seguretat, que serà d'un 20%:

$$F = 1.20 \cdot (10 + 8 + 16 + 10) = 1.20 \cdot 44 = 52.8 \text{ N} \quad [4.9]$$

Per últim, el parell motor dependrà del radi de la politja, donada pel fabricant en les fulles de característiques de la unitat lineal:

$$M = F \cdot r = 52.8 \cdot 0.0153 = 0.808 \text{ Nm} = 808 \text{ mNm} \quad [4.10]$$

4.2.3 Selecció dels motors

A partir dels càlculs realitzats, hem trobat que els motors han de complir les següents especificacions:

- Velocitat nominal en buit: $\omega=312.5$ rpm
- Parell nominal a la sortida de l'eix: $M=808$ mNm

El fabricant Maxon ens oferta el model RE35 GB 90W KL 2WE. Hem de seleccionar una reductora de forma que el conjunt motor-reductora compleixi els compromisos de velocitat i força donades per les especificacions del sistema.

Aquests motors tenen les següents característiques:



Figura 4.3. Motor Maxon RE

Motor Data		
Values at nominal voltage		
1	Nominal voltage	V 30.0
2	No load speed	rpm 7220
3	No load current	mA 123
4	Nominal speed	rpm 6420
5	Nominal torque (max. continuous torque)	mNm 92.4
6	Nominal current (max. continuous current)	A 2.50
7	Stall torque	mNm 949
8	Starting current	A 24.4
9	Max. efficiency	% 84
Characteristics		
10	Terminal resistance	Ω 1.23
11	Terminal inductance	mH 0.340
12	Torque constant	mNm / A 38.9
13	Speed constant	rpm / V 246
14	Speed / torque gradient	rpm / mNm 7.76
15	Mechanical time constant	ms 5.50
16	Rotor inertia	gcm ² 67.6

Donat que, com veurem més endavant, l'alimentació del sistema serà a 24 V_{dc} , variarà la velocitat angular del motor respecte de la seva velocitat nominal a 30 V.

- $V_{dc}=24$ V
- Constant velocitat=246 rpm/V

La velocitat angular de cada motor en aquestes condicions serà:

$$\omega = 24 \cdot 246 = 5904 \text{ rpm} \quad [4.11]$$

Per tant, la reductora acoblada al motor haurà de tenir un valor inferior o igual a:

$$N = \frac{\omega_{motor}}{\omega_{eix}} = \frac{5904}{312.5} = 18.89 \quad [4.12]$$

Escollim la reductora immediatament inferior, amb un valor de reducció de 18:1, amb la que obtindrem una seva velocitat de sortida sense càrrega de:

$$\omega_{eix} = \frac{\omega_{motor}}{N} = \frac{5904}{18} = 328 \text{ rpm} \quad [4.13]$$

El fabricant ens dona les corbes característiques del motor amb la reductora seleccionada, i alimentat a 24 Vdc, a partir de les quals podrem trobar el corrent i la velocitat de funcionament amb càrrega:

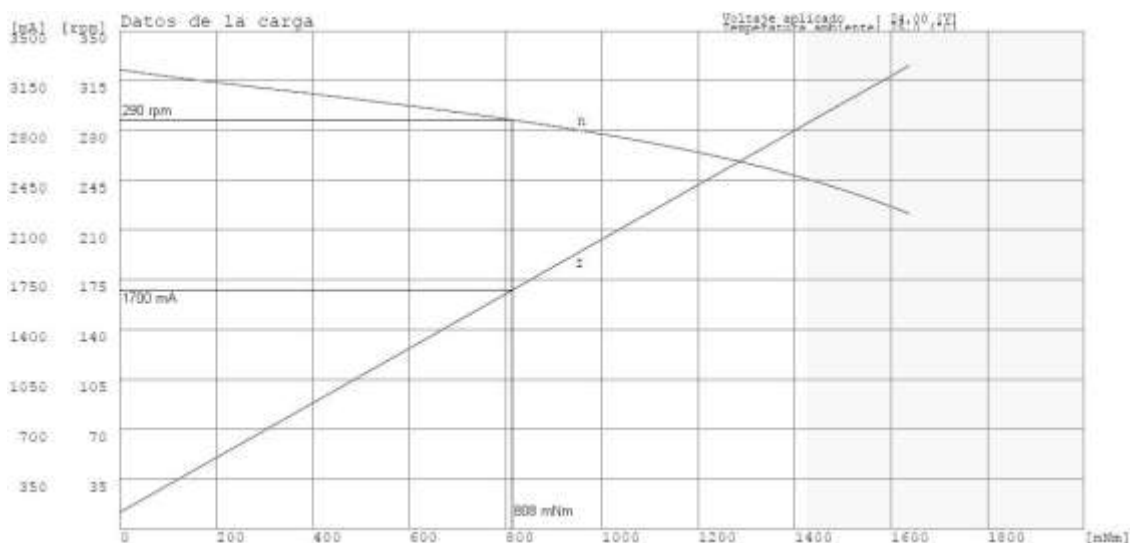


Figura 4.4. Corbes del motorreductor

Així doncs veiem que, amb el parell inicialment fixat de 808 mNm necessari per moure la carrega màxima, la velocitat del motor serà de 290 rpm i el corrent de 1700 mA.

Per tant, segons les dades del motor i la gràfica anterior, la velocitat de cada una de les unitats lineals treballant en buit serà de:

$$v_b = 328 \text{ rpm} \cdot \frac{0.096 \text{ m}}{1 \text{ rev}} \cdot \frac{1 \text{ min}}{60 \text{ seg}} = 0.525 \text{ m/s} \quad [4.14]$$

Valor que compleix amb els requeriments mínims del sistema, pel qual es desitjava una velocitat lineal en buit de 500 mm/s.

El valor de la velocitat a plena càrrega serà:

$$v_c = 290 \text{ rpm} \cdot \frac{0.096 \text{ m}}{1 \text{ rev}} \cdot \frac{1 \text{ min}}{60 \text{ seg}} = 0.464 \text{ m/s} \quad [4.15]$$

A partir d'aquí, i per càlculs posteriors, agafarem una velocitat lineal igual a la mitja entre els dos valors:

$$v = \frac{v_b + v_c}{2} = \frac{525 + 464}{2} = 0.4945 \text{ m/s} \cong 500 \text{ mm/s} \quad [4.16]$$

4.3 Encoders i finals de carrera

Per conèixer en tot moment la posició de les unitats lineals, així com per definir els límits del recorregut, serà necessari l' utilització d'encoders, en el primer cas, i finals de carrera en el segon.

4.3.1 Encoders

Un encoder a cada motor serà suficient per conèixer, mitjançant la conversió adequada, la posició del carro en les unitats lineals d'ambdós eixos.

Per evitar al màxim errors produïts per una baixa resolució, i donat que disminuir aquesta es tan senzill com seleccionar un encoder amb el número de polsos per volta (*ppv*) adequats, farem que el seu valor sigui de 0.01 mm.

El número de comptes per volta de l'encoder serà de:

$$R = \frac{1 \text{ pols}}{0.01 \text{ mm}} \cdot \frac{96 \text{ mm}}{1 \text{ rev}_{eix}} \cdot \frac{1 \text{ rev}_{eix}}{18 \text{ rev}_{motor}} = 533 \text{ ppv} \quad [4.17]$$

Maxon, el mateix fabricant dels motors, ens ofereix dins dels seus productes el model HEDS5540. Els valors disponibles en aquest model d'encoder més propers al desitjat són el de 500 ppv i el de 2000 ppv. Comprovem si ens ajustem al valor de la resolució amb el més petit dels dos:

$$R = \frac{1 \text{ rev}_{motor}}{500 \text{ comptes}} \cdot \frac{1 \text{ rev}_{eix}}{18 \text{ rev}_{motor}} \cdot \frac{96 \text{ mm}}{1 \text{ rev}_{eix}} = 0.0107 \text{ mm} \cong 0.01 \text{ mm} \quad [4.18]$$

Observem que la resolució és lleugerament superior a la especificada inicialment en els nostres requeriments, però ja que aquesta havia estat sobredimensionada, i la diferència és notablement petita, donem aquest encoder per bo.

4.3.2 Finals de carrera

Per tal d'inicialitzar les unitats lineals, i com a sistema de protecció, serà necessari afegir interruptors de final de carrera. Inclourem dos per cada eix, un d'ells a cada extrem.

Hepco, el mateix fabricant de les unitats lineals ens ofereix la solució dins del seu catàleg, amb els interruptors inductius Proxistor IPO002VSF.

Aquest interruptors permeten una alimentació de 5 a 30 V_{dc} i són configurables en quant a polaritat, NPN o PNP, i funcionament, normalment obert (NO) o normalment tancat (NC), la qual cosa els fa ideals per la nostra aplicació.

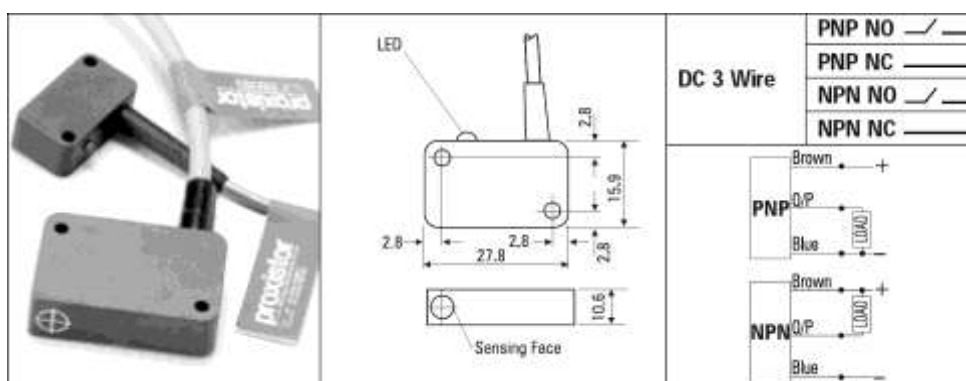


Figura 4.5. Proxistor IPO002VSF

4.4 Etapes de potència i controladors

Malgrat el control últim dels motors, i per tant, de les unitats lineals, es farà des d'un PC, serà necessari utilitzar etapes de potència i controladors. Actualment, existeixen al mercat multitud de dispositius amb aquestes dues funcionalitats integrades, tot i que molts d'ells que acostumen a ser útils únicament per aplicacions de baixa potència com la nostra.

Aquests controladors s'encarregaran de interpretar les ordres donades per l'ordinador i donar la potència necessària als motors per moure'ls.

Per tant, les seves característiques hauran de complir les especificacions que volem que tingui el nostre sistema, especialment en el que fa referència a potència subministrada i mètodes de control.

4.4.1 Potència subministrada

La potència que ha de ser capaç de donar cada controlador serà la necessària per alimentar el motor que hagi de controlar.

Tal com s'ha calculat en l'apartat dedicat al dimensionament dels motors, el corrent del motor a càrrega màxima serà de 1700 mA. Alimentat a 24 Vdc, necessitarem uns controladors capaços de donar una potència màxima de 40.8 W.

4.4.2 Mètodes de control

Per obtenir el funcionament desitjat del sistema, el controlador ha de ser capaç de fer un control per posició i per velocitat, el que significa que ha de tenir entrades per la connexió d'un encoder.

A més, tant per conèixer la posició de repòs o inicial, com per l'aturada al assolir la posició final als dos extrems de la unitat lineal, haurà de tenir entrades per connectar els dos interruptors de final de carrera.

4.4.3 Selecció dels controladors

Segons les dades

- Tensió d'alimentació del motor: $V_{\text{motor}}=24 \text{ V}$
- Corrent d'alimentació del motor: $I_{\text{motor}}=1.7 \text{ A}$
- Potència requerida: $P_{\text{motor}}=40.8 \text{ W}$

A més ha de constar d'entrades suficients pels interruptors de final de carrera i encoder. I el dispositiu que existeix al mercat i que més s'adapta a les nostres necessitats és el fabricat per Faulhaber, model MCDC2805.



Figura 4.6. Controlador Faulhaber MCDC2805

General Specifications			
Supply Voltage	U_s	12 ... 28	V DC
PWM Switching Frequency	f_{PWM}	62,5	kHz
Efficiency	η	95	%
Max. Continuous Current	I_{cont}	5	A
Max. Peak Current	I_{max}	10	A
Current Consumption in the Electronics	I_e	0,06	A
Velocity Range		10 ... 20000	rpm
Output Voltage for External Use	VCC	5	V DC
-Max. Load Current	I_{CC}	60	mA
Input No. 1)	Input Resistance	18	k Ω
Command Analog Velocity	Voltage Signal	± 10	V
	Slope of the Curve	10000	rpm/V
Command Digital Velocity	PWM Signal	low 0...0.5 / high 4...30	V
	Frequency Range	100 ... 2000	Hz
	Pulse Duty Ratio 50%	0	rpm
	Pulse Duty Ratio <50%	left turning	
	Pulse Duty Ratio >50%	right turning	
Step Frequency	f_{step}	100	kHz
Fault Output (Input No. 2)	Open collector	max. U_s / 30mA	
	No Error	Switched to GND	
	Programmed as an input	low 0...0.5 / high 4... U_s	V
Input No. 3 / No. 4 / No. 5	Logic Level	low 0...0.5 / high 4...30	V
Encoder Input	Logic Level	low 0...0.5 / high 4...30	V
	Max. Frequency	200	kHz
Port	RS-232	9600 (1200,2400,4800,19200)	Baud
Memory for Programs	Serial EEPROM	7936	Bytes

Cada controlador pot alimentar un motor a 24 Vdc i un corrent màxim de 5 A. Per tant, cada un d'ells es capaç de subministrar una potència màxima de 120 W, sobradament suficient pels nostres requeriments.

4.5 Font d'alimentació

Per seleccionar la font d'alimentació adequada al sistema partim de la base de que aquesta ha de donar una tensió de 24 V_{dc}. La potència que ha de ser capaç de donar la podem trobar a partir del corrent total que necessiten els diferents dispositius que es connecten a ella.

Al tenir dues unitats lineals elèctricament idèntiques, podem calcular el consum de cada una d'elles, amb tots els seus accessoris:

$$I_{unitat} = I_{motor} + I_{encoder} + 2I_{interruptor} + I_{control} \quad [4.19]$$

$$I_{unitat} = 1.7 + 0.02 + 0.25 + 0.06 = 2.03 \text{ A} \quad [4.20]$$

$$I_{total} = 2I_{unitat} = 2 \cdot 2.03 = 4.06 \text{ A} \quad [4.21]$$

La potència requerida total serà igual a:

$$P_{total} = V_{cc} \cdot I_{total} = 24 \cdot 4.06 = 97.44 \text{ W} \quad [4.22]$$

Escollim la font d'alimentació de la casa Puls de 100W, que ens dóna la potència necessària a més de poder-se muntar en carril DIN, el que la fa encara més adequada per les nostres necessitats.



Figura 4.7. Font d'alimentació PULS 100W, 24V

4.6 Interfície amb el PC

La comunicació entre els controladors i el PC es farà mitjançant el protocol RS232. I donat que els ports més habituals en els computadors actuals són els ports USB, utilitzarem un convertidor USB a RS232 per poder tenir connectivitat entre tots els dispositius.



Figura 4.8. Cable convertidor de USB a RS232

5. Integració de sistemes

La integració de sistemes té com a finalitat la creació d'estructures compostes de dispositius de diferents tipus i procedències que operin entre ells de forma transparent. I aquest serà l'objectiu en el disseny del nostre sistema, en el que han d'interconnectar-se sensors, actuadors, controladors i ordenadors.

