



Universitat de Girona
Escola Politècnica Superior

Projecte/Treball Final de Carrera

Estudi: Enginyeria Tècn. Ind. Electrònica Ind. Pla 2002

Títol:

Sistema de manteniment de l'equilibri
per a un robot bípede

Document: 1. MEMÒRIA

Alumne: Pere Amadó Codony

Director/Tutor: Marc Carreras i Pérez / Lluís Magí i Carceller

Departament: Electrònica, Informàtica i Automàtica

Àrea: Arquitectura i Tecnologia de Computadors

Convocatòria (mes/any): Gener 2006

ÍNDEX

1. INTRODUCCIÓ.....	4
1.1. Antecedents.....	4
1.2. Objecte.....	5
1.3. Especificacions i abast.....	5
2. DESCRIPCIÓ DEL ROBOT.....	7
2.1. Estructura.....	7
2.1.1. Plataforma superior.....	8
2.1.2. Cuixa.....	10
2.1.3. Tíbia i peroné.....	11
2.1.4. Peu.....	12
2.2. Actuadors.....	12
2.2.1. Articulacions comes.....	12
2.2.2. Sistema d'estabilitat.....	15
2.3. Sensors.....	16
2.3.1. Potenciòmetres.....	16
2.3.2. Encoders.....	17
2.3.3. Finals de cursa.....	17
2.4. Dimensions i pesos.....	18
2.5. Estabilitat.....	19
2.6. Energia.....	19
2.7. Connexionat.....	20
2.8. Placa electrònica del robot.....	21
2.9. Software de control robot.....	22
3. SISTEMA D'ESTABILITZACIÓ.....	23
3.1. Disseny pla X-Y.....	23
3.2. Estructura.....	24
3.2.1. Moviment transversal.....	24
3.2.2. Moviment frontal.....	27
3.2.3. Sistema de transmissió.....	27
3.2.4. Motors.....	30
3.2.5. Lectura de posició.....	34
3.2.6. Limitadors de recorregut.....	38
3.2.7. Dimensions i pesos.....	40
3.2.8. Interconnexió dels components que formen l'estructura.....	42
3.3. Física del sistema.....	43

3.3.1. Distribució de pesos	45
3.3.2. Parts mòbils	45
3.3.3. Càlcul de l'estabilitat del robot	46
4. PLACA CONTROL DELS ACTUADORS	55
4.1. Disseny	55
4.1.1. Objectius	56
4.2. Especificacions	57
4.3. Elecció de components	58
4.3.1. FPGA	58
4.3.2. Components de potència.....	59
4.4. Comunicació entre el software i el robot.....	62
4.5. Control de velocitat	64
4.5.1. Generació de PWM	64
4.5.2. Resultats	65
4.6. Control del sentit	69
4.7. Connectors.....	69
4.8. Programació FPGA	75
4.8.1. Disseny del programa	76
4.8.2. Transmissió programa.....	78
5. SOFTWARE DE CONTROL DEL ROBOT	81
5.1. Disseny	81
5.2. Tarja PCL-812PG.....	82
5.2.1. Connexionat	85
5.3. Interfície gràfica.....	86
5.4. Control actuadors.....	87
5.5. Lectura sensors.....	87
5.5.1. Articulació frontal.....	88
5.5.2. Articulació transversal	90
5.5.3. Articulació maluc	91
5.5.4. Articulació genoll	93
5.5.5. Articulació turmell.....	94
5.6. Resultats.....	96
5.6.1. Gràfic GdLI frontal	96
5.6.2. Gràfic GdLI transversal.....	98
5.6.3. Gràfic GdLI maluc.....	99
5.6.4. Gràfic GdLI genoll.....	101
5.6.5. Gràfic GdLI turmell	104

6. RESUM DEL PRESSUPOST	107
7. CONCLUSIONS	108
8. RELACIÓ DE DOCUMENTS.....	110
9. BIBLIOGRAFIA	111

1. INTRODUCCIÓ

Aquest projecte s'ha realitzat per donar un pas més en el desenvolupament realitzat a la Universitat de Girona en el camp dels robots bípedes i alhora satisfer les necessitats de formació docent en l'àmbit de la robòtica en estudis d'enginyeria industrial en general i concretament en les especialitats d'electrònica i d'informàtica.

1.1. Antecedents

Durant els últims anys, el camp de la robòtica ha gaudit d'un gran interès per part del públic en general i això s'ha reflectit en el desenvolupament de nous robots. En concret, els robots bípedes, robots articulats que es mouen amb només les dues extremitats inferiors imitant el moviment dels humans, estan gaudint d'una gran evolució, en part, gràcies a les milionàries inversions realitzades en I+D (Investigació i Desenvolupament) per part de grans firmes comercials. En els casos més avançats podem veure robots bípedes imitant molt fidelment moviments humans. Poden caminar, ballar i fins i tot fer petits salts, tot això sense perdre l'equilibri i amb gran precisió de moviments.

Pel que fa a la Universitat de Girona, en el departament d'Electrònica, Informàtica i Automàtica s'han dissenyat i construït dues plataformes bípedes per a l'ús docent. La més evolucionada d'elles, finalitzada l'any 1999, està composta per dues cames d'alumini amb tres actuadors lineals cada una, simulant la funció del turmell, del genoll i del maluc (Salellas, 1999). Per mantenir l'equilibri es va dissenyar un grau de llibertat accionat per un motor – reductor de corrent continu, DC, i format per un contrapès que es movia seguint una trajectòria circular. El control de cada un dels motors de corrent continu es realitzava amb una placa de potència que transformava valors analògics a senyals PWM (Pulse-Width Modulation).

La principal deficiència d'aquesta plataforma bípede és la dificultat per mantenir l'equilibri. És força inestable i per tant, no és vàlida per a l'ús docent a la qual està destinada. Això es podria millorar canviant el grau de llibertat del contrapès rotatiu per un pla XY, el qual permetria situar el contrapès en qualsevol punt d'aquest pla.

A part del problemes d'equilibri, també es detecten problemes de grandària en la placa de control i en la de potència. Aquestes dimensions fan que sigui impossible imaginar que aquesta plataforma bípeda sigui autònoma, ja que per dimensions, la plataforma no pot fer-

se càrrec de la placa de control. En la mateixa placa de control es detecten deficiències de robustesa en el control dels motors, fet que obliga a la continua substitució de components.

1.2. Objecte

Com dèiem, la última versió d'aquesta plataforma bípeda va finalitzar el 1999. L'objectiu d'aquest projecte és evolucionar la plataforma bípeda per tal que en un futur se'n pugui donar un ús destinat a la docència.

Per complir amb aquest objectiu, s'han encarregat a aquest projecte una sèrie d'objectius concrets que cal assolir. Aquests es detallen a continuació:

Dissenyar dos graus de llibertat lineals en forma de pla XY per moure el pes que convingui per assegurar l'equilibri durant el moviment de la plataforma bípeda. Per realitzar aquesta tasca s'utilitzaran dos motors DC equipats amb reduccions i encoders.

Dissenyar una placa amb una FPGA que generi senyals PWM pels vuit motors disponibles, els sis de les extremitats inferiors i els dos del pla XY, que llegeixi els dos encoders dels motors del pla XY i que es comuniqui amb un PC equipat amb una tarja específica.

Dissenyar una placa de potència pel control dels motors que sigui robusta i d'unes dimensions adequades a l'ús que se li pretén donar.

Realitzar un programa per comprovar el correcte funcionament de les plaques, dels actuadors i dels sensors utilitzats en la plataforma bípeda.

1.3. Especificacions i abast

El projecte inclou el disseny teòric de cada objectiu, per poder complir els objectius es parteix d'una quantitat econòmica que limita l'elecció del material, per això, a partir de plànols i esquemes se seleccionen els components que compleixen les condicions: econòmiques, qualitatives i funcionals. Així doncs, complint les especificacions seleccionem el material i els components necessaris per poder dur a terme la seva implementació.

Una vegada seleccionat el material es procedirà a la seva compra i verificació.

Seguidament es farà el disseny de la placa de potència per tal d'integrar totes les parts que la componen, es durà a terme la creació del programa intern de la FPGA i es prosseguirà amb la seva programació.

Es realitzarà el muntatge real de tots els components, integrant-los en el conjunt de la plataforma bípeda i realitzant tot el cablejat necessari per garantir el seu correcte funcionament.

Per finalitzar, es realitzarà una comprovació de tota la plataforma amb un programa creat amb LabWindows que ens permetrà comprovar i demostrar el correcte funcionament de la plataforma bípeda i de cadascun dels seus components independentment.

2. DESCRIPCIÓ DEL ROBOT

En aquest capítol s'explica tot allò que fa referència a l'elecció del material utilitzat, als actuadors utilitzats i a l'estructura del robot. Val a dir que l'estructura i les característiques del sistema d'estabilitat les tractarem més detalladament en el següent capítol, ja que és una part que mereix una menció especial dins el nostre projecte.

2.1. Estructura

Tal com s'ha comentat en la introducció del projecte, es parteix de les extremitats inferiors creades en un projecte anterior (Salellas, 1999). El material utilitzat en la creació de les cames va ser l'alumini, concretament perfils d'alumini de dimensions 20 x 40 mm. Pel que fa a la seva longitud, cal dir que depèn de la part del robot a la que ens referim trobem una longitud o una altra, sempre amb l'objectiu de crear unes proporcions similars a les de l'ésser humà.

Cada cama està formada per tres graus de llibertat. Les cames s'uneixen per la seva part superior a través del pla XY, amb quatre cargols per cama, simulant la funció del maluc. De centre a centre, les cames, estan separades 200mm, aquesta distància ens proporciona una bona estabilitat i, a més, al no ser excessiva fa que quan aixequem una cama, l'altre no generi un moment excessiu. Tractarem amb més detall la distribució de pesos i la física del sistema al subapartat 3.3. L'estructura de cada cama intenta imitar l'estructura de la cama d'un ésser humà. Començant per la part inferior, el primer grau de llibertat el trobem al turmell, en aquest cas podem moure el peu amunt i avall, però no cap a un costat i cap a l'altre. El segon grau de llibertat és l'equivalent al genoll humà, permet moure l'equivalent a la tibia i el peroné, més el peu, endavant i endarrera. Per últim trobem l'articulació del maluc que ens permet desplaçar tota la cama endavant o endarrera.

Pel que fa a la plataforma superior, encarregada de mantenir l'equilibri del robot bípede, es substitueix l'antiga plataforma per una altra totalment nova, es passa d'un moviment rotatiu a dos moviments lineals, un de transversal i un altre de frontal. Aquests moviments es realitzen a dos nivells, el primer nivell reservat pel moviment transversal i el segon nivell dedicat al moviment frontal.

A la figura 1 podem comparar les tres versions de robot bípede que s'han fet a la Universitat de Girona.

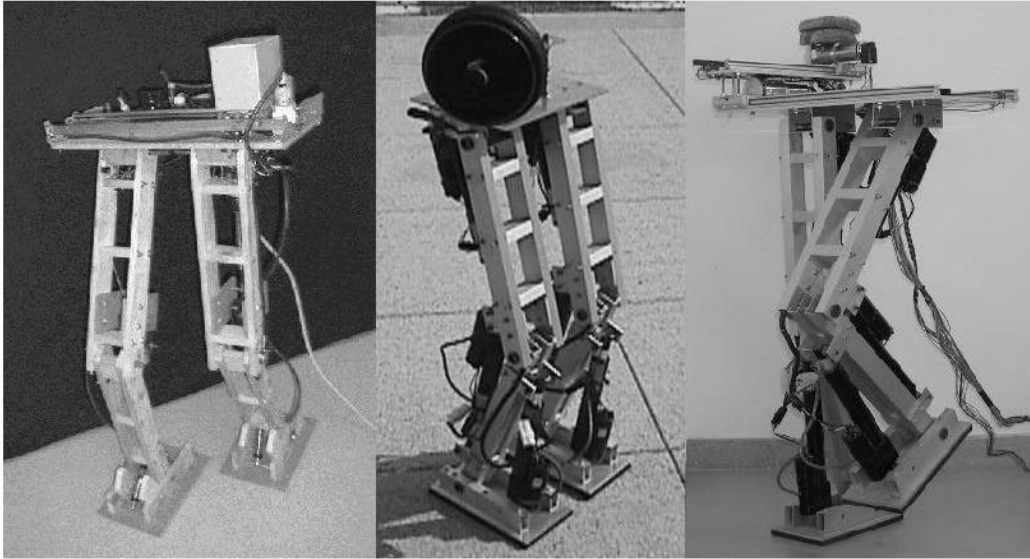


Figura 1. Evolució plataformes bípedes: 1998, 1999 i 2006

2.1.1. Plataforma superior

Mecànicament ens hem centrat en el disseny de la plataforma superior, ja que en l'anterior projecte era la part que presentava més deficiències. Es va comprovar que el moviment rotatiu no proporcionava una bona estabilitat al conjunt.

Val a dir que per dissenyar la plataforma superior ens hem inspirat amb la primera versió de robot bípede, realitzada a la Universitat de Girona l'any 1998. En aquest cas es disposava d'un contrapès que realitzava un desplaçament transversal per garantir l'equilibri, d'aquesta manera no es va aconseguir una estabilitat correcte. Evolucionant aquesta idea, s'ha cregut que col·locant un altre contrapès perpendicular al primer, és a dir, de forma frontal, podem millor l'equilibri del conjunt.

La plataforma superior està muntada sobre una placa d'alumini, i realitza dues funcions: unir les dues cames i fer de base de tot el sistema d'estabilització. Les dimensions d'aquesta

base són 500 x 250 x 5 mm. En aquesta base hi van acoblades dues guies paral·leles de desplaçament horitzontal amb els seus corresponents carros, amb les quals s'aconsegueix el moviment transversal del contrapès. A més, aquestes guies serviran per aguantar una altra base d'alumini de dimensions 350 x 100 x 5 mm que servirà com a base de muntatge de l'últim grau de llibertat, el que proporcionarà el moviment frontal al contrapès.

Pel que fa als desplaçaments transversal i frontal, venen donats pel conjunt format pel motor, el reductor, el pinyó i la cremallera de cada un dels dos graus de llibertat. A la figura 2 podem veure tota la plataforma d'estabilització.

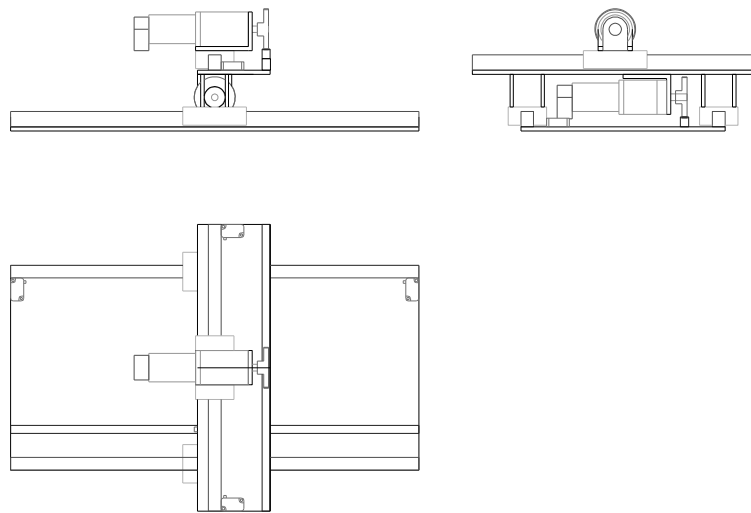


Figura 2. Plataforma superior

2.1.2. Cuixa

En quant a la cuixa, cal dir que està formada per dos perfils d'alumini de 400 mm de longitud col·locats verticalment, un està a la part exterior i l'altre a la part interior de la cama. Per unir aquestes dues peces hi ha tres perfils d'alumini de 60 mm de longitud col·locats com a travessers distribuïts equitativament al llarg de la cuixa. Finalment, tenim un perfil de 60x40x120 mm per unir la cama del robot amb la plataforma superior.

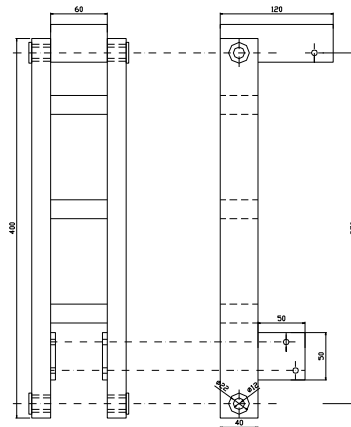


Figura 3. Estructura de la cuixa

Tal i com podem veure a la figura 3, a cada extrem de la cuixa hi ha els eixos que uniran les diferents parts de la cama, permetent un moviment rotatori. Per facilitar aquest moviment, es van col·locar unes dolles en els eixos, aconseguint així, menys fregament. La distància entre els dos eixos és de 370 mm.

2.1.3. Tíbia i peroné

Seguidament, ens trobem amb la part equivalent a la tíbia i el peroné. Aquesta part va muntada a la part interior dels perfils de la cuixa, buscant la similitud de gruixos que té la cama humana. Els perfils estan col·locats seguint la mateixa lògica que en la cuixa. Disposem de dos travessers de longitud 20 mm que uneixen els perfils, estan col·locats un a la part superior i l'altre a la part inferior del conjunt tíbia peroné. Igual que en el cas de la cuixa, la longitud dels perfils és de 400 mm, la separació entre eixos de 370 mm i es segueixen utilitzant dolles per facilitar el moviment. A la figura 4 en veiem la representació gràfica.

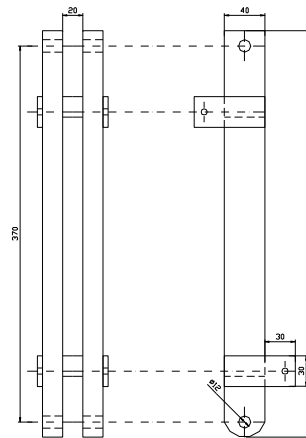


Figura 4. Estructura del conjunt tíbia-peroné

2.1.4. Peu

El peu del robot està format per una placa d'alumini de 5 mm de gruix i de dimensions 240 x 130 mm. La part inferior d'aquesta placa, la que té contacte amb el terra, s'ha recobert amb un material esponjós perquè el contacte amb el terra no sigui tan brusc. A més, disposem de dos perfils de 200 mm de longitud, separats 60 mm entre ells que serviran per unir el peu amb la resta de la cama. Aquesta part també inclou dues dolles a la part posterior del peu per facilitar la rotació de l'eix.

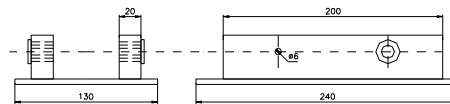


Figura 5. Estructura del peu

A la cuixa i al conjunt tibia-peroné, hem pogut observar que les estructures porten incorporades unes petites plaques d'alumini per fixar-hi els actuadors lineals. L'actuador lineal del turmell es fixa a l'estructura del peu gràcies al forat de diàmetre 6 mm que veiem a la figura 5.

2.2. Actuadors

Disposem de dos tipus d'actuadors: els motors lineals de les cames i els motors rotatius de la plataforma superior. Per evitar problemes i facilitar l'alimentació s'han escollit els motors rotatius de 12v DC, ja que els motors lineals que ja incorpora el robot, també ho són.

2.2.1. Articulacions cames

Els actuadors lineals són motors elèctrics que distribueix la firma comercial DANAHER MOTION. Aquesta firma comercial ofereix varies gammes d'actuadors lineals, es van escollir els de la gamma LA1, ja que són els que s'ajusten més a les nostres necessitats de mides, recorreguts i forces. Com ja hem dit, els motors dels actuadors lineals són de corrent continu i s'alimenten a 12v. El fabricant ens ofereix la possibilitat que l'actuador incorpori un

potenciòmetre o dos finals de carrera. Es va optar per la opció d'incorporar potenciòmetre, ja que, per nosaltres, és imprescindible conèixer la posició de l'actuador en tot moment. Donat que no disposen de interruptors final de carrera, s'ha de limitar el recorregut per software utilitzant els potenciòmetres.

Dins la gamma LA1 s'escullen els de 100 mm de recorregut i 110 N de força per tots els graus de llibertat. A continuació mostrarem les característiques d'aquests motors.

Dispositiu anti-rotació per extensió del tub en els modes SP.

Protecció per a la utilització tant en espais interiors com exteriors.

Proporcionen una forma de treball constant i repetitiva.

Dimensions comparables a les dels cilindres hidràulics o pneumàtics.

Dissenyats per l'ús sota condicions de treball difícils.

Connexió per connector.

Construcció molt compacte.

Motor amb protecció tèrmica.

Resistent a altes temperatures i grans variacions de tensió d'entrada.

No necessiten manteniment ni reglatges.

Dissenyats per aplicacions d'ús intermitent.

A la taula 1 podem veure les especificacions principals dels actuadors lineals seleccionats.

Model SP12-09A4	
Voltatge (V DC)	12
Intensitat màxima (A)	5,6
Càrrega dinàmica màxima (N)	110
Càrrega estàtica màxima (N)	1300
Carrera útil (mm)	100
Temperatura funcionament (°C)	De -25 a +65
Pes (g)	630

Taula 1. Especificacions principals actuadors lineals

Les característiques dels actuadors són perfectament compatibles amb l'ús que se'ls hi donarà.

A la figura 6, podem veure un parell de gràfics: el primer relaciona la càrrega amb la velocitat i el segon amb la intensitat.

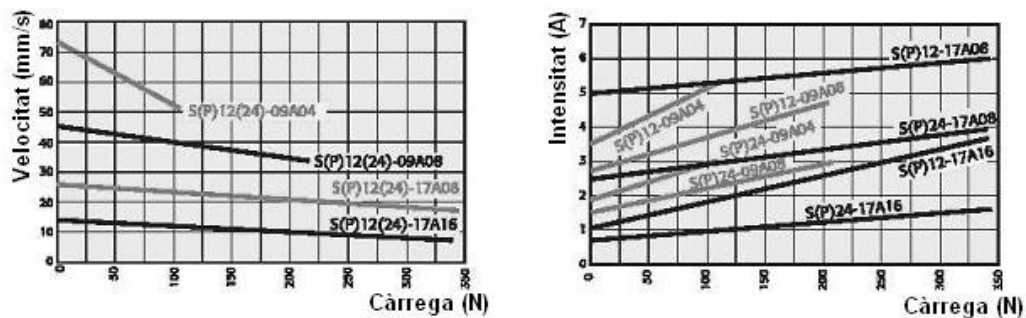


Figura 6. Velocitat - Càrrega i Intensitat - Càrrega

Si ens fixem amb el model escollit, l'SP12-09A4, observem que podem aconseguir velocitats de 70mm/s sense càrrega i de 50mm/s amb uns 10Kg. Pel que fa al consum, a càrrega

màxim estarem consumint 5,6A, ho haurem de tenir en compte en l'elecció dels components de la placa de control.

A la figura 7, veiem un plànol de l'actuador que ens serveix per localitzar el connector de l'alimentació de l'actuador i el de l'alimentació i lectura del potenciòmetre.

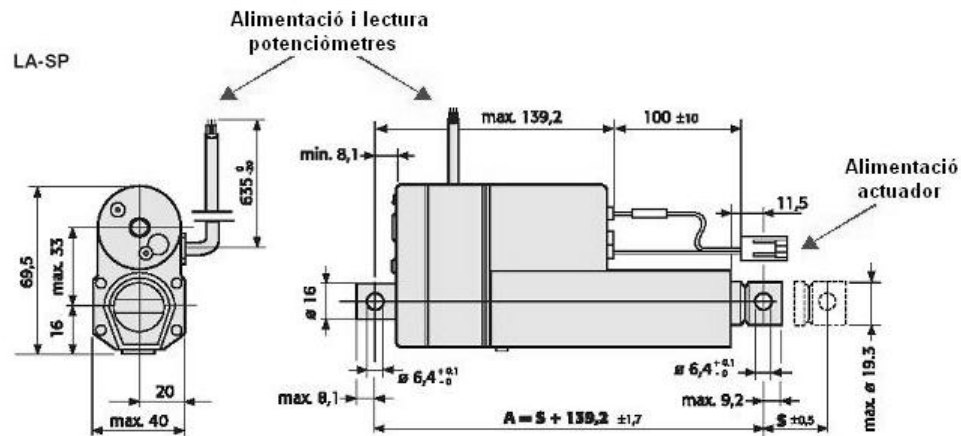


Figura 7. Plànol actuador lineal i situació alimentacions

Finalment, a la figura 8, observem l'aspecte extern de l'actuador.



Figura 8. Actuador lineal

2.2.2. Sistema d'estabilitat

En aquest cas, hem seleccionat els motors partint d'unes característiques molt específiques: dimensions reduïdes, prou parell per desplaçar tot el sistema d'estabilitat, velocitat adequada, control de posició i alimentació de 12v DC. Partint d'aquestes especificacions es van buscar varies opcions dins de diferents firmes comercials.

Finalment, per proporcionar moviment al sistema d'estabilitat s'han seleccionat dos motors idèntics de 12v DC, que acoblats a un reductor de relació 32:1, és a dir que per cada 32 voltes que fa l'eix del motor, l'eix del reductor en dóna 1, proporcionen una velocitat en càrrega de 123 rpm amb un parell nominal de 1.2 Nm i un corrent nominal de 2.3 A. Gràcies a la figura 9 podem veure l'aspecte extern del conjunt motor-reductor.



Figura 9. Fotografia del conjunt motor-reductor

2.3. Sensors

En aquest robot utilitzem tres tipus de sensors: els potenciòmetres dels actuadors lineals, els encoders dels motor-reductors i els finals de cursa de la plataforma superior.

2.3.1. Potenciòmetres

Com ja sabem, cada actuator lineal porta acoblat un potenciòmetre. Aquest ens permet conèixer la posició de l'actuator en qualsevol moment.

La placa de control els alimenta a 5v DC i envia el valor de referència obtingut per cada potenciòmetre cap al PC, que al disposar d'una tarja d'adquisició de dades (PCL-812PG), captura els valors de referència i els converteix en valors digitals.

L'esquema elèctric de cada potenciòmetre el veiem a la figura 10:

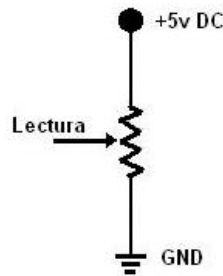


Figura 10. Esquema elèctric del potenciòmetre

Com podem veure, cada potenciòmetre té tres terminals, dos per l'alimentació i un tercer per llegir el seu valor analògic amb la tarja del PC.

Tal i com s'explica en el capítol 5, amb el software de control, partint del valor digital de cada articulació trobarem el seu equivalent amb angle.

2.3.2. Encoders

El control de posició del sistema d'estabilitat es fa amb encoders. Aquests, van acoblats a l'eix del motor i generen 500 polsos per cada volta de l'eix del motor, per tant, en la nostre aplicació no ens hem de preocupar per la precisió d'aquest posicionament. S'ha escollit el model HEDS 5540 de la marca HP. Tractarem detalladament el seu funcionament i les seves característiques en el subapartat 3.2.5.

