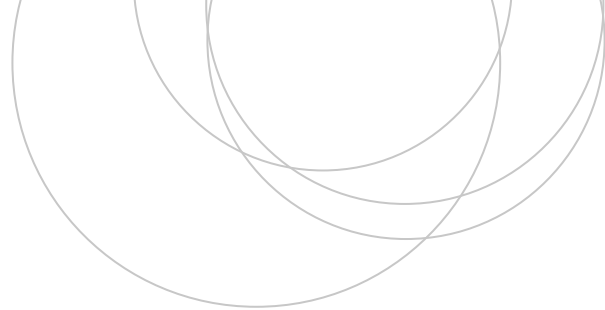




Universidad
del País Vasco

Euskal Herriko
Unibertsitatea

ZIENTZIA
ETA TEKNOLOGIA
FAKULTATEA
FACULTAD
DE CIENCIA
Y TECNOLOGÍA



Gradu Amaierako Lana / Trabajo Fin de Grado
Ingeniaritza Elektronikoko Gradua / Grado en Ingeniería Electrónica

Elektroimanen posizionamendu angularra erresonantzia bidezko irudien teknika berrientzat

Egilea:
Julen Urtaza Larrañaga
Zuzendaria:
Estibaliz Asua Uriarte
Zuzendarikidea:
Javier Alonso Valdesueiro

© 2019, Julen Urtaza Larrañaga

Leioa, 2019ko ekainaren 20a /Leioa, 20 de junio de 2019

Laburpena

Magic Angle Field Spinning (MAFS) Erresonantzia Magnetiko Nuklearra (EMN) lortzeko teknika bat da. Teknika honen oinarria, aztertu nahi den lagina iman baten barruan kokatzea da, zeinak laginaren ardatz nagusiarekiko $54,74^\circ$ orientatutako eremu magnetikoa sortzen duen. MAFS teknikaren bidez lortzen diren emaitzak erabilgarriak izan daitezten eremu magnetikoa homogeneoa izatea ezinbestekoa da.

Lan honen helburu nagusia, eremu magnetiko hau sortzen duten elektroimanen posizionamendu mikrometrikoko angeluarra ahalbidetuko duen sistema eraikitzea eta kontrolatzea izango da, modu honetan eremu homogeneo bat lortzeko. Honetarako, elektroimanen mugimendua posible egingo duen plataforma eraiki da eta plataforma honen posizionamenduaren kontrola ahalbidetuko duen sistema osoa diseinatu, inplementatu eta programatu da. Amaitzeko, erabiliko den sistema-rekin eremu magnetiko homogeneo bat lortzen dela ebaluatuko da.

Gaien Aurkibidea

1	Sarrera	1
1.1	Erresonantzia Magnetiko Nuklearra	1
1.2	MAS eta MAFS	3
1.2.1	<i>Magic Angle Spinning</i> (MAS)	3
1.2.2	<i>Magic Angle Field Spinning</i> (MAFS)	4
2	Eremu magnetikoaren norabidea biratzeko sistema	5
2.1	Luzetarako eremu magnetikoa sortzeko harilak	6
2.2	Errotazio eremu magnetikoa sortzeko harilak	7
2.3	B_{ROT} intentsitatearen homogeneotasuna	8
3	B_{ROT} homogeneoa lortzeko lerrokatze sistema	11
3.1	Elektroimanak	11
3.2	Motorra	11
3.2.1	Kontrolatzailetik seinalea bidaltzea	15
3.3	Kodetzailea	18
3.4	Plataforma eta motorra eta kodetzailearen loturak	18
4	Sistemaren kontrola	23
4.1	Programaketa	24
4.1.1	Motorra abiaraztea	25
4.1.2	Kodetzailetik informazioa jasotzea	25
4.1.3	PID kontrolagailua	26
5	Emaitzak	29
5.1	PID kontrolatzailearen sintonizazioa	29
5.2	Sistemaren mugimenduaren karakterizazioa	31
5.3	B_{ROT} -en homogeneotasunaren neurketa	31
5.3.1	DHD-en sintonizazioa	32
5.3.2	DHD-en lerrokatze zuzena	33