Com a nexce d'unió entre tots ells s'ha fabricat un armari de control, a on es reparteixen les etapes de potència i de comunicació. Segons els dispositius referenciats en capítols anteriors i aquest nou armari de control, el sistema s'esquematisa de la següent manera:

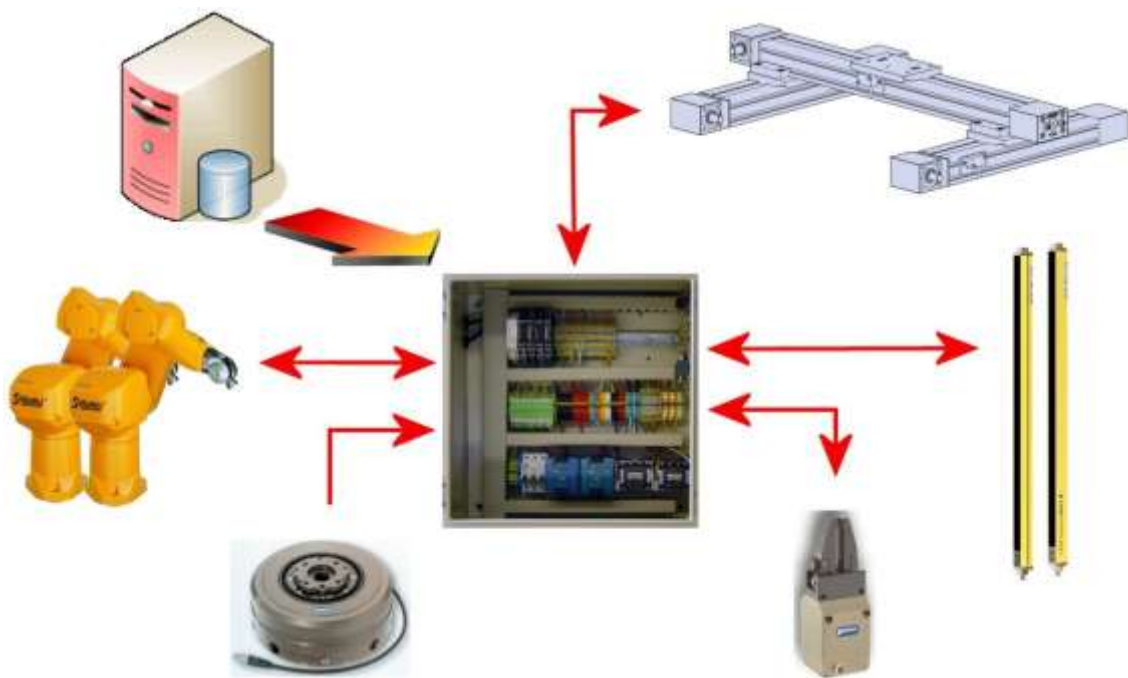


Figura 5.1. Integració de sistemes, sistema complert

5.1 Armari de control

L'armari de control és el nexce d'unió entre tots els dispositius del sistema. En ell és fa la separació entre les etapes de potència i alimentació, encarregades de donar tensió a cada un dels elements, i la etapa de comunicació o dades, encarregada de la comunicació entre ells.

Per fer-ho s'ha subdividit en tres línies principals, situant l'entrada de corrent altern i fonts d'alimentació el més allunyats possible dels senyals de control, amb la finalitat d'evitar interferències.

5.1.1 Línia de potència

La part inferior de l'armari conté l'interruptor de connexió i desconnexió de tot el sistema, les fonts d'alimentació necessàries per alimentar tots els dispositius i els dos controladors dels motors del robot XY.



Figura 5.2. Armari de control; línia de potència

D'esquerra a dreta tenim:

- Entrada de corrent alterna
- Interruptor de connexió
- Fonts d'alimentació 24 Vdc, 100 W
- Controladors de motor Faulhaber MDC2805

S'observa la inclusió de les dues fonts d'alimentació, una corresponent al robot XY, dimensionada a partir dels càlculs realitzats en el capítol anterior; i una altra pels dispositius accessoris ressenyats en el capítol 3.

L'utilització de dues fonts independents ens permet aïllar sistemes, de manera que els motors i els dispositius auxiliars del robot XY (controladors, finals de carrera i encoders) no interfereixin en el funcionament dels altres components. Aquesta configuració és especialment interessant quan es treballa amb motors elèctrics amb moviments intermitents, ja que la possibilitat de interferències i soroll és més elevada.

5.1.2 Línia de connexions

A la part mitja, en aquest carril es troben els connectors utilitzats per transmetre senyals dels connectors d'entrada als de sortida, i els optoacobladors que adapten els senyals per ser compatibles entre si.

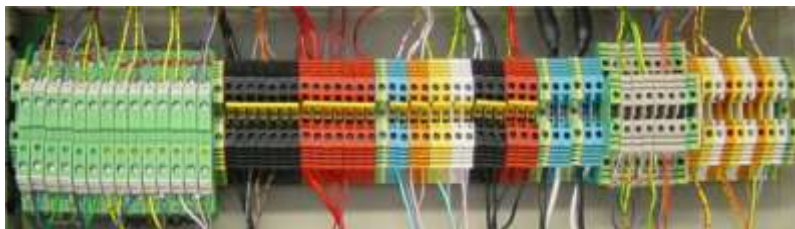


Figura 5.3. Armari de control; línia de connexions

Veiem els optoacobladors de color verd, a la part dreta del carril. La resta de borneres són connexions directes per distribuir les alimentacions (en negre i en vermell) i els senyals.

5.1.1 Línia de senyals

A la part superior situem els dispositius més sensibles a les possibles interferències, per , el més allunyat de la principal font de soroll que tenim que és l'entrada de corrent alterna.



Figura 5.4. Armari de control; línia senyals

D'esquerra a dreta, tenim el relé d'emergència que desconnecta el circuit quan es tallen les barreres òptiques, els dos controladors de les pinces i els senyals provinents de les targes d'entrades i sortides instal·lades al PC de control.

5.1.3 Panel lateral; connexions externes

A un lateral es situen els connectors dels diferents dispositius, alguns connectors sense ús en previsió de properes aproximacions i els comandaments de les barreres fotoelèctriques.



Figura 5.5. Panel lateral; connexions

1. No connectat
2. No connectat
3. No connectat
4. No connectat
5. Pinça A
6. Pinça B
7. Potenciòmetres
8. No connectat
9. Sensor de força FTC (entrada)
10. Sensor de força FTC (sortida)
11. Controlador unitat X
- 12.. Controlador unitat Y
13. Motor unitat X
14. Motor unitat Y
15. Finals de carrera unitat X
16. Finals de carrera unitat Y
17. Barrera òptica emissora
18. Barrera òptica receptora
19. Sortida A relé barrera òptica
20. Sortida B relé barrera òptica

La sortida duplicada del relé de les barreres òptiques (connectors 19 i 20) es deguda a que són dos braços els que s'han de desconectar en cas de paro d'emergència.

Els connectors 1 a 4 i els connectors DBi estan sense utilitzar i s'han instal·lat en previsió de posteriors ampliacions del sistema.

El selector d'anul·lació de barreres òptiques permet anul·lar el dispositiu de seguretat, ja que el fet de realitzar treballs d'investigació amb els braços robòtics, requereix aquesta possibilitat. Mentre que el botó de *resetejar* les barreres desactiva el seu relé, posant el sistema en condicions de funcionament.

5.2 Sensor de força

És la connexió més senzilla de tots els dispositius, ja que únicament necessita alimentar-se a 24 Vdc i la comunicació amb l'ordinador via RS232.



Figura 5.6. Diagrama de connexió del sensor de força Schunk FTCL

5.3 Pinces

Bastant complex, tenim per un costat l'alimentació dels dos controladors. Aquests controladors es connecten directament cada un d'ells a la seva pinça corresponent.

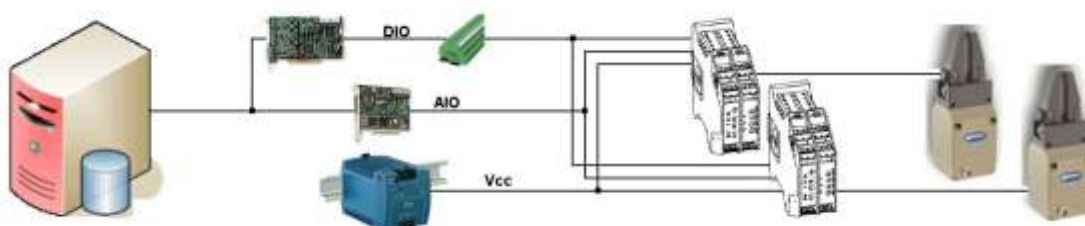


Figura 5.7. Diagrama de connexió de les pinces Schunk

Els senyals de control els dona l'ordinador mitjançant les dues targetes PCI integrades, amb sortides i entrades analògiques i digitals. I donat que les sortides digitals del PC són TTL, i per tant amb nivells de 5 Vdc, i les dels controladors de les pinces tenen nivells de 24 Vdc, s'ha hagut d'afegir uns optoacobladors que transformes ambdós nivells de senyals per fer-los compatibles entre ells.

5.4 Barreres òptiques

Les barreres òptiques es connecten al seu relé de seguretat, el qual ha d'estar alimentat a 24 Vdc. Aquest relé talla el circuit dels braços Stäubli quan el feix entre la barrera emissora i la receptora s'interrompeix en algun punt.



Figura 5.8. Diagrama de connexió de les barreres òptiques Leuze

Es l'únic sistema independent que no es troba connectat al PC.

5.5 Robot XY

Ja vist en l'anterior capítol, les connexions del robot XY són les que més complexitat tenen.

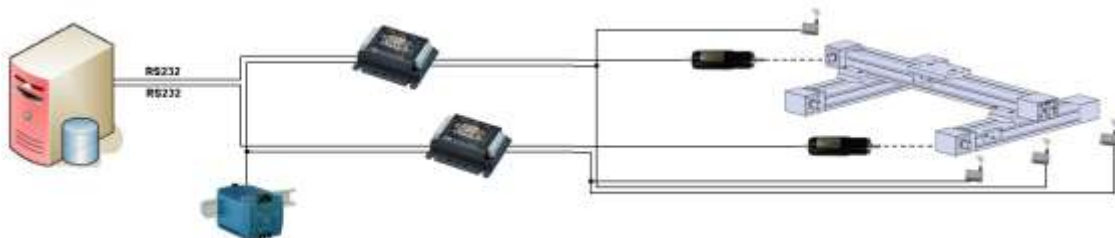


Figura 5.9. Diagrama de connexió del robot XY

Cada un dels controladors, alimentats per una font de 24 Vdc independent de la resta d'alimentacions del sistema, es comunica amb el PC via RS232.

A cada controlador es connecten els finals de carrera, el motor i el encoder integrat corresponents a una unitat lineal.

6. Software

Es desenvoluparà un software específic per l'utilització del manipulador que constarà de tres parts ben diferenciades:

- Software de baix nivell que ens ofereixi la comunicació i el control entre el Pc i els controladors dels motors.
- Software d'alt nivell, en entorn visual, que permeti l'utilització amigable i intuïtiva, per part de l'usuari, del software de baix nivell.
- Aplicacions predefinides, realitzant-se la programació de tasques com la realització per part del manipulador de trajectòries prefixades, el seguiment del dispositiu òptic del Pc o la importació de dades des de fitxer.

S'ha utilitzat un ordinador de control amb un sistema operatiu Linux Ubuntu, i el C++ com a llenguatge de programació sota entorn KDevelop. La interfície gràfica ha estat desenvolupada en Qt3 i afegida al codi del programa.

6.1 Arquitectura

La forma més lògica d'explicar l'arquitectura de l'aplicació desenvolupada és prenent com a unitat bàsica cada una de les classes emprades.

Per un costat, i aprofitant la possibilitat de la reutilització de codi, s'ha partit d'unes classes existents dedicades exclusivament al control dels motors. A partir d'aquestes, s'ha desenvolupat classes adjacents per tal de controlar les unitats lineals i, com a nexa d'unió entre elles, una classe central de control del robot.

S'ha dissenyat altres classes complementàries per tal de implementar el seguiment de trajectòries i el seguiment del braç robòtic, així com la classe de l'entorn gràfic i les que aquesta hereta.

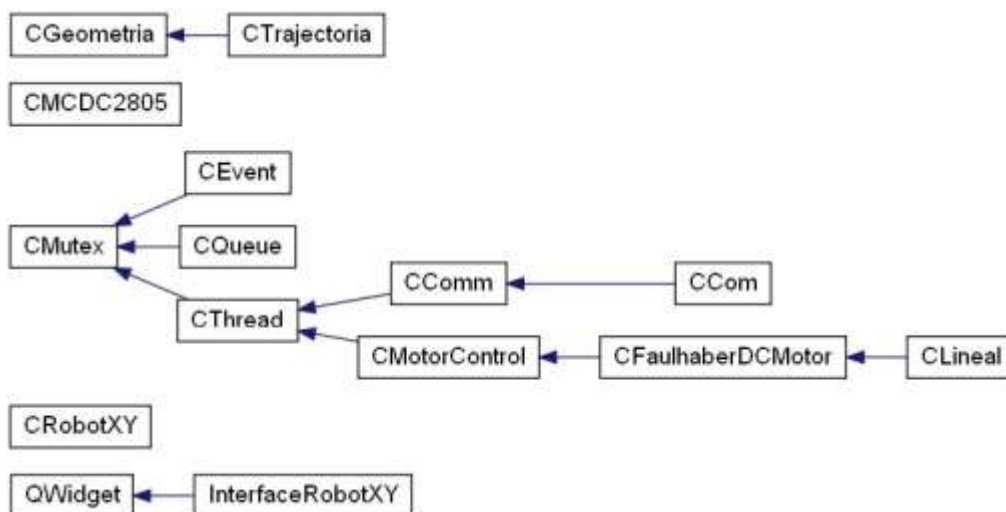


Figura 6.1. Diagrama de jerarquia de classes

6.1.1 Codi reutilitzat

El codi reutilitzat està format per unes classes ja implementades, i dedicades al control d'un altre tipus de motors Faulhaber, però amb els mateixos controladors.

- *CCom*: port sèrie. Aquesta classe hereta tota la funcionalitat d'un dispositiu de comunicacions de la classe *CComm*, i només re-implemента aquelles funcions que són específiques del port sèrie (*UserHandleEvent* per gestionar els events del port sèrie, *HardWrite* per escriure al port sèrie, *HardRead* per llegir del port sèrie i *HardGetNumData* per saber el número de dades pendents).
- *CComm*: dispositiu genèric de comunicacions. Aquesta classe implementa tota la funcionalitat de baix nivell d'accés a un dispositiu de comunicacions d'un ordinador. Per millorar l'eficiència en la transmissió i recepció de dades, l'objecte crea un *thread* intern i completament transparent que s'encarrega de gestionar tots els events del dispositiu (error, recepció i final de transmissió). Aquesta classe permet l'accés de múltiples *threads* al dispositiu físic, i per tant implementa mètodes d'exclusió mútua per evitar problemes de memòria compartida i de bloqueig.
- *CEvent*: event lògic de senyalització. Aquesta classe implementa la funcionalitat d'un event lògic per a Linux. Aquests tipus d'events es poden utilitzar per sincronitzar diferents processos o bé per indicar el compliment d'una certa condició. Per tal de realitzar esperes coordinades de dos o més events i també per tal de combinar esperes d'events d'altre naturalesa amb events lògics, aquesta classe proporciona dos mecanismes addicionals.
- *CFaulhaberDCMotor*: control d'un motor de contínua amb un controlador Faulhaber. Aquesta classe permet realitzar el control d'un motor de contínua ja que hereta tota la funcionalitat de control de motors de la classe *CMotorControl* i redefineix el comportament d'algunes de les seves funcions per que s'ajustin a les característiques dels motors que s'utilitzin.
- *CMCDC2805*: controlador del motor de contínua. Aquesta classe implementa tota la funcionalitat dels controladora MCDC2805 de Faulhaber. Està dissenyada per poder enviar comandes i rebre les respostes sense haver de preocupar-se pel format d'aquestes.
- *CMotorControl*: interfície bàsica de control d'un motor. Aquesta classe implementa la interfície bàsica de control de baix nivell de qualsevol motor. Aquesta classe ha de ser la classe base de qualsevol classe de més alt nivell que s'encarregui del control d'un motor.
- *CMutex*: mecanisme d'exclusió mútua. La classe *CMutex* implementa la funcionalitat necessària per al mètode més senzill de memòria compartida (exclusió mútua). En classes on existeixi la possibilitat

que diversos processos accedeixin simultàniament a uns mateixos atributs, s'ha d'utilitzar una classe d'aquest tipus.

- *CQueue*: cua circular. Aquesta classe implementa la funcionalitat d'una cua circular de longitud ajustable per l'usuari. La longitud de la cua només es pot determinar en el moment de crear l'objecte. Implementa mètodes d'exclusió mútua per permetre l'accés a múltiples *threads* sense produir ninguna condició d'excepció ni tampoc conflictes de memòria compartida.
- *CThread*: fil d'execució paral·lel. Aquesta classe implementa tota la funcionalitat necessària per poder manipular múltiples fils d'execució de forma senzilla. Quan sigui necessari que diverses funcions o funcionalitats d'una classe s'executin de forma concurrent serà necessari que aquesta classe derivi de *CThread*.
- *TComConfig*: estructura de configuració del port sèrie.
- *TThread*: estructura amb la informació de cada un dels *threads* que gestiona la classe.

6.1.2 *CLineal*

Aquesta classe permet realitzar el control d'una unitat lineal controlada per controladors Faulhaber MCDC2805, ja que hereta tota la funcionalitat de control de motors de la classe *CMotorControl* i *CFaulhaberDCMotor*.

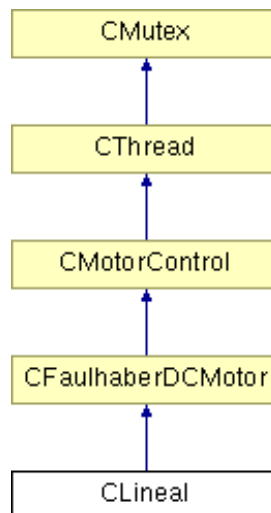


Figura 6.2. Diagrama d'herència de la classe *CLineal*

Redefineix el comportament d'algunes de les funcions de les classes de les que hereta per que s'ajustin a les noves característiques de les unitats lineals, com pot ser la conversió d'unitats angulars a unitats lineals.