2.3.3. Finals de cursa

Per seguretat, s'ha col·locat un final de cursa a cada extrem de les guies encarregades de realitzar els moviments frontal i transversal. En cas que s'arribi a l'extrem, el final de cursa actuarà interrompent l'alimentació de l'actuador. S'ha escollit el model T0037 de la firma comercial MARQUARDT. Els detalls els explicarem al subapartat 3.2.6.

2.4. Dimensions i pesos

Una vegada comentada l'estructura de cada una de les parts que formen el robot i els actuadors escollits, a la taula 2 es mostren, detalladament, les dimensions i els pesos de les cames del robot, i a la taula 3 el pes total del grau de llibertat transversal i el grau de llibertat frontal. Al següent capítol ens centrarem amb les dimensions i els pesos de tot el sistema d'estabilització. Amb les dades de les taules 2 i 3 podrem determinar el pes total del robot bípede.

Estructura	Dimensions (mm)	Pes (g)	Quantitat (1 cama)	Pes total (g)
Peu	240 x 130	685	1	685
Tíbia – peroné	400	700	1	700
Cuixa	400	745	1	745
Motor lineal	de 150 a 254	820	3	2.460
Pes total d'una cama				4.590

Taula 2. Dimensions i pesos cames

Estructura	Dimensions (mm)	Pes (g)	Quantitat	Pes total (g)
Moviment transversal	500 x 250 x 69	4.797	1	4.797
Moviment frontal	350 x 100 x 76,5	2.049	1	2.049
Contrapès		5.000	1	5.000
Pes total del sistema d'estabilització				11.846

Taula 3. Dimensions i pesos estructura estabilització

Així doncs el pes total del robot és de 21.026g.

2.5. Estabilitat

És possible que no ens hàgim plantejat mai el moviment de la projecció del centre de gravetat dels humans en el pla on es recolzen el peus quan caminem. És un procés molt complex que desenvolupem intuïtivament, però en el cas dels robots bípedes no és així, els creadors hem de tenir en compte cadascun dels moviments i les condicions que cal complir per tal que el robot mantingui l'equilibri.

La condició imprescindible perquè una estructura qualsevol es pugui mantenir en equilibri és que el seu centre de masses estigui situat sobre la superfície de contacte amb el terra.

En el nostre cas, per garantir l'equilibri del robot quan té una cama aixecada, la projecció del centre de masses haurà de coincidir amb la superfície del peu que toca al terra. Com més gran sigui la base del peu més fàcil és que estigui en posició d'equilibri. Quan no està fent cap pas i per tant, els dos peus toquen al terra, el centre de masses estarà situat entre els dos peus.

Explicarem més detalladament les condicions d'estabilitat en l'apartat 3.3.Física del sistema.

2.6. Energia

Tota l'energia emprada en el funcionament del robot s'obté d'una font d'alimentació que disposa d'una sortida regulable que va dels 1,3vDC als 13,8vDC, es tracte del model FARV-15/20A. Així assegurem una alimentació constant a 12v DC. Cal dir que podem alimentar la placa amb qualsevol font que garanteixi les mateixes condicions tècniques i de qualitat que la utilitzada. A través de la placa de control alimentem els motors a 12v DC i els potenciòmetres, els encoders i la FPGA a 5v DC. Cal dir que la placa de control conté un regulador de voltatge L7805, que quan li apliquem 12v DC ens en proporciona 5v DC per alimentar els components que ho requereixen.

A la figura 11 podem veure la imatge de la font d'alimentació utilitzada.

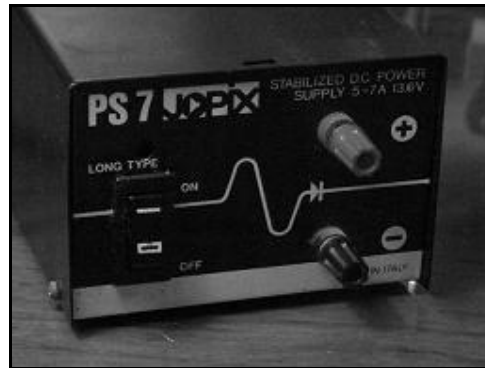


Figura 11. Font d'alimentació

2.7. Connexionat

Per explicar les connexions entre totes les parts: robot, placa de control, PC i alimentació utilitzarem l'esquema de la figura 12, visualment ho entendrem molt millor.

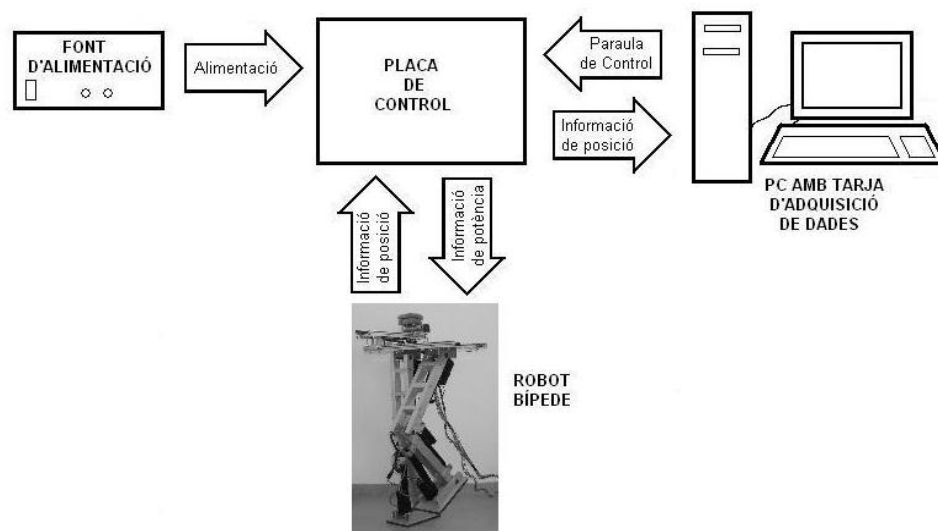


Figura 12. Connexions entre les diferents parts

Així doncs, a partir de la font d'alimentació, alimentem la placa de control. A la placa hi ha la FPGA i tots els elements de potència i de control interconnexionats convenientment. De la placa de control surt cap al robot bípede el que anomenem "Informació de potència", que

està composta per la consigna d'alimentació dels motor (de M0 a M7), l'alimentació dels sis potenciòmetres, l'alimentació dels dos encoders i la connexió dels quatre finals de cursa.

A través de la placa de control, el PC rep la "Informació de posició" del robot, que no és res més que els valors que llegim dels potenciòmetres i dels encoders, mentre que la "Informació de posició" dels encoders és tractada per la FPGA avanç de ser enviada cap a la tarja, la "Informació de posició" dels potenciòmetres no rep cap tractament a la placa de control, per tant, podria anar directament a la tarja del PC, però per una qüestió d'ordre es concentren tots els connectors que venen del robot a la placa de control.

El PC envia, sempre que ho sol·liciti l'usuari, la "Paraula de Control" cap a la placa de control que conté tota la "Informació de Potència" que cal aplicar al robot

2.8. Placa electrònica del robot

Inicialment, en els objectius del projecte, es va estipular que es dissenyaria una placa amb una FPGA per generar senyals PWM pels 8 motors, que llegís els 2 encoders i que es comunicés amb el PC i una altre placa, la de potència, que serviria per controlar els motors.

Evolucionant aquesta idea s'ha cregut més convenient dissenyar una única placa que contingui les dues parts. Així doncs, hem aconseguit crear una única placa de dimensions molt ajustades, aproximadament 205x138mm, que incorpora la part de potència i la part de control del robot.

Al projecte anterior (Salellas, 1999), es disposava de dues plaques: la d'interfasament amb unes dimensions de 160x100mm i la de potència, que governava set motors i mesurava 200x155mm. Per tant, hem reduït en aproximadament el 60% l'espai que ocupaven les plaques de control. Aquest fet provoca que no sigui difícil imaginar aquesta nova placa incorporada en el robot fent un pas més cap a la seva autonomia total.

A més, hem guanyat amb fiabilitat buscant alternatives als elements que causaven deficiències.

A la figura 13 mostrem l'aparença física de l'actual placa de control.

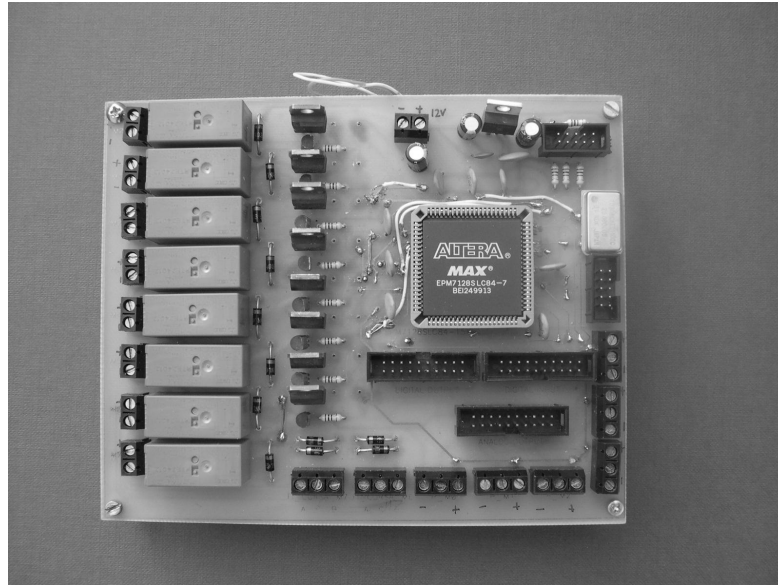


Figura 13. Placa de control

Aquesta part, juntament amb el sistema d'estabilitat, són les parts que més hem fet evolucionar respecte l'anterior projecte, per això la tractarem detalladament en el capítol 4.

2.9. Software de control robot

Ens hem centrat en la realització d'un software senzill que permeti comprovar el correcte funcionament de cada un dels actuadors i dels seus sensors independentment.

Així doncs, s'ha optat per realitzar el software de control amb el programa LabWindows, ja que és senzill i molt gràfic, i això és el que ens convenia. En el capítol 5 exposarem tots els detalls del software de control.

3. SISTEMA D'ESTABILITZACIÓ

Aquesta és una de les parts més importants del projecte, mecànicament parlant no té res a veure amb l'anterior sistema d'estabilització. Entre altres avantatges, el sistema actual ens permet col·locar el pes per estabilitzar el robot en qualsevol punt del pla X-Y, mentre que l'anterior només podia seguir una única trajectòria circular.

3.1. Disseny pla X-Y

Arribar al disseny definitiu del pla X-Y no ha sigut gens fàcil. La premissa inicial era que es necessitava crear una solució als problemes d'estabilitat del robot. Per fer-ho es va pensar en un pla X-Y. Hi ha altres sistemes, potser més efectius, basats en pèndols i sistemes complexos que per preu i complexitat hem de descartar.

Inicialment es van buscar catàlegs d'eixos accionats per corretges dentades, aquests sistemes ofereixen diferents longituds, la possibilitat d'acoblar-hi un motor entre un gran ventall d'opcions disponibles i nombrosos accessoris per ser muntats en sistemes de varis eixos com el nostre. Vam posar-nos en contacte amb comercials d'empreses distribuïdores d'aquests productes, i després d'avaluar el nostre objectiu, vam arribar a la conclusió que aquest tipus d'accionaments no s'acabaven d'adaptar a les nostres necessitats. Els principals inconvenients són: preus desorbitats pel nostre pressupost i el poc aprofitament de la carrera útil de l'eix, ja que el patí no pot arribar a l'extrem de l'eix per culpa de la seva estructura. Aquests sistemes tan robustos estan dissenyats per aplicacions industrials en que la velocitat, la precisió, la vida útil, i altres aspectes relacionats amb la producció, primen davant del pes i del tamany, en canvi, en aquest projecte són dues característiques molt importants.

La següent opció ha sigut mirar quines possibilitats hi ha de fer un muntatge a la carta a partir de components que podem trobar en catàlegs d'empreses que es dediquen a la comercialització de tot tipus de productes mecànics i electrònics de diferents marques. Així doncs, es comença la recerca en un gran catàleg amb més de 60.000 productes de primeres marques.

Després de varis plànols i de temptejar diferents configuracions es troba la solució desitjada.

3.2. Estructura

En aquest apartat ens centrarem en explicar detalladament com estan configurades les diferents parts de la plataforma superior, que té la funció de donar estabilitat al robot. Com hem comentat anteriorment tota la plataforma superior va muntada sobre una planxa d'alumini de dimensions 500x250x5mm, la base X. La part que proporciona el moviment frontal va muntada a un segon nivell, en una planxa d'alumini de dimensions 350x100x5mm. A la figura 14 aclarim visualment aquesta estructura.

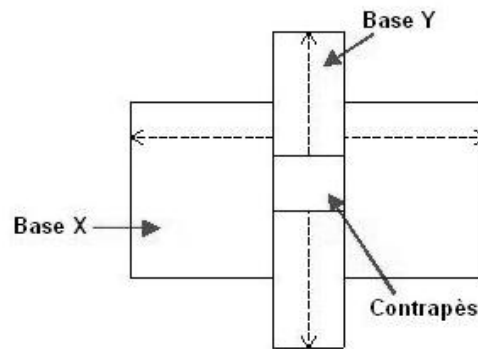


Figura 14. Disseny del pla XY

3.2.1. Moviment transversal

Sobre la plataforma anomenada base X s'hi acoblen tots els elements encarregats de proporcionar el moviment transversal. Tot el desplaçament es realitza gràcies a dues guies de desplaçament lineal amb un patí cada una. Les dues guies estan col·locades paral·lelament i en direcció perpendicular al desplaçament del robot, una a cada extrem de la base.

El conjunt guies més patí que s'encarrega del moviment transversal i el que s'encarrega del moviment frontal, del qual parlem en el següent subapartat, tenen les següents característiques comunes.

Tant els patins com les guies són productes de la marca HOERBIGER ORIGA, es tracte d'una prestigiosa firma internacional que produeix equipaments pneumàtics i electrònics amb una gran varietat d'accessoris i muntatges possibles. A nosaltres ens ha interessat, concretament, la seva gamma de guies linears.

Pel que fa als patins, al utilitzar components d'alumini lleuger en les seves estructures, es redueixen les masses en moviment i les velocitats de recorregut augmenten. Estan dissenyats per transportar càrregues de pes mig fins a 10m/s sobre rodaments d'agulla, que circulen per pistes d'acer embotides en el perfil d'alumini, aquestes pistes són el que anomenem guies, les trobem disponibles amb tres amplades de patí i diverses longituds de perfil. Per la nostra aplicació seleccionem els patins amb menys amplada d'entre els disponibles, els de 15 mm. Aquests ens proporcionen garanties tècniques suficients, veiem les característiques tècniques a les taules 4 i 5.

Càrrega (N)		Massa (g)	
Dinàmica	Estàtica	Rail/metre	Patí
5.900	7.200	800	300

Taula 4. Característiques tècniques patins

Velocitat màxima (m/s)	10
Acceleració / desacceleració màxima (m/s²)	40
Coefficient de fregament	0,001
Temperatura de funcionament (°C)	De -10 a +80

Taula 5. Característiques tècniques patins

Cada patí porta incorporats quatre forats roscats de mètric 5 que ens permeten fixar-hi l'element que ens convingui. En el cas dels patins encarregats del moviment transversal, aprofitem aquests forats roscats per collar-hi quatre barres roscades de 65 mm de longitud que faran de pilars per la base del moviment frontal.

Les guies venen foradades de fàbrica verticalment cada 60 mm amb forats passants de 4,5 mm de diàmetre i amb forats no passants de diàmetre 8 mm que serveixen per tapar el cap

del cargol que colla la guia. Aquests forats ens permetran fixar les guies fàcilment a la base corresponent. A la figura 15 es mostra una perspectiva del conjunt guia més patí.

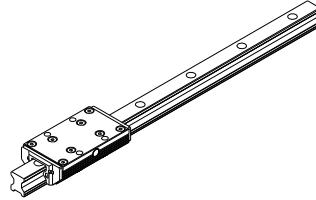


Figura 15. Guia més patí

Pel que fa a les dimensions de les guies podem escollir entre diferents longituds segons ens interressi, en aquest cas, pel moviment transversal, les hem hagut d'escollir de 600mm de longitud, i posteriorment les hem reduït a 500mm perquè s'adaptin perfectament a la base X. Veiem com van col·locades aquestes dues guies a la figura 16.

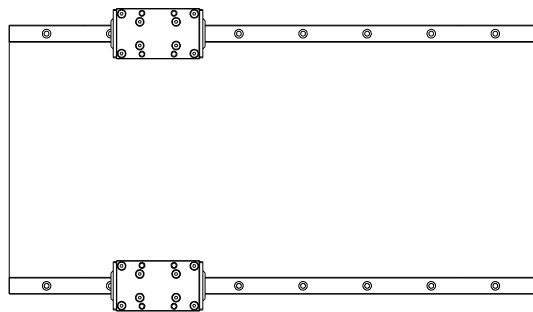


Figura 16. Guies i patins base X

3.2.2. Moviment frontal

Aquest grau de llibertat es confia a una altra guia amb les mateixes característiques que les dues del moviment transversal, però en aquest cas la longitud és inferior, concretament és de 350 mm, suficient per establir aquest moviment. Aquesta guia va col·locada en la mateixa direcció que el desplaçament del robot, i a un extrem de la segona base. Com ja hem comentat, aquesta base, la base Y, va fixada sobre els patins de les guies del moviment transversal, així doncs, a més de la seva funció, tota aquesta estructura fa la funció de contrapès del moviment transversal. A la figura 17 observem com va col·locada la guia i el patí de la base Y.

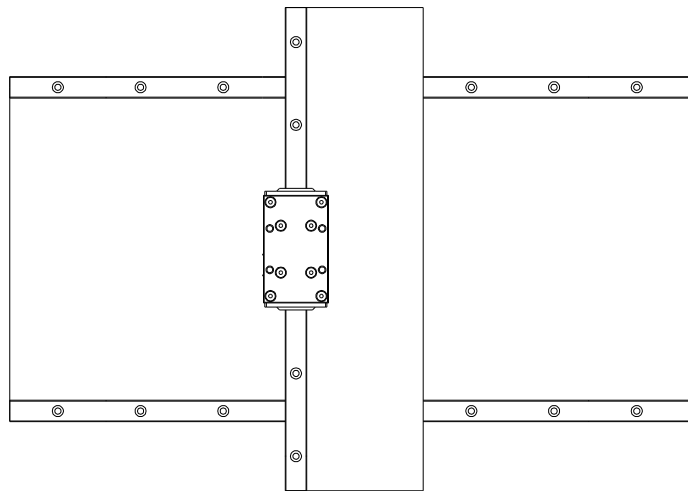


Figura 17. Vista superior estructura estabilització

3.2.3. Sistema de transmissió

Aquest ha estat un altre apartat delicat en la construcció del sistema d'estabilització ja que és imprescindible pel bon funcionament del robot. Després d'estudiar varies opcions s'ha decidit que el més segur i robust és crear un conjunt cremallera - pinyó. S'ha creat un conjunt pel desplaçament transversal i un altre pel frontal amb gairebé les mateixes característiques, la única que canvia és la longitud del desplaçament.

Dins l'oferta que se'ns ofereix de cremalleres mètriques dentades n'hi ha de diferents materials, hem escollit les d'acer, sobretot per la seva resistència al desgast. Aquestes

cremalleres es poden utilitzar amb diferents engranatges dentats. També disposem de la possibilitat d'escollir entre diferents mòduls, que repercuteix en la mida de les dents, s'han escollit de mòdul 1 ja que és el més petit que trobem en aquest model de cremallera i és correcte per la nostra aplicació. Les característiques tècniques de la cremallera dentada les mostrem a continuació, a la taula 6.

Material	S45C (acer)
Resistència mínima a la tensió (N/mm²)	569
Punt mínim d'elasticitat (N·mm²)	343
Allargament (%)	20+
Acabat	Òxid negre
Longitud (mm)	500

Taula 6. Característiques tècniques cremallera

Pel que fa a la longitud, és correcte pel moviment transversal (500mm), mentre que l'haurem de tallar per adaptar-la al moviment frontal (350mm). L'aspecte real de la cremallera el veiem a la figura 18.



Figura 18. Cremallera acer

Seguidament ha calgut buscar un pinyó que s'adaptés a la cremallera i a l'eix del motor. Hem buscat engranatges mètrics de mòdul 1 amb un diàmetre interior de 8 mm, gràcies al mòdul s'adapten perfectament a la cremallera i gràcies al seu diàmetre interior s'adapten perfectament al reductor. A més, aquests engranatges són cilíndrics de dentadura recta modelats per injecció i proporcionen tota una gamma de relacions.

Tenen un angle de pressió de 20° . A la figura 19 veiem l'aspecte d'aquest tipus de pinyó.

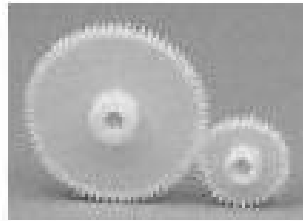


Figura 19. Pinyons duracon

Variant la mida del diàmetre del pinyó podem modificar el comportament del sistema, com més petit escollim el pinyó, a menys velocitat es desplaçarà el conjunt, però amb més força. I si escollim un pinyó amb un diàmetre superior obtenim més velocitat lineal i menys força. Per tant, hem de buscar un pinyó que satisfaci les nostres necessitats de velocitat i de força.

Finalment, després d'estudiar varies possibilitats, seleccionem un engranatge de 48 dents que acoblat al conjunt motor-reductor seleccionat provoca que a velocitat nominal tardem uns 2 segons a fer el recorregut transversal i aproximadament 1,1 segons a fer el frontal. Són uns temps més que acceptables en la nostra aplicació. El pinyó té les característiques tècniques de la taula 7.

Material	Duracon
Resistència mínima a la tensió (N/mm²)	94
Allargament (%)	entre 20 i 60
Duresa (Rockwell R)	entre 110 i 120
Gruix de la cara (mm)	entre 3 i 4

Taula 7. Característiques tècniques pinyons

Així doncs queda solucionat el problema de la transmissió.

3.2.4. Motors

Tal i com hem comentat anteriorment hem escollit motors de 12v DC equipats amb un reductor mecànic i un encoder digital. El conjunt motor-reductor és de la firma comercial FAULHABER i estan fets a alemanya.

Acoblant un reductor a l'eix del motor aconseguim augmentar el parell que ens proporciona el motor i disminuir-ne la velocitat. Passem de tenir un parell de 50 mN·m a l'eix del motor a tenir-ne 1.2 N·m a l'eix del reductor, que és on hi fixarem el nostre engranatge. Pel que fa a la velocitat passem de 3940 rpm a l'eix del motor a 123 rpm a l'eix del reductor. A la taula 8 mostrem les característiques tècniques d'aquest motor.

Tensió d'alimentació U_a (V)	12
Velocitat a corrent nominal (rpm)	3940
Parell a corrent nominal I_n (mN·m)	50
Corrent permanent màxima I_n (mA)	2260
Velocitat en buit a $U_a \pm 10\%$ (rpm)	5400
Corrent en buit a $U_a \pm 50\%$ (mA)	125
Parell d'arrencada a U_a (mN·m)	185
Corrent d'arrencada a U_a (mA)	8955
Constant del parell (mN·m/A)	20,9
Constant de velocitat (rpm/V)	456
Pendent velocitat/parell (rpm/mN·m)	29,2
Velocitat límit (rpm)	5000

Taula 8. Característiques tècniques actuador rotatiu (I)

A la taula 9 seguim amb més característiques tècniques de l'actuador rotatiu.

Potència útil màxima a U_a (W)	26
Rendiment màxim (%)	79
Constant de temps electromecànica (ms)	16
Inèrcia ($\text{g}\cdot\text{cm}^2$)	52
Resistència en borns (Ω)	1,34
Resistència tèrmica carcassa/ambient K/W	1,5

Taula 9. Característiques tècniques actuador rotatiu (II)

Hem pogut veure la fulla de característiques dels motors del sistema d'estabilització, ara veurem la imatge del motor a la figura 20.



Figura 20. Motor rotatiu

Pel que fa al reductor, seguidament veiem les seves característiques a la taula 10.

Relació de reducció	32
Número d'etapes	2
Rendiment (%)	75
Velocitat en buit (rpm)	169
Velocitat en càrrega (rpm)	123
Parell nominal (N·m)	1,2
Corrent nominal (A)	2,3

Taula 10. Característiques tècniques reductor

A la figura 21, mostrem, gràficament, el comportament del conjunt motor-reductor i, a més, hi localitzem: la velocitat amb càrrega, el parell nominal i la intensitat nominal.

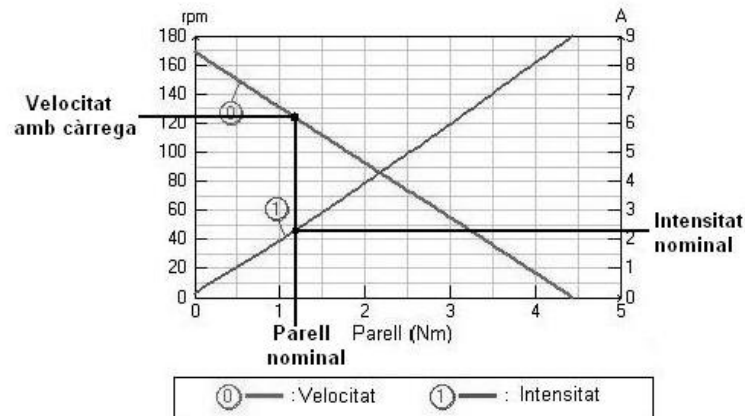


Figura 21. Comportament conjunt motor-reductor

Veiem com la velocitat i el parell són inversament proporcionals, per tant, teòricament, a màxima velocitat, 170rpm, no disposaríem de parell. I a mesura que la velocitat disminueix, mantenint l'alimentació constant, el parell augmenta fins a obtenir un parell màxim teòric de gairebé 4.5N·m. Pel que fa a la relació entre el parell i la intensitat, és proporcional, com més parell, més consum.

I tot seguit mostrem les generalitats del conjunt motor-reductor a la taula 11.

Tensió d'alimentació U_a (V)	12
Commutació	Grafit
Imant	ALNICO
Tipus de reductor	Planetari
Coixinets	Rodaments de boles
Material dels pinyons etapa d'entrada	Delrin
Material dels pinyons etapa de sortida	Metàl·lics
Càrrega axial màxima (N)	150
Joc axial màxim (mm)	0,3
Càrrega radial màxima (N)	250
A 20mm de la brida joc radial (mm)	0
A 25mm de la brida força inserció (N)	150
Joc angular en buit (°)	1
Joc angular en càrrega (°)	1,5
Temperatura ambient mínima de funcionament (°C)	0
Temperatura ambient màxima de funcionament (°C)	60
Pes mínim (g)	715

Taula 11. Generalitats conjunt motor-reductor

Finalment veurem un plànol del conjunt motor-reductor a la figura 22.