6	Ondorioak	35
A	Arduino kodea	39
A.1	Motorra abiaraztea	39
A.2	Kodetzailetik informazioa jasotzea	41
A.3	PID kontrolagailua	42

1. Kapituluia

Sarrera

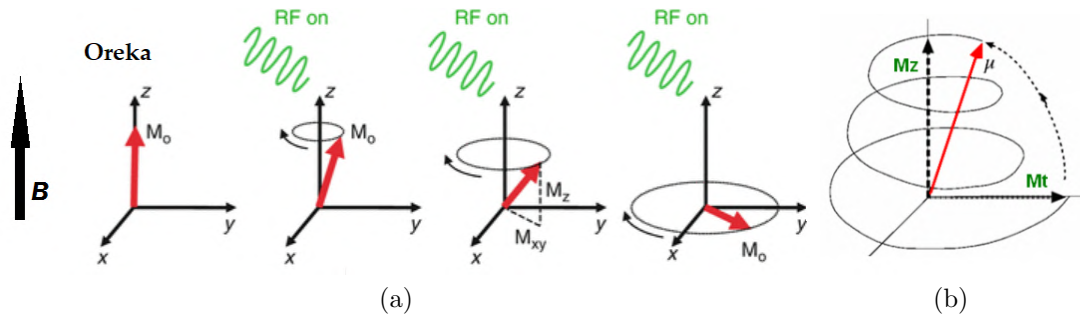
1.1 Erresonantzia Magnetiko Nuklearra

Erresonantzia Magnetiko Nuklearraren (EMN) espektroskopia teknika baliagarria da zenbait alorretan, hala nola kimikan, biologian eta medikuntzan: gorputz osoko patologia ezberdinen (minbizia, Alzheimerra...) diagnosirako erabiltzen da, bertatik gaixotasunek molekula mailan duten portaeraren informazioa jasotzen baita.

EMN-aren oinarritzko funtzionamenduaren ideia orokor bat 1.1 Irudian ageri da. Demagun EMN-aren bidez lagin bat aztertu nahi dela: lagin honetako nukleoek eremu magnetiko estatiko bat jasatean (1.1a Irudiko B) magnetizazio (M) bat sufritzen dute, B eremuaren norabide berdinarekin. Behin magnetizazioa sortuta dagoela, irrati-maiztasuneko haril baten bidez B eremuarekiko perpendikularra den eremu oszilakor txiki bat (B_1) ezartzen da. B_1 eremu honen maiztasuna prezesio nuklearraren maiztasunaren berdina denean (Larmor-en maiztasuna), xurgapen erresonantea gertatzen da eta honek magnetizazioaren errotazioa sortzen du XY planorantz [1] 1.1a Irudian adierazten den bezala. Larmor-en maiztasuna $w_0 = -\gamma B_0$ bezala definitzen da, B_0 aplikatutako eremua eta γ elementuaren berezkoa den konstante bat izanik.

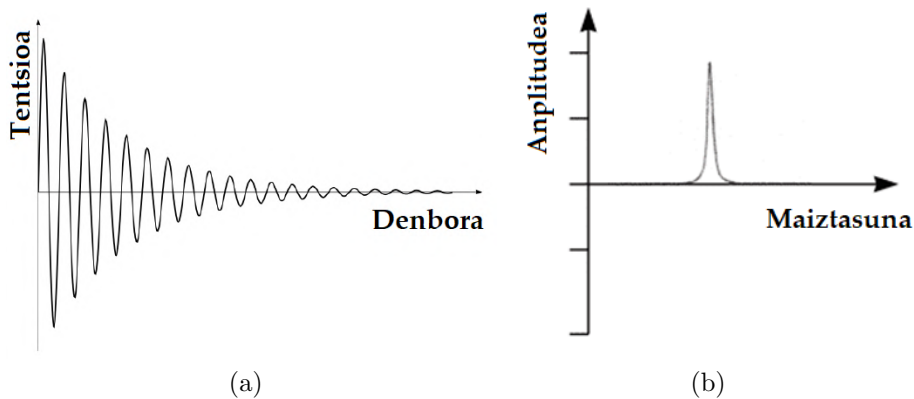
RF pulsu hau kentzean, magnetizazioa bere oreka egoerara itzultzen da, 1.1b Irudian adierazten den bezala. Prozesu honetan, magnetizazioak B eremuaren inguruan prezesatzen du Larmor-en maiztasunean. Laginaren inguruan kokatutako haril baten bidez prezesio hau antzeman daiteke. Faraday-ren indukzio legea erabiliz magnetizazioaren ondorioz sortutako tentsio induzituaren seinalea (1.2a Irudia) neurtu daiteke.