La seva utilització és mitjançant la crida a les seves rutines. Internament es fa la conversió en graus, radians o comptes d'encoder, segons es necessiti, per tal de cridar a les rutines de les classes inferiors.

Implementa els mètodes *Init()* i *Load()* modificats de la classe *CFaulhaberDCMotor* per adaptar el codi corresponents als controladors de contínua a les unitats lineals. Les rutines *Start()*, *Notify()*, *WaitFor()*, *Abort()* i *Close()* les hereta directament de les classes inferiors, ja que no necessiten ser modificades.

Implementa també rutines per fixar i conèixer el desplaçament, velocitat i acceleració màxims, així com funcions per transformar les unitats lineals (mil·límetres) en unitats angulars (graus o comptes d'encoder) i viceversa, i la crida a rutines de les classes de més baix nivell.

6.1.3 *CGeometria*

Aquesta classe implementa un seguit de figures geomètriques. Permet, mitjançant la definició prèvia de la figura, trobar un punt corresponent a aquesta en funció de la distància al origen.

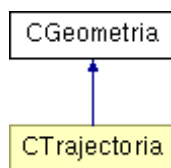


Figura 6.3. Diagrama d'herència de la classe *CGeometria*

Per utilitzar-la, s'ha de definir una figura geomètrica utilitzant la rutina *SetFigura()*, a on es defineix el tipus, l'origen i, segons s'escaigui, el costats o el radi.

Cridant a la rutina *Punt()*, tornarà les coordenades del punt desitjat.

També conté funcions públiques de conversió d'unitats polars, de graus a radians i viceversa.

S'ha desenvolupat les geometries corresponents a una recta, un rectangle i un arc, ja que a partir d'aquestes, o del seu mètode de programació, es molt senzill desenvolupar geometries complementàries.

6.1.4 *CTrajectoria*

Aquesta classe permet definir una trajectòria a seguir definint inicialment la figura geomètrica que seguirà, així com el seu punt d'origen, el seu punt final i la velocitat a la que s'ha d'efectuar.

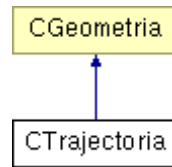


Figura 6.4. Diagrama d'herència de la classe CTrajectoria

Hereta de la classe *CGeometria*, amb lo que pot tornar punts de qualsevol figura geomètrica que es trobi definida en aquesta darrera.

Per utilitzar-la, s'ha de definir la figura a seguir mitjançant la rutina *SetFigura()*, heretada de *CGeometria*. A continuació s'ha de definir la trajectòria a seguir mitjançant la rutina *SetTrajectoria()*.

Una vegada s'hagin definit figura i trajectòria, serà suficient amb la crida a la rutina *SeguentPunt()* per rebre punts consecutius. Si es desitja conèixer la posició d'un punt situat dins de la trajectòria i al qual s'arribarà en un temps donat a la velocitat configurada, s'haurà de cridar la rutina *Punt()*.

6.1.5 CRobotXY

Aquesta classe permet realitzar el control del robot planar de dos graus de llibertat, format per dues unitats lineals. No hereta de cap classe, per el que no existeix diagrama d'herència associat.

Instancia dues unitats lineals de la classe *CLineal*, amb el que permet la crida a les rutines públiques d'aquesta classe, suficients per realitzar el control total del robot.

Es dedica principalment a la traducció de les rutines en les que intervenen coordenades cartesianes (X,Y) a la crida a les rutines corresponents a cada un dels eixos o, en aquest cas, unitats lineals.

A la vegada, conté una instància de la classe *CGeometria*, per tal de poder realitzar crides a les rutines d'aquesta classe en el control de trajectòries.

Afegeix una rutina de control de trajectòries. Mitjançant les instàncies a cada una de les unitats lineals i a la classe *CGeometria*, permet conèixer i controlar la posició de les unitats lineals en un instant de temps donat per la trajectòria a seguir.

La rutina encarregada de realitzar la trajectòria es crida mitjançant un *Thread*, de forma que aquesta no prengui el control del programa mentre s'estigui realitzant. Per conèixer l'estat del robot, i saber si s'està movent o està llest per rebre ordres, s'ha de consultar la variable *State*.

6.1.6 Estructures

Es defineixen dues noves estructures:

- *TComponents*: estructura formada per dos valors decimals, utilitzada per poder treballar amb parells de números. Aquest parell de números correspon a les dues components, en X i en Y, de qualsevol valor de magnituds del pla en el que treballa el robot.
- *TFigura*: guarda la configuració de la figura.

```
typedef struct TFigura{
    int tipus;
    TComponents origen;
    TComponents costats;
    float radi;
}TFigura;
```

6.2 Estratègia de control

Donades les especials característiques del sistema, es important desenvolupar unes bones estratègies de control. Això significa:

- Una discretització de la trajectòria en els segments més petits possibles.
- Una estratègia de canvi de velocitat en la concatenació de segments consecutius.
- La notificació de posició quan aquesta sigui assolida.
- La precàrrega de les noves consignes
- El càlcul de les acceleracions i velocitats de cada motor per aconseguir el sincronisme entre les unitats lineals.

6.2.1 Discretització de la trajectòria

Per tal de que el robot planar segueixi la trajectòria donada amb la major fiabilitat possible, s'ha optat per realitzar un control per posició. Aquest mètode es basa en la divisió de la trajectòria en els segments lineals més petits possible, i ens assegura que sempre arribem a la posició desitjada.

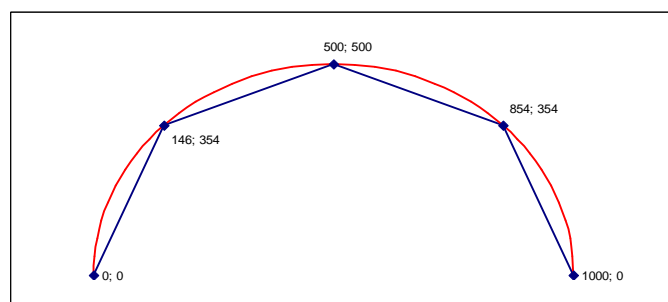


Figura 6.5. Divisió en segments d'una trajectòria

Cada segment tindrà un punt d'inici, que serà la posició del robot en l'instant inicial, i un punt final o objectiu. La velocitat a la que haurà de realitzar aquest segment es coneix a partir de la velocitat desitjada per la trajectòria sencera.

La longitud del cada segment vindrà donada per les característiques físiques del sistema. Per un costat tenim que quan més llarg sigui cada segment, més elevat pot ser el error al apartar-se de la seva trajectòria, quan aquesta no sigui lineal. Per un altre, donat que per cada segment han de ser enviades les noves consignes als motors, el temps que es necessiti per recorre'ls no pot ser menor que el temps ocupat en la comunicació.

En apartats posteriors calculem aquesta longitud mínima de segment per que s'ajusti als nostres requeriments.

6.2.2 Concatenació de segments

El canvi de velocitats i acceleracions en la concatenació de segments és crític pel bon funcionament del sistema, ja que una mala resolució pot provocar parades, oscil·lacions o trajectòries indesitjades.

Analitzarem tres possibles estratègies de les múltiples possibles.

6.2.2.1 Segments independents

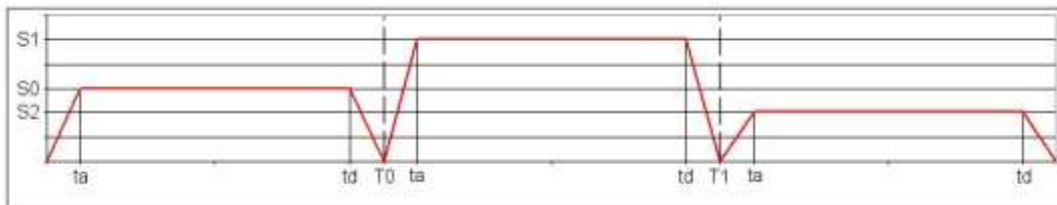


Figura 6.6. Representació gràfica de l'estratègia de segments independents

Es pren cada segment independentment i es calculen la seva velocitat i acceleració com si no formés part de cap trajectòria.

Assignant una part del temps total a l'acceleració i la desacceleració, i coneixent la velocitat a la que es vols anar, el càlcul del paràmetres es realitza de forma senzilla.

Aquesta estratègia presenta un problema important, ja que al realitzar els motors una parada al finalitzar cada segment, la trajectòria es realitza amb moltes estrebades, en lloc de ser contínua i fluida com seria desitjable.

6.2.2.2 Estratègia predictiva

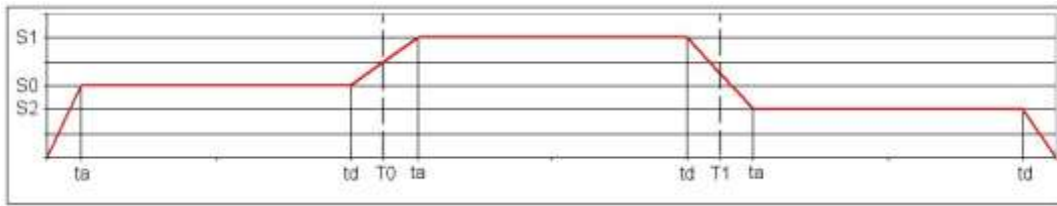


Figura 6.7. Representació gràfica de l'estratègia predictiva

L'acceleració de cada segment es calcula a partir de la posició a on el motor començarà a desaccelerar, per tal d'arribar a la velocitat nominal del següent segment en el temps previst.

Malgrat aquesta estratègia redunda en un millor comportament del sistema, la relativa complexitat en el càlcul dels paràmetres fa que tampoc aquesta sigui la més adient.

6.2.2.3 Desacceleració nul·la

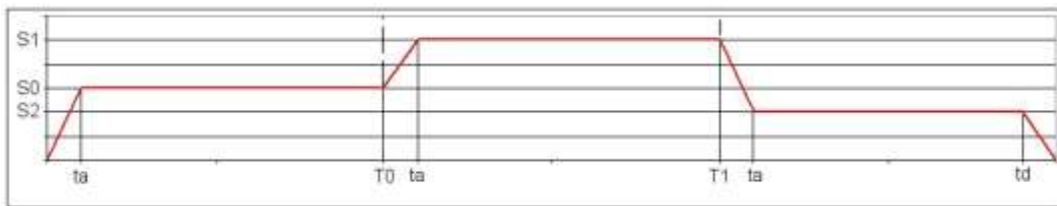


Figura 6.8. Representació gràfica de l'estratègia de desacceleració nul·la

En aquest mètode, s'espera a que el motor arribi a la seva posició objectiu a la velocitat nominal, sabent que no ha de realitzar cap parada excepte en el segment final. L'acceleració es calcula per assolir la nova velocitat en el temps previst.

6.2.3 Càlcul de la longitud del segment

La longitud mínima dels segments en els quals es discretitzarà la trajectòria vindrà donada per tres importants limitacions:

- Temps de comunicació
- Error de posició
- Temps d'acceleració i assoliment de la velocitat final

Segons aquest requeriments, podem realitzar els càlculs necessaris per arribar a un valor de compromís en la longitud mínima del segment.

6.2.3.1 Estimació del segment mínim segons el temps de comunicació

En primer lloc, necessitem tenir una estimació, al menys aproximada, del temps de comunicació emprat en l'enviament de totes les comandes utilitzades en el control del sistema.

Les instruccions necessàries i la seva longitud en bits són:

Comanda	Bytes comanda	Bytes resposta
Ordre de notificació	<16	-
Ordre de moviment	1	-
Anul·lació de desacceleració	2	-
Carregar posició	<16	-
Carregar velocitat	5	-
Carregar acceleració	6	-
Notificació de posició	-	1
TOTAL: 47 bytes		

Per tant, i segons la velocitat de comunicació dels controladors, configurats a 19200 bps, tenim que es necessitarà un temps igual a:

$$T_{com} = 47 \text{ bytes} \cdot \frac{8 \text{ bits}}{1 \text{ byte}} \cdot \frac{1 \text{ s}}{19200 \text{ bits}} = 0.01958 \text{ s} = 19.58 \text{ ms} \quad [6.1]$$

A la velocitat nominal desitjada del sistema, això implica una distància mínima igual a:

$$d = 0.01958 \text{ s} \cdot \frac{500 \text{ mm}}{1 \text{ s}} = 9.79 \text{ mm} \cong 10 \text{ mm} \quad [6.2]$$

6.2.3.2 Estimació del segment mínim segons l'error de posició

Òbviament, en trajectòries rectilínies l'error per discretització de la trajectòria serà nul. Quan més corba sigui aquesta més error és produirà. Per poder-lo dimensionar, representem gràficament l'error producte de la discretització de la trajectòria per circumferències de diferents radis.

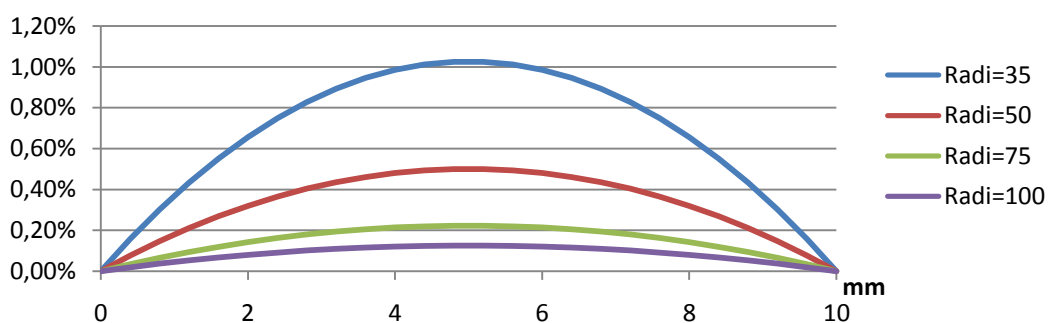


Figura 6.9. Error de posició amb segments de longitud 10 mm

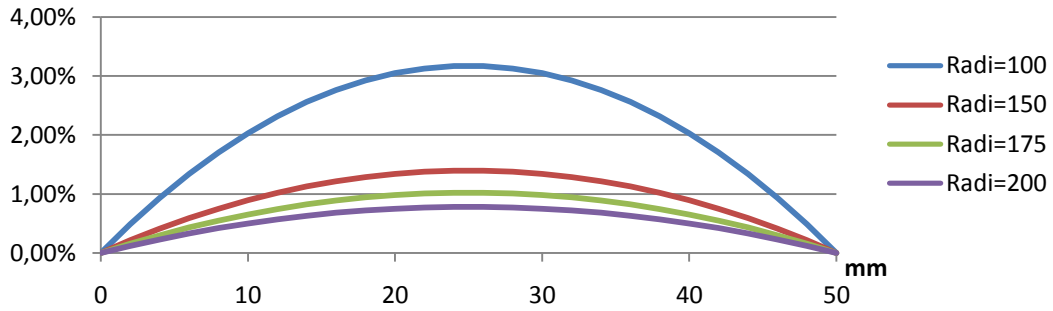


Figura 6.10. Error de posició amb segments de longitud 50 mm

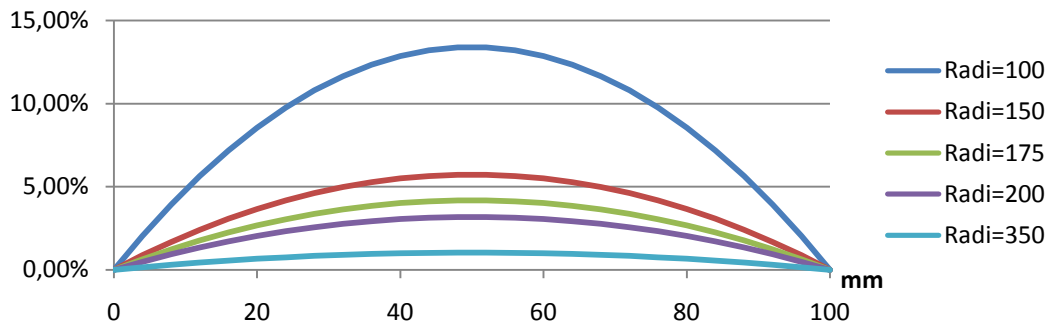


Figura 6.11. Error de posició amb segments de longitud 100 mm

Observem que l'error ràpidament augmenta conforme el segment es fa més gran, i que un error més petit del 1% només el tenim per segments de 10 mm.

Per segments de 50 mm, l'error és més gran que aquest 1% només que el radi de la trajectòria circular sigui major de 175 mm; i l'error es dispara per segments de 100 mm, on només és d'un 1% si el radi de la trajectòria circular és igual o superior a 350 mm.

6.2.3.3 Estimació del segment mínim segons el temps d'acceleració

Segons la velocitat i l'acceleració del sistema, la longitud mínima del segment ha de ser:

$$X = x_0 + v_0 \cdot t + \frac{1}{2} a \cdot t^2 = \frac{1}{2} a \cdot t^2 \quad [6.3]$$

Coneixem la velocitat, de 500 mm/s, per que ens ve donada per les especificacions del sistema i que hem calculat a partir de les dades dels motors seleccionats treballant en buit i a plena càrrega (fórmula 4.16).