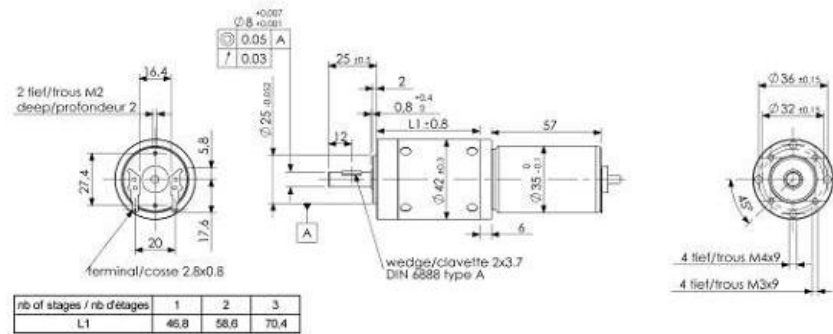


Figura 22. Plànol conjunt motor-reductor

Per tenir el conjunt complet ens falta acoblar-hi l'encoder que ens servirà per realitzar la lectura de posició del sistema.

3.2.5. Lectura de posició

Com ja hem dit, per fer la lectura de posició utilitzarem encoders de la firma HP, en concret el model HEDS 5540. Aquests encoders òptics incrementals de 3 canals (A, B i índex), són accessoris imprescindibles per la lectura de la velocitat o posició en un servosistema. Ens el subministren montats sobre l'eix del motor amb un cable de connexió de 1 m.

Els encoders són transductors rotatius que transformen un moviment angular en una sèrie d'impulsos digitals. Aquests impulsos generats es poden utilitzar per controlar els desplaçaments de tipus angular o de tipus lineal si s'associen a cremalleres, com és el nostre cas. Les aplicacions principals d'aquests transductors les trobem en màquines eina d'elaboració de materials, en robots, en sistemes de motors i en aparells de medicació i control.

A la figura 23 podem veure un esquema de l'interior de l'encoder.

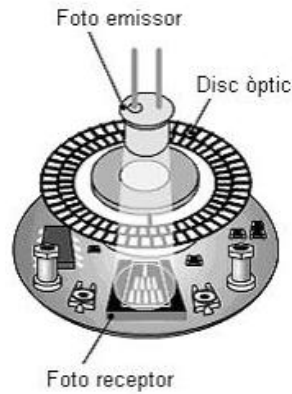


Figura 23. Esquema intern encoder

L'encoder incremental proporciona dues formes d'ona quadrades i desfasades 90° elèctrics entre si, el "canal A" i el "canal B". Amb la lectura d'un sol canal es disposa de la informació corresponent a la velocitat de rotació, mentre que si es capta també la senyal "B" és possible determinar el sentit de rotació en base a la seqüència de dades que produeixen les dues senyals. A més, tenim disponible el canal índex, que proporciona la posició absoluta de zero de l'eix de l'encoder, no utilitzarem aquest canal ja que la informació que proporciona, per nosaltres no és rellevant.

A la figura 24 representem els senyals incrementals de l'encoder en el disc òptic i respecte el temps. Aquest senyal es presenta amb una forma de pols quadrat amb fase i amplitud centrat al canal A.

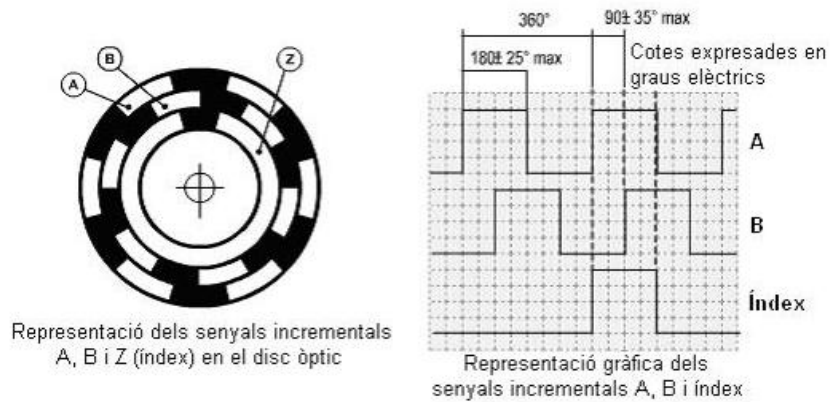


Figura 24. Representació senyals encoder

La precisió d'un encoder incremental depèn de factors mecànics i elèctrics entre els quals: l'error de divisió de l'excentricitat del disc, el dels rodaments, el produït per la electrònica de lectura i imprecisions de tipus òptic.

La unitat de mesura per definir la precisió d'un encoder és el grau elèctric, aquest determina la divisió d'un impuls generat per l'encoder. En efecte, els 360° elèctrics corresponen a la rotació mecànica de l'eix necessària per fer que es realitzi un cicle complet del senyal de sortida.

Interpretarem el senyal de sortida donat per l'encoder amb la FPGA. Les característiques d'aquest encoder són les que podem veure a la taula 12.

Tensió d'alimentació (V)	5
Senyal de sortida	TTL
Corrent de sortida (mA)	5
Velocitat en Número de polsos per volta	500
Número de canals	2
Canal zero (índex)	Si
Desfasament entre canals (°)	90
Freqüència màxima d'utilització (KHz)	100
Inèrcia del disc (g·cm²)	0,60
Sortida per trigger Schmitt	Si
Temperatura mínima d'utilització (°C)	-40
Temperatura màxima d'utilització (°C)	100
Pes (g)	20

Taula 12. Característiques tècniques encoder

Aquest encoder té unes dimensions força reduïdes, adequades pel nostre muntatge, aproximadament fa 41 x 30 mm, tot seguit una foto de l'encoder a la figura 25.



Figura 25. Encoder heds5540

3.2.6. Limitadors de recorregut

Per la limitació del recorregut dels moviments transversal i frontal utilitzarem uns finals de cursa de dimensions molt ajustades, 28 x 16 x 10,3 mm. A la figura 26 podem veure una foto del limitador.

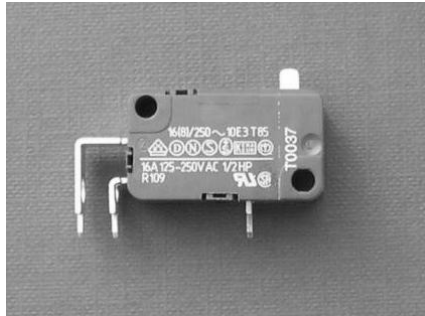


Figura 26. Final de carrera

En col·locarem un a cada extrem d'una de les guies encarregades del desplaçament transversal i repetirem l'operació en la guia encarregada del desplaçament frontal. Podem veure'ls senyalitzats a la figura 27.

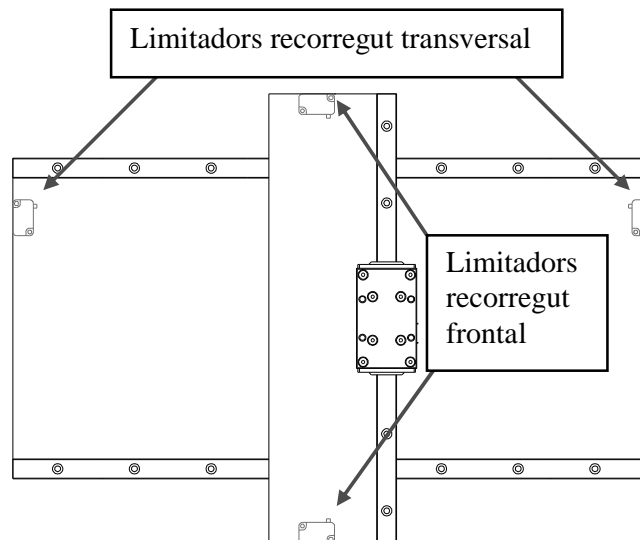


Figura 27. Col·locació limitadors de recorregut

Per aconseguir que cada patí es desplaci totalment fins al final del recorregut de la seva guia s'han confeccionat unes petites palanques de ferro fixades al patí, pel desplaçament transversal, i al suport del motor de l'eix frontal, pel desplaçament frontal amb la única finalitat d'activar el final de cursa quan el patí corresponent està al màxim del recorregut possible. Així s'aconsegueix aprofitar al màxim el desplaçament que ens ofereixen els dos graus de llibertat del sistema d'estabilització.

Les característiques dels finals de cursa són ben simples, tenim l'opció de muntar-los en posició normalment oberta o normalment tancada. Ho farem en posició normalment tancada tal com es mostra a la figura 28, ja que així, mentre el corresponent patí no toqui el final de cursa, el circuit elèctric estarà tancat i el motor rebrà alimentació. Per tant, es pot dir que en condicions normals alimentarem els motors, mentre que quan el patí activi el final de cursa es tallarà l'alimentació dels motors automàticament.

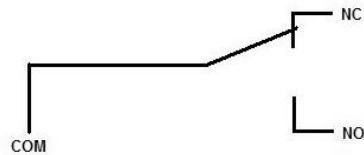


Figura 28. Esquema elèctric final de cursa

3.2.7. Dimensions i pesos

En aquest subapartat exposarem la dimensió i el pes de cada una de les peces que componen l'estructura del sistema d'estabilització. Per facilitar la comprensió de totes les dades ho exposarem en forma de taula a la taula 13.

En la mesura del possible ordenarem les peces seguint un ordre ascendent, és a dir, començarem exposant les peces que estan més a prop de les cames per acabar amb les més allunyades. A la taula 13 no s'hi han representat elements de poc pes com són els finals de carrera o els cargols de fixació, perquè no es considera rellevant.

Peça	Dimensions (mm)	Pes (g)	Unitats	Pes total (g)
Base X	500x250x5	1650	1	1650
Guia transversal	500x15,5x18,7	420	2	840
Patí guia	78x47x22	160	3	480
Cremallera transversal	500x10x12	420	1	420
Conjunt motor	160xØ42	750	2	1500
Pinyó	14xØ50	20	2	40
Fixació conjunt motor	60x60x2,5	125	2	250
Barres rosca mètric 5	70xM5	84	8	672
Base Y	350x100x5	460	1	460
Guia frontal	350x15,5x18,7	240	1	240
Cremallera frontal	350x10x12	294	1	294
Contrapès	45x15x50	5.000	1	5.000
PES TOTAL SISTEMA ESTABILITZACIÓ SENSE CONTRAPÈS				6.846
PES TOTAL SISTEMA ESTABILITZACIÓ AMB CONTRAPÈS				11.846

Taula 13. Detall dimensions i pesos estructura estabilitzadora

A partir de la taula 13 crearem les taules 14 i 15, que serveixen per veure millor de quin contrapès es disposa per equilibrar el robot frontalment i transversalment.

Peça	Pes (g)	Unitats	Pes total (g)
Contrapès	5.000	1	5.000
Guia transversal	420	2	840
Patí guia	160	3	480
Conjunt motor	750	2	1.500
Pinyó	20	2	40
Fixació conjunt motor	125	2	250
Barilla mètric 5	84	8	672
Base Y	460	1	460
Guia frontal	240	1	240
Cremallera frontal	294	1	294
PES TOTAL CONTRAPÈS TRANSVERSAL			9.776

Taula 14. Contrapès transversal

Així doncs, amb la taula 14 es veu com es disposa d'un contrapès total de 9.776g per establitzar el robot transversalment quan aquest aixeca una cama.

Peça	Pes (g)	Unitats	Pes total (g)
Contrapès	5.000	1	5.000
Patí guia	160	1	160
Conjunt motor	750	1	750
Pinyó	20	1	40
Fixació conjunt motor	125	1	125
PES TOTAL CONTRAPÈS FRONTAL			6.075

Taula 15. Contrapès frontal

A la taula 15 s'observa que es disposa de menys contrapès que pel moviment transversal, a més el desplaçament que pot fer en aquest sentit és més curt i per tant genera menys moment, però és suficient per estabilitzar el robot en aquest sentit.

3.2.8. Interconnexió dels components que formen l'estructura

Acabarem aquest apartat resumint com van acoblats entre si els diferents components que formen l'estructura d'estabilització.

Començant per la base X, la plataforma d'alumini de 500 x 250 x 5 mm, va collada a les dues cames per la seva part inferior, cada cama es colla amb 4 cargols de mètric 5 i longitud 15mm i femelles de seguretat. Per la part superior va collada a cada guia lineal de 500mm de longitud amb 2 cargols de mètric 4 i longitud 30mm i a la cremallera, de també 500 mm, amb 2 cargols de mètric 5 i longitud 15mm. A cada cremallera se li han fet dos forats roscats de mètric 5 per collar-hi els 2 cargols que la fixen a la base X. Per últim, cada final de cursa va collat a la base X amb 2 cargols de mètric 3 i 25mm de longitud.

Com ja sabem, de cada patí de les guies que proporcionen el moviment transversal en surten 4 barres roscades de M5 que van collades a la base Y. Amb aquestes barres podem modificar l'alçada de l'estructura del moviment frontal fàcilment.

A la part inferior de la base Y, hi collarem un perfil d'alumini de 90° que servirà per fixar el conjunt motor-reductor-encoder que dotarà l'estructura de moviment transversal. A l'eix del reductor hi acoblarem el pinyó de 48 dents, que s'ajustarà perfectament a la cremallera gràcies a la precisió que ofereixen les barres roscades.

A la part superior de la base Y, hi collem la guia lineal de 350 mm i la cremallera adaptada de 350 mm.

Collarem un altre perfil d'alumini de 90° al patí de la guia de 350 mm que fixarà el conjunt motor-reductor-encoder que dotarà l'estructura de moviment frontal. A l'eix del reductor també hi acoblarem el pinyó, aquest s'ajustarà perfectament a la cremallera jugant amb l'alçada de la cremallera a través dels cargols de mètric 5.

Així doncs queda explicada l'estructura del sistema d'estabilització. A la figura 29 es pot observar una fotografia de l'aspecte definitiu del sistema d'estabilització amb tots els elements convenientment acoblats.



Figura 29. Sistema estabilització

3.3. Física del sistema

Una vegada ja coneixem el pes, les dimensions i la ubicació de cada peça que componen el sistema, estudiarem com es comporta el sistema des de la basant estàtica. Per tant analitzarem com estan distribuïts en diferents situacions els pesos fixes, els mòbils i com aconseguirem l'estabilitat del robot.

Inicialment posarem algunes definicions imprescindibles per entendre l'estabilitat estàtica de l'estructura que anomenem robot.

Anomenem estructura a una organització de material que defineix i manté una forma determinada.

El centre de gravetat d'un cos és, per a tota possible orientació del cos en l'espai, el punt lligat al cos per al qual el moment resultant del sistema de forces format per totes les forces pes de cada una de les partícules constituents del cos és zero.

Una estructura és estable quan la projecció del centre de gravetat cau dins la seva base.

Quan una estructura és inestable cau cap al costat on hi ha més pes.

Algunes estructures que semblen inestables, com per exemple les grues de les obres que serveixen per pujar i baixar càrregues, no cauen. Són estables gràcies a un contrapès que equilibra la força que fa cada braç cap al terra.

Aquesta mateixa tècnica s'utilitzarà per calcular que el robot sigui estable en totes les seves posicions.

L'estàtica dels sòlids és la part de la mecànica que estudia les condicions d'equilibri que han de complir les forces exteriors o les càrregues aplicades sobre cossos, als que es considera infinitament resistents i indeformables. Quan les forces exteriors estan distribuïdes de tal forma que s'equilibren entre si, podem dir que el cos es troba en repòs o en equilibri estàtic.

L'objectiu de l'estàtica és determinar la força resultant i el moment resultant de totes les forces que actuen sobre un cos per poder determinar les seves condicions d'equilibri.

Així doncs, queda clar que perquè un cos o estructura (el robot) estigui en equilibri s'ha de complir, simultàniament, que la força resultant i el moment resultant siguin nuls. En el nostre cas, la base de l'estructura serà el peu dret o l'esquerra. Donat que el peu es recolza sobre el terra i que no hi ha cap fixació entre ells, el terra només pot exercir una força normal cap amunt per mantenir l'equilibri estàtic. Així doncs si la projecció del centre de gravetat surt de l'àrea mínima que engloba els peus en contacte amb el terra, la força normal del terra no serà suficient per mantenir l'equilibri i el robot caurà.

A la figura 30 veiem una representació gràfica de la posició del centre de gravetat global del robot en diferents situacions.

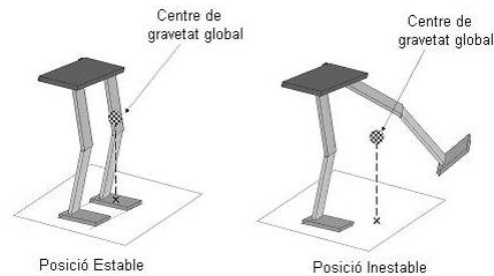


Figura 30. Posició Estable i Posició Inestable

3.3.1. Distribució de pesos

Per construir l'estructura del robot s'ha tingut molt en compte la distribució final de pesos del robot. Sempre buscant la màxima simetria entre dreta i esquerra i davant i darrera. Així podrem tractar el desplaçament cap a un costat i cap a l'altre igualment.

Al tenir dues guies lineals aconseguim que geomètricament la base X sigui pràcticament simètrica, l'únic element que trenca la simetria és la cremallera que no queda totalment centrada, però a nivell de pesos no té importància ja que queda molt propera al centre de l'estructura i té poc pes.

Pel que fa a la distribució de la base Y, només hi tenim una guia, per tant geomètricament no pot ser simètric. Però aconseguirem la simetria de pesos desplaçant la base segons convingui fins a trobar l'estabilitat.

3.3.2. Parts mòbils

Considerem que les parts mòbils són aquelles que podem controlar per aconseguir l'estabilització del robot.

Disposem del contrapès que genera el moviment transversal i del que genera el moviment frontal. Amb aquests dos graus de llibertat estabilitzarem el robot.

El moviment transversal és imprescindible per establir el robot lateralment quan aquest aixeca una cama. En el moment en que la cama que aixequem deixa de tenir contacte amb el terra el robot es comporta com un sistema inestable i necessitem desplaçar tot el pes disponible cap a l'extrem oposat, per fer que el centre de masses caigui sobre la base del peu que està en contacte amb el terra. Sinó fos així, el robot perdria l'equilibri i cauria.

Per evitar que el robot caigui endavant o endarrera disposem d'un altre grau de llibertat, el de moviment frontal, que funciona igual que el de moviment transversal.

Com que el contrapès que hem de fer per equilibrar el desplaçament lateral és més gran que el que cal fer pel desplaçament frontal, la guia del moviment transversal és més llarga, així, aplicant la llei de la palanca, aconseguim generar el moment necessari per establir el robot.

3.3.3. Càlcul de l'estabilitat del robot

L'estabilitat l'hem calculat a partir de les situacions més desfavorables que ens podem trobar, corresponen al moment en que el robot aixeca una cama i ha d'aguantar tot l'equilibri amb l'altra. Per determinar l'estabilitat del robot s'estudien quatre casos: dos relacionats amb el desplaçament transversal, quan té la cama esquerra aixecada i quan té la cama dreta aixecada (hi ha petites diferències entre un cas i l'altre) i dos pel desplaçament frontal, quan té una cama endavant o endarrera (en aquest cas és completament indiferent de quina cama es tracti).

Per evitar entrar en càlculs diferencials s'han fet tots els càlculs considerant que el pes de cada component està situat al centre geomètric del mateix. Els components que no generen moment (la distància del seu centre geomètric fins al centre de gravetat global és nul) no s'inclouran a les taules.

Per calcular l'estabilitat quan el robot té la cama esquerra aixecada i s'aguanta amb la dreta hem configurat la taula 16, amb els pesos i les distàncies dels components respecte al centre de gravetat ideal, que és el centre de la base del peu.

Part	Pes (N)	Distància(mm)	Moment(N·m)
Cama esquerra	45,03	200,00	9,01
Suma a 200mm	45,03	200,00	9,01
Base X	16,19	100,00	1,62
Guies X	8,24	100,00	0,82
Cremallera X	4,12	100,00	0,41
Suma a 100mm	28,55	100,00	2,85
Patins X	3,14	-111,00	-0,35
Conjunt motor X	7,36	-111,00	-0,82
Suport motor X	1,23	-111,00	-0,14
Base Y	4,51	-111,00	-0,50
Conjunt motor Y	7,36	-111,00	-0,82
Contrapès	49,05	-111,00	-5,44
Suma a -111mm	72,65	-111,00	-8,07
Guia Y	2,35	-68,75	-0,16
Patí Y	1,57	-68,75	-0,11
Suma a -0,07mm	3,92	-68,75	-0,27
Cremallera Y	2,88	-156,25	-0,45
Suma a -0,05mm	2,88	-156,25	-0,45
MOMENT TOTAL A L'EIX DE LA CAMA DRETA			3,39

Taula 16. Distribució de pesos cama esquerra aixecada

Hi podem observar el moment resultant al centre de la base frontal del peu, que és de 3,39N·m, a partir de les dades de la taula 16 i aplicant l'equació 1 trobem la distància

respecte el centre de la cama en que tenim un sumatori de moments nul i, per tant el robot està en equilibri.

$$\sum M = \sum F \cdot d \quad (\text{Eq. 1})$$

Adaptant l'equació 1 a les necessitats actuals s'obté l'equació 2. Igualant l'equació 1 a zero trobem el valor de la X, que significa la distància que hi ha des del centre d'estabilitat ideal fins al centre d'estabilitat real.

$$45,03 \cdot (0,200 - X) + 28,55 \cdot (0,100 - X) + 72,65 \cdot (-0,111 - X) + 3,92 \cdot (-0,06875 - X) + 2,88 \cdot (-0,15625 - X) = 0 \quad (\text{Eq. 2})$$

Resolent l'equació s'obté que el valor de la X és de 20,10mm.

Així doncs, tot i que l'estructura té certa tendència a caure lateralment en sentit de les agulles del rellotge, el centre de gravetat real en aquesta posició està situat a 20,10mm del centre geomètric del peu en direcció transversal i en sentit positiu, com que aquest punt està situat dins l'àrea del peu, el robot està en equilibri.

Per estudiar on es troba el centre de gravetat quan la cama aixecada és la dreta i el robot s'aguanta amb l'esquerra es configura la taula 17.

Part	Pes (N)	Distància(mm)	Moment(N·m)
Cama esquerra	45,03	200,00	9,01
Suma a 200mm	45,03	200,00	9,01
Base X	16,19	100,00	1,62
Guies X	8,24	100,00	0,82
Cremallera X	4,12	100,00	0,41
Suma a 100mm	28,55	100,00	2,85
Patins X	3,14	-111,00	-0,35
Conjunt motor X	7,36	-111,00	-0,82
Suport motor X	1,23	-111,00	-0,14
Base Y	4,51	-111,00	-0,50
Conjunt motor Y	7,36	-111,00	-0,82
Contrapès	49,05	-111,00	-5,44
Suma a -111mm	72,65	-111,00	-8,07
Guia Y	2,35	-68,75	-0,36
Patí Y	1,57	-68,75	-0,24
Suma a -68,75mm	3,92	-68,75	-0,60
Cremallera Y	2,88	-65,75	-1,90
Suma a -65,75mm	2,88	-65,75	-1,90
MOMENT TOTAL A L'EIX DE LA CAMA DRETA			1,30

Taula 17. Distribució de pesos cama dreta aixecada

Amb els resultats de la taula 17 i adaptant l'equació 1 a les necessitats pròpies d'aquest cas s'obté l'equació 3.

$$45,03 \cdot (0,200 - X) + 28,55 \cdot (0,100 - X) + 72,65 \cdot (-0,111 - X) + 3,92 \cdot (-0,06875 - X) + 2,88 \cdot (-0,06575 - X) = 0 \quad (\text{Eq. 3})$$

Resolent l'equació 3 es troba que el centre de gravetat exacte, està situat a 8,49mm del centre geomètric del peu en direcció transversal i en sentit negatiu.

A la figura 31 s'observa com, utilitzant un contrapès de 5.000g, els punts d'equilibri reals tan d'una com de l'altra cama estan situats dins l'àrea del peu del robot, per tant, el robot es manté en equilibri.

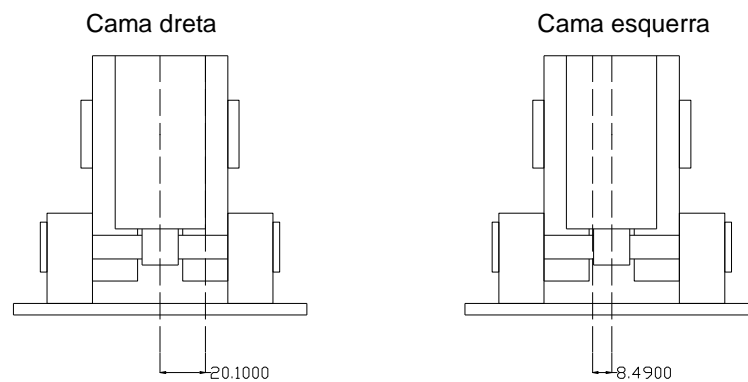


Figura 31. Situació centres d'estabilitat reals

Una vegada resolt el problema de l'estabilitat transversal, s'analitza l'estabilitat frontal. Primer s'estudiarà l'estabilitat quan una de les cames està endavant i seguidament quan la cama està endarrera.

Inicialment els càlculs es faran agafant com a centre d'estabilitat la línia central de la base X indicada a la figura 32 i llavors es calcularà el centre d'estabilitat real tal i com s'ha fet en els dos casos anteriors.

Per estudiar el moviment cap endavant es considera que el pitjor cas en que ens podem trobar és quan el robot està en la posició que mostra la figura 32.

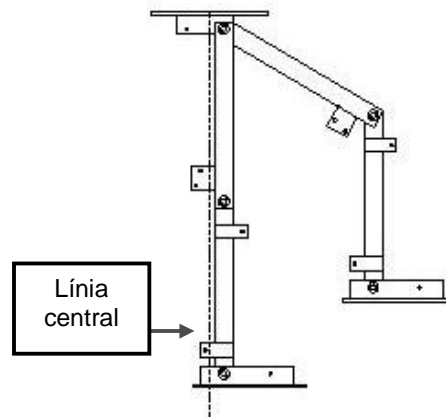


Figura 32. Cama endavant

La cuixa es troba al límit del seu desplaçament a $+60^\circ$ (a la figura 54 del subapartat 5.5.3 es pot veure aquesta dada clarament) i el conjunt tíbia-peroné està situat en posició perpendicular a la superfície de contacte.

Per tant, tenint en compte les dimensions i els angles de cada part es confecciona la taula 18.

Part	Pes (N)	Distància(mm)	Moment(N·m)
Cuixa	14,76	190,715	2,81
Tíbia-peroné	14,91	350,93	5,23
Peu	15,35	405,93	6,23
Cremallera X	4,12	82,75	0,34
Patí Y	1,57	-136,00	-0,21
Conjunt motor Y	7,36	-136,00	-1,00
Contrapès	49,05	-136,00	-6,67
Suma a -136mm	57,98	-136,00	-7,89
MOMENT TOTAL CAMA ENDAVANT			6,73

Taula 18. Distribució de pesos cama endavant

Per trobar el centre de gravetat real es resol l'equació 4 generada a partir de la taula 18 i l'equació 1.

$$14,76 \cdot (0,190715 - X) + 14,91 \cdot (0,35093 - X) + 15,35 \cdot (0,40593 - X) + 4,12 \cdot (0,08275 - X) + 57,98 \cdot (-0,136 - X) = 0 \quad (\text{Eq. 4})$$

Com veiem el centre no coincideix amb el que havíem suposat, aplicant l'equació s'ha trobat el centre de gravetat real, situat a 6,29mm del punt suposat en direcció transversal i en sentit positiu. En aquest sentit, el robot seguiria sent estable mentre la distància trobada no superi els 205,5mm, que és la longitud de peu que queda per davant del centre fictici de gravetat que s'ha considerat inicialment.

Per acabar, s'estudia l'estabilitat quan la cama està endarrera, també en les pitjors condicions mostrades a la figura 33.

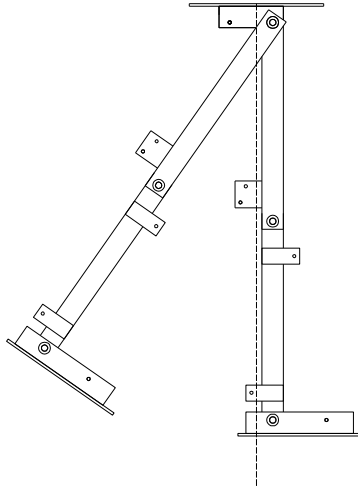


Figura 33.Cama endarrera

Com es pot observar, el maluc està en el seu límit, situat a -35° i la resta de la cama segueix la mateixa direcció. Tenint en compte les dimensions i els angles de cada part es confecciona la taula 19.

Part	Pes (N)	Distància(mm)	Moment(N·m)
Cuixa	14,76	181,72	2,68
Tíbia-peroné	14,91	287,83	4,29
Peu	15,35	364,67	5,60
Cremallera X	4,12	-82,75	-0,34
Patí Y	1,57	-136	-0,21
Conjunt motor Y	7,36	-136	-1,00
Contrapès	49,05	-136	-6,67
Suma a -136mm	57,98	-136	-7,89
MOMENT TOTAL CAMA ENDAVANT			4,35

Taula 19.Distribució de pesos cama endarrera

De nou, a partir de l'equació 5 i de la taula 19 es troba que el centre de gravetat.

$$14,76 \cdot (0,18172 - X) + 14,91 \cdot (0,28783 - X) + 15,35 \cdot (0,36467 - X) + \\ + 4,12 \cdot (0,08275 - X) + 57,98 \cdot (-0,136 - X) = 0 \quad (\text{Eq. 5})$$

En aquest cas el centre de gravetat real està situat a 4,58mm en direcció transversal i en sentit positiu del centre de gravetat fictici que ens permet fer el càlcul inicial.

Cal dir que per tal que els càlculs fossin del tot correctes s'haurien de recalculer tots els centres de gravetat una vegada s'ha trobat el real, hi ha peces que passarien a generar moment, i s'obtidria un altre centre de gravetat més correcte. Com que el que es pretén és agafar una idea dels valors pels quals ens estem movent per poder determinar el contrapès necessari per equilibrar el robot transversal i frontalment, es consideren bons els resultats obtinguts i correcte el contrapès de 5.000g.

4. PLACA CONTROL DELS ACTUADORS

Per poder governar i dirigir tots els actuadors del robot ha calgut dissenyar circuiteria electrònica específica.

4.1. Disseny

S'ha dissenyat una placa de dimensions el màxim de reduïdes possible i buscant la màxima simplicitat, però sense perdre de vista que l'objectiu principal és que funcioni perfectament.

El disseny s'ha fet amb el programa P-CAD, primer s'ha utilitzat l'esquemàtic que veiem a la figura 34 (es pot observar amb més resolució i més tamany al Plànol 4 del document plànols) i finalment el PCB per crear la placa.

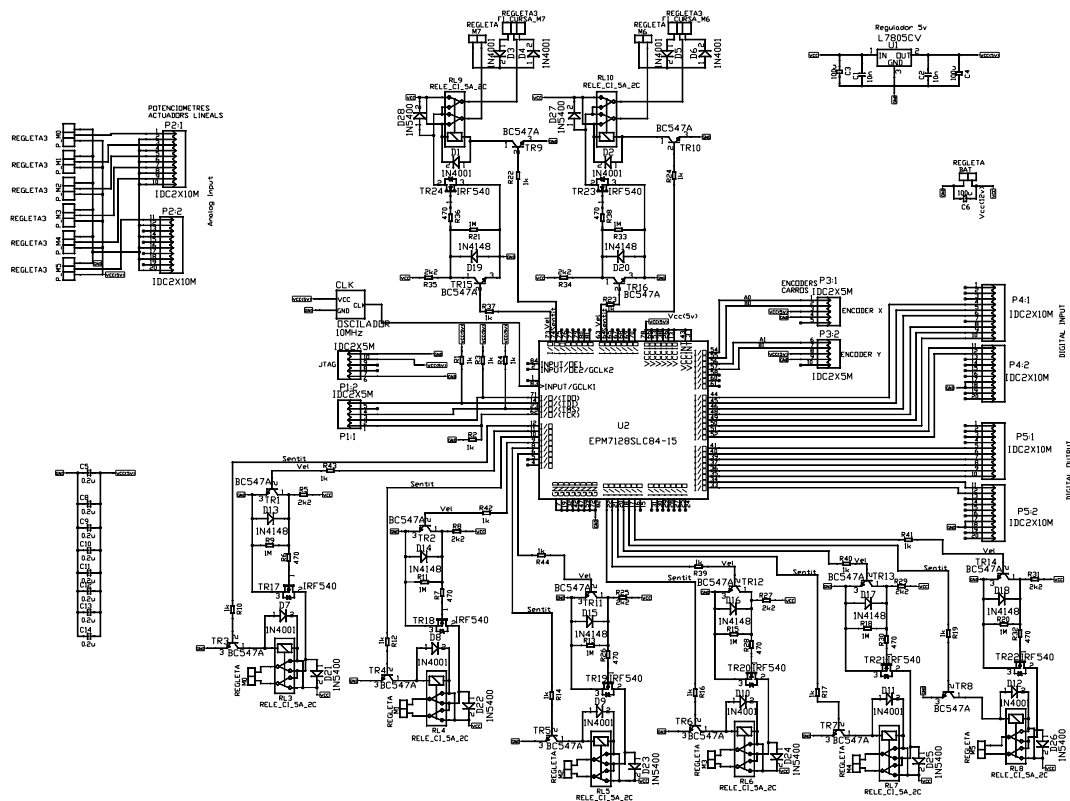


Figura 34. Esquemàtic placa de control

Es va partir de la base que el cor de la placa seria una FPGA, que programaríem segons les nostres necessitats, a l'apartat 4.8 explicarem com s'ha realitzat la seva programació.

En el projecte anterior, per invertir el sentit de gir de cada motor es va utilitzar el pont en H. S'ha descartat la seva utilització degut als problemes de fiabilitat que ocasionaven. Aquesta vegada hem buscat uns relés de potència adequats per fer aquesta feina.

Com que la placa s'alimenta a 12v DC, ha calgut incorporar un regulador L7805 per proporcionar 5v DC a la FPGA, als potenciòmetres i als encoders.

4.1.1. Objectius

Per alimentar i controlar els motors del robot bípede s'ha de disposar d'una placa que ofereixi unes característiques molt concretes, és impossible trobar una placa comercial que faci, justament, les funcions que es necessiten. Per això ha calgut dissenyar una placa amb uns objectius molt concrets.

Com ja em comentat, el tamany ha de ser el màxim reduït possible per poder crear, a la llarga un robot bípede autònom.

Fiabilitat, els components han de permetre una llarga vida útil a la placa amb el mínim manteniment.

Prestacions, es necessiten unes prestacions molt determinades que s'han d'aconseguir amb la correcte elecció de components.

Efectivitat, cal buscar la màxima efectivitat evitant pèrdues de potència i distorsió dels senyals al llarg del seu recorregut per la placa.

Velocitat, els components escollits han de garantir que la informació pugui ser tractada en temps real, per tant el processament de les consignes i la informació ha de ser pràcticament instantani.

Per poder complir amb els objectius fixats ha calgut tenir molt en compte els components utilitzats. A l'apartat 4.3 justifiquem l'elecció dels components més determinants de la placa de control.

4.2. Especificacions

La placa de control que s'ha creat té unes característiques molt concretes ja que ha estat dissenyada específicament per aquesta aplicació. Les especificacions de la placa de control són les següents:

La placa s'alimenta a +12v DC a través d'una regleta de 2 vies.

Pot alimentar altres components o plaques a +5v DC.

Pot alimentar fins a un màxim de 8 motors DC entre 0v i 12v amb quatre nivells d'alimentació.

Pot llegir 6 entrades analògiques.

Pot interpretar la lectura de 2 encoders incrementals de 2 canals.

Té un connector JTAG de reprogramació del component lògic programable (FPGA).

Disposa d'un rellotge de 20MHz.

Les seves dimensions són 205x138x30mm.

I disposa de tots els components electrònics necessaris per garantir el seu correcte funcionament.

La FPGA que incorpora, porta gravat un programa que difícilment serviria per una altra aplicació. En aquest sentit es pot dir que la placa és flexible dins d'aquest projecte, podem reprogramar la FPGA i modificar el programa de control sempre que calgui, però com dèiem, difícilment es podria utilitzar en un altra aplicació.

4.3. Elecció de components

D'entre tots els components de la placa de control farem una menció especial dels que trobem més singulars, com poden ser: la FPGA, el relé bipolar, el MOSFET IRF540 o el regulador L7805.

4.3.1. FPGA

Podem considerar que el component més important de la placa és la FPGA (Field-Programmable Gate Array) o matriu de portes programables, ja que realitza vàries funcions, entre elles, les més rellevants són: generar els PWM bàsics per la nostra aplicació, marcar el sentit de gir dels motors i interpretar els senyals enviats pels encoders. Per tant, farem una petita introducció de les característiques de les FPGAs.

Quan abordem el disseny d'un sistema electrònic i sorgeix la necessitat d'implementar una part amb hardware disposem de diferents possibilitats. S'han representat les principals aproximacions ordenades en funció dels paràmetres: cost, flexibilitat, prestacions i complexitat a la figura 35. Com es pot veure, les millors prestacions les proporciona un disseny full-custom, aconseguit a base d'elevats preus i enorme complexitat de disseny. A l'altre extrem del ventall de possibilitats s'hi troba la implementació software, que és molt econòmica i flexible, però que en determinats casos no és vàlida per arribar a un nivell de prestacions relativament alt.

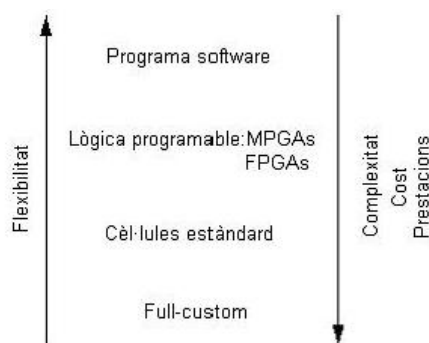


Figura 35. Flexibilitat – complexitat, cost i prestacions

Entre aquestes dues opcions es pot triar la fabricació d'un circuit electrònic realitzat mitjançant disseny semi-custom, utilitzant cèl·lules estàndard, o recórrer a un circuit ja

fabricat que es pugui “programar” in situ, com són les FPGAs. D'aquestes dues opcions, la primera proporciona millors prestacions, encara que es més cara i exigeix un cicle de disseny relativament llarg. Per altra banda, els dispositius lògics programables constitueixen una bona oferta per realitzar dissenys electrònics digitals amb un bon compromís preu - prestacions. I el que és millor, permeten obtenir una implementació en un temps de disseny increïblement curt. Un altre aspecte a tenir en compte per decidir-se per aquest tipus d'implementació és que el cost de realització és molt baix, pel que sol ser una bona opció per la realització de prototips com el nostre.

Altera informa que cal tenir en compte alguns requisits alhora de dissenyar la circuiteria que conté la FPGA. Un dels requisits més importants és col·locar un condensador de $0,2\mu\text{F}$ entre cada alimentació (V_{cc} i GND) del dispositiu lògic programable. I a més, s'ha de col·locar un condensador de gran capacitat al punt on alimentem la placa, per estabilitzar l'alimentació. Com que la tensió que rep la FPGA prové del regulador de voltatge, cal posar el condensador a la sortida del regulador entre ($V_{cc}+5v$ i GND).

Arribem a la conclusió que les característiques més importants de les FPGAs són: flexibilitat, capacitat de processar en paral·lel i a alta velocitat. Per tant, són ideals per la nostre aplicació.

4.3.2. Components de potència

Un altre element important és el relé, el qual rep el sentit i la velocitat que s'han d'aplicar al motor. El relé és un dispositiu que a través d'un electroiman acciona un o varis contactes que permeten obrir o tancar circuits elèctrics.

En aquest cas ha sigut imprescindible escollir relés bipolars per poder alimentar els motors en els dos sentits. Una altra característica important és el seu tamany ($29 \times 15,7 \times 12,7$ mm), són relés de potència compactes que ocupen el mateix espai a la placa que els relés en miniatura per a PCB. Disposen d'una estructura totalment sellada, contactes de plata níquel lliures de cadmi, bobina d'alta sensibilitat pels canvis de polaritat i la categoria de protecció segons VDE 0700 és de 2.

A la figura 36 veiem la imatge externa i interna del relé.

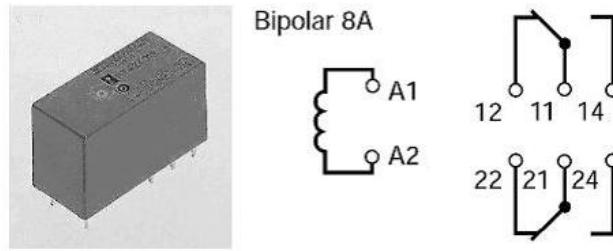


Figura 36. Imatges externa i interna del relé

Per controlar aquest relé ens arriben dues senyals de la FPGA: la de sentit i la de velocitat. La de sentit s'aplica a la base d'un transistor BC547 i el col·lector controla la bobina A1 del relé fent que aquest commuti quan rep el senyal. Els pins A2, 22 i 14 estan connectats a V_{cc} , a l'11 i al 21 s'hi connecten els motors i al 12 i 24 el senyal de PWM amplificat pel MOSFET.

Per tal que els motors utilitzats interpretin correctament els PWM, cal amplificar els senyal. Per fer-ho, s'ha escollit un transistor MOSFET (Metal Oxide Semiconductor Field Effect Transistor) IRF540 que alimenta i governa directament el relé i indirectament els motors, a la següent figura, la figura 37, veiem la imatge externa i el símbol característic del MOSFET en qüestió, i seguidament, a la taula 20 podem veure el seu pin-out.

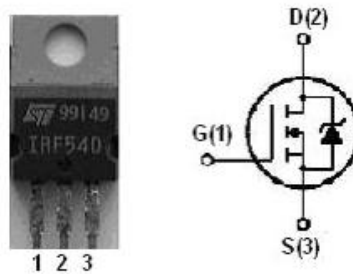


Figura 37. Imatge externa i símbol del MOSFET IRF540

Pin	Funció
1	Porta (Gate)
2	Drenador (Drain)
3	Font (Source)

Taula 20. Taula pin-out MOSFET IRF540

Com qualsevol amplificador, la seva funció és augmentar el nivell del senyal, en aquest cas actuant sobre la seva amplitud. Els MOSFETs, creen un efecte de camp gràcies a la unió d'un semiconductor format per la parella metall-òxid. Des de la seva creació s'utilitzen molt, perquè asseguren una distorsió molt baixa ja que controlen el desprendiment tèrmic que es produeix durant el processament del senyal. En el subapartat 4.5.2 podem veure el PWM abans i després de ser amplificat pel MOSFET corresponent. Per aconseguir fer commutar el MOSFET no en tenim prou amb el senyal de PWM que obtenim de la FPGA. Per això necessitem l'ajuda d'un transistor BC547 i d'alguns components discrets. Així doncs, apliquem la consigna de PWM a la base d'aquest transistor que commuta d'acord amb el senyal aplicat a la seva base. El col·lector es connecta a la porta del MOSFET i l'emissor del

BC547 a la font, per últim es connecta el drenador als pins 12 i 24 del relé. Amb aquest muntatge s'aconsegueix fer commutar el MOSFET perfectament.

Com a últim element a comentar tenim el regulador L7805. Es tracta d'un regulador de voltatge positiu. Ens permet alimentar components a +5vDC a partir dels +12vDC amb que alimenten la placa de control. Gràcies a aquest component, ens estalviem haver de tenir dues fonts d'alimentació o bé una que disposi d'una sortida de 12vDC i una altra de 5vDC. Mostrarem una imatge exterior i el pin-out del component a la figura 38.

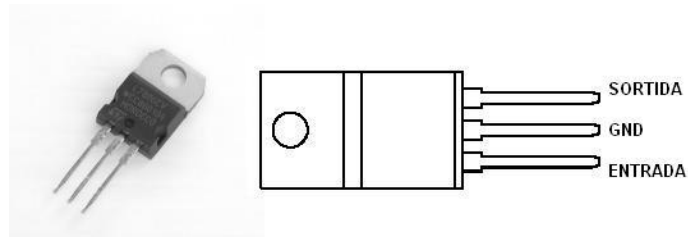


Figura 38. Imatge L7805 i pin-out

Observant la figura, pel pin ENTRADA hi aplicarem els +12vDC, el pin GND el connectarem amb la resta de masses del circuit i pel pin SORTIDA obtindrem els +5vDC necessaris per alimentar alguns components importants com pot ser la FPGA.

Cal dir que s'han seguit i complert les característiques tècniques que indiquen els fulls de característiques de tots i cadascun dels components de la placa de control.

4.4. Comunicació entre el software i el robot

La comunicació entre el software i el robot es realitza a través del que anomenem paraula de control. Les nostres paraules de control estan formades per 8 bits que contenen la informació necessària pel control del robot.

La paraula de control que surt de la tarja pel connector 3 està distribuïda segons mostra la taula 21.

Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
CE	Selecció 3	Selecció 2	Selecció 1	Selecció 0	Sentit	Velocitat 1	Velocitat 0

Taula 21. Estructura paraula de control

CE: el bit de Check Enable és l'indicador que la paraula de control es pot enviar, quan la FPGA detecta el flanc de pujada d'aquest bit és sinònim de que es validen les noves entrades.

Selecció 3, Selecció 2, Selecció 1 i Selecció 0: aquests 4 bits permeten seleccionar el motor que volem controlar o l'acció que volem realitzar, al tenir 4 bits de selecció podem definir fins a 16 (2^4) seleccions.

Sentit: bit que proporciona el sentit de gir del motor.

Velocitat 1 i Velocitat 0: són els 2 bits que proporcionen el nivell de PWM, o velocitat, al motor seleccionat. Al tenir dos bits de velocitat disposem de 4 velocitats (2^2).

Així doncs, la comunicació entre el software i el robot es realitza a través de paraules de control. Cal recordar, que cada paraula de control només pot moure un sol motor alhora, per tant, per modificar les consignes de varis motors simultàniament cal una seqüència de paraules de control. A nivell pràctic, a la pantalla del PC podem canviar totes les característiques de velocitat i sentit que vulguem, quan enviem les instruccions, el software s'encarrega de generar la seqüència de paraules de control seleccionada i l'usuari veu complertes totes les ordres que ha enviat al robot.

4.5. Control de velocitat

El procés de control de la velocitat comença a la pantalla del PC quan l'usuari escull la velocitat d'un motor, i acaba quan el motor rep el PWM desitjat.

4.5.1. Generació de PWM

D'entre diferents sistemes per regular la velocitat dels motors elèctrics s'ha escollit la modulació per amplada de pulsacions, PWM. És una tècnica que té la particularitat de mantenir el parell del motor constant i no suposa un desaprofitament de l'energia elèctrica.

La consigna de velocitat que seleccionem amb l'interfície gràfica del LabWindows l'enviem cap a la FPGA a través de la tarja d'adquisició de dades, aquests bits els trobem als pins 4 i 5 del connector 4 de la placa de control. Perquè la FPGA interpreti i generi el PWM relacionat amb la consigna rebuda cal seguir un procés concret.

El funcionament intern de la FPGA és molt simple. A cada flanc de pujada del senyal CE de la paraula, la FPGA rep la informació enviada des de la pantalla del PC. Dins el bloc decode de la FPGA se selecciona el motor al que se li ha d'aplicar la velocitat gràcies a la informació que conté la paraula de control.

A través d'un línia de dades la velocitat seleccionada va cap a un altre bloc de la FPGA, anomenat PWM. A partir del senyal del rellotge, de la consigna de velocitat i d'un comptador intern es genera el PWM necessari. Si volem modificar el valor del PWM o els intervals de la velocitat només cal accedir al bloc PWM, modificar el valor del límit del comptador que volem canviar i reprogramar la FPGA amb el connector JTAG, així de simple.

A la taula 22 indiquem per quin pin de la FPGA surt el PWM de cada motor.

Pin	11	9	5	4	18	16	63	73
Motor	0	1	2	3	4	5	6	7

Taula 22. Localització PWMs en els pins de la FPGA

4.5.2. Resultats

Aquí visualitzarem els PWM generats a la FPGA per a cada velocitat i el senyal que arriba al motor, que no és res més que el PWM amplificat amb el MOSFET IRF540 perquè els motors funcionin correctament. A la figura EPC ubiquem el primer punt de visualització, el PWM tot just sortir de la FPGA, el veurem pel canal 1 (CH1) de l'oscil·loscopi calibrat a 2v/divisió i el segon punt de visualització, el PWM després de ser amplificat el veurem pel canal 2 (CH2) de l'oscil·loscopi calibrat a 5v/divisió.

La figura 39 és la representació gràfica del senyal obtingut quan tenim seleccionada la velocitat 0.

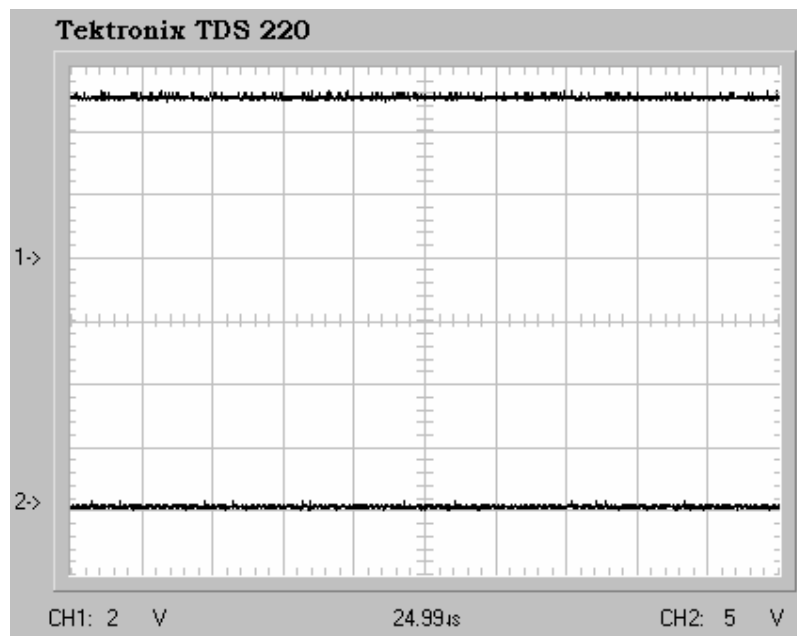


Figura 39. Representació velocitat 0

Com podem veure, de la FPGA, representat al CH1, en surt un senyal de 5v, la màxima, per tant el motor rep una consigna d'exactament 0v, que corresponen a la velocitat 0.

A la figura 40 veiem la representació de la velocitat 1.

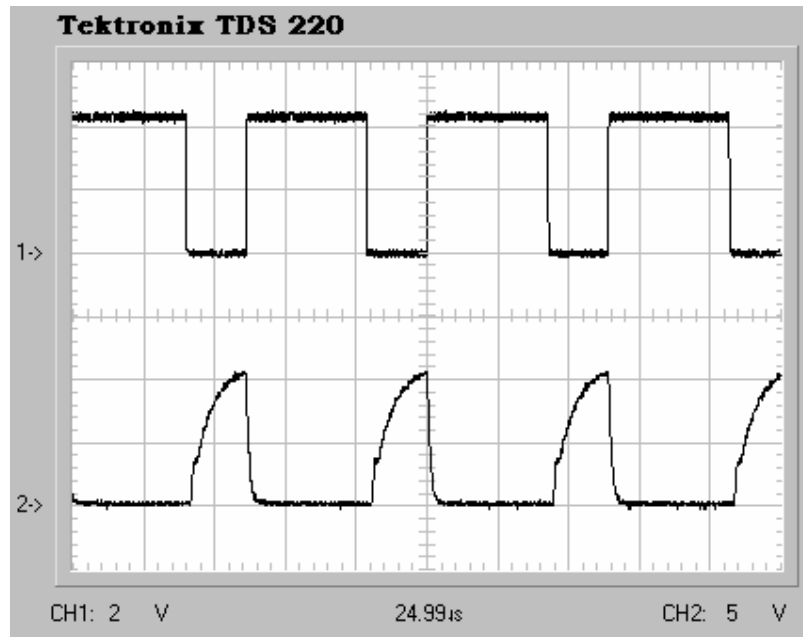


Figura 40. Representació velocitat 1

Veiem com en el canal 1 tenim un PWM amb una freqüència de pràcticament 40KHz (període=25μs) i amb una amplitud de 5 volts donats per la FPGA, mentre que en el canal 2 tenim, aproximadament, el mateix senyal, invers i ampliat fins a una amplitud de 12v.

Seguidament, a la figura 41 mostrem la velocitat 2.

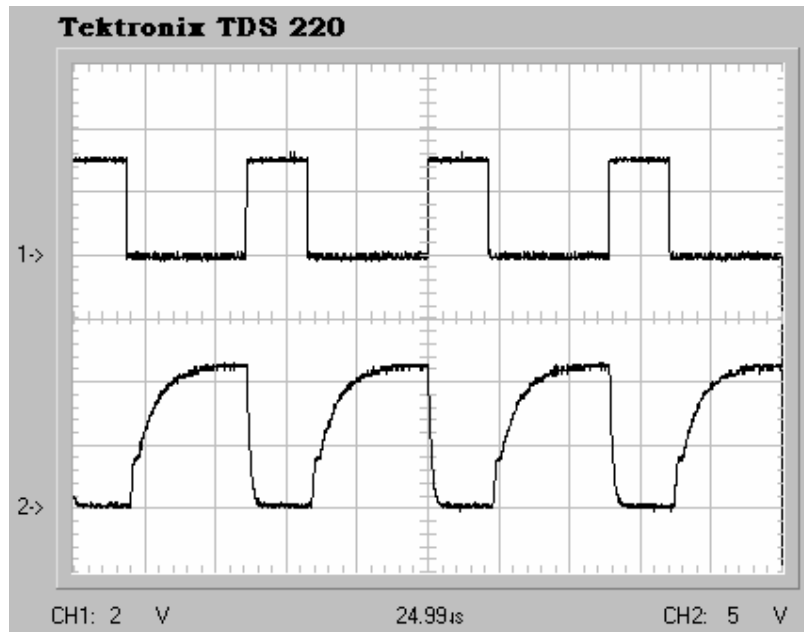


Figura 41. Representació velocitat 2

En aquest cas a partir del senyal de PWM mostrat en el canal 1 obtenim la sortida mostrada en el canal 2. Observem com la mitjana de l'alimentació dels motors cada vegada és més propera a 12v, ja que ens estem acostant a l'alimentació màxima de 12v, i per tant, també a la velocitat màxima.