I s'ha trobat l'acceleració màxima a partir de diferents proves empíriques, a partir de les quals es veu com, per acceleracions més grans de 2000 mm/s^2 , l'estructura que suporta el robot pateix fortes sotragades. Recordem que les especificacions inicials del sistema exigien una acceleració de 1 m/s^2 , per la qual cosa, amb 2 m/s^2 es compleixen sobradament.

En aquestes condicions:

$$v = a \cdot t \Rightarrow t = \frac{v}{a} = \frac{500}{2000} = 250 \text{ ms} \quad [6.4]$$

I substituint [6.4] en [6.3] obtenim:

$$X_{acc} = \frac{1}{2} \cdot 2000 \cdot (0.250)^2 = 62.5 \text{ mm} \quad [6.5]$$

Prendríem aquesta com a longitud mínima del segment, tenint en compte que el sistema necessitarà el doble de longitud en cas d'acceleració i desacceleració encadenades.

A partir dels valors obtinguts, optem per un segment de 50 mm de longitud mínima, ja que és el que ens dona un menor error de posició en les trajectòries corbes, a la vegada que ens permet perfectament les feines de comunicació del sistema.

D'aquesta forma s'introdueix un error en cas d'haver d'accelerar des de velocitat nul·la a velocitat màxima, o al revés, al no tenir la longitud necessària per assolir la velocitat final; però s'assumeix, donat que és el pitjor cas possible, i en canvi ens dona un millor comportament en la resta de casos.

6.2.4 Notificació de posició

Una vegada decidida la manera en que el sistema realitzarà les trajectòries, la forma en que concatenarem els segments i la longitud mínima d'aquests, ens faltaria conèixer la posició de les unitats lineals per tal de poder enviar ordres de moviment.

Aquesta posició ens interessarà conèixer-la en cada moment, o si més no, en l'instant en que aquest assoleixi la posició objectiu.

Els controladors utilitzats ens donen, a través del port sèrie, una notificació asíncrona quan el motor arribi a la posició enviada com a paràmetre.

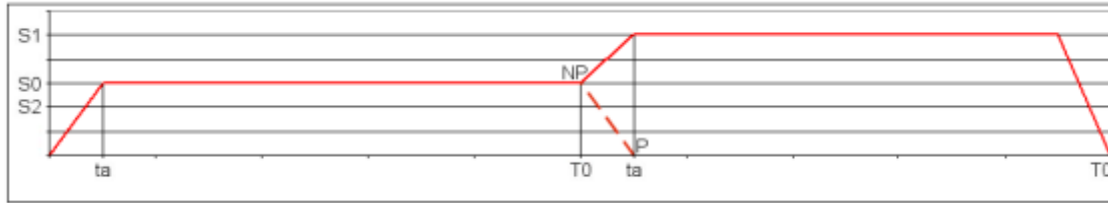


Figura 6.12. Representació gràfica dels instants de notificació de posició

El mode de funcionament dels controladors utilitzats, no permet el control de posició amb trajectòries enllaçades, però sí que admeten la recepció de noves consignes mentre s'està efectuant un moviment.

Això significa que, per poder realitzar la concatenació de segment tal com s'ha descrit en l'apartat anterior, haurem d'enviar com a consigna de posició un punt per davant de la posició desitjada (P). Es configurarà, però, el controlador, per tal de que envii la notificació de posició aquella que nosaltres desitgem com a objectiu (NP), que correspon amb la de la de finalització del segment.

D'aquesta manera, podrem enviar les noves consignes de posició, velocitat i acceleració just en el moment en que començaria la desacceleració.

6.2.5 Precàrrega de paràmetres

Cada càrrega de noves consignes als controladors implica l'enviament, via sèrie, de les instruccions i valors de les noves posicions, velocitats i acceleracions. Malgrat la velocitat de comunicació amb els controladors sigui elevada, s'ocupa un temps en la seva transmissió i recepció.

Per la forma de funcionament del sistema, i especialment per la estratègia de concatenació de segments escollida, les unitats lineals segueixen desplaçant-se mentre s'efectua aquesta comunicació.

Si a baixes velocitats i acceleracions, aquest desplaçament pot ser rebutjat, quan aquestes són altes provoca que la distància final al punt objectiu sigui massa elevada; recordem que el punt objectiu s'ha configurat més enllà del punt real de finalització del segment, com ha estat explicat a l'apartat anterior.

Per evitar-ho, aprofitem la característica del controlador de poder emmagatzemar les consignes, i no actualitzar-les fins que no es rebi un nou ordre de moviment.

De forma general, fem servir el següent algorisme de control:

```

Inici_seqüència
{
  Primer_punt (posició)
  Calcula (increment, velocitat, acceleració)
  Carrega (posició + increment, velocitat, acceleració)
  Ordre_notificació (posició)
  Ordre_moviment
  Mentre no(últim)
  {
    Següent_punt (posició)
    Calcula (increment, velocitat, acceleració)
    Carrega (posició + increment, velocitat, acceleració)
    Espera_notificació
    Ordre_notificació (posició)
    Ordre_moviment
  }
  Fi_mentre
}
Fi_seqüència

```

Com es pot veure, un cop donada la primera ordre de moviment, es carreguen les noves consignes i s'espera a que el robot arribi a posicionar-se a la consigna anterior, moment en que es torna a donar la nova ordre de moviment.

D'aquesta manera aconseguim reduir el temps de comunicació en el moment crític a l'enviament de l'ordre de notificació, que per característiques del codi no es pot efectuar abans, i d'inici de moviment.

6.2.6 Sincronització dels motors

Per tenir un bon sincronisme entre ambdós motors, els dos que mouen les unitats lineals, es imprescindible el càlcul de les acceleracions de forma que totes dues unitats assoleixin la velocitat donada al mateix moment. Fer-ho d'una altra forma implicaria que un dels motors podria arribar abans al punt objectiu que l'altre motor, observant-me desacceleracions i parades no desitjades.

A continuació, podem veure les gràfiques corresponents a dos motors no sincronitzats. El fet de tenir una mateixa acceleració fa que cada un d'ells assoleixi la seva velocitat nominal en un temps diferent, el que es tradueix en una resultant XY no lineal.

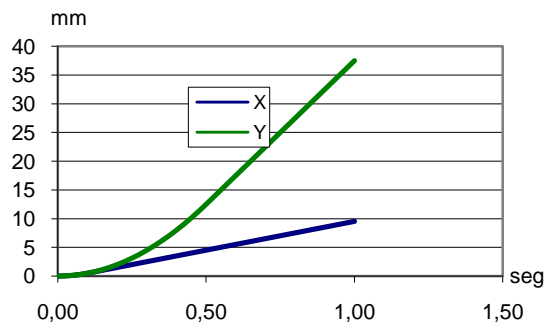


Figura 6.13 Velocitats de les unitats lineals

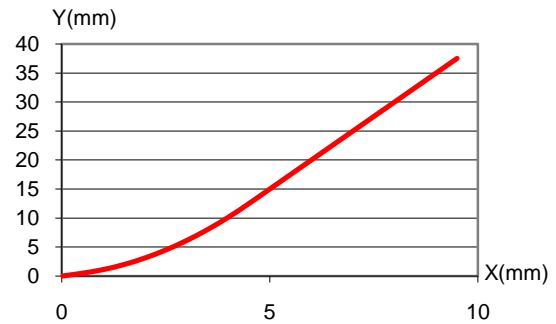


Figura 6.14 Velocitat del carro

En canvi, si es calculen les acceleracions per arribar en el mateix instant de temps a la velocitat desitjada, la resultant de les dues components es, efectivament, una funció lineal:

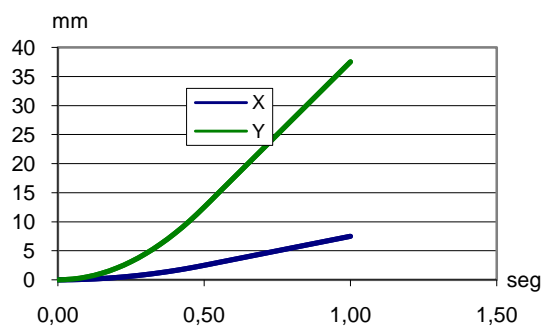


Figura 6.15 Velocitats de les unitats lineals

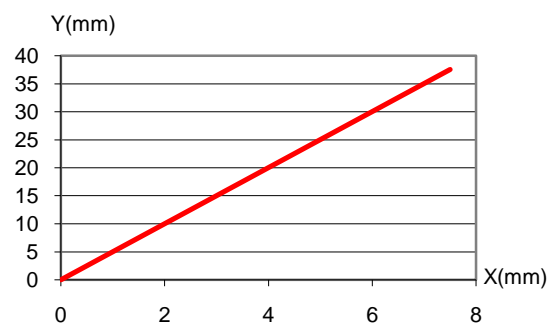


Figura 6.16 Velocitat del carro

6.2.7 Càlcul dels paràmetres

Tenint en compte les restriccions imposades en els apartats anteriors, pel càlcul del paràmetres s'utilitzaran les següents fórmules:

- Posició: ve donada per la longitud dels segments en que dividim la trajectòria.
- Velocitat: ve donada a partir de la velocitat final a la que es vol realitzar la trajectòria.
- Acceleració: es calcula a partir de la velocitat a la que es mou el sistema i la velocitat que es vol assolir:

$$acc = \frac{vel - vel_{actual}}{temps_{acceleració}} \quad [5.1]$$

6.2.7.1 Estimació del temps d'acceleració

Per facilitar els càlculs de velocitats i acceleracions, es suposarà, en el càlcul dels paràmetres a enviar al controlador, que el sistema té acceleració infinita, calculant i dimensionant l'error que aquesta hipòtesi ens introdueixi.

En el moment de calcular velocitats i acceleracions, prendrem com a temps d'acceleració una fracció del temps total en el que el robot ha de recórrer el segment de trajectòria en el que es trobi. Grafiquem la velocitat respecte del temps per diferents valors d'aquesta fracció (0%, 10% i 50%):

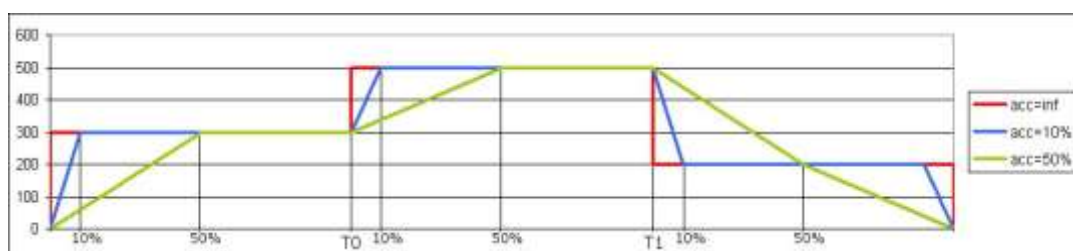


Figura 6.17. Representació gràfica de diferents temps d'acceleració

Amb aquests valors, i prenent:

- e1: espai recorregut en el temps d'acceleració (moviment rectilini uniformement accelerat)
- e2: espai recorregut amb acceleració nul·la (moviment rectilini a velocitat constant)

calculem l'espai recorregut en un lapse de temps de 10 s, des de la posició de parat fins a assolir la velocitat màxima. Trobem l'error percentual a fons d'escala segons diferents valors del temps d'acceleració:

	e1	e2	eTotal	Error
acc=inf	0	5000	5000	0
acc=10%	250	4500	4570	5%
acc=50%	1250	2500	3570	25%

Figura 6.18. Errors de posició per diferents temps d'acceleració

Veiem per tant que l'error és molt elevat quan el temps d'acceleració és la meitat del temps necessari per recórrer el segment, mentre que es redueix al 5% quan el temps d'acceleració és del 10%.

De fet, un error del 5% seria també inadmissible en la majoria de casos; en el nostre sistema, però, s'ha de tenir en compte que aquest error del 5% el tenim únicament en el pitjor cas possible, amb el sistema passant de parat a velocitat màxima.

A més, aquest error de posició està calculat per un sol segment, de forma que a major quantitat de segments per trajectòria, més petit serà l'error final, ja que no és acumulable.

Donat que la longitud de l'eix X, el més curt, és d'aproximadament 1500 mm, el número de segments de 50 mm de longitud en que es divideix és de 30. Per un temps d'acceleració del 10%, l'error es divideix entre tots ells, resultant un error total del 0.17%; quan tractem d'un temps del 50%, l'error total serà del 0.83%.

En el cas de la trajectòria més llarga, la del eix Y, d'aproximadament 2000 mm, es tindrà un número total de 40 segments. Per un temps d'acceleració del 10%, tindrem un error total de 0.125%; per un temps dels 50%, l'error en la totalitat de la trajectòria serà de 0.625%.

Òbviament, es tracta d'errors assumibles quan la trajectòria és llarga, però comencen a ser crítics quan és de poca longitud. Si per un temps d'acceleració del 10%, l'error es redueix al 1% per trajectòries amb 5 segments (250 mm), per un temps d'acceleració del 50% encara es manté al 5%.

Així doncs, agafarem un temps d'acceleració del 10% del temps total que triga el sistema en recórrer el segment actual, mentre que el 90% del temps restant, el robot es mourà a velocitat constant.

6.2.7.2 Gradient màxim de velocitat entre segments consecutius

Sabent que l'acceleració màxima del sistema és de 2000 mm/s² i que la velocitat nominal és de 500 mm/s, el temps necessari per recórrer el segment, de 50 mm de longitud, serà:

$$t = \frac{e}{v} = \frac{50}{500} = 0.1 \text{ s} = 100 \text{ ms} \quad [5.2]$$

Per tant, en un 10% d'aquest temps, la velocitat màxima assolible serà de:

$$vel = a \cdot t = 2000 \cdot 0.1 = 200 \text{ mm/s} \quad [5.3]$$

Això no implica un mal funcionament del sistema; indica que serà incapaç d'assolir la velocitat desitjada en un sol segment de la trajectòria quan el gradient de velocitat entre un segment i un altre sigui superior a 200 mm/s.

En última instància, serà l'error total el que ens dirà si l'estratègia i els paràmetres seleccionats són vàlids pels nostres requeriments.

6.3 Entorn visual

S'ha desenvolupat una interfície gràfica per tal de fer més còmode per l'usuari la utilització del sistema.

Aquesta interfície gràfica consta de dues parts, clarament diferenciades i seleccionables mitjançant pestanyes:

- Funcions de baix nivell: permet el posicionament del robot a partir d'ordres de control de velocitat i de posició
- Funcions d'alt nivell: permet la realització de trajectòries i el seguiment de diferents dispositius, entre ells, el dispositiu òptic de l'ordinador, o ratolí.

Per tal de que la posició del robot sigui coneguda en cada instant de temps, es compta amb una rutina d'execució periòdica. A partir d'un temporitzador, inclòs al dissenyador de classes gràfiques Qt3, es pren la lectura de posició del robot i s'actualitzen els controls corresponents, com els quadres de text o els gràfics a on es representa la posició real del dispositiu.

A la vegada, aquest temporitzador ens permet controlar l'ús del port sèrie, ja que cada pregunta de posició consumeix part dels seus recursos, disminuint així els necessaris per la tasca més crítica de control de trajectòries.

Donat que la única finalitat de mostrar les dades per pantalla és la d'informar a l'usuari, es configura per una execució periòdica cada 100 ms. Amb aquesta freqüència d'execució els controls estan contínuament actualitzats, l'usuari coneix en tot moment quina és la posició del robot i el port sèrie no es carrega amb comunicacions no bàsiques pel control del sistema.

Per altra banda, l'entorn visual ens dona la possibilitat de bolcar dades a un fitxer de sortida, lo que serà útil a l'hora de comprovar els errors comesos en la realització de les trajectòries.

6.3.1 Funcions de baix nivell

Són les dues funcions bàsiques que ens ofereix el controlador Faulhaber MCDC2805: consigna de posició i consigna de velocitat.

6.4.1.1 Control de posició

Actuant sobre els controls lliscants o *sliders* enquadrats en el marc de 'control de posició', es fixa la posició a la que es vol situar la unitat lineal corresponent.