Per últim, a la figura 42 podem observar la velocitat màxima, que evidentment, es dona quan l'alimentació del motor és d'aquests 12v que veiem representats al canal 2 de la pantalla de l'oscil·loscopi.

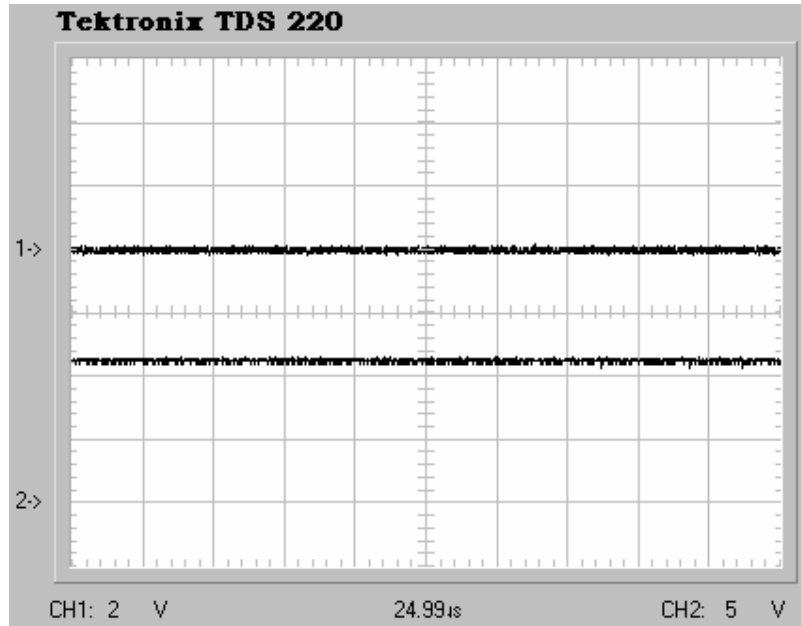


Figura 42. Representació velocitat 3

Com hem pogut observar a les figures 39, 40, 41 i 42, els PWM que surten de la FPGA són inversos als que rep el motor. Això, és degut a que amb el circuit electrònic utilitzat per fer l'amplificació del senyal, aquest es gira. Per això, amb el programa de la FPGA fem sortir els PWM completament oposats, així funciona perfectament.

4.6. Control del sentit

Com en el cas del control de la velocitat, el control del sentit comença quan, des del PC, seleccionem un sentit determinat pel motor que estem controlant. Aquest control és més simple que el de velocitat ja que només hi ha dos sentits disponibles, segons el motor: amunt i avall, o bé, esquerra i dreta. El sentit pot prendre valor lògic "1" (+5v) o "0" (0v).

La consigna de sentit, enviada des del PC, arriba a la FPGA pel pin 6 del connector 4 de la placa de control a través de la tarja d'adquisició de dades. Aquesta consigna entra al bloc decode de la FPGA i surt pel pin que li correspon, la taula 23 ens serveix per localitzar el sentit de cada motor al pin de la FPGA.

Pin	12	10	8	22	20	17	64	74
Motor	0	1	2	3	4	5	6	7

Taula 23. Localització dels sentits en els pins de la FPGA

Segons el motor, el senyal surt pel pin indicat a la taula cap a la base del transistor BC547 que s'encarrega de modificar, si cal, la polaritat de l'alimentació del motor en qüestió a través del relé de potència que li correspongui.

4.7. Connectors

La placa de control disposa de 21 connectors: 1 per l'alimentació de la placa, 8 per alimentar els motors (de M0 a M7), 2 per l'alimentació i control dels finals de cursa (FI_CURSA_M7 i FI_CURSA_M6), 6 per l'alimentació i el control dels potenciòmentres (de P_M0 a P_M5), 3 per comunicar-nos amb la placa d'adquisició de dades (CN1, CN3 i CN4), 1 per l'alimentació i control dels encoders (ENCODERS) i un altre per programar la FPGA sempre que convingui (JTAG).

Tots els cables i connectors estan convenientment etiquetats per evitar errors de connexió entre components i facilitar la connexió entre components.

Per començar aquest apartat podem veure la distribució dels tots i cadascun dels connectors a la placa de control en la figura 43.

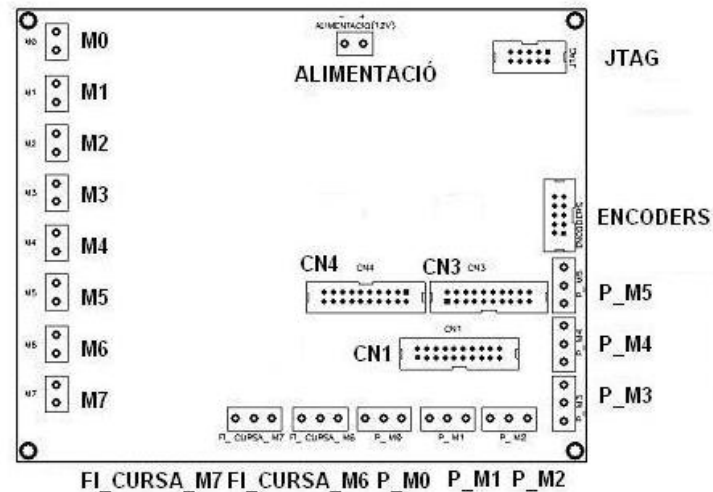


Figura 43. Ubicació connectors a la placa de control

El connector per l'alimentació no és res més que una regleta de 2 vies que ens permet alimentar la placa entre 0v i +12v.

Seguint amb les regletes de 2 vies, en tenim 8 per alimentar els motors (de M0 a M7). Segons la velocitat seleccionada els alimentem amb diferents PWM, per tant amb diferents voltatges que van dels 0v a velocitat 0 fins als 12v a velocitat 3.

Com ja sabem, els dos motors del pla XY disposen de finals de cursa. Cada motor en té dos, un per cada sentit. Doncs bé, connectarem cada fi de cursa (FI_CURSA_M7 i FI_CURSA_M6) a una regleta de 3 vies agafant un terminal comú, així, quan s'obre un contacte, el motor segueix alimentat a través de l'altre final de cursa i, al canviar el sentit per software, el circuit ja està a punt per funcionar.

A la figura 44 veiem com fem aquesta connexió entre els finals de cursa i la regleta de 3 vies.

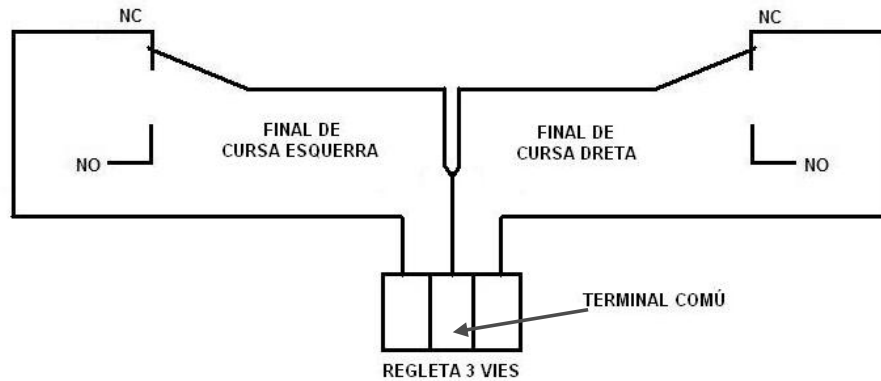


Figura 44. Esquema elèctric finals de cursa amb regleta

Per l'alimentació i control dels potenciòmetres (de P_M0 a P_M5) disposem de regletes de 3 vies. Les entrades 1 i 3 es fan servir per alimentar els potenciòmetres entre 0v i +5v, mentre que per la segona entrada llegim el valor analògic del potenciòmetre. Tots aquests valors s'envien directament a la placa d'adquisició de dades a través del CN1.

El connector 1 (CN1), igual que el CN3 i el CN4 es tracte d'un connector de 20 pins mascle tipus IDC (Insulation Displacement Connector). El CN1 està destinat a transmetre els valors analògics dels potenciòmetres dels actuadors lineals des de la placa de control fins a la tarja d'adquisició de dades. És l'entrada analògica de la tarja.

A la taula 24 veiem la descripció i utilització de cada pin d'aquest connector.

Pin	Descripció	Utilització	Pin	Descripció	Utilització
1	Entrada Analògica 0	Valor potenciòmetre M0	2	GND Analògic	GND
3	Entrada Analògica 1	Valor potenciòmetre M1	4	GND Analògic	GND
5	Entrada Analògica 2	Valor potenciòmetre M2	6	GND Analògic	GND
7	Entrada Analògica 3	Valor potenciòmetre M3	8	GND Analògic	GND
9	Entrada Analògica 4	Valor potenciòmetre M4	10	GND Analògic	GND
11	Entrada Analògica 5	Valor potenciòmetre M5	12	GND Analògic	GND
13	Entrada Analògica 6	No Connectat	14	GND Analògic	GND
15	Entrada Analògica 7	No Connectat	16	GND Analògic	GND
17	Entrada Analògica 8	No Connectat	18	GND Analògic	GND
19	Entrada Analògica 9	No Connectat	20	GND Analògic	GND

Taula 24. Pin-outs connector 1

El connector 3, CN3, és la sortida digital de la tarja d'adquisició de dades, les consignes per controlar el funcionament de cadascun dels motors. Les paraules de control surten de la tarja d'adquisició de dades per aquest connector.

A la següent taula, la taula 25, veiem la descripció i utilització de cada pin d'aquest connector.

Pin	Descripció	Utilització	Pin	Descripció	Utilització
1	Sortida Digital 0	No Connectat	2	Sortida Digital 1	No Connectat
3	Sortida Digital 2	No Connectat	4	Sortida Digital 3	Check Enable (CE)
5	Sortida Digital 4	Selecció 3	6	Sortida Digital 5	Selecció 2
7	Sortida Digital 6	No Connectat	8	Sortida Digital 7	Selecció 1
9	Sortida Digital 8	Selecció 0	10	Sortida Digital 9	Sentit
11	Sortida Digital 10	Velocitat 1	12	Sortida Digital 11	Velocitat 0
13	Sortida Digital 12	No Connectat	14	Sortida Digital 13	No Connectat
15	Sortida Digital 14	No Connectat	16	Sortida Digital 15	No Connectat
17	GND Digital	GND	18	GND Digital	GND Digital
19	Sortida de +5v	No Connectat	20	Sortida de +12v	No Connectat

Taula 25. Pin-outs connector 3

El següent connector és el CN4, per ell arriba a la tarja d'adquisició de dades el valor de posició del grau de llibertat de l'estructura d'estabilització seleccionat, per tant, és l'entrada digital de la tarja del PC.

A la taula 26 veiem la descripció i utilització de cada pin d'aquest connector.

Pin	Descripció	Utilització	Pin	Descripció	Utilització
1	Entrada Digital 0	No Connectat	2	Entrada Digital 1	No Connectat
3	Entrada Digital 2	No Connectat	4	Entrada Digital 3	No Connectat
5	Entrada Digital 4	Bit Posició 0	6	Entrada Digital 5	Bit Posició 1
7	Entrada Digital 6	Bit Posició 2	8	Entrada Digital 7	Bit Posició 3
9	Entrada Digital 8	Bit Posició 4	10	Entrada Digital 9	Bit Posició 5
11	Entrada Digital 10	Bit Posició 6	12	Entrada Digital 11	Bit Posició 7
13	Entrada Digital 12	No Connectat	14	Entrada Digital 13	No Connectat
15	Entrada Digital 14	No Connectat	16	Entrada Digital 15	No Connectat
17	GND Digital	GND	18	GND Digital	GND Digital
19	Sortida de +5v	No Connectat	20	Sortida de +12v	No Connectat

Taula 26. Pin-outs connector 4

Pel que fa al connector dels encoders i el de programació JTAG, cada un d'ells és un connector de 10 pins mascle tipus IDC.

El connector del encoders, té els pins distribuïts tal com indica la taula 27.

Pin	Utilització	Pin	Utilització
1	Canal A – encoder X	2	Canal B – encoder X
3	+5v encoder X	4	GND encoder X
5	No Connectat	6	Canal A – encoder Y
7	Canal B – encoder Y	8	+5v encoder Y
9	GND encoder Y	10	No Connectat

Taula 27. Pin-outs connector encoder

Al connector de programació JTAG hi tenim els pins distribuïts com es mostra a la taula 28.

Pin	Descripció	Pin	Descripció
1	TCK	2	TDO
3	TMS	4	No Connectat
5	TDI	6	GND
7	No Connectat	8	No Connectat
9	+5V	10	GND

Taula 28. Pin-outs connector JTAG

Explicarem, detalladament, la funció i connexió de cada pin del connector JTAG al següent apartat.

4.8. Programació FPGA

Per programar aquest component només cal fer el codi de programa que hi vols implementar amb el software facilitat pel fabricant. El software és el Quartus II i el fabricant Altera. Per programar aquest tipus de dispositius es segueix l'esquema que mostrem a la figura 45.

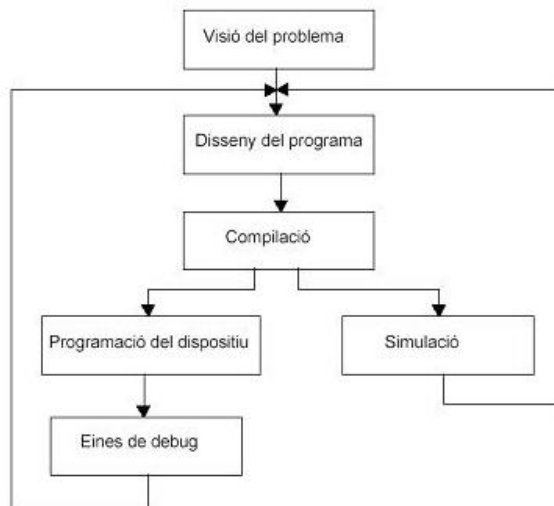


Figura 45. Esquema disseny programa

4.8.1. Disseny del programa

Una vegada hem estudiat el problema a resoldre i en coneixem la solució entrem en el disseny del programa. El disseny es realitza de forma descendent, per tant, primer de tot creem un nou projecte que contindrà tots els arxius del programa, en el nostre cas tenim: els arxius de text .vhd i l'arxiu esquemàtic .bdf.

El disseny l'hem separat amb cinc blocs: clkdiv, periode, decode, posicio i PWM. Cada bloc té un número determinat d'entrades i sortides que corresponen a pins I/O (Input/Output, entrada/sortida) de la FPGA, per tant, es pot utilitzar cada un d'aquests pins com una entrada o una sortida segons interressi.

El bloc clkdiv adapta la freqüència del rellotge de la placa de control, que és de 20MHz, a la que ens va més bé per l'aplicació, que és de 10MHz. Modificant aquesta funció es pot transformar la freqüència del rellotge de la placa a la que més ens convingui. Per tant aquest bloc, té una única entrada, el senyal de rellotge de la placa de control i una única sortida, clkout, que és el senyal de rellotge adaptat a les nostres necessitats

El bloc periode va a continuació del bloc clkdiv i la seva única funció és crear un període de PWM cada 255 rising edge (flanc de pujada) del senyal de rellotge que surt del bloc clkout.

Com a bloc principal hi ha el bloc decode, com la resta de blocs té entrades i sortides. Els bits d'entrada són: els vuit bits de la paraula de control, els vuit bits de posició de l'encoder de l'eix transversal i els vuit bits de posició de l'encoder de l'eix frontal. La funció del bloc decode és detectar el CE de la paraula de control i posteriorment donar a cada motor la consigna que marca la paraula de control.

El següent a analitzar és el bloc posicio, disposem de dos d'aquests blocs, un per cada encoder, en ell interpretem el senyal rebut des de l'encoder i l'adaptem al format que pot interpretar el bloc decode. Dins aquest bloc hi ha un contador de pulsos de l'encoder de 16 bits que es va incrementant o decrementant automàticament segons el motor giri cap a un sentit o cap a l'altre. Tenint en compte el recorregut de 500mm del desplaçament transversal i les característiques de l'encoder HEDS 5540, estem obligats a treballar amb 16 bits (veiem els motius a la taula 29) per poder tenir coneixement de la posició del contrapès en qualsevol punt d'aquest recorregut. Encara que després suprimirem la part baixa, no

necessitem resolució d'una centèsima de mil·límetre, la precisió que ens ofereixen 8 bits es tradueix a 2mm en el desplaçament lineal, amb aquesta precisió ja en tenim prou.

Eix	Voltes pinyó=Voltes reductor	Voltes eix motor	Pulsos necessaris	Bits requerits
X	3,25	104	52.000	16
Y	2,25	72	36.000	16

Taula 29. Bits requerits

Per últim tenim el bloc PWM, que com el seu nom indica s'encarrega de proporcionar la velocitat al motor especificat pel decode, per tant tenim 8 blocs PWM, ni més ni menys que un per a cada motor. Per configurar el PWM necessari necessitem dues entrades: el senyal de PWM que obtenim del bloc període i la consigna de velocitat que obtenim del bloc decode. A partir d'aquests dos senyals i de les quatre velocitats configurades dins el bloc s'obté l'amplitud del PWM desitjat.

Seguidament unim aquests blocs convenientment amb línies i busos de dades en l'arxiu esquemàtic principal del projecte, és l'arxiu potencia.bdf. Ja per acabar assignem quin pin de la FPGA correspon a cada entrada i a cada sortida del projecte. L'aspecte final de l'arxiu potencia.bdf és el de la figura 46.

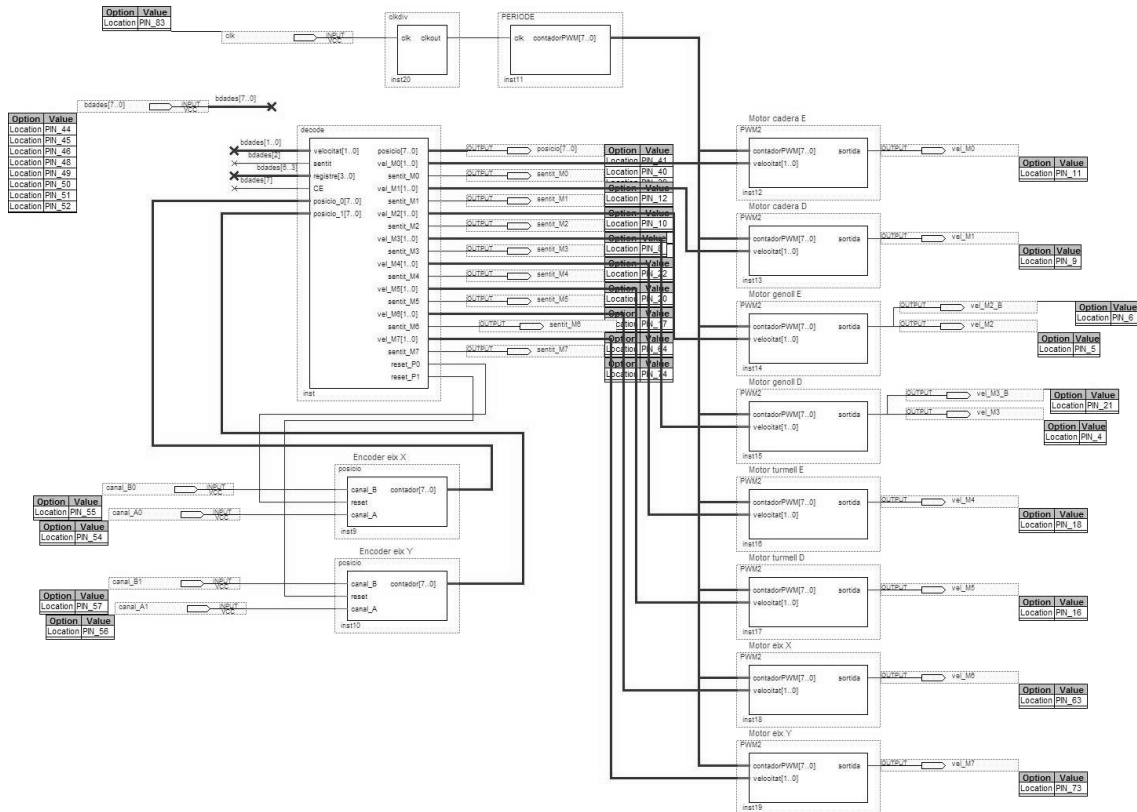


Figura 46. Esquema programa FPGA

4.8.2. Transmissió programa

Finalment cal compilar totalment l'esquema del projecte i ja estarà a punt per ser volcat al dispositiu a programar. Amb el procés de compilació, a més, segons el tamany que ocupa el projecte, el mateix Quartus II t'indica la capacitat i característiques de la FPGA d'Altera que has d'utilitzar. En aquest cas, pel tamany del nostre projecte, el programa selecciona la EPM7128SLC84-15 que té 84 pins dels quals 68 són utilitzables, la resta són pins d'alimentació, programació i control. Aquesta FPGA pertany a la família MAX 7000, basada en la segona generació d'arquitectura MAX d'Altera. Està fabricada amb avançada tecnologia CMOS. La família MAX 7000 està basada en una EEPROM que disposa d'entre 600 i 5000 portes, la resta d'especificacions les podem trobar en la fulla de característiques

d'aquest component. A la figura 47 es mostra la imatge real de la FPGA d'Altera utilitzada, la EPM7128SLC84-15 i la distribució dels pins amb la funció que té cadascun d'ells.

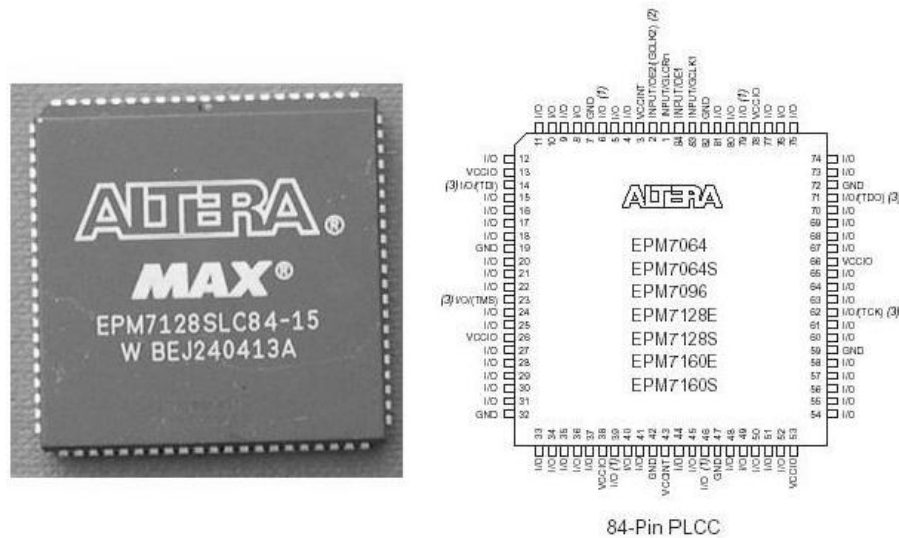


Figura 47. Imatge exterior i Pins-funció FPGA

Hi ha varies opcions per bolcar el programa a la FPGA, nosaltres ho hem fet a través del connector JTAG implementat a la mateixa placa de control, així tenim la possibilitat de reprogramar la FPGA molt fàcilment. Una interfície JTAG és una interfície especial de quatre o cinc pins inclosa al xip, els pins d'aquest connector són els següents:

El TDI (Test Data In), és d'entrada sèrie per dades i instruccions, són desplaçats cap dins cada rising edge del senyal TCK. Aquest senyal necessita ser posat externament a nivell alt

El TDO (Test Data Out), es tracte del pin de sortida sèrie per instruccions i dades. Les dades són expulsades cada falling edge (flanc de baixada) del.TCK. Aquest senyal està en el tercer estat si les dades no han sigut expulsades de l'aparell. (Test Clock),

TMS (Test Mode Select), aquest pin d'entrada controla l'IEEE Std. 1149.1 i TRST (Test Reset) que és opcional i nosaltres no fem servir.

Des de que només una línia de data està disponible, el protocol és necessàriament en sèrie com SPI (Serial Peripheral Interface Bus). El senyal d'entrada del rellotge és al pin TCK. La

configuració es realitza manipulant un bit de la màquina d'estats a través del pin TMS. Un bit de data entra i surt pels pins TDI i TDO, respectivament, per cada pols del rellotge TCK. Es poden carregar diferents tipus d'instruccions: per llegir la identificació del xip, per tenir pins d'entrada, funcions de manipulació del xip... La freqüència d'operació del TCK varia en funció del xip, però típicament va de 10 a 100MHz (10-100ns per bit). A la figura 48 veiem com es connecten aquests senyals al connector IDC de 10 pins.

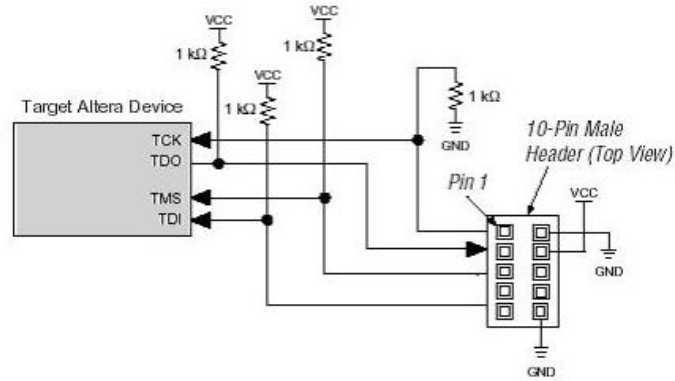


Figura 48. Connexió JTAG

5. SOFTWARE DE CONTROL DEL ROBOT

Pel control del robot necessitem un software senzill que ens permeti visualitzar determinats resultats gràficament. Per exemple, el valor de l'angle de cada part de les cames o el valor de la posició del sistema d'estabilitat.

Per això s'ha trobat convenient seleccionar el software de programació LabWindows CVI.

5.1. Disseny

El programa de control del robot està distribuït amb cinc arxius, quatre de text i un de gràfic. Tots els arxius estan interrelacionats, és a dir, a través d'inclusions es comuniquen entre ells. Hi ha els arxius .c, els .h i el .uir que és el gràfic.

Per dissenyar el software comencem per l'arxiu Robot.h. Conté l'adreça de memòria del PC on hi ha connectada la tarja d'adquisició de dades PCL-812PG, el que anomenem adreça base, hi ha definits tots els motors i articulacions, l'estructura de dades del robot i finalment les funcions del robot. L'arxiu Placa.h conté totes les característiques del panell de control i es genera automàticament al compilar l'arxiu Placa.uir.