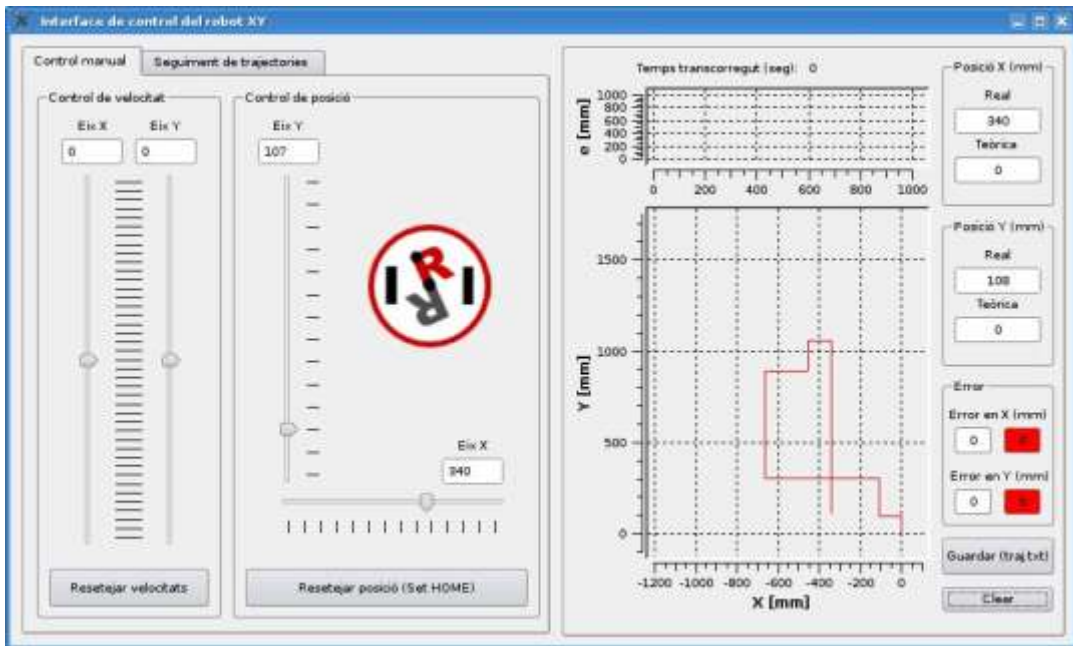


Figura 6.19. Pantalla de control de posició i gràfica corresponent

6.4.1.2 Control de velocitat

De la mateixa forma que en el control de posició de baix nivell vist en el darrer punt, actuant sobre els controls lliscants o *sliders* enquadrats en el marc de 'control de velocitat', es fixa la velocitat a la que es vol que es mogui la unitat lineal corresponent.

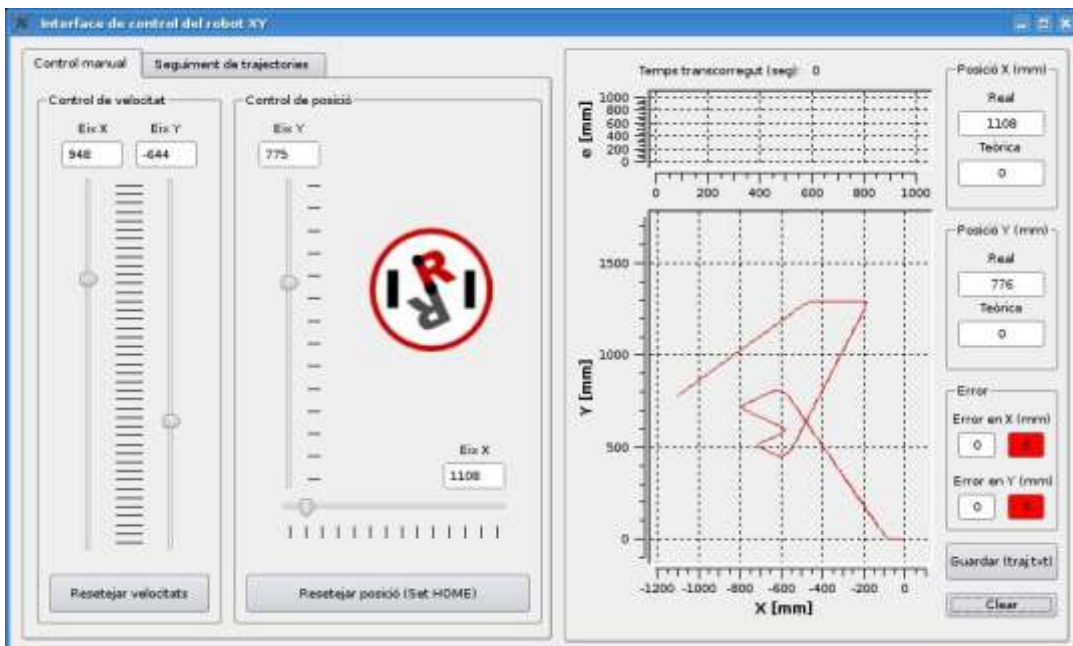


Figura 6.20. Pantalla de control de velocitat i gràfica corresponent

6.4.2 Funcions d'alt nivell

Utilitzant les funcions de baix nivell de control de posició, s'ha desenvolupat rutines més complexes de control de trajectòries.

6.4.2.1 Trajectòria lineal

El sistema realitza una trajectòria rectilínia a partir dels paràmetres passats: punt d'origen, punt final i velocitat lineal.

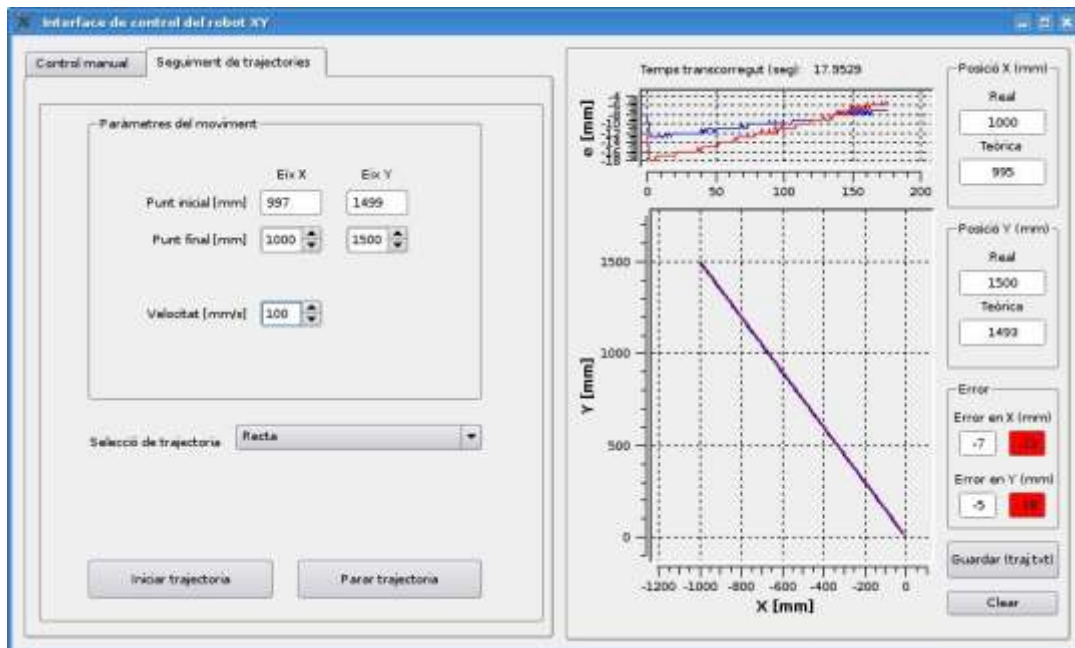


Figura 6.21. Pantalla de control de trajectòries rectilínies

6.4.2.2 Trajectòria circular

Realitza una trajectòria circular, ja sigui cercle o arc, a partir de la configuració donada: punt central, punt inicial de la trajectòria, velocitat angular i longitud de l'arc (360° per una circumferència completa).

Permet seleccionar el sentit de la marxa, horari o anti-horari.

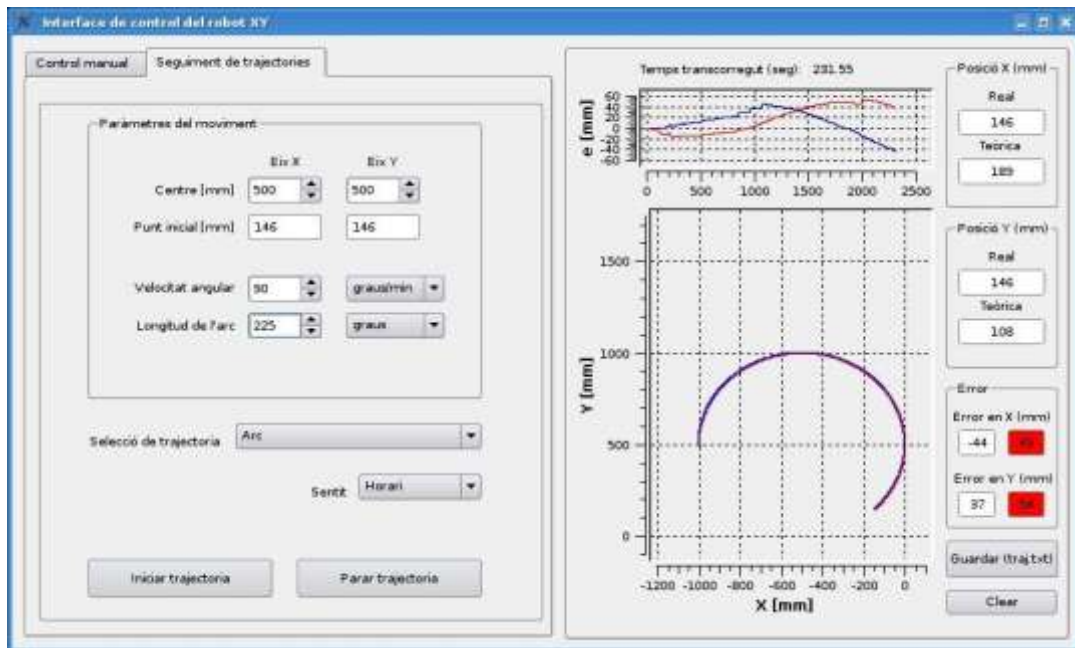


Figura 6.22. Pantalla de control de trajectòries circulars

6.4.2.3 Importació des de fitxer

Realitza trajectòries rectilínies concatenades a partir de les dades importades d'un fitxer de text, en el qual es defineixen els punts finals del segment i la velocitat a la que s'ha de realitzar. El punt inicial del segment és aquella posició en la que es trobi el carro en el moment d'iniciar-lo.

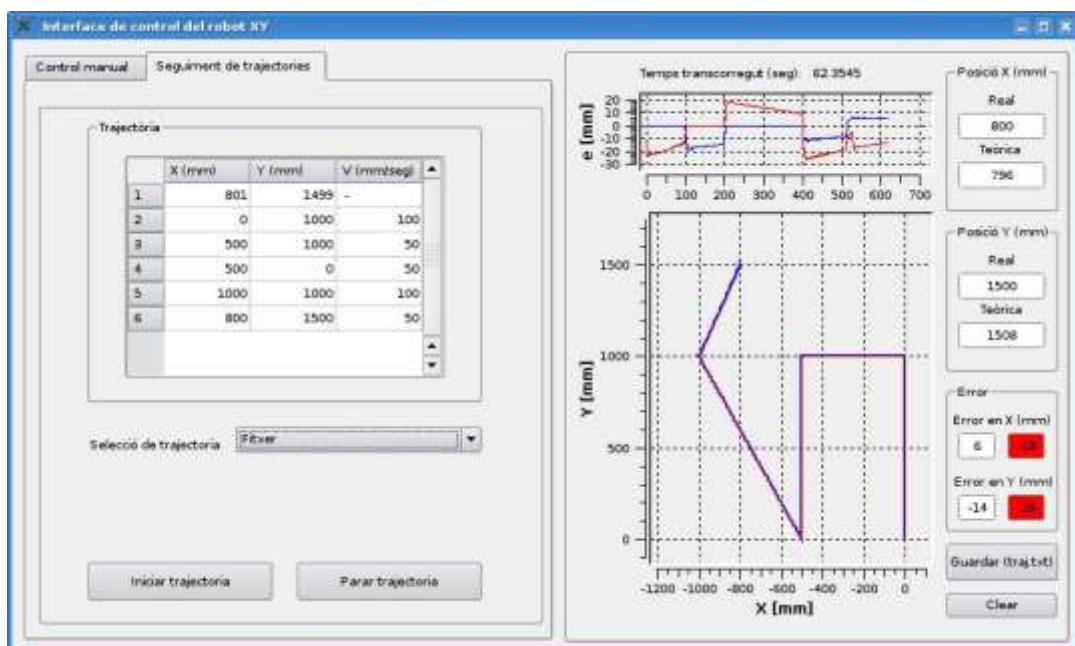


Figura 6.23. Pantalla de control de trajectòries importades de fitxer

6.4.2.4 Seguiment de ratolí

Permet la selecció de punts objectius mitjançant pulsacions del ratolí sobre la finestra gràfica. Realitza els segments a una velocitat fixada de 100 mm/s; igual que en la importació des de fitxer de text, el punt inicial del segment és aquella posició en la que es trobi el carro en el moment d'iniciar-lo.

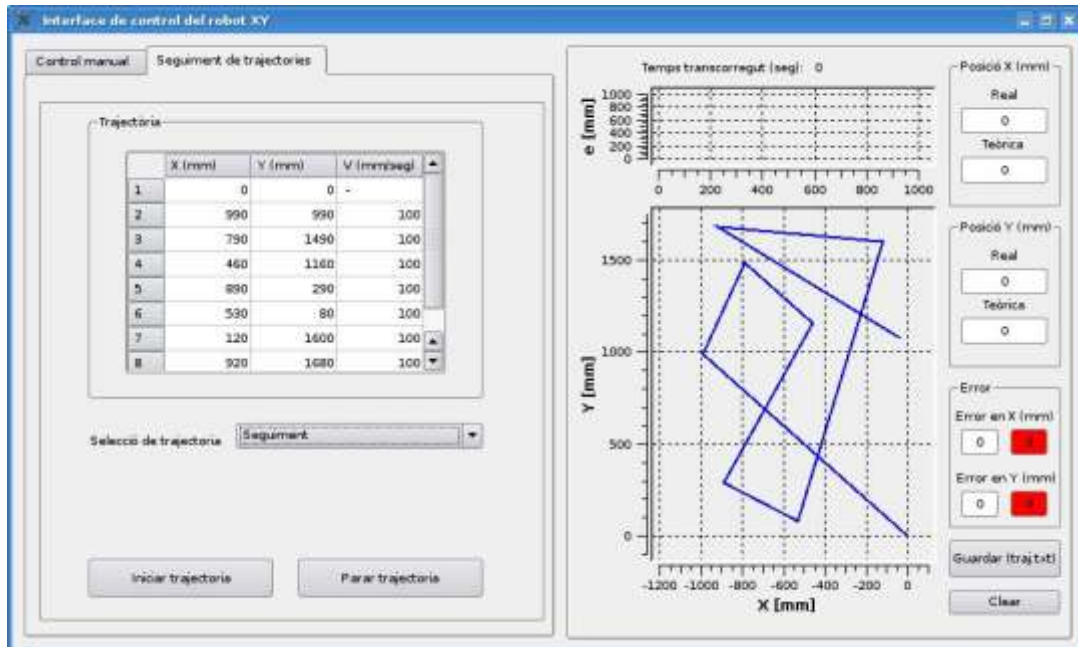


Figura 6.24. Pantalla de control de trajectòries configurades per dispositiu òptic

7. Experiments i resultats

Per comprovar el funcionament del sistema s'han realitzat diferents trajectòries, amb la finalitat de trobar els errors que s'obtenen tant en desplaçament com en temps.

7.1 Trajectòria rectilínia única

Analitzem el comportament del sistema al realitzar una trajectòria rectilínia, del punt (0,0) al punt (1000,1500) a una velocitat de 100 mm/s. En aquestes condicions, s'obté la següent gràfica, en dues escales diferents:

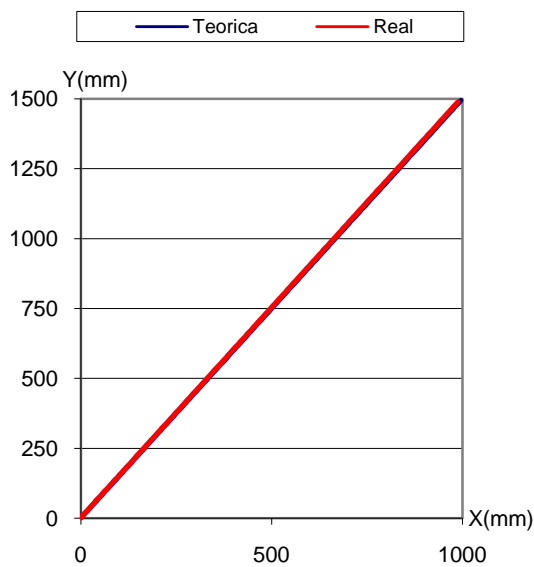


Figura 7.1 Trajectòria rectilínia

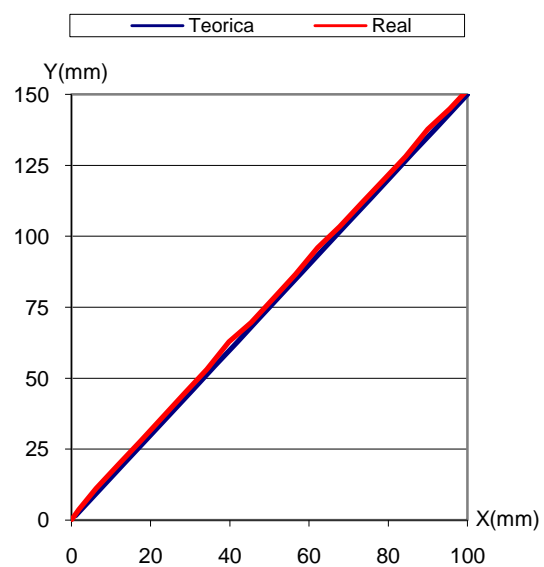


Figura 7.2 Trajectòria rectilínia ampliada

Com s'observa als gràfics, inclús al de la dreta, amb l'escala ampliada, l'error comés al seguir la trajectòria donada és molt petit, del rang del mil·límetre.