L'arxiu Placa.uir és l'arxiu gràfic o el panell de control. Aquest és l'únic arxiu que veu l'usuari final, des d'ell enviem i visualitzem tots els moviments del robot. El panell disposa d'un botó per enviar la o les paraules de control necessàries pel control del robot, un botó per parar tots els motors i un altre per tancar l'aplicació. A més hi ha un timer que a efectes pràctics fa la mateixa funció que un botó, és com si anéssim pitjant un botó repetidament. A l'apartat 5.3 s'explica l'estructura detallada d'aquesta interfície gràfica.

A l'arxiu Placa.c hi ha descrites quines funcions s'executen per cada botó del panell de control.

I per últim, l'arxiu Robot.c conté el cos de les funcions que es criden des de l'arxiu Placa.c.

Així doncs tenim el software dissenyat a partir de varis arxius amb la intenció de crear una estructura ordenada que facilita la modificació del programa. Tot el codi del programa es troba a l'annex B.

5.2. Tarja PCL-812PG

La tarja d'adquisició de dades PCL-812PG ens permet comunicar-nos amb el robot bípede. És una tarja d'alta qualitat, ens proporciona molt bon rendiment, alta velocitat i és multifunció. A la figura 49 podem veure el seu diagrama de blocs.

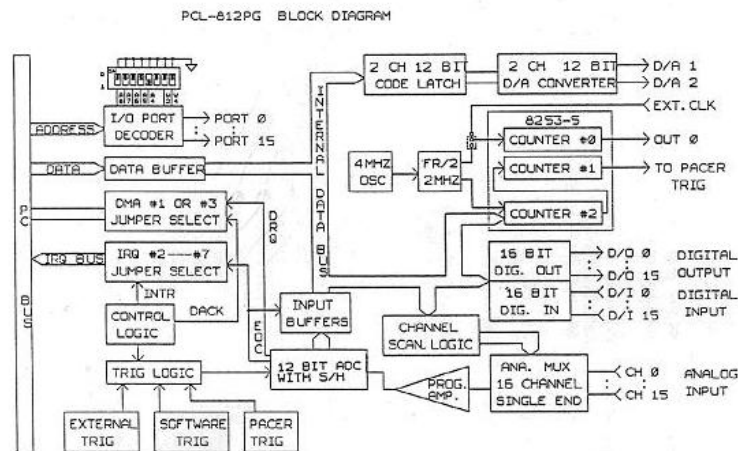


Figura 49. Diagrama de blocs de la tarja PCL-812PG

Les característiques més importants d'aquesta placa són:

16 canals d'entrades analògiques.

Un convertidor industrial estàndard de 12 bits que converteix les entrades analògiques a digitals. El màxim rati de mostreig és de 30KHz en mode DMA.

Rang d'entrades analògiques programable per software.

Bipolar: $\pm 10V$, $\pm 5V$, $\pm 2.5V$, $\pm 1.25V$, $\pm 0.625V$, $\pm 0.3125V$

Un Timer/Counter Programable Intel 8253-5 amb un canal de 16 bits reservat per aplicacions d'usuari.

Dos canals de sortida D/A de 12 bits monolítics.

16 TTL/DTL entrades digitals compatibles i 16 canals digitals de sortida.

En aquest projecte utilitzem 6 entrades analògiques, són necessàries per la lectura dels potenciómetres que porten incorporats els actuadors lineals.

Segons la documentació de la tarja d'adquisició de dades PCL-812PG, l'adreça de port d'entrades i sortides base es selecciona amb un switch de 8 posicions. La tarja necessita 16 adreces consecutives de memòria en l'espai d'entrades i sortides. El rang d'adreces vàlides va des de l'adreça hexadecimal 200 fins a la 3F0. En el nostre cas hem configurat la tarja perquè treballi en el rang d'adreces que van de la 220 a la 22F en hexadecimal.

A la taula 30, podem veure la posició de cada registre de la tarja mapejat a les adreces de port I/O relativa a l'adreça base, i el seu ús. En el nostre cas, hem utilitzat el registres seleccionats de la taula 30.

Posició	Lectura	Escriptura
Base + 0	Counter 0	Counter 0
+ 1	Counter 1	Counter 1
+ 2	Counter 2	Counter 2
+ 3	N/U*	Counter Control
+ 4	A/D byte baix	CH1 D/A byte baix
+ 5	A/D byte alt	CH1 D/A byte alt
+ 6	D/I byte baix	CH2 D/A byte baix
+ 7	D/I byte alt	CH2 D/A byte alt
+ 8	N/U	Clear interrupt request
+ 9	N/U	Gain control
+ 10	N/U	MUX control
+ 11	N/U	Mode control
+ 12	N/U	Software A/D trigger
+ 13	N/U	D/O byte baix
+ 14	N/U	D/O byte alt
+ 15	N/U	N/U

Taula 30. Posició registres respecte adreça BASE

A la figura 50 podem veure l'aspecte extern de la tarja d'adquisició de dades utilitzada, la PCL-812PG.

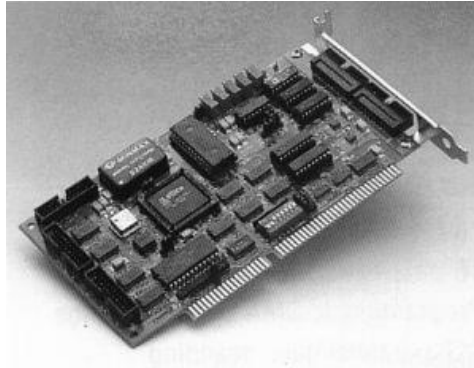


Figura 50. Tarja adquisició de dades PCL-812PG

5.2.1. Connexionat

El connexionat entre la tarja i el PC és d'allò més simple, només cal connectar la tarja al bus ISA (Industry Standard Architecture) del PC, per tant, és imprescindible que el PC que s'utilitza disposi d'aquest tipus de connexió. L'ampla de banda màxim del bus ISA de 16 bits és de 16Mbytes/segon. Aquest ample de banda és insuficient per les necessitats actuals, tals com, targetes de vídeo d'alta resolució, però per a la nostra aplicació és suficient. En els PCs actuals (2006) el bus ISA no existeix, s'ha substituït pel bus PCI (Peripheral Component Interconnect). Per tant, ha calgut buscar un PC amb bus ISA i connectar-hi la tarja.

Per altre banda, com ja hem comentat, la tarja va connectada a la nostre placa de control mitjançant tres connectors tipus IDC amb cables plans de 20 fils.

5.3. Interfície gràfica

Es tracta de l'arxiu Placa.uir del software LabWindows. S'ha creat una pantalla de control on hi tenen cabuda tots els actuadors i sensors del robot. La informació mostrada per pantalla s'actualitza automàticament cada 300ms (ho hem configurat així amb un timer), així podem seguir tots els moviments que està fent el robot, pràcticament en temps real, gràcies a la constant lectura dels seus sensors.

Al centre de tots els indicadors de posició hi tenim una imatge frontal del robot, els indicadors estan col·locats d'acord amb la seva posició real en el robot, així doncs, hi ha l'articulació a governar pràcticament al costat del seu panell de control. Això facilita que en tot moment sapiguem amb quin actuador estem treballant. Podem veure aquesta distribució a la figura 51.

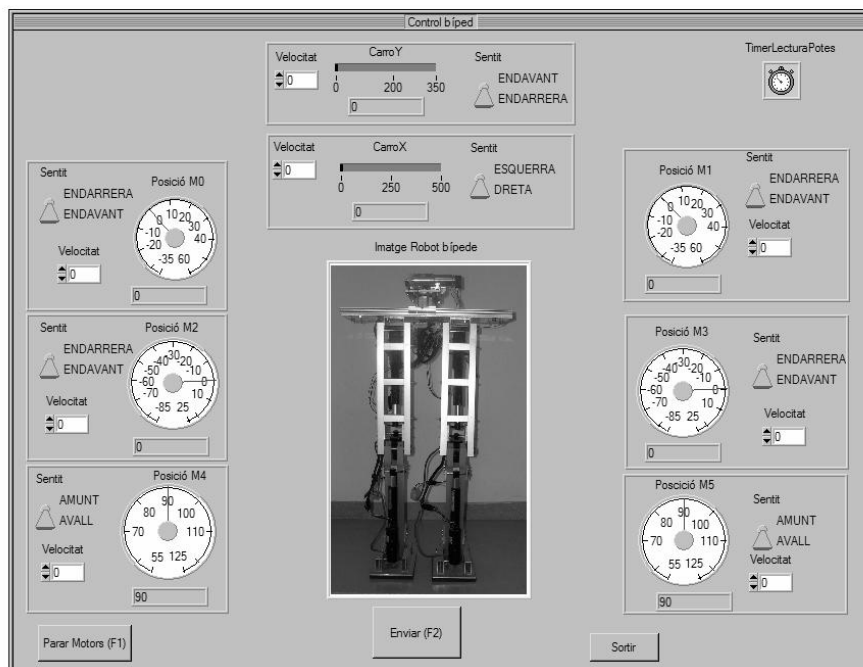


Figura 51. Panell de control del LabWindows

Com veiem, disposem d'un visualitzador circular per l'angle de cadascun dels sis graus de llibertat de les cames, tres a cada cama. Com a valor inicial tenen la seva posició de repòs. Per apreciar el desplaçament longitudinal que realitza el sistema d'estabilitat disposem de dos barres, la primera es desplaça imitant el desplaçament de l'articulació transversal i la segona imitant el desplaçament de l'articulació frontal.

5.4. Control actuadors

Per controlar els actuadors ho fem amb la paraula de control, de la que ja coneixem les característiques.

Els actuadors es controlen amb la funció `Activar_Motor`, se'n pot veure el codi a l'annex B. Aquesta funció s'activa cada vegada que es pitja el botó `Enviar` del panell de control. Primer de tot resseteja el contingut de la paraula de control, el `Valor_Registre`, i seguidament agafa els paràmetres: velocitat, sentit i selecció de la pantalla del PC, els ordena segons la paraula de control, quan tot està ben posat, activa el CE i els envia a la FPGA perquè puguin ser interpretats.

Així doncs, amb aquesta funció tan simple s'envia la consigna de velocitat i sentit a cadascun dels motors, ja siguin lineals o rotatius.

5.5. Lectura sensors

Les funcions de lectura i visualització dels valors obtinguts pels sensors es criden amb un interval de 100ms fixat per un timer. Aquest valor es pot modificar fàcilment segons convingui.

Al disposar de dos tipus de sensors diferents: encoders digitals pels actuadors rotatius i potenciómetres analògics pels actuadors lineals, estem obligats a crear dues funcions diferents per tractar les dades.

Per a les articulacions frontal i transversal s'ha creat la funció `Llegir_Posició_Carro`, cada vegada que es rep l'ordre, es llegeix la posició dels dos eixos consecutivament. Primer es llegeixen els 8 bits de posició de l'eix transversal i s'envia el valor perquè pugui ser visualitzat per pantalla i seguidament es fa el mateix per l'eix frontal. Per calcular la posició en la que es troba el contrapès ho fem amb els encoders (HEDS 5540) que són de tipus incremental. Per llegir una posició amb un encoder incremental és imprescindible tenir un punt de referència conegut des del qual farem tots els càlculs. Quan s'inicialitza el software de control, el PC es queda esperant durant 5 segons que serveixen perquè el motor del grau de llibertat frontal i el del grau de llibertat transversal se situïn a un extrem de la guia a velocitats 1 i 2 respectivament. Seguidament la FPGA resseteja el valor dels encoders i a partir d'aquest punt conegut s'aniran sumant o restant polsos segons el contrapès es

desplaci cap a una direcció o cap a l'altra, així es coneix la posició del contrapès en tot moment.

La funció `Llegir_Angle_Articulacions` és l'equivalent per les articulacions del maluc, del genoll i del turmell. La seva funció és llegir el valor de totes les articulacions consecutivament. En aquest cas com que els valors dels potenciòmetres són analògics s'han de convertir a valors digitals, per fer-ho s'utilitza el convertidor A/D de la base +12, el bit 4 de la base +5 ens avisa quan la conversió analògica – digital s'ha completat. En aquest punt ja disposem del valor digital de l'articulació, però per poder fer-ho més entenedor ens interessa disposar del valor de l'angle de l'articulació. Per últim apliquem l'equació de la recta que relaciona el valor digital amb el valor en graus, així es pot visualitzar per pantalla el desplaçament de l'articulació en graus.

5.5.1. Articulació frontal

Com ja hem comentat, aquesta articulació serveix per dotar de moviment frontal a l'estructura d'estabilització del robot. La longitud de la guia d'aquest actuator és de 350 mm i tot i que es necessiten 16 bits per poder llegir tota la longitud, disposem de 8 bits reals de precisió. Això fa que tinguem una resolució de 1,36 mm, $(350/256)$ més que suficient per la nostre aplicació.

A la figura 52 podem observar on es troba el patí del desplaçament frontal a mesura que es van llegint pulsos.

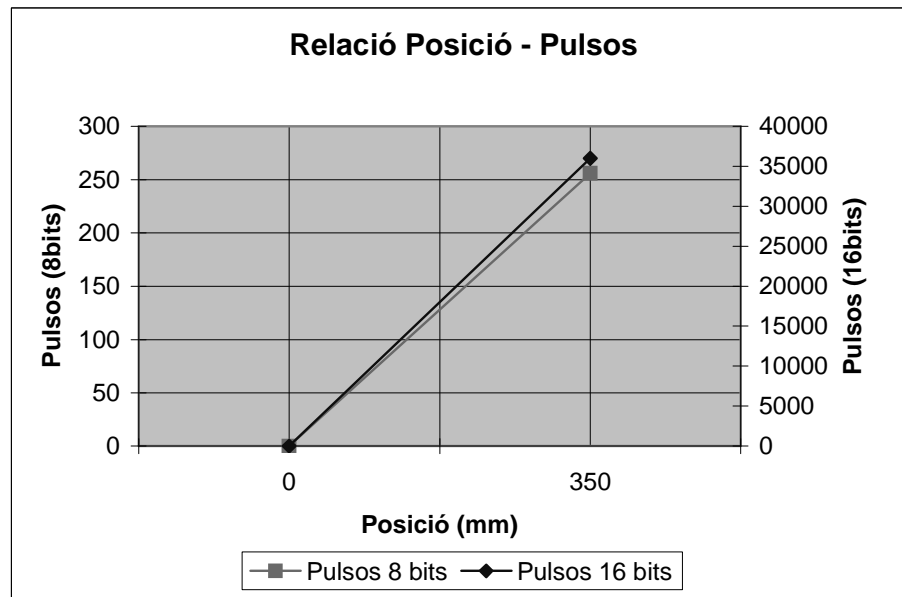


Figura 52. Gràfic relació Posició-Pulsos 8bits-Pulsos 16 bits articulació frontal

Com es pot observar tenim dues lectures de pulsos, una amb 8bits i una altra amb 16bits. La de 16bits és la que fem amb la FPGA, llegim tots els pulsos, per tant necessitem aquests 16bits per poder controlar els 350mm de desplaçament frontal, però com que a la placa de control hi ha 8bits, que ja són suficients, el bloc posicio de la FPGA el passa a 8bits eliminant els 8 bits baixos. A la figura 52 també s'hi aprecia la representació dels 8bits.

Per aconseguir que per pantalla es visualitzi la posició amb mil·límetres d'aquest grau de llibertat cal trobar l'equació que relaciona la posició de l'articulació frontal en mil·límetres amb la lectura de l'encoder adaptada a 8 bits, es tracte de l'Equació 6.

$$Posició(mm) = lectura(8bits) \cdot 1,758793 \quad (\text{Eq. 6})$$

Aplicant l'Equació 1 al lloc que li correspon de l'arxiu "Robot.c" del programa LabWindows, es visualitzarà la posició del carro en el panell de control, concretament on hi ha la barra anomenada Carro Y (s'ha observat a la figura 51).

5.5.2. Articulació transversal

Aquesta és l'articulació que dota de moviment transversal a l'estructura d'estabilització del robot. Aquesta articulació disposa d'una longitud de 500mm pel desplaçament transversal, per tant, adaptant els càlculs del subapartat anterior tindrem una precisió d'aproximadament 2mm, també suficient.

Com en el cas de l'articulació frontal també s'utilitzen, primer 16 bits i seguidament el bloc posicio de la FPGA els adapta als 8 bits disponibles, a la figura 53 podem observar aquesta relació.

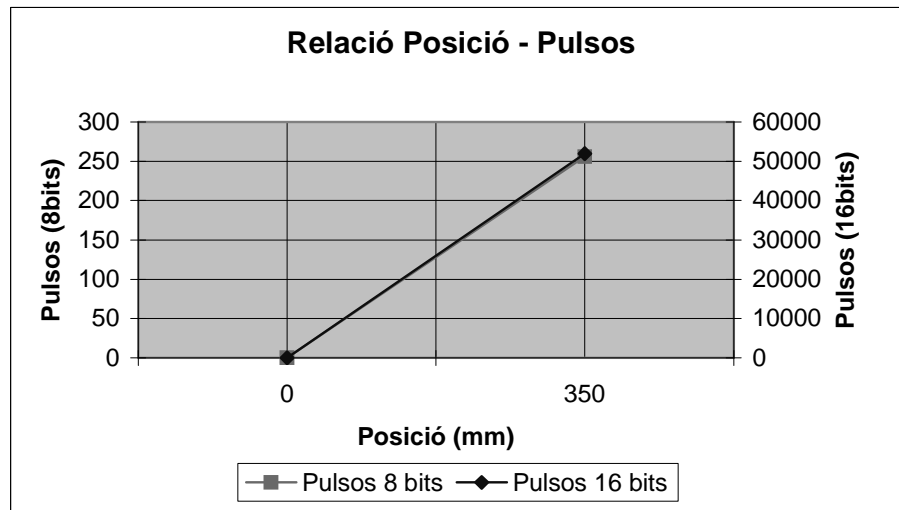


Figura 53. Gràfic relació Posició-Pulsos 8bits-Pulsos 16 bits articulació transversal

A nivell de representació, la única diferència entre aquest grau de llibertat i el frontal és que aquest té un desplaçament disponible de 500mm.

En aquest cas, l'Equació 7 serveix per relacionar la posició de l'articulació transversal en mil·límetres amb la lectura de l'encoder adaptada a 8 bits.

$$Posició(mm) = lectura(8bits) \cdot 2,369668 \quad (\text{Eq. 7})$$

5.5.3. Articulació maluc

Primer de tot cal dir que considerem les dues articulacions del maluc com una de sola ja que tenen idèntiques característiques, també ho farem amb les articulacions dels genolls i amb les dels turmells.

A la figura 54 veiem un petit esquema per situar el rang de desplaçament que té l'articulació, és a dir, veiem a quina posició ens estem referint quan es parla d'un angle determinat.

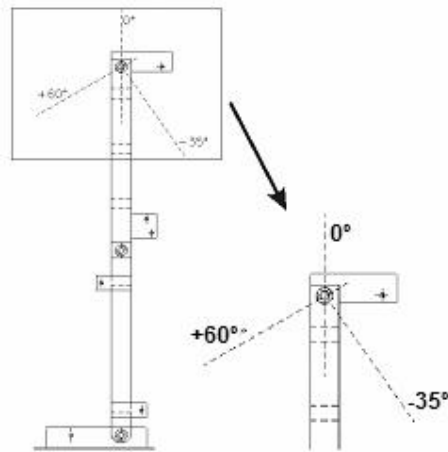


Figura 54. Rang desplaçament Maluc

Com veiem a la figura, el grau de llibertat del maluc és de 95° de recorregut i quan s'activa desplaça tota la cama.

Disposem de 12 bits per realitzar aquesta lectura, així doncs tenim 4096 nivells de posició, per un recorregut de 95° , per tant tenim una precisió d'aproximadament $0,02^\circ$. Tenint en compte que els actuadors són de corrent continu, la precisió queda limitada per la capacitat de posicionament dels motors, no per la lectura de la seva posició. Per tant, la precisió obtinguda és més que suficient per nosaltres.

A la figura 55 s'hi observa el gràfic que mostra com varia el valor analògic del potenciòmetre de l'articulació del maluc a mesura que aquest modifica la seva posició.

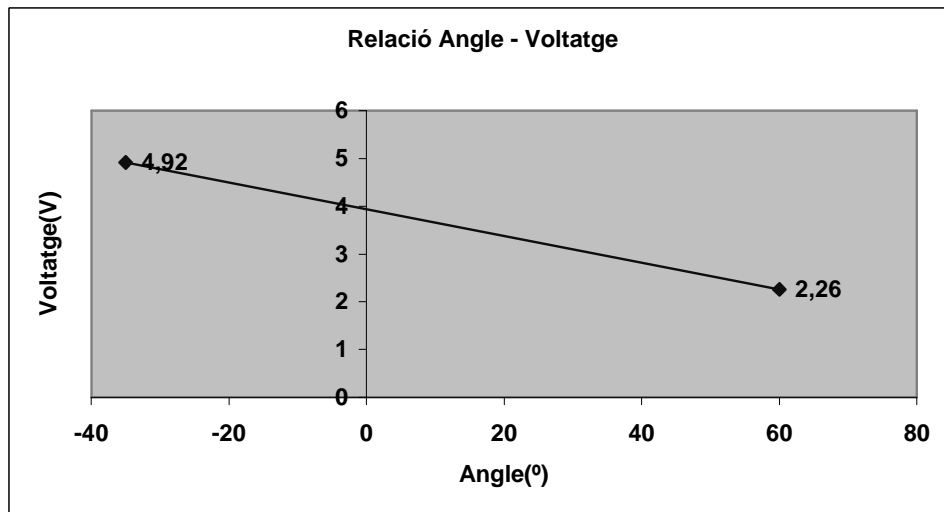


Figura 55. Relació Angle – Voltatge articulació maluc

S'observa que a -35° es llegeix un valor analògic del potenciòmetre de 4,92V, mentre que a l'altre extrem, a 60° llegim 2,26V.

En totes les articulacions de les cames s'agafa el valor analògic amb la tarja d'adquisició de dades i, com sabem, la tarja converteix aquest valor analògic amb un de digital de 8bits.

Finalment, col·locant convenientment l'Equació 8 a l'arxiu "Robot.c" del software de LabWindows s'obté el valor en graus de l'articulació que representem a les figures Relació Angle – Voltatge de cada articulació i que podem visualitzar en el Panell de control del LabWindows.

$$Posició(^{\circ}) = valor_digital(8bits) \cdot (0,0859) + 314,68 \quad (\text{Eq. 8})$$

5.5.4. Articulació genoll

L'articulació del genoll es llegeix exactament igual que la del maluc. Amb la única diferència que aquesta té un recorregut de 110° . Així doncs, si repartim aquests 110° amb els 4096 nivells de posició disponibles ens surt que tenim una precisió d'aproximadament $0,02^\circ$. Superior a la que ens poden donar els actuadors.

A la figura 56 veiem la representació gràfica del moviment disponible per l'articulació del genoll.

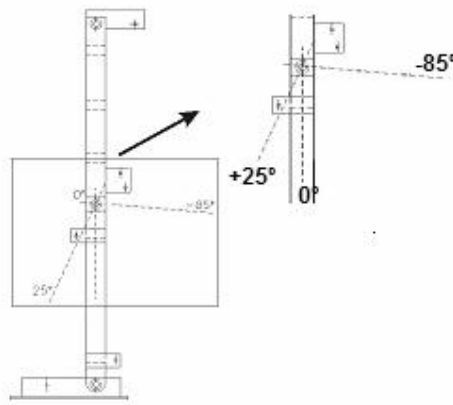


Figura 56. Rang desplaçament Genoll

En aquest cas, quan l'articulació del genoll es troba a -85° es llegeix, del potenciòmetre, un valor analògic de 4,92v i quan l'articulació està a l'altre límit, 25° , s'obtenen 2,26v. Es veu representat a la figura 57.

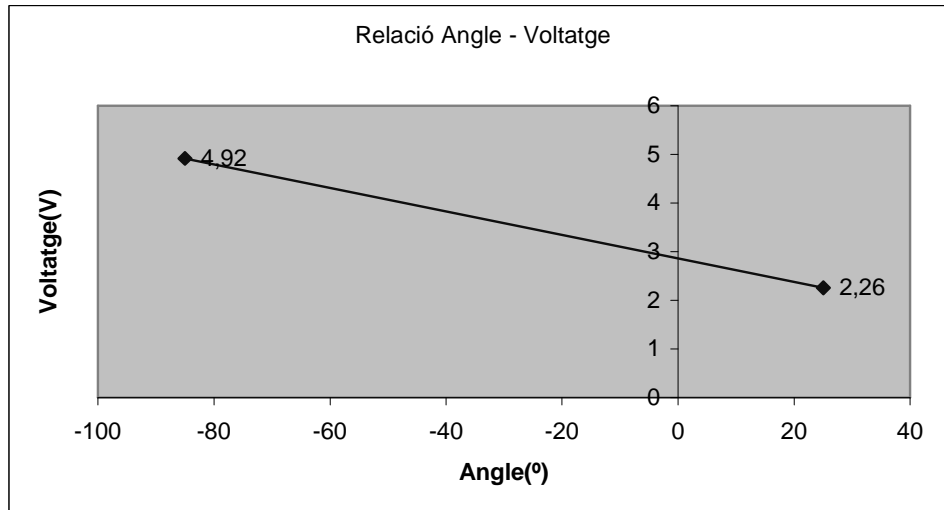


Figura 57. Relació Angle – Voltatge articulació genoll

L'Equació 9 relaciona l'Angle i el Voltatge d'aquesta articulació. Com en el cas del maluc caldrà introduir-la al software de control per visualitzar per pantalla la posició en graus de l'articulació del genoll.

$$Posició(^{\circ}) = valor_digital(8bits) \cdot (-0,0995) + 319,46 \quad (\text{Eq. 9})$$

5.5.5. Articulació turmell

La lectura funciona igual que en els dos casos anteriors. El recorregut d'aquesta articulació és de 70° , per tant encara tenim un valor més elevat de precisió.

Per últim tenim el grau de llibertat del turmell, seguirem el mateix mètode que a les altres dues articulacions. A la figura 58 s'observa el rang de desplaçament d'aquesta articulació i a la figura 59 la relació entre l'angle de posició i el valor analògic que s'obté.