Malgrat això, si grafiquem l'error existent en l'eix X tenint en compte la posició que ha de tenir en un cert instant de temps, aquest error és inicialment molt més elevat del que sembla.

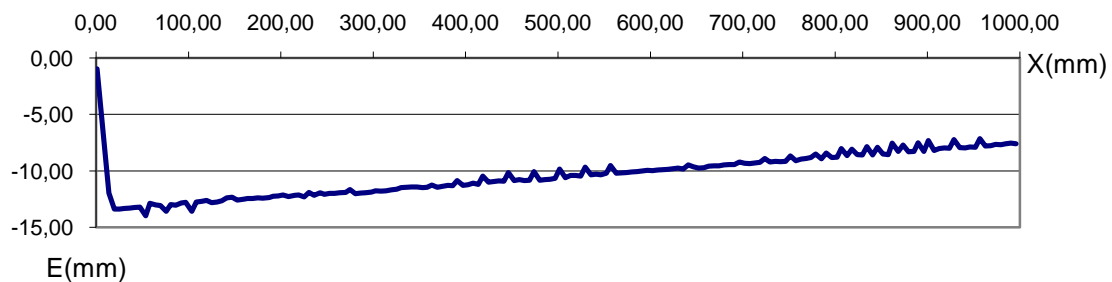


Figura 7.3. Error en l'eix X en la realització d'una trajectòria rectilínia

El mateix passa si grafiquem l'error trobat a l'eix Y.

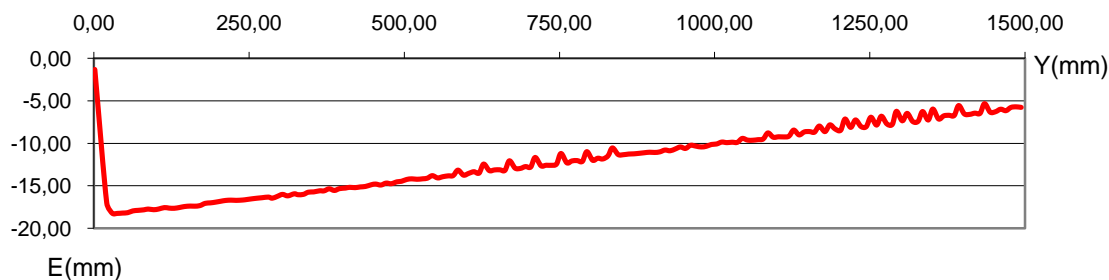


Figura 7.4. Error en l'eix Y en la realització d'una trajectòria rectilínia

Troblem doncs, un primer error inherent al sistema, causat per no assolir la posició donada en l'instant de temps donat. I que va disminuint conforme passa el temps.

Aquest error es degut principalment a les característiques mecàniques del sistema, especialment al temps d'arrencada. S'ajusta a partir de la configuració dels reguladors PID integrats en els controladors dels motors.

En el nostre cas, vist que el comportament del sistema és correcte i dintre de les especificacions requerides, hem mantingut els valors per defecte que ha programat el fabricant.

7.2 Trajectòria rectilínia múltiple a igual velocitat

Per saber si l'error comés, és un error de repetibilitat, s'han realitzat un seguit de trajectòries iguals, a la mateixa velocitat, en una quantitat de 5 trajectòries d'anada i 5 trajectòries de tornada.

La velocitat escollida ha estat de 100 mm/s per estar al mig del marge de funcionament, i la longitud de 1000 mm a cada eix, es a dir, 1.414 mm de longitud total.

En aquestes condicions s'han trobat els següents resultats:

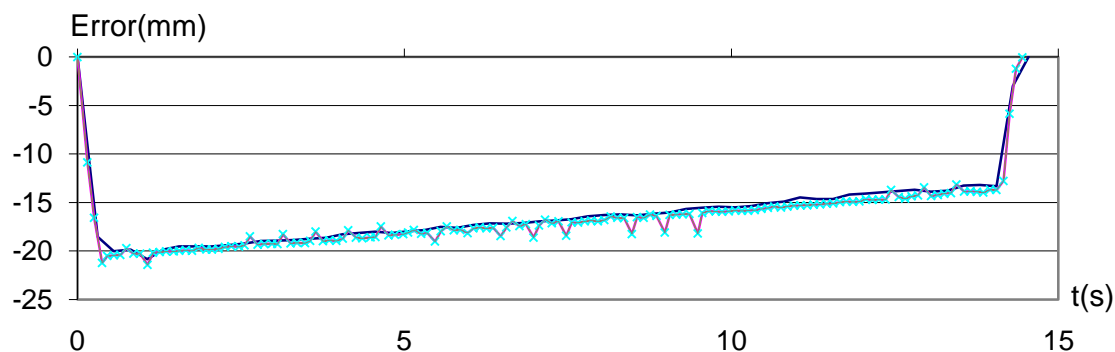


Figura 7.5. Error de repetibilitat en l'eix X; velocitat positiva

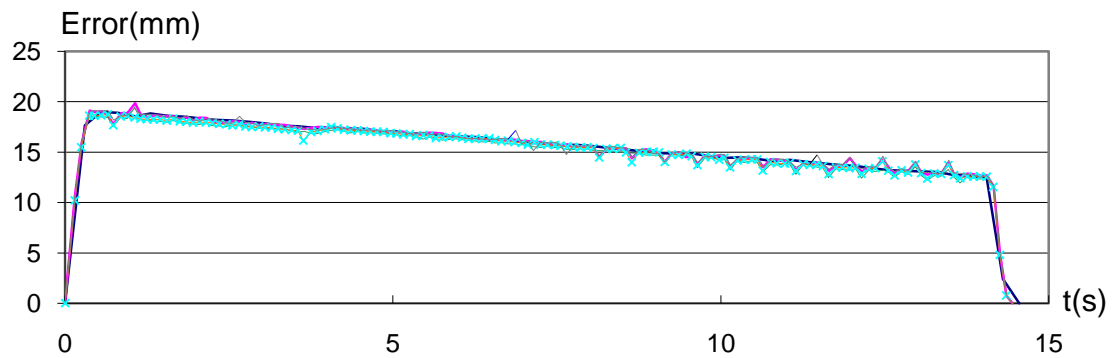


Figura 7.6. Error de repetibilitat en l'eix Y; velocitat negativa

Veiem doncs que, aparentment, l'error és sistemàtic i, per tant evitable. Les trajectòries segueixen les mateixes posicions tant en l'eix X com en l'eix Y, amb un error de repetibilitat despreciable.

Els resultats obtinguts al realitzar les mateixes trajectòries en sentit invers han estat similars.

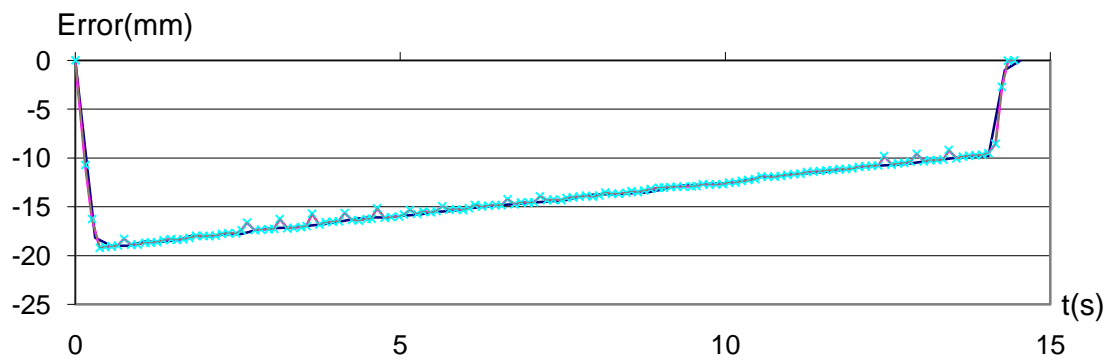


Figura 7.7. Error de repetibilitat en l'eix X; velocitat positiva

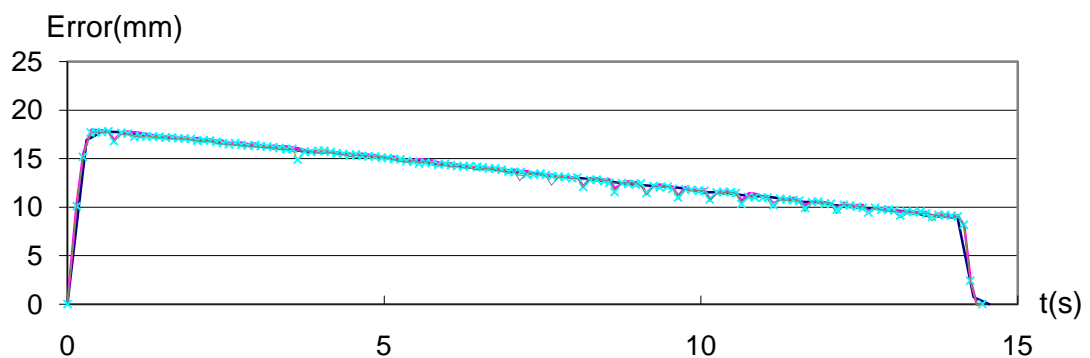


Figura 7.8. Error de repetibilitat en l'eix Y; velocitat positiva

7.3 Trajectòria rectilínia múltiple a distintes velocitats

Per poder conèixer millor i dimensionar els errors, s'han realitzat un seguit de trajectòries rectilínies, del punt (0,0) al (1000,1000), a diferents velocitats.

Donat que en la comprovació de l'error de repetibilitat, en el darrer apartat, s'ha observat que aquest és despreciable en els dos sentits de funcionament, i que ambdues unitats lineals es comporten igual sigui quin sigui el signe de la velocitat, analitzarem tan sols en un sentit de la marxa.

7.3.1 Representació del error de posició en funció del temps

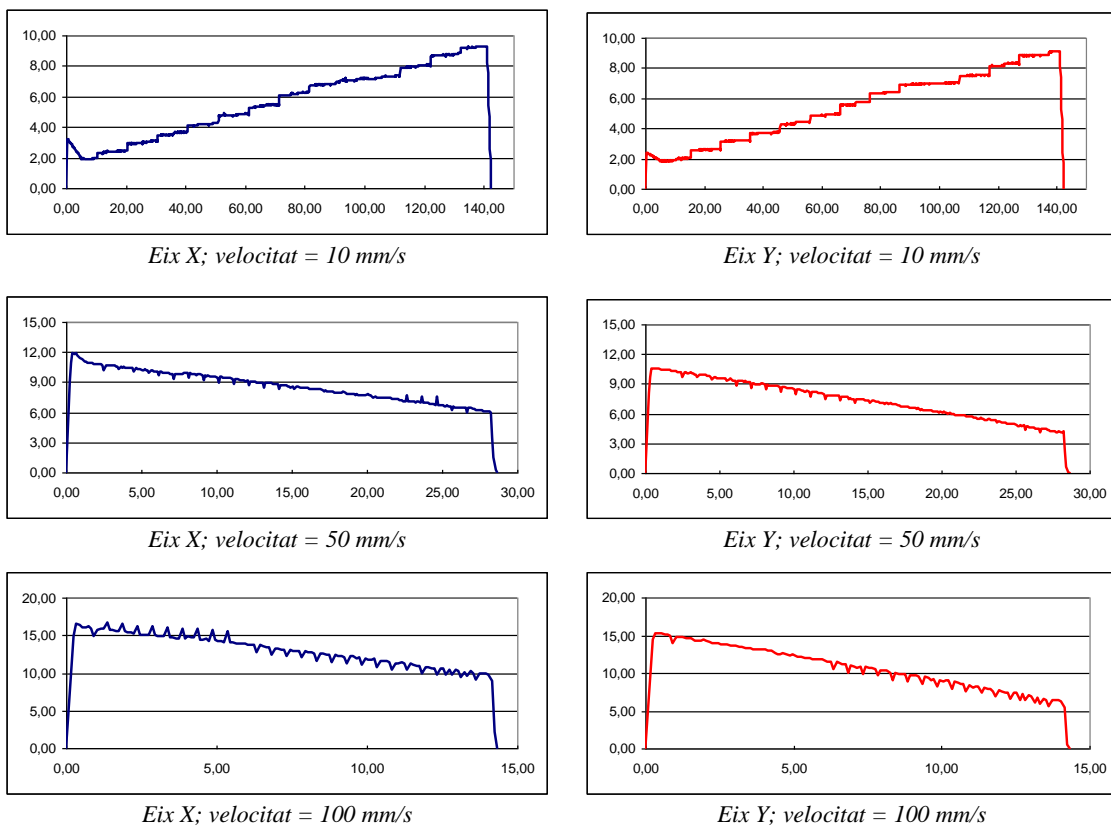


Figura 7.9. Errors de posició en funció del temps
- eix X: temps (s); eix Y: error(mm) -

A partir de les dades obtingudes, podem calcular els errors màxims de posició pel fons d'escala:

Velocitat (mm/s)	Eix X (mm)	Eix Y (mm)	Error X (%)	Error Y (%)
10	9.29	9.11	0.93	0.91
50	11.85	10.57	1.18	1.06
100	16.48	15.30	1.65	1.53

7.3.3 Representació del error del temps en funció de la posició

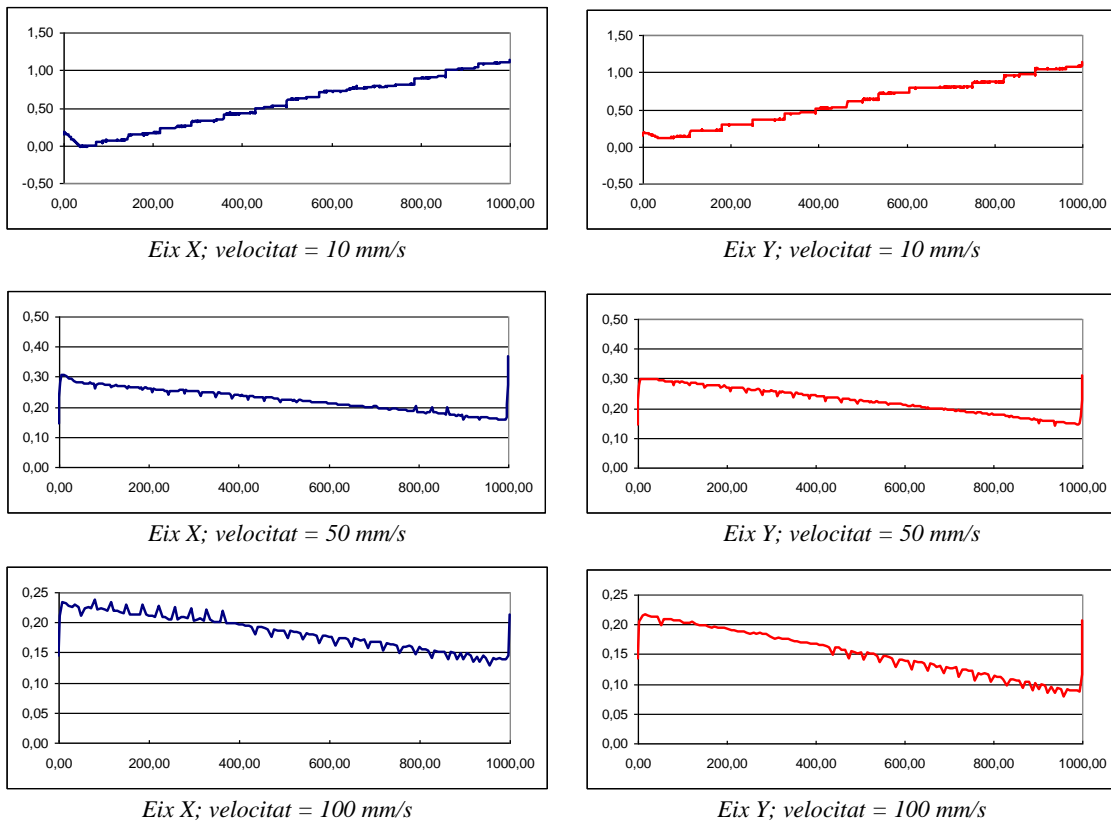


Figura 7.10. Errors de temps en funció de la posició
- eix X: posició (mm); eix Y: temps (s) -

Calculem ara els errors màxims en temps:

Velocitat (mm/s)	Eix X (mm)	Eix Y (mm)	Error X (%)	Error Y (%)
10	1.13	1.14	0.80	0.91
50	0.37	0.31	1.31	1.10
100	0.24	0.21	1.70	1.48

7.4 Anàlisi del errors

Una vegada realitzats els experiments pertinents, observem que l'error final que presenta el sistema és en el domini del temps, i no pas en el de posició. Es a dir, el sistema assoleix sempre la posició consignada, encara que no ho fa en l'instant de temps desitjat.