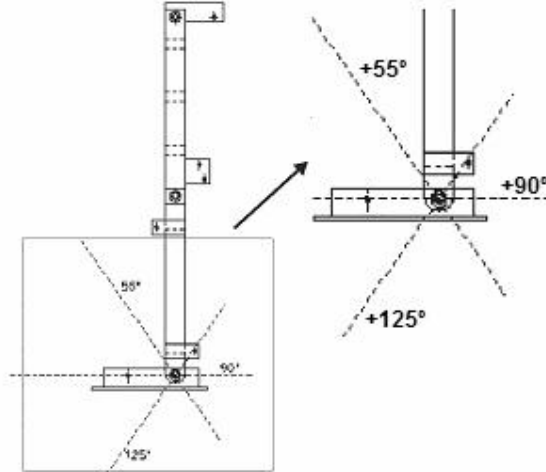


Figura 58. Rang desplaçament Turmell

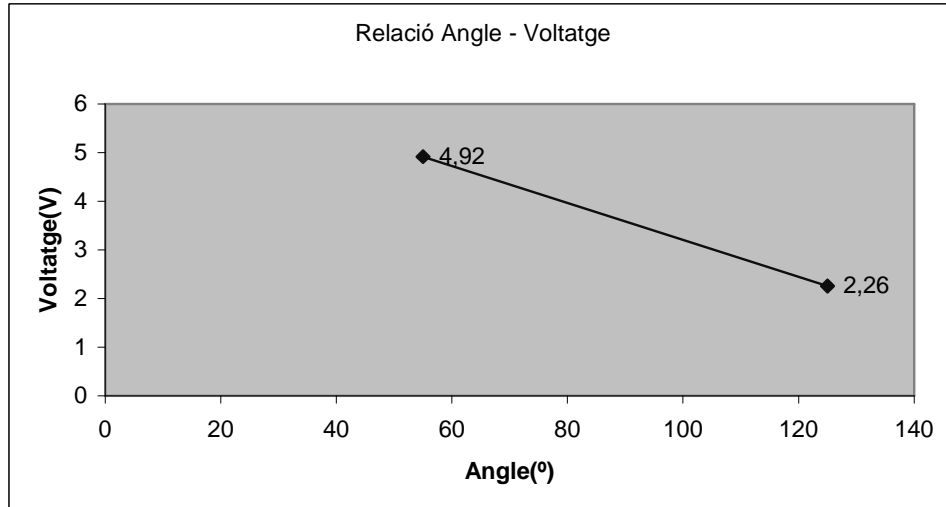


Figura 59. Relació Angle – Voltatge articulació turmell

A la figura 59 es pot observar com a $+55^{\circ}$ obtenim una lectura de 4,92V, i a $+125^{\circ}$ es llegeixen 2,26V

L'Equació 10 relaciona l'Angle i el Voltatge d'aquesta articulació.

$$Posició(^{\circ}) = valor_digital(8bits) \cdot (-0,0621) + 308,04 \quad (\text{Eq. 10})$$

5.6. Resultats

En aquest apartat mostrarem les gràfiques que s'han realitzat a partir de dades reals. Relacionen el desplaçament de cada grau de llibertat amb el temps.

Per adquirir els valors s'ha afegit una comanda d'escriptura dins les instruccions que es realitzen cada activació del timer ajustat a 0,1segons.

Al disposar de 8bits per seleccionar el PWM tenim 255 valors disponibles de velocitat. Amb l'arxiu PWM de la FPGA marquem els límits que ens interessen. Aquests límits estan fixats a 85 per la velocitat 1, 170 per la velocitat 2 i 255 per la velocitat 3.

A nivell de representació de les articulacions de les cames, s'ha optat per no diferenciar entre les articulacions de la cama dreta i les de la cama esquerra, per tant, la representació d'un grau de llibertat engloba la representació de les dues cames. A nivell teòric i a nivell pràctic es comporten exactament igual.

5.6.1. Gràfic GdLI frontal

En aquest cas, com que es comporta exactament igual en els dos sentits, s'han pres mesures amb les tres velocitats disponibles cap a un dels sentits.

A la figura 60 es pot observar com evoluciona la posició del contrapès al llarg del temps segons la velocitat seleccionada.

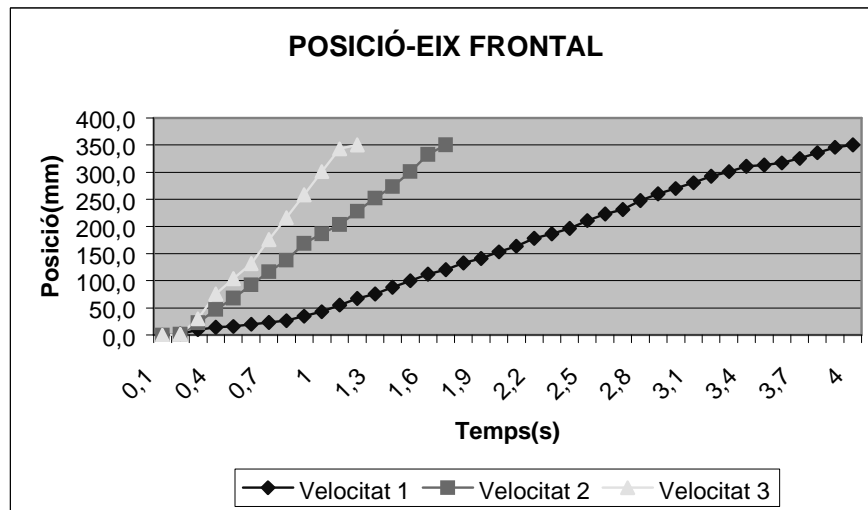


Figura 60. Gràfic velocitats eix frontal

Analitzant el gràfic podem extreure algunes conclusions. Amb la velocitat 1 el contrapès tarda uns 4 segons a fer el desplaçament de 350mm de l'eix frontal, mentre que amb la velocitat 2 tarda uns 2 segons i amb la velocitat 1 una mica més d'1 segon. Pel que fa al pendent de la recta, s'observa com va augmentant a mesura que seleccionem una velocitat més elevada, per tant la velocitat mitjana del desplaçament està augmentant tal i com es mostra a la taula 31.

Velocitat	Desplaçament (mm)	Temps total (s)	Velocitat mitjana(mm/s)
1	350	4,0	87,5
2	350	1,8	194,4
3	350	1,1	318,2

Taula 31. Velocitats mitjanes eix frontal

Si es necessita un moviment suau i precís haurem de seleccionar la velocitat 1, mentre que si el que necessitem és un moviment ràpid i no tan precís seleccionarem la velocitat 2 o 3 segons convingui.

5.6.2. Gràfic GdLI transversal

En aquest cas els resultats són molt semblants als anteriors, amb la diferència que aquest grau de llibertat, el transversal, disposa de 500mm de recorregut i per tant el temps que tarda el contrapès en realitzar tot el recorregut disponible augmenta. A la figura 61 s'observa la representació de les tres velocitats disponibles per aquest desplaçament.

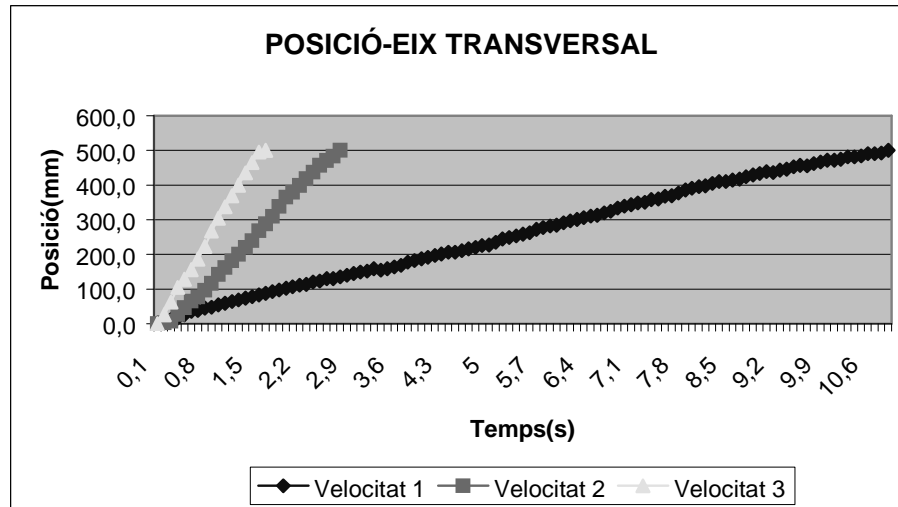


Figura 61. Gràfic velocitats eix transversal

En aquest cas la velocitat 1 és considerablement superior al doble de la velocitat 2, això indica que la potència que s'està donant a velocitat 1, tot i ser correcte, és una mica justa per realitzar el desplaçament. Si fos necessari es podria augmentar la velocitat molt fàcilment, només s'hauria de canviar el límit de la velocitat 1 de l'arxiu PWM de la FPGA i reprogramar la FPGA pel JTAG. En aquests casos queda justificada la utilització d'un dispositiu d'aquestes característiques.

A la figura 60 i a la 61 hem pogut observar el bon funcionament dels encoders. Es pot dir que no hi ha cap error de lectura, els pulsos van augmentant a mesura que hi ha desplaçament d'una forma continua i regular.

A la taula 32 podem observar les velocitats mitjanes del grau de llibertat transversal.

Velocitat	Desplaçament (mm)	Temps total (s)	Velocitat mitjana(mm/s)
1	500	10,9	45,9
2	500	2,8	178,6
3	500	1,6	312,5

Taula 32. Velocitats mitjanes eix transversal

Veiem com les velocitats mitjanes de les velocitats 2 i 3 són molt semblants a les de l'eix frontal. Al disposar d'un reductor acoblat al motor de DC obtenim un parell elevat que fa que el sistema es torni més robust a canvis de pes. Cal tenir en compte que el sistema del moviment frontal té un pes de 2.049g.

5.6.3. Gràfic GdLI maluc

Per donar més informació sobre el funcionament dels graus de llibertat del maluc, del genoll i del turmell s'han realitzat dos tipus de mesures. S'ha mesurat el desplaçament de cada grau de llibertat en un sentit i en l'altre, a diferència dels graus de llibertat frontal i transversal. En el cas del maluc, del genoll i del turmell, el temps d'execució del moviment varia si es realitza en un sentit o en un altre.

La figura 54 (Rang desplaçament Maluc) pot ajudar a interpretar el gràfic de la figura 62.

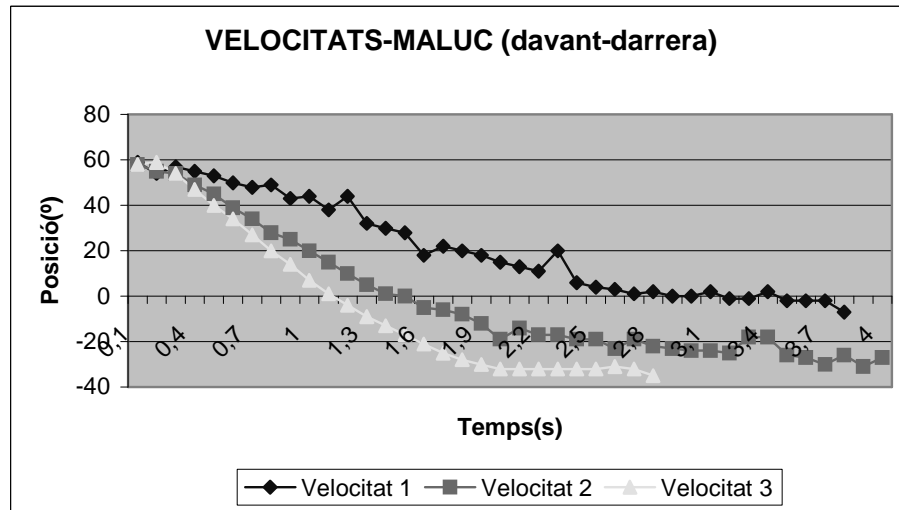


Figura 62. Gràfic velocitats MALUC davant-darrera

Sobretot a la velocitat 1 s'observa com algun dels valors és erroni, ja que el desplaçament es continu i lineal. Aquestes oscil·lacions són causa dels jocs mecànics que tenen les cames. Per tant, es consideren erronis els valors que no segueixen la tendència marcada pel desplaçament de l'articulació.

Analitzant el gràfic de la figura 62 s'observa com a velocitat 1 i 2 no s'arriba al límit del desplaçament posterior de l'articulació. A aquestes velocitats no es disposa de la suficient potència com per arribar al límit. En aquest grau de llibertat les característiques varien molt si el desplaçament és en un sentit o en l'altre, ha de desplaçar el pes de tota la cama i la gravetat hi juga un paper molt important. També es pot observar com després del pas pels 0° (la vertical), la recta que representa les velocitats perd pendent, l'articulació està començant a notar la força de la gravetat adversa. Només a velocitat 3 podem assolir els 95° de recorregut de l'articulació.

A la figura 63, s'observa el desplaçament de les articulacions en sentit invers, de darrera a davant.

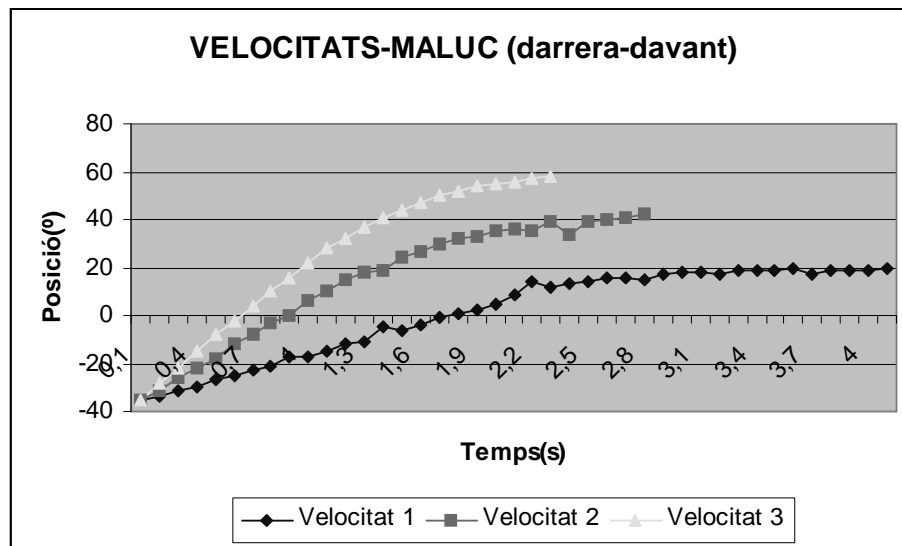


Figura 63. Gràfic velocitats MALUC darrera-davant

En aquest cas també observem una disminució de la velocitat a partir del pas per 0° . Cal dir que totes les mesures d'aquesta articulació s'han fet amb la resta de la cama completament estirada i això genera una gran resistència a desplaçaments en contra de la gravetat.

Dins el rang que va de -4° a 20° , el més important per caminar, podem fer funcionar el grau de llibertat a qualsevol de les tres velocitats disponibles.

5.6.4. Gràfic GdLI genoll

Pel que fa a les mesures d'aquest grau de llibertat s'han realitzat amb el maluc fix a 0° i seguint el mateix mètode emprat en el desplaçament del maluc, primer de davant a darrera i posteriorment de darrera a davant. En aquest cas, la figura 56 (Rang desplaçament Genoll) pot ajudar a interpretar els resultats.

A la figura 64 s'observa la posició de l'angle de l'articulació respecte la variable temps a diferents velocitats.

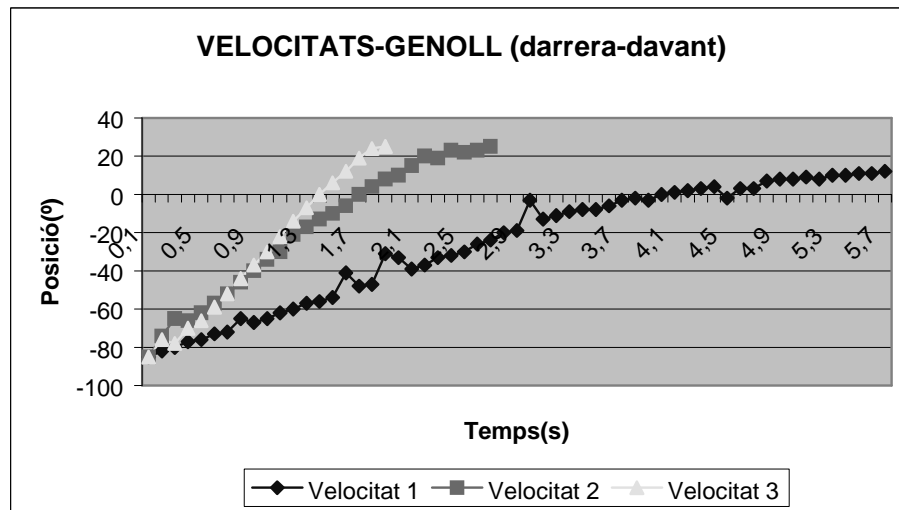


Figura 64. Gràfic velocitats GENOLL darrera- davant

Si s'observa el gràfic de la figura 64 veiem com en aquest cas també trobem tres valors erronis a la velocitat 1, els despraciem. Com a la resta de casos, al passar pel punt 0° observem una lleugera disminució del pendent de les rectes, la força de la gravetat va en contra del desplaçament de l'articulació.

A la figura 65 s'observa el desplaçament del genoll en sentit invers, de darrera a davant.

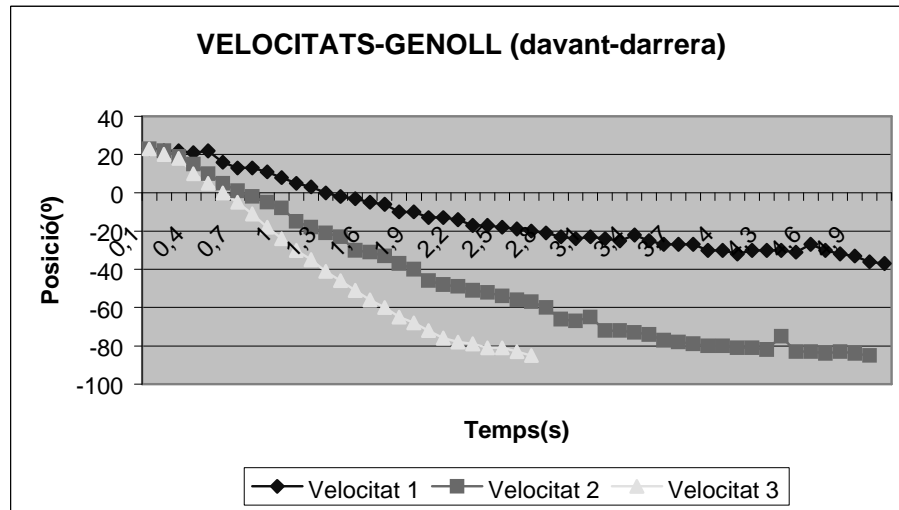


Figura 65. Gràfic velocitats GENOLL davant-darrera

Com a conclusió d'aquest grau de llibertat se'n pot extreure que al desplaçar menys pes (conjunt tibia-peroné més peu) que l'articulació del maluc tenim un rang de desplaçament més ampli amb totes les velocitats, de -37° a 12° , adequat per caminar.

5.6.5. Gràfic GdLI turmell

A la figura 66 observem el gràfic de les velocitats en sentit avall-amunt d'aquest grau de llibertat, per entendre millor el desplaçament es pot observar la figura 58 (Rang desplaçament Turmell).

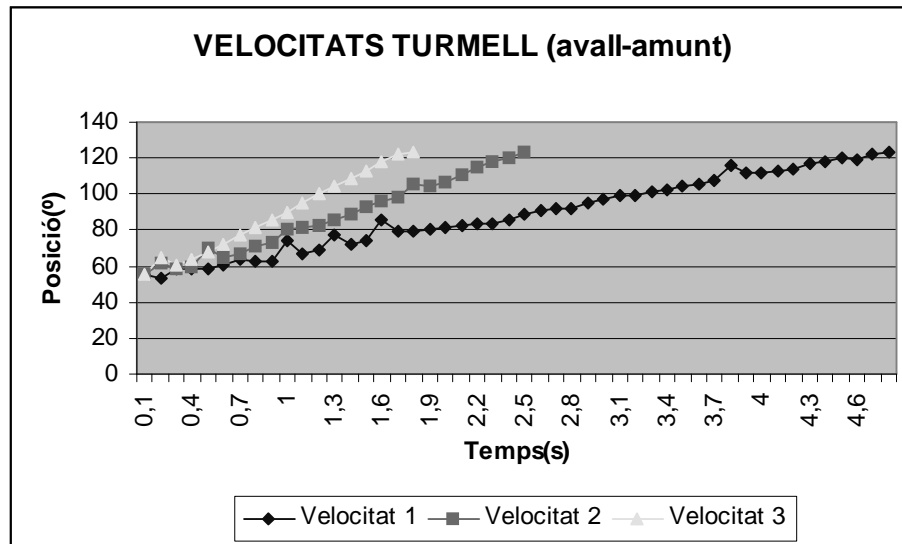


Figura 66. Gràfic velocitats TURMELL avall-amunt

A la figura 67 s'observa la lectura del potenciòmetres convertida a graus respecte el temps de l'articulació del turmell quan es desplaça de $+125^{\circ}$ a $+55^{\circ}$.

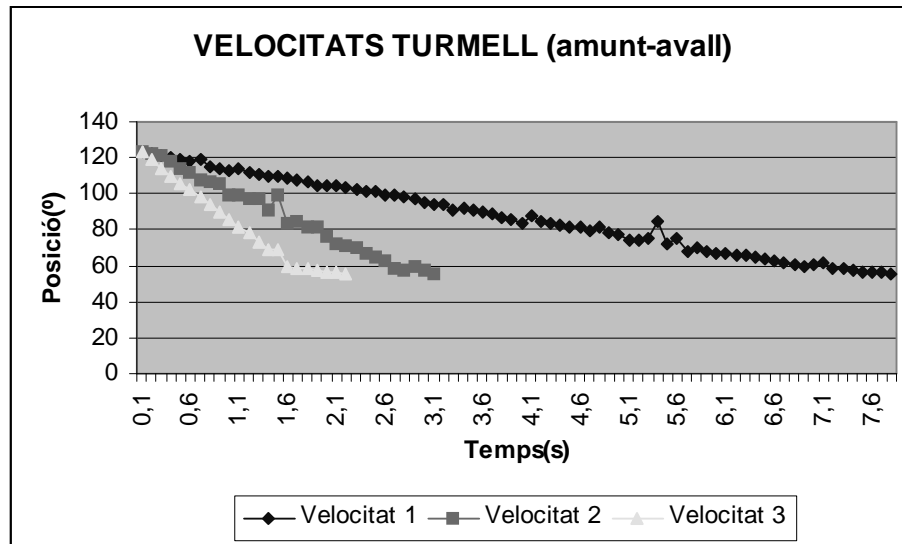


Figura 67. Gràfic velocitats TURMELL amunt-avall

Degut al poc pes que ha de desplaçar aquest grau de llibertat les gràfiques són molt lineals, gairebé no els afecta la força de la gravetat en cap de les velocitats. Això provoca que es disposi d'un rang de desplaçament del 100% a totes les velocitats disponibles.

Per acabar cal dir que totes les dades s'han pres amb una de les cames, a efectes teòrics i pràctics s'obtenen resultats molt semblants i que les cames estaven situades en suspensió gràcies al suport dissenyat per poder fer proves sense tocar al terra tal com es pot observar a la figura 68.

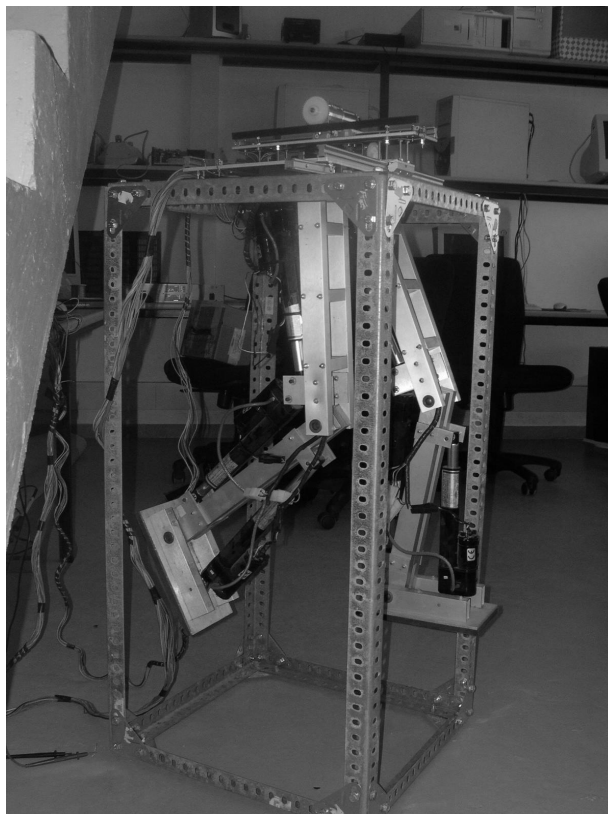


Figura 68. Suport proves i Robot Bípede

6. RESUM DEL PRESSUPOST

Seguint els preus unitaris i pressupostos parcials detallats en el document PRESSUPOST, el cost del present projecte sense IVA és de quatre mil cent trenta-cinc euros amb setanta-un cèntims.

7. CONCLUSIONS

Aquest projecte representa una etapa més dins la investigació i recerca sobre plataformes bípedes que durant els darrers anys s'està duent a terme al departament d'Electrònica, Informàtica i Automàtica de la Universitat de Girona. Amb aquesta tercera remodelació del robot bípede s'ha comprovat que calia introduir millores en l'apartat mecànic i electrònic per poder gaudir d'un òptim funcionament del conjunt.

Dels objectius concrets marcats en aquest projecte se'n poden extreure les següents conclusions:

S'han dissenyat dos graus de llibertat lineals en forma de pla XY per moure un contrapès. Per fer-ho s'ha construït un sistema de guies i patins d'alumini a dos nivells, un nivell pel desplaçament transversal i un altre pel lateral. Per donar tracció al conjunt s'ha utilitzat un mecanisme de pinyó cremallera que proporciona uns resultats molt òptims en quan a seguretat, precisió i suavitat de funcionament. Cadascun dels dos pinyons van accionats per un motor-reductor de corrent continu equipat amb un encoder digital incremental que proporciona molta precisió en la lectura de posició del contrapès.

S'ha dissenyat una placa amb una FPGA Altera de 84 potes que, a més de generar senyals PWM pels vuit motors, llegir els dos encoders i comunicar-se amb un PC proporciona la potència necessària al robot bípede perquè aquest pugui funcionar perfectament. Respecte l'anterior projecte (Salellas, 1999), s'ha aconseguit reduir l'espai ocupat per les plaques de control i potència en aproximadament el 60%, aconseguint donar un pas més cap a l'autonomia total del robot bípede. A més, en quan al seu funcionament, després d'haver realitzat totes les proves necessàries, la placa no presenta cap tipus d'anomalia ni cap falta de fiabilitat. Gràcies al seu bon funcionament es pot dir que s'ha dissenyat una placa que supera els objectius del projecte.

Pel que fa a la realització del software de comprovació, s'ha realitzat amb LabWindows. Per capturar i enviar les consignes a la plataforma bípede s'ha utilitzat la tarja d'adquisició de dades PCL-812PG. S'ha realitzat un programa de control ben estructurat que facilita la manipulació del mateix i permet veure gràficament i en temps real tots els moviments que realitza el robot bípede. S'ha aconseguit verificar el funcionament òptim de tots els actuadors i sensors que té el robot bípede.

S'han confeccionat plànols de totes les parts mecàniques que formen la plataforma bípede, de l'esquemàtic de la placa de control i de les diferents capes que la formen. I s'ha realitzat aquesta memòria explicativa, on queda detallat el funcionament de totes les parts del robot, perquè pugui ser utilitzada com a referència per a futurs projectistes que treballin i/o millorin aquesta plataforma bípede.

Durant la realització del projecte han sorgit problemàtiques que s'han anat resolent aplicant, majoritàriament, nous coneixements, que s'han adquirit durant la realització del projecte i que s'han intentat reflectir en aquesta memòria. Podem dir doncs, que s'han complert satisfactòriament tots els objectius fixats en aquest projecte.

Pere Amadó Codony
Enginyer Tècnic Industrial

Girona, 10 de gener de 2006

8. RELACIÓ DE DOCUMENTS

El projecte consta de 5 documents diferenciats: memòria, plànols, plec de condicions, estat d'amidaments i pressupost.

El primer document, és la memòria del projecte, consta de 111 planes distribuïdes un 5 capítols. Conté un annex amb els codis de programa realitzats, el primer apartat conté el codi de la FPGA realitzat amb el Quartus II i el segon el codi del programa de control fet amb el software LABWINDOWS.

L'apartat de plànols es troba en el segon document i consta de 7 plànols. D'entre el total, 3 són de les diferents estructures que formen el robot i 4 de la placa que s'ha realitzat pel control del robot bípede. Els plànols número 1, número 3 i el número 4 estan en format DIN A3, la resta està en format DIN A4. El document plànols també consta d'un annex que conté 4 plànols en format DIN A4 de les tres parts de cada cama del robot bípede i de l'actuador lineal utilitzat.

El plec de condicions consta de 7 planes on s'expliquen les diferents condicions d'utilització i funcionament de les diferents parts del robot bípede.

L'estat d'amidaments és el document número quatre i consta de 5 planes amb el llistat de tot el material necessari per la realització del projecte ordenat segons la part a la qual corresponen.

L'últim document, el document número cinc, és el pressupost, consta de 9 planes que inclouen els preus unitaris dels material, els pressupostos parcials, el pressupost total i l'annex del pressupost.

9. BIBLIOGRAFIA

ADVANTECH. Documents varis PCL-812PG. (http://www.advantech.com/products/Model_Detail.asp?model_id=1-UC834&BU=&PD=IO), octubre 2005.

ALTERA. Software Quartus II. (https://www.altera.com/support/software/download/soft-download_center.html), setembre 2005.

ALTERA. Documents varis de la FPGA. (<http://www.altera.com/literature/lit-m7k.jsp>), octubre 2005.

ELMEQ. Documents del productes comprats. (<http://www.elmeq.es>), setembre 2005.

LOPEZ VALLEJO, M.L i AYALA RODRIGO J.L. FPGA: Nociones básicas e implementació. (http://www-lsi.die.upm.es/~marisa/docencia/fpga_a2_2004.pdf), novembre 2005.

MAGÍ, LI. Programació de dispositius FPGA's d'Altera. (<http://eia.udg.es/~llmagi/>), setembre 2005.

NATIONAL INSTRUMENTS CORPORATION. LabWindows/CVI. User Interface Reference Manual. Austin. 1994.

ORIGA. Documents de les guies lineals. (<http://www.linearguides.com/LinearGuides.htm>), octubre 2005.

RS AMIDATA. Documents dels productes comprats. (<http://www.amidata.es>), juny - desembre 2005.

SALELLAS, J. Robot bípede walker II. Projecte/ Treball Fi de Carrera. Enginyeria Informàtica. Escola Politècnica Superior. Universitat de Girona. Setembre 1999.

ST. Datasheet MOSFET IRF 540. (<http://www.st.com/stonline/products/literature/ds/9387.pdf>), octubre 2005.

A. ANNEX CODI PROGRAMES

Aquest annex conté el codi dels dos programes que ha calgut fer per tal de poder controlar el robot bípede. El primer capítol conté el codi del programa de la FPGA i el segon el codi del programa de control.

A.1. Programa FPGA

Aquest capítol conté tot el programa que incorpora la FPGA. S'ha realitzat amb el software Quartus II. Està estructurat amb diferents arxius amb la seva part de codi. Cadascun d'aquests arxius estan interconnectats convenientment.

A.1.1. Arxiu clkdiv.vhd

Aquest arxiu adapta el senyal del rellotge de la placa de control que és de 20MHz a 10MHz, que és el que interessa. Amb aquest arxiu podem adaptar el rellotge de la placa de control a la freqüència que ens interressi.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;

ENTITY clkdiv IS
    PORT(
        clk: in std_logic;
        clkout: out std_logic
    );
END clkdiv;

ARCHITECTURE arch_clkdiv OF clkdiv IS
    signal contador: integer range 0 to 4;
BEGIN
    process(CLK)
    begin
        if rising_edge(CLK) then
            if contador<2 then
                contador<=contador+1;
                clkout<='1';
            elsif contador<4 then
                contador<=contador+1;
                clkout<='0';
            else
                contador<=0;
            end if;
        end if;
    end process;
end arch_clkdiv;
```

```

                contador<=0;
            end if;
        end if;
    end process;

```

```
end arch_clkdiv;
```

A.1.2. Arxiu decode.vhd

La funció de l'arxiu decode és detectar el CE de la paraula de control i posteriorment donar a cada motor la consigna que marca la paraula de control.

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;

ENTITY decode IS
    PORT(
        velocitat: in std_logic_vector (1 downto 0);
        sentit: in std_logic;
        registre: in std_logic_vector (3 downto 0);
        CE: in std_logic;
        posicio_0: in std_logic_vector (7 downto 0);
        posicio_1: in std_logic_vector (7 downto 0);

        posicio: out std_logic_vector (7 downto 0);
        vel_M0: out std_logic_vector (1 downto 0);
        sentit_M0: out std_logic;
        vel_M1: out std_logic_vector (1 downto 0);
        sentit_M1: out std_logic;
        vel_M2: out std_logic_vector (1 downto 0);
        sentit_M2: out std_logic;
        vel_M3: out std_logic_vector (1 downto 0);
        sentit_M3: out std_logic;
        vel_M4: out std_logic_vector (1 downto 0);
        sentit_M4: out std_logic;
        vel_M5: out std_logic_vector (1 downto 0);
        sentit_M5: out std_logic;
        vel_M6: out std_logic_vector (1 downto 0);
        sentit_M6: out std_logic;
        vel_M7: out std_logic_vector (1 downto 0);
        sentit_M7: out std_logic;
        reset_P0: out std_logic;
        reset_P1: out std_logic
    );
END decode;

ARCHITECTURE arch_decode OF decode IS

BEGIN

```

```
process(CE)
begin
    if rising_edge(CE) then
        case registre(3 downto 0) is
            when "0000" =>
                vel_M0<=velocitat;
                sentit_M0<=sentit;

            when "0001" =>
                vel_M1<=velocitat;
                sentit_M1<=sentit;

            when "0010" =>
                vel_M2<=velocitat;
                sentit_M2<=sentit;

            when "0011" =>
                vel_M3<=velocitat;
                sentit_M3<=sentit;

            when "0100" =>
                vel_M4<=velocitat;
                sentit_M4<=sentit;

            when "0101" =>
                vel_M5<=velocitat;
                sentit_M5<=sentit;

            when "0110" =>
                vel_M6<=velocitat;
                sentit_M6<=sentit;

            when "0111" =>
                vel_M7<=velocitat;
                sentit_M7<=sentit;

            when "1000" =>
                posicio<=posicio_0;

            when "1001" =>
                posicio<=posicio_1;

            when "1010" =>
                if velocitat(1 downto 0)="01" then
                    reset_P0<='1';

                    elsif velocitat(1 downto 0)="10" then
                        reset_P1<='1';

                    else reset_P0<='0';
                        reset_P1<='0';

                end if;

            when others =>
                null;
        end case;
    end if;
end process;
```

```
end process;

end arch_decode;
```

A.1.3. Arxiu periode.vhd

L'arxiu periode va a continuació del bloc clkdiv i la seva única funció és crear un període de PWM cada 255 rising edge (flanc de pujada) del senyal de rellotge que surt de l'arxiu clkout.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;

ENTITY PERIODE IS
    PORT(
        clk: in std_logic;
        contadorPWM: buffer integer range 0 to 255
    );
END PERIODE;

ARCHITECTURE arch_PERIODE OF PERIODE IS

BEGIN

    process(CLK)
    begin
        if rising_edge(CLK) then
            if contadorPWM<255 then
                contadorPWM<=contadorPWM+1;
            else
                contadorPWM<=0;
            end if;
        end if;
    end process;

end arch_PERIODE;
```

A.1.4. Arxiu posicio.vhd

En aquest arxiu s'interpreta el senyal rebut des de l'encoder i s'adapta al format que pot interpretar l'arxiu decode.

```
LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;

ENTITY posicio IS
```

```

PORT(
    canal_B: in std_logic;
    reset: in std_logic;
    canal_A: in std_logic;

    contador: buffer unsigned (7 downto 0)

);

end posicio;

ARCHITECTURE arch_posicio OF posicio IS
    signal contap: unsigned (15 downto 0);           --contador pulsos

BEGIN

    process(canal_A,reset)
    begin
        if reset='1' then
            contap<=conv_unsigned(0,16); --posem el contap a 0
        elsif rising_edge(canal_A) then           --si detectem rising
                                                    --edge a
canalA i canal B                                     --és 0
incrementem el contador                               --i si és
1 el decrementem
            if canal_B='0' then
                contap<=contap+1;
            else
                contap<=contap-1;
            end if;
        end if;
    end process;

    contador (7 downto 0)<=contap (15 downto 8);    --com que realment
disposem
    --de 8 bits de posció digital,
    --els 8 bits baixos del contap els suprimim
    --i ens quedem amb els 8 més significatius
end arch_posicio;

```

A.1.5. Arxiu PWM.vhd

Aquest arxiu s'encarrega de proporcionar la velocitat al motor especificat per l'arxiu decode.

```

LIBRARY IEEE;
USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_ARITH.ALL;

ENTITY PWM IS
    PORT(

```

```
        contadorPWM: in integer range 0 to 255;
        velocitat: in std_logic_vector(1 downto 0);
        sortida: out std_logic

    );

END PWM;

ARCHITECTURE arch_PWM OF PWM IS

    signal limit: integer range 0 to 255;

BEGIN

    process(velocitat)
    begin
        if velocitat(1 downto 0)="00" then
            limit<=255;
        elsif velocitat(1 downto 0)="01" then
            limit<=170;
        elsif velocitat(1 downto 0)="10" then
            limit<=85;
        else
            limit<=0;
        end if;

    end process;

    process(contadorPWM, limit)
    begin
        if limit=0 then
            sortida<='0';
        elsif contadorPWM<=limit then
            sortida<='1';
        else
            sortida<='0';
        end if;
    end process;

end arch_PWM;
```

A.2. CODI DEL PROGRAMA DE CONTROL LABWINDOWS

Aquest apartat conté els tots i cadascun del arxius que componen el programa de control realitzat amb el software LabWindows.

A.2.1. Arxiu Placa.c

A l'arxiu Placa.c hi ha descrites quines funcions s'executen per cada botó del panell de control.

```
#include <ansi_c.h>
#include <utility.h>
#include <userint.h>
#include "Placa.h"
#include "Robot.h"

Vbiped robot;
int panel;
int panel0;

void main(void)
{
    int i;
    outp(BASE + 11, 0x01); //mode trigger per software
    for (i=0;i<6;i++)      // resetegem el valor de tots els angles
        robot.Angle[i]=0;

    panel0 = LoadPanel(0, "Placa.uir",INICIAL);
    DisplayPanel(panel0);
    Delay (5);                //la màquina es queda
esperant 5segons
    Activar_Motor(CARRO_X, 2, 0);      // mentre,els dos motors van a
la posició inicial
    Activar_Motor(CARRO_Y, 1, 0);     //el transversal a velocitat 2 i el
frontal a velocitat 1
    HidePanel (INICIAL);

    panel = LoadPanel (0, "Placa.uir", PANEL);
    DisplayPanel(panel);
    RunUserInterface();
}

int CVICALLBACK Enviar (int panel, int control, int event,
                        void *callbackData, int eventData1, int eventData2)
{
    int i;

    switch (event)
    {
        case EVENT_COMMIT:
```

```

        GetCtrlVal (panel, PANEL_VELOCITAT_M0,
&robot.Velocitat[0]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M1,
&robot.Velocitat[1]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M2,
&robot.Velocitat[2]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M3,
&robot.Velocitat[3]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M4,
&robot.Velocitat[4]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M5,
&robot.Velocitat[5]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M6,
&robot.Velocitat[6]);
        GetCtrlVal (panel, PANEL_VELOCITAT_M7,
&robot.Velocitat[7]);
        GetCtrlVal (panel, PANEL_SENTIT_M0, &robot.Sentit[0]);
        GetCtrlVal (panel, PANEL_SENTIT_M1, &robot.Sentit[1]);
        GetCtrlVal (panel, PANEL_SENTIT_M2, &robot.Sentit[2]);
        GetCtrlVal (panel, PANEL_SENTIT_M3, &robot.Sentit[3]);
        GetCtrlVal (panel, PANEL_SENTIT_M4, &robot.Sentit[4]);
        GetCtrlVal (panel, PANEL_SENTIT_M5, &robot.Sentit[5]);
        GetCtrlVal (panel, PANEL_SENTIT_M6, &robot.Sentit[6]);
        GetCtrlVal (panel, PANEL_SENTIT_M7, &robot.Sentit[7]);
        for(i=0;i<MAXMOTORS;i++)
            Activar_Motor(i, robot.Velocitat[i],
robot.Sentit[i]);

        break;
    }
    return 0;
}

int CVICALLBACK Sortir (int panel, int control, int event,
void *callbackData, int eventData1, int eventData2)
{
    int i;

    switch (event)
    {
        case EVENT_COMMIT:
            for(i=0; i<MAXMOTORS;i++)
                Activar_Motor(i, 0, 0);
            QuitUserInterface (0);
            break;
    }
    return 0;
}

int CVICALLBACK LecturaPotes (int panel, int control, int event,
void *callbackData, int eventData1, int eventData2)
{
    switch (event)
    {
        case EVENT_TIMER_TICK:
            SetCtrlAttribute (panel, PANEL_TIMER, ATTR_ENABLED, 0);
            // OFF timer
            Llegir_Angle_Articulacions();
            // actualitza el valor de l'angle de les articulacions

```



```

        Llegir_Posicio_Carro();
        // actualitza la posició XY del carro
        Comprovar_Limits_Motors();
        //comprova que els motors estiguin dins els seus límits

        SetCtrlVal (panel, PANEL_ANGLE0, robot.Angle[MALUC_E]);
        // mostrem per pantalla el valor del angles
        SetCtrlVal (panel, PANEL_VOLTS0,
robot.Voltatge[MALUC_E]);
        SetCtrlVal (panel, PANEL_ANGLE1, robot.Angle[MALUC_D]);
        // i de la posició dels carros
        SetCtrlVal (panel, PANEL_VOLTS1,
robot.Voltatge[MALUC_D]);
        SetCtrlVal (panel, PANEL_ANGLE2, robot.Angle[GENOLL_E]);
        SetCtrlVal (panel, PANEL_VOLTS2,
robot.Voltatge[GENOLL_E]);
        SetCtrlVal (panel, PANEL_ANGLE3, robot.Angle[GENOLL_D]);
        SetCtrlVal (panel, PANEL_VOLTS3,
robot.Voltatge[GENOLL_D]);
        SetCtrlVal (panel, PANEL_ANGLE4, robot.Angle[TURMELL_E]);
        SetCtrlVal (panel, PANEL_VOLTS4,
robot.Voltatge[TURMELL_E]);
        SetCtrlVal (panel, PANEL_ANGLE5, robot.Angle[TURMELL_D]);
        SetCtrlVal (panel, PANEL_VOLTS5,
robot.Voltatge[TURMELL_D]);

        SetCtrlVal (panel, PANEL_CARRO0, robot.Real[CARRO_X]);
        SetCtrlVal (panel, PANEL_POLSX,
robot.Posicio_Carro[CARRO_X]);
        SetCtrlVal (panel, PANEL_CARRO1, robot.Real[CARRO_Y]);
        SetCtrlVal (panel, PANEL_POLSY,
robot.Posicio_Carro[CARRO_Y]);

        SetCtrlAttribute (panel, PANEL_TIMER, ATTR_ENABLED, 1);
        // ON timer

        break;
    }
    return 0;
}

int CVICALLBACK Parar_Tots_Motors (int panel, int control, int event,
void *callbackData, int eventData1, int eventData2)
{
    switch (event)
    {
        case EVENT_COMMIT:
            Activar_Motor(MALUC_E, 0, 0);
            Activar_Motor(MALUC_D, 0, 0);
            Activar_Motor(GENOLL_E, 0, 0);
            Activar_Motor(GENOLL_D, 0, 0);
            Activar_Motor(TURMELL_E, 0, 0);
            Activar_Motor(TURMELL_D, 0, 0);
            Activar_Motor(6, 0, 0);
            Activar_Motor(7, 0, 0);
            break;
    }
    return 0;
}

```

```
}

```

A.2.2. Arxiu Robot.c

L'arxiu Robot.c conté el cos de les funcions que es criden des de l'arxiu Placa.c.

```
#include <userint.h>
#include <utility.h>
#include "Robot.h"

Vbiped robot;

// funcio per activar el canal del AD
void Seleccionar_Canal_AD(int NCanal)
{
    int i;

    outp(BASE + 10, NCanal);    // activem canal AD a llegir
    for(i=0;i<3;i++);          // petit retard
}

// actualització de l'angle de les articulacions
void Llegir_Angle_Articulacions(void)
{
    unsigned int resultat, arti;

    // lectura dels angles de les articulacions
    outp(BASE + 11, 0x01);    //trigger per soft
    for(arti=0;arti<6;arti++) // per totes les articulacions...

    {
        Seleccionar_Canal_AD(arti); // seleccionem el canal AD
        corresponent a l'articulació
        outp(BASE + 9, 0x00);    //Guany a 1
        outp(BASE + 12, 0x01);    //disparem el convertidor
        resultat=inp(BASE + 5);    // llegim
        el valor de data ready i esperem fins
        while ((resultat & 0x10)!=0) //que es posa a
        0, ja ha fet la conversió

        {
            resultat=inp(BASE + 5);
        }

        robot.Voltatge[arti]=inpw(BASE + 4) & 0xFFF;
        //llegim la base +4 i la superior o sigui la +5

        // passem de tensió(12bits) a angle (funcio que passa)
        robot.Angle[MALUC_E]=robot.Voltatge[MALUC_E]*(-0.0859)+314.68;
        //articulacions maluc
    }
}

```

```

        robot.Angle[MALUC_D]=robot.Voltatge[MALUC_D]*(-0.0859)+314.68;
        robot.Angle[GENOLL_E]=robot.Voltatge[GENOLL_E]*(-
0.0995)+319.46;           //articulacions genoll
        robot.Angle[GENOLL_D]=robot.Voltatge[GENOLL_D]*(-
0.0995)+319.46;
        robot.Angle[TURMELL_E]=robot.Voltatge[TURMELL_E]*(-
0.0621)+308.04;           //articulacions turmell
        robot.Angle[TURMELL_D]=robot.Voltatge[TURMELL_D]*(-
0.0621)+308.04;

    }
}

// lectura de la posicio del carro XY
void Llegir_Posicio_Carro(void)
{
    int eix, Valor_Registre;
    unsigned char pos;

    // lectura de la posicio del carro XY
    for(eix=0;eix<2;eix++)
    {
        Valor_Registre = eix<<7;           // segons sigui
eixX o eixY aquest valor val '0' o '1'
        outp(BASE + 13, Valor_Registre);   // part baixa
        Valor_Registre=0x04;               // part Alta per
demanar encoder 0 i encoder 1 ja aquest valor no canvia
        outp(BASE + 14, Valor_Registre);   // part Alta

        Valor_Registre |= 0x08;
        outp(BASE + 14, Valor_Registre);   // CE=1

        Valor_Registre &=0x07;
        outp(BASE + 14, Valor_Registre);   // CE=0
//
ja tenim demanat la posició del carro
//
anem a llegir-la pel canal DI

        pos=inp(BASE + 6);                 // llegim
la part baixa (4 bits situats del D7 al D4)
        robot.Posicio_Carro[eix]=pos>>4;
        pos=inp(BASE + 7);                 // llegim
la part alta (4 bits situats del D0 al D3)
        pos=pos<<4;
        robot.Posicio_Carro[eix]|=pos;     // els ajuntem
amb els baixos

        robot.Real[CARRO_X]=robot.Posicio_Carro[CARRO_X]*(2.369668);
        robot.Real[CARRO_Y]=robot.Posicio_Carro[CARRO_Y]*(1.758793);

    }
}

void Activar_Motor(int mot, int velocitat, int sentit)
{
    int Valor_Registre=0, temp;

```

```
Valor_Registre |= velocitat<<3;
Valor_Registre |= sentit<<5;
Valor_Registre |= mot<<7;
outp(BASE + 13, Valor_Registre);           // part baixa

Valor_Registre=0;
temp=mot>>1;
Valor_Registre |= temp;
outp(BASE + 14, Valor_Registre);         // part Alta

Valor_Registre |= 0x08;
outp(BASE + 14, Valor_Registre);         // CE=1

Valor_Registre &=0x07;
outp(BASE + 14, Valor_Registre);         // CE=0
}

//parar motors si cal
void Comprovar_Limits_Motors(void)
{
    if((robot.Angle[MALUC_E]>60)|| (robot.Angle[MALUC_E]<-35))
//controlem els límits dels malucs
        Activar_Motor(MALUC_E, 0, 0);

    if((robot.Angle[MALUC_D]>60)|| (robot.Angle[MALUC_D]<-35))

        Activar_Motor(MALUC_D, 0, 0);

    if((robot.Angle[GENOLL_E]>25)|| (robot.Angle[GENOLL_E]<-85))
//controlem els límits dels genolls
        Activar_Motor(GENOLL_E, 0, 0);

    if((robot.Angle[GENOLL_D]>25)|| (robot.Angle[GENOLL_D]<-85))

        Activar_Motor(GENOLL_D, 0, 0);

    if((robot.Angle[TURMELL_E]>125)|| (robot.Angle[TURMELL_E]<55))
//controlem els límits dels turmells
        Activar_Motor(TURMELL_E, 0, 0);

    if((robot.Angle[TURMELL_D]>125)|| (robot.Angle[TURMELL_D]<55))
        Activar_Motor(TURMELL_D, 0, 0);
}
```

A.2.3. Arxiu Placa.h

L'arxiu Placa.h conté totes les característiques del panell de control i es genera automàticament al compilar l'arxiu Placa.uir.

```

/*****
/
/* LabWindows/CVI User Interface Resource (UIR) Include File
*/
/* Copyright (c) National Instruments 2006. All Rights Reserved.
*/
/*
/*
/* WARNING: Do not add to, delete from, or otherwise modify the contents
*/
/*      of this include file.
*/
*****/
/

#include <userint.h>

#ifdef __cplusplus
extern "C" {
#endif

    /* Panels and Controls: */

#define INICIAL 1
#define INICIAL_TEXTMSG 2

#define PANEL 2
#define PANEL_VELOCITAT_M6 2
#define PANEL_VELOCITAT_M7 3
#define PANEL_VELOCITAT_M5 4
#define PANEL_VELOCITAT_M4 5
#define PANEL_VELOCITAT_M3 6
#define PANEL_VELOCITAT_M2 7
#define PANEL_VELOCITAT_M1 8
#define PANEL_VELOCITAT_M0 9
#define PANEL_SENTIT_M6 10
#define PANEL_SENTIT_M7 11
#define PANEL_SENTIT_M5 12
#define PANEL_SENTIT_M4 13
#define PANEL_SENTIT_M3 14
#define PANEL_SENTIT_M2 15
#define PANEL_SENTIT_M1 16
#define PANEL_SENTIT_M0 17
#define PANEL_ENVIAR 18 /* callback function:
Enviar */
#define PANEL_SORTIR 19 /* callback function:
Sortir */
#define PANEL_ANGLE0 20
#define PANEL_ANGLE1 21
#define PANEL_ANGLE2 22
#define PANEL_ANGLE3 23
#define PANEL_ANGLE4 24

```

```

#define PANEL_ANGLE5 25
#define PANEL_CARRO0 26
#define PANEL_CARRO1 27
#define PANEL_PARAR_TOTS_MOTORS 28 /* callback function:
Parar_Tots_Motors */
#define PANEL_DECORATION 29
#define PANEL_DECORATION_2 30
#define PANEL_DECORATION_3 31
#define PANEL_DECORATION_4 32
#define PANEL_DECORATION_5 33
#define PANEL_DECORATION_6 34
#define PANEL_DECORATION_7 35
#define PANEL_DECORATION_8 36
#define PANEL_TIMER 37 /* callback function:
LecturaPotes */
#define PANEL_PICTURE 38

/* Menu Bars, Menus, and Menu Items: */

/* (no menu bars in the resource file) */

/* Callback Prototypes: */

int CVICALLBACK Enviar(int panel, int control, int event, void
*callbackData, int eventData1, int eventData2);
int CVICALLBACK LecturaPotes(int panel, int control, int event, void
*callbackData, int eventData1, int eventData2);
int CVICALLBACK Parar_Tots_Motors(int panel, int control, int event, void
*callbackData, int eventData1, int eventData2);
int CVICALLBACK Sortir(int panel, int control, int event, void
*callbackData, int eventData1, int eventData2);

#ifdef __cplusplus
}
#endif

```

A.2.4. Arxiu Robot.h

Aquest arxiu conté l'adreça de memòria del PC on hi ha connectada la tarja d'adquisició de dades PCL-812PG, el que anomenem adreça base, hi ha definits tots els motors i articulacions, l'estructura de dades del robot i finalment les funcions del robot.

```

#ifndef ROBOT_H
#define ROBOT_H

#define BASE 0x220 // posicio on hi ha la tarja PCL812

#define M_MALUC_E 0
#define M_MALUC_D 1
#define M_GENOLL_E 2
#define M_GENOLL_D 3
#define M_TURMELL_E 4

```

```
#define M_TURMELL_D    5
#define M_EIX_X        6
#define M_EIX_Y        7

#define CARRO_X        0
#define CARRO_Y        1

#define MAXMOTORS      8

#define MALUC_E          0
#define MALUC_D          1
#define GENOLL_E         2
#define GENOLL_D         3
#define TURMELL_E        4
#define TURMELL_D        5

// estructura de dades del robot
typedef struct{
    int Canal_AD;
    int Angle[6];
    unsigned int Voltatge[6];
    int Posicio_Carro[2];
    int Real[2];
    int Motor[8];
    int Velocitat[8];
    int Sentit[8];
}Vbiped;

// funcions del robot
void Seleccionar_Canal_AD(int NCanal);
void Llegir_Angle_Articulacions(void);
void Llegir_Posicio_Carro(void);
void Activar_Motor(int mot, int velocitat, int sentit);
void Comprovar_Limits_Motors(void);

#endif
```