Aquest error en el temps es degut, en part, a les característiques mecàniques del sistema i, en part, a l'estratègia de control utilitzada.

S'observa que l'error disminueix per les velocitats de 50 i 100 mm/s, mentre que augmenta a velocitats inferiors, com la de 10 mm/s. Aquest diferent comportament és degut a que, encara que existeixi la realimentació en posició, no existeix realimentació en velocitat o en temps.

La rutina de control de trajectòries envia una comanda de posició, velocitat i acceleració als motors; quan el dispositiu ha assolit la posició objectiu, envia un event a l'aplicació amb la informació de que el robot es troba en aquesta posició; malgrat tot, l'aplicació no pren lectura del temps que s'ha trigat en assolir aquesta posició, per la qual cosa que no és capaç de saber si la velocitat a la que arribat es efectivament la consignada.

Per què no existeix aquesta realimentació de la velocitat? Per la estratègia de control escollida, on es prioritza la posició per sobre de la velocitat o el temps en assolir-la.

Quan el sistema arriba a la posició objectiu, i com s'ha vist al llarg de la present memòria, els controladors ja tenen les noves consignes pre carregades, i no es dóna opció a la càrrega de les noves consignes corregides per eliminar o minimitzar l'error.

Malgrat tot, aquest error és inferior al 1'5% en els pitjors dels casos, el que és perfectament acceptable per les característiques del sistema.

8. Conclusions i treball futur

8.1 Conclusions

Al llarg del desenvolupament de l'actual Projecte Final de Carrera, s'han aconseguit els diferents objectius que s'havien proposat.

Pel que fa referència a la part hardware del projecte:

- Robot XY: s'ha dissenyat i construït un robot cartesià per la cel·la de braços robòtics. Una vegada fetes les proves de funcionament, s'ha comprovat que compleix els requeriments inicials i que té un bon comportament en quant a errors.

En el carro mòbil d'aquest robot poden ser instal·lats diferents dispositius, com càmeres de visió o punters làser, i permetrà el seu posicionament en qualsevol punt de la superfície de treball.

- Armari de control: s'ha dissenyat i implementat un armari de control per centralitzar tota la operativa corresponent als accessoris de la cel·la de braços robòtics. A ell es connecten tots els dispositius que formen part dels robots articulats Staübli RX60 que formen la cel·la robòtica (sensor de força i pinces), les cortines òptiques i el nou robot XY dissenyat en el present projecte. Compleix les funcions de nus de comunicacions i armari d'alimentació de tots ells.
- Barreres òptiques: s'han instal·lat cortines òptiques a l'espai accessible per l'usuari de la cel·la de braços robòtics. D'aquesta forma, qualsevol intromissió d'un usuari dins de la zona de treball quan aquests es troben en moviment automàtic provoca la seva parada.

Aquesta parada d'emergència pot ser anul·lada quan el moviment dels braços és en mode manual, per facilitar les tasques d'investigació que amb ells es desenvolupen.

De la part del software:

- Controladors del robot: s'han desenvolupat controladors del robot cartesià que permeten el seu posicionament en qualsevol punt del seu espai de treball, mitjançant ordres de baix nivell consistents en consignes de posició i/o velocitat.

A més, s'han desenvolupat diferents aplicacions d'alt nivell que permeten al robot la realització de trajectòries, ja siguin aquestes figures geomètriques, seqüències de dades importades des de fitxer o seqüències de dades introduïdes des del dispositiu òptic del Pc (ratolí).

- Aplicació en entorn visual: per facilitar les tasques a l'usuari, s'ha desenvolupat una aplicació en entorn gràfic que, fent ús dels controladors del robot desenvolupats, permeten operar amb ell d'una manera senzilla.

Es presenta com un paquet informàtic tancat que, de forma transparent, fàcil i intuïtiva, explota les capacitats dels controladors del robot implementats.

A la vegada, permet la representació gràfica per pantalla de les trajectòries seguides pel carro mòbil del robot i de la posició d'aquest en tot moment, el càlcul d'errors en temps real i el bolcat de trajectòries realitzades a fitxer.

8.2 Treball futur

Donat que la idea del robot posicionador XY és la de dotar de noves funcionalitats als braços robòtics, es preveu continuar el seu desenvolupament:

- Control de velocitat: de forma que a més de tenir una realimentació en posició, es tingui una realimentació en velocitat, com a mètode per minimitzar els errors existents.
- Seguiment del braç robòtic mitjançant la comunicació entre els controladors del braç i els controladors del robot: afegint a l'aplicació de control desenvolupada les classes corresponents al control del braç robòtic, serà possible la comunicació entre ambdós dispositius, poden així realitzar diverses tasques de col·laboració.
- Seguiment del braç robòtic mitjançant càmeres de visió: mitjançant processament d'imatges adquirides a partir d'una càmera de visió instal·lada en el carro mòbil del robot, es considerarà fer el seguiment d'aquest.
- Desenvolupament de noves trajectòries: pel mètode de disseny seguit a l'aplicació informàtica, és possible l'addició d'infinitud de trajectòries possibles amb petits canvis del programa. Així, podrien ser ampliades les actualment existents (recta, rectangle, arc i rastreig) amb altres de noves, com el·lipsis, espirals o qualsevol altra que un proper disseny pugui requerir.

9. Bibliografia

- [1] Raúl Valls, 2006. Control de posició i força d'un braç robòtic industrial. Projecte Final de Carrera en el Institut de Robòtica Industrial – UPC.
- [2] Antonio Larrosa, 2003. El tutorial del desarrollo visual. larrosa@kde.org
- [3] A. D. Marshall, 1994. Programming in C (UNIX system calls and subroutines using C). <http://www.cs.cf.ac.uk/Dave/C/>
- [4] Héctor Tejeda Villela, 2005. Manual de C. <http://fismat.umich.mx/mn1/manual/manual.html>
- [5] WikiArde. Asociación de Robótica y Domótica de España. <http://wiki.webdearde.com/index.php/Portada>
- [6] Qt-Designer Tutorial. <http://www.trolltech.com>
- [7] KDevelop Tutorial. <http://www.kdevelop.org>

Annex I
Esquema de l'armari de control

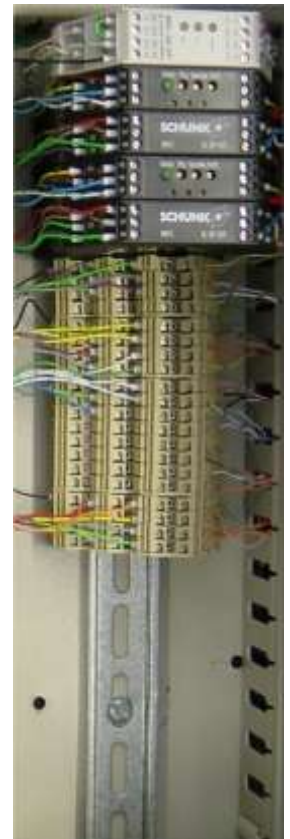
Armari de control

Vista frontal:



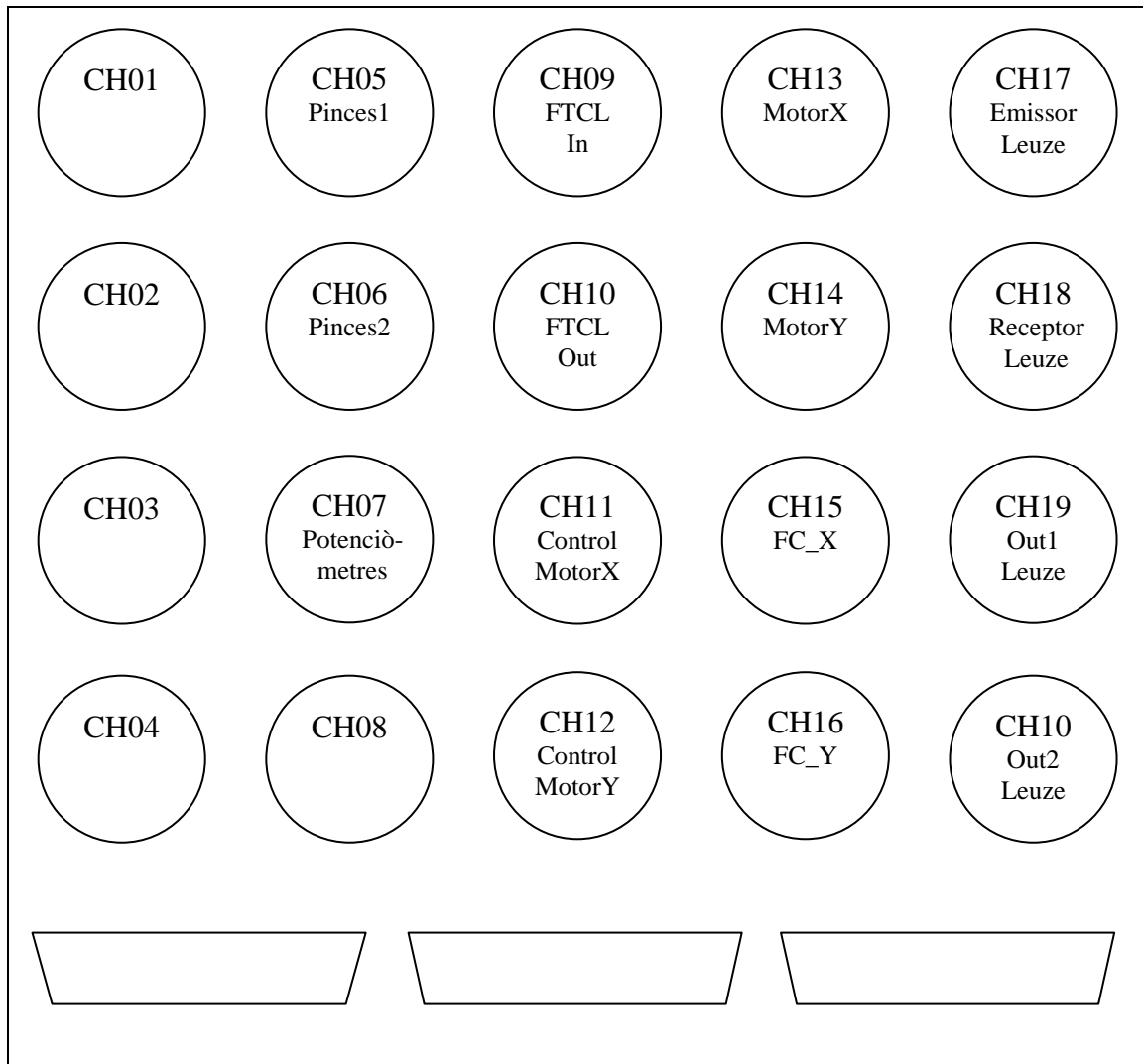
Armari de control – Carril superior

Borne34	14	MSI.S22	S21	MSI	A1	Borna25	13	Borna32
	S11	MSI.S21	S22		S33	Pulsador.4	23	Borna36
CH18.4	S12	CH18.2	S31		S34	Pulsador.3	24	Borna38
	32	Borne18	A2		S35		31	
OPTO5.Out2	19	OPTO24.1	13	PINZA1	6		12	Borna17
OPTO5.Out3	20	OPTO24.2	14		5		11	OPTO5.Out1
OPTO5.Out4	21	NI6220.AI0	15		4		10	Borna24
NI6722.AO0	22	Borna26	16		3	CH5.2	9	CH5.4
NI6722.AO1	23	Borna20	17		2		8	NI6220.AGND
NI6722.AO2	24	OPTO24.3	18	1	CH5.1	7	CH5.3	
OPTO5.Out6	19	OPTO24.4	13	PINZA2	6		12	Borna18
OPTO5.Out7	20	OPTO24.5	14		5		11	OPTO5.Out5
OPTO5.Out8	21	NI6220.AI	15		4		10	Borna25
NI6722.AO3	22	Borna27	16		3	CH6.2	9	CH6.4
NI6722.AO4	23	Borna21	17		2		8	NI6220.AGND
NI6722.AO5	24	OPTO24.6	18		1	CH6.1	7	CH6.3
Borne23	Pinça1.22	Pinça1.23	1	AO1		AO0	AOGND	NI-PCI6722
	Pinça1.24	Pinça2.22	2	AO3		AO2		
	Pinça2.23	Pinça2.24	3	AO5		AO4		
			4	AO7		AO6		
OPTO5.In	OPTO5.In1	OPTO5.In2	5	DO1		DO0	DOGND	NI-PCI6220
	OPTO5.In3	OPTO5.In4	6	DO3		DO2		
	OPTO5.In5	OPTO5.In6	7	DO5		DO4		
	OPTO5.In7	OPTO5.In8	8	DO7		DO6		
CH7.7	Pinza1.15	Pinza2.15	9	AI1		AI0	AIGND	NI-PCI6220
	CH7.4	CH7.5	10	AI3		AI2	Pinza1.8	
	CH7.6		11	AI5		AI4	Pinza2.8	
			12	AI7		AI6		
			13	AI9		AI8		
			14	AI11		AI10		
			15	AI13		AI12		
			16	AI15		AI14		
OPTO24.Out	OPTO5.Out	CH7.1	17	+5V		+5V	DIGND	
	OPT24.1	OPT24.2	18	D11		D10		
	OPT24.3	OPT24.4	19	D13		D12		
	OPT24.5	OPT24.6	20	D15		D14		



	GND	1	GND		
OPTO24	Borna 19	Pinça1.13	2	NI6220_DI0	NI6220 5V
		Pinça1.14	3	NI6220_DI1	
		Pinça1.18	4	NI6220_DI2	
		Pinça2.13	5	NI6220_DI3	
		Pinça2.14	6	NI6220_DI4	
		7	NI6220_DI5		
OPTO5	NI6722 DGND	NI6722_DO0	8	Pinça1.11	Borna 30
		NI6722_DO1	9	Pinça1.19	
		NI6722_DO2	10	Pinça1.20	
		NI6722_DO3	11	Pinça1.21	
		NI6722_DO4	12	Pinça2.11	
		NI6722_DO5	13	Pinça2.19	
		NI6722_DO6	14	Pinça2.20	
		15	Pinça2.21		
	GND	16	GND		
	FE1_0 Vdc	17	Pinça1.12		
	MSI.32	18	Pinça2.12		
	OPTO24.In	19	Opto5.Out		
	Pinça1.17	20	CH17.3		
	Pinça2.17	21	CH18.3		
		22	CH9.7		
		23	NI6722.AOGND		
	FE1_24 Vdc	24	Pinça1.10		
	MSI.13	25	Pinça2.10		
	Pinça1.16	26	CH18.1		
	Pinça2.16	27	CH17.1		
		28	CH17.4		
		29	CH9.6		
		30	OPTO5.OUT		
	GND	31	GND		
	MSI.13	32	CH19.1		
	Selector1.4	33	CH20.1		
	MSI.14	34	CH19.2		
	Selector1.3	35	CH20.2		
	MSI.23	36	CH19.3		
	Selector2.4	37	CH20.3		
	MSI.24	38	CH19.4		
	Selector2.3	39	CH20.4		
		40	CH15.4		
	MCDC1.GND	41	CH16.4		
	MCDC2.GND	42	FE2_0 Vdc		
		43	CH15.1		
	MCDC1.+24V	44	CH16.1		
	MCDC2.+24V	45	FE2_24 Vdc		
	GND	46	GND		
	MCDC1.MOT-	47	CH13.2		
	MCDC1.MOT+	48	CH13.1		
	GND	49	GND		
	MCDC2.MOT-	50	CH14.2		
	MCDC2.MOT+	51	CH14.1		
GND	GND	52	GND	GND	GND
MCDC1.ChA	MCDC1.+5V	53	CH15.7		CH15.6
MCDC1.SGND	MCDC1.ChB	54	CH15.5		CH15.8
MCDC1.ChA	MCDC1.+5V	55	CH16.7		CH16.6
MCDC2.SGND	MCDC2.ChB	56	CH16.5		CH16.8
MCDC1.3.In	MCDC1.Fault	57	CH15.3		CH15.2
MCDC2.3.In	MCDC2.Fault	58	CH16.3		CH16.2
GND	GND	59	GND	GND	GND
	MCDC1.RXD	60	CH11.2		
	MCDC1.TXD	61	CH11.3		
		62	CH11.5		
	MCDC2.RXD	63	CH12.2		
	MCDC2.TXD	64	CH12.3		
		65	CH12.5		
	CH9.2	67	CH10.2		
	CH9.3	68	CH10.3		
	CH9.5	69	CH10.5		
	GND	70	GND		

Armari de control: connectors (vista exterior)



Armari de control: canals

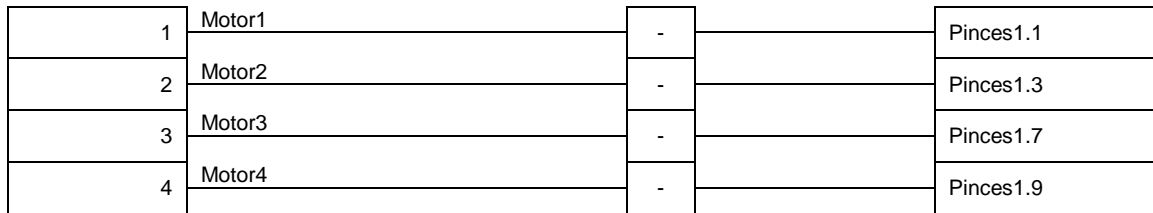
Canal 1: sense utilitzar.

Canal 2: sense utilitzar.

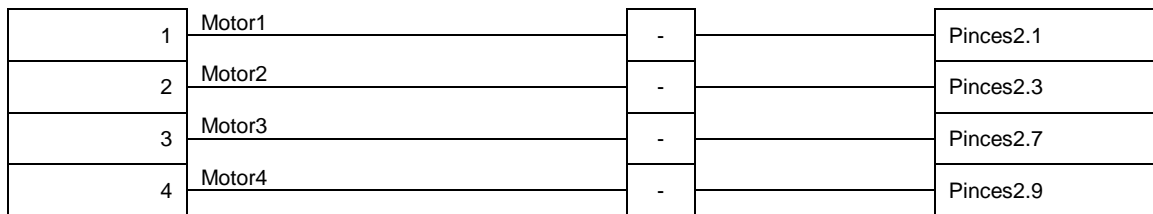
Canal 3: sense utilitzar.

Canal 4: sense utilitzar.

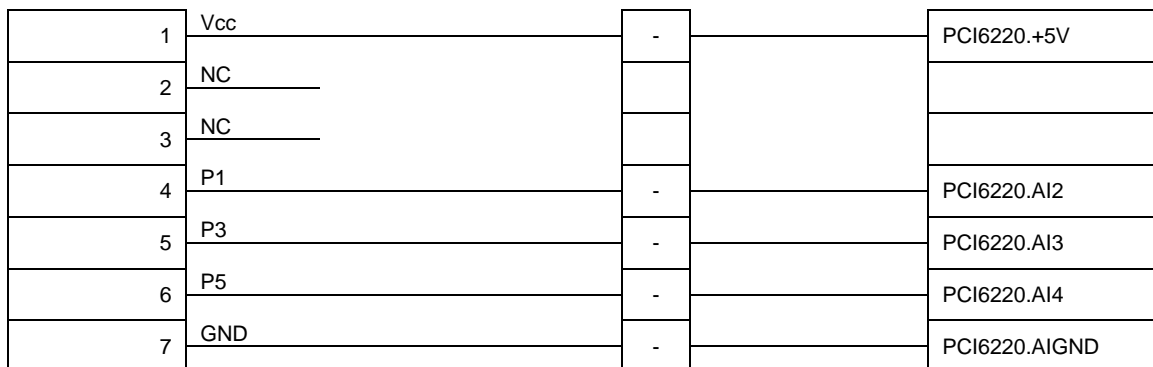
Canal 5: pines01.



Canal 6: pines02.



Canal 7: potenciómetres.



Canal 8: sense utilitzar.

Canal 9: sensor de força Schunk FTCL entrada.

1	NC	-	
2	RX	67	CH10.2
3	TX	68	CH10.3
4	NC	-	
5	SGND	69	CH10.5
6	Vcc	29	FE1.24Vdc
7	GND	22	FE1.0Vdc

Canal 10: sensor de força Schunk FTCL sortida (RS232).

1	NC	-	
2	RX	67	CH9.2
3	TX	68	CH9.3
4	NC	-	
5	SGND	69	CH9.5

NOTA: RX y TX des de l'instrument connectat, no des del Pc.

Canal 11: Control MotorX (RS232).

1	NC	-	
2	RX	60	MCDC1.RXD
3	TX	61	MCDC2.TXD
4	NC	-	
5	SGND	62	

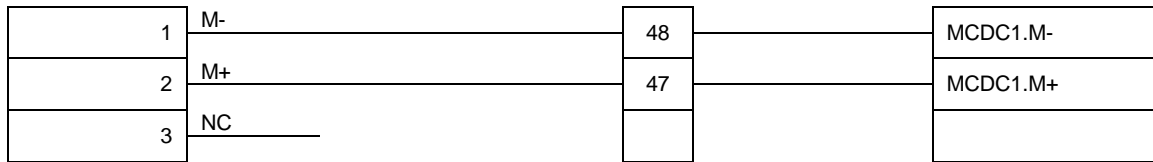
NOTA: RX y TX des de l'instrument connectat, no des del Pc.

Canal 12: Control MotorY (RS232).

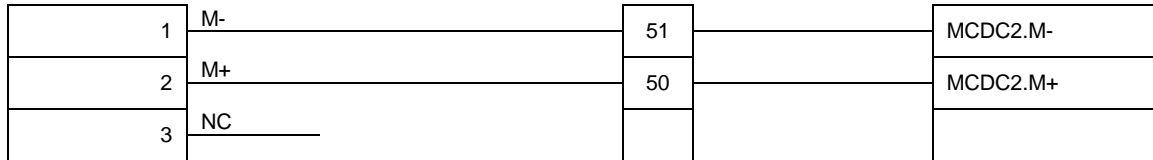
1	NC	-	
2	RX	63	MCDC2.RXD
3	TX	64	MCDC2.TXD
4	NC	-	
5	SGND	65	

NOTA: RX y TX des de l'instrument connectat, no des del Pc.

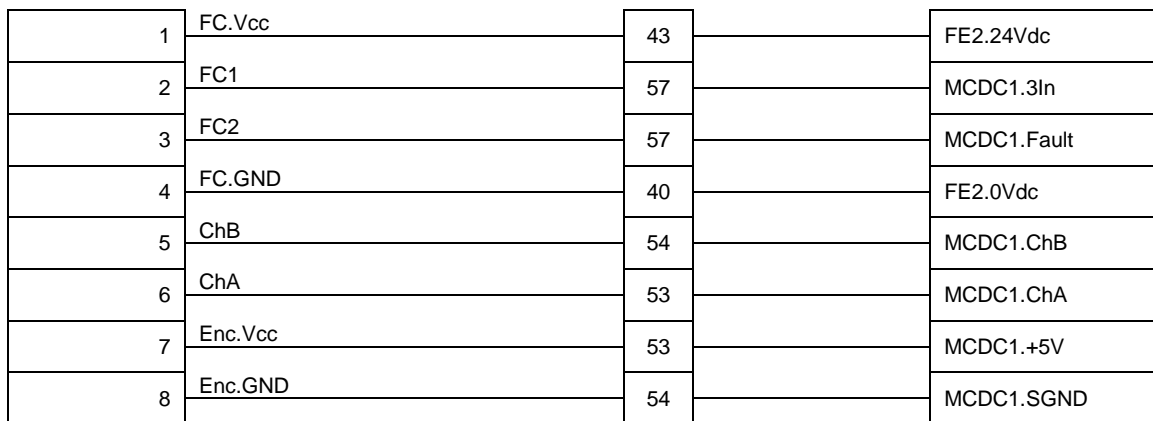
Canal 13: Motor X.



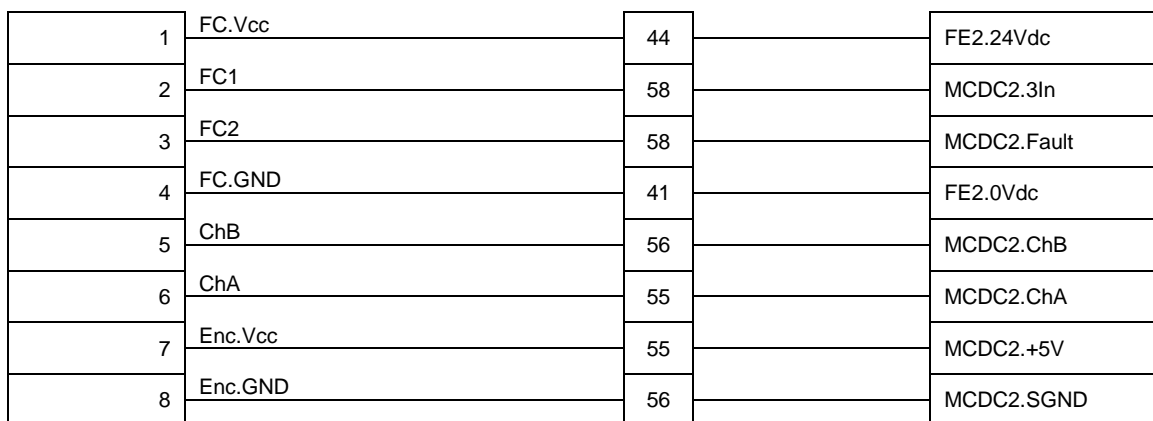
Canal 14: Motor Y.



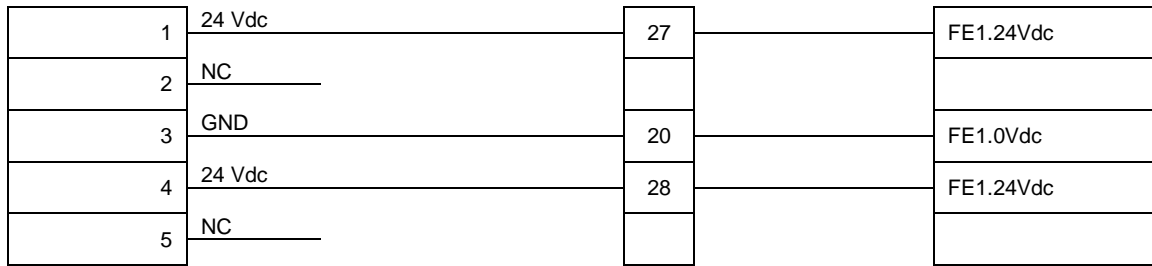
Canal 15: senyals motor X.



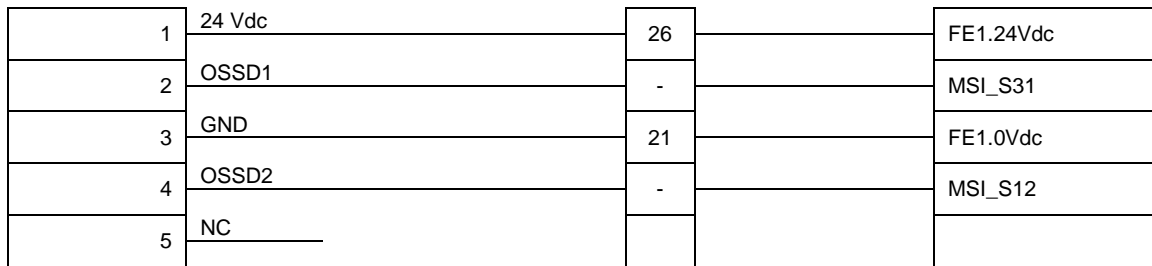
Canal 16: senyals motor Y.



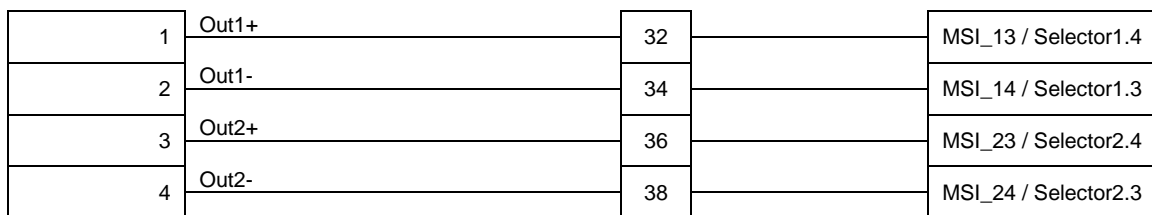
Canal 17: barrera òptica Leuze; emissor.



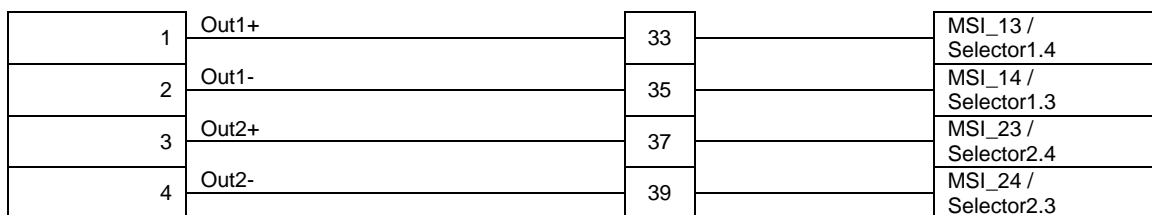
Canal 18: barrera òptica Leuze; receptor.



Canal 19: Leuze MSI out1.



Canal 20: Leuze MSI out2.



Cables

Leuze MSI a cadena de parades d'emergència dels controladors Stäubli (Connector Harting)

1	Vermell	57
2	Negre	69
3	Groc	58
4	Blanc	70

Connexió pines-tarja DAQ

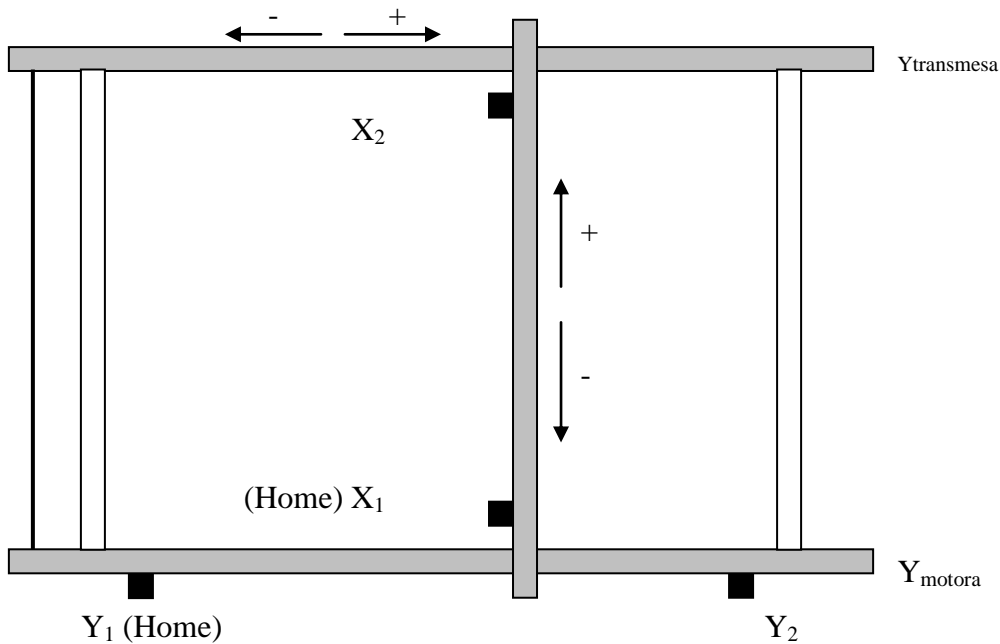
	Pin		Pinza1	Pinza2
Digital output	13	Gripper stopped	Dev1.DIG0	Dev1.DIG3
	14	Reference movement carried out	Dev1.DIG1	Dev1.DIG4
	18	Target Position Reached	Dev1.DIG2	Dev1.DIG5
Analogue output	15	Analogue Position Output	Dev1.AI0	Dev1.AI1
Digital input	11	External reset	Dev2.DIG0	Dev2.DIG4
	19	Reference movement	Dev2.DIG1	Dev2.DIG5
	20	Gripper close	Dev2.DIG2	Dev2.DIG6
	21	Gripper open	Dev2.DIG3	Dev2.DIG7
Analogue input	22	Speed	Dev2.AO0	Dev2.AO3
	23	Steps / Position	Dev2.AO1	Dev2.AO4
	24	Force	Dev2.AO2	Dev2.AO5

Dev1: NI-PCI6220

Dev2: NI-PCI6722

Annex II
Posicionador XY; notes addicionals

Robot XY



Esquema mecànic, vista superior.

Connectem la sortida dels sensors X_1 i Y_1 (*Homes*) a V_8 (In3) del controlador per fer-les servir com a *home switch*.

Connectem la sortida dels sensors X_2 i Y_2 (Límits finals) a V_4 (Fault) per fer-los servir com a finals de carrera.

Configuració dels controladors Faulhaber MCDC2805

1. Configuració de la resolució de l'encoder Maxon HEDS5540 de 500 polsos per volta:

```
ENCRES2000 [enter]      (4 · 500 polsos/volta = 2000)
EEPSAV [enter]
```

2. Configuració d'entrada per la consigna de velocitat per RS232:

```
SOR0 [enter]
EEPSAV [enter]
```

3. Configuració de pin Fault com entrada digital per detecció del final de carrera:

```
REFIN [enter]
EEPSAV [enter]
```


Barcelona, 3 de juny de 2009