Behin hau izanik, Fourier-en transformatuaren bidez EMN-aren espektroa lor daiteke maiztasunaren domeinuan (1.2b Irudia). Espektro honetatik, erlaxazio denboren eta Larmor-en maiztasunen informazioa lor daiteke, eta hauekin



1.1 Irudia: (a) Ereku estatikoari perpendikularra den RF pulsu baten ondoriozko magnetizazioaren errotazioa. (b) RF pulsuaren ondorengo magnetizazioaren erlaxazioa.

laginaren zenbait ezaugarri: konposizioa, egitura, etab.



1.2 Irudia: (a) Induzitutako tentsioaren adierazpena. (b) EMN espektroa: induzitutako tentsioaren Fourier transformatua.

Hala ere, EMNren espektroskopiaren berezko diren zenbait arazok espektroaren bereizmen galera ekar dezakete:

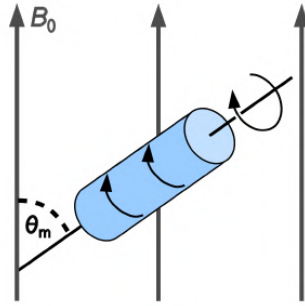
- Desplazamendu kimikoa: spin nuklearrek eta hauek inguratzen duten ingurune kimikoaren arteko elkarrekintzaren ondoriozko erresonantzia maiztasunaren desplazamendua.
- Akoplamendu dipolarra: spin nuklear ezberdinen (dipolo-dipolo) elkarrekintza magnetikoak.
- Akoplamendu kuadripolarra: $I > 1/2$ -ko nukleoetan, nukleoaren eta inguruko eremu elektrikoaren arteko elkarrekintzak.

Erabiltzen den lagina likido isotropo bat bada, molekulen translazio zein errotazioak spin nuklearren elkarrekintza hauen batezbestekoa deuseztatu egiten dute, mugimendu hauek berdinak baitira edozein norabidetan. Solidoetan, ordea, mugimendu hauek mugatuta daude eta ez dira isotropoak: elkarrekintza anisotropoen eraginez EMN espektroan norabide berezi batzuk sortzen dira, eta hauen ondorioz espektroko lerroak zabaltzen dira, bereizmena kaltetuz. Bereizmen hau hobetzeko zenbait teknika existitzen dira, hauen artean *Magic Angle Spinning* (MAS) eta *Magic Angle Field Spinning* (MAFS) deritzenak.

1.2 MAS eta MAFS

1.2.1 *Magic Angle Spinning* (MAS)

Egoera solidoko laginen EMNaren bereizmen espektrala hobetzeko sorturiko tekniketako bat *Magic-Angle Spinning* (MAS) [2] delakoa da. Metodo honetan, laginak errotazio azkar bat du aplikaturiko B_0 norabidearekiko $54,74^\circ$ (Angelu Magikoa, θ_m) osatzen duen ardatzarekiko, 1.3 Irudian ageri den bezala.



1.3 Irudia: MAS teknikako laginaren eta eremuaren eskema.

Ondoren ikusiko den bezala, modu honetan espektroen bereizmen hobe lortzen da, lagin hauetan ematen diren elkarrekintza anisotropoen batezbestekoa zerora eramaten du eta. Honen arrazoia ikusteko, kontuan izan behar da lagina θ angeluan orientatutako ardatz batekiko biratzen denean, EMN espektroaren zabaltzea eragiten duten elkarrekintzak deskribatzen dituen Hamiltondarraren batezbestekoa denboran Legendreren bigarren polinomioarekiko proportzionala dela [2]:

$$H \propto \frac{1}{2}(3\cos^2(\theta) - 1) \quad (1.1)$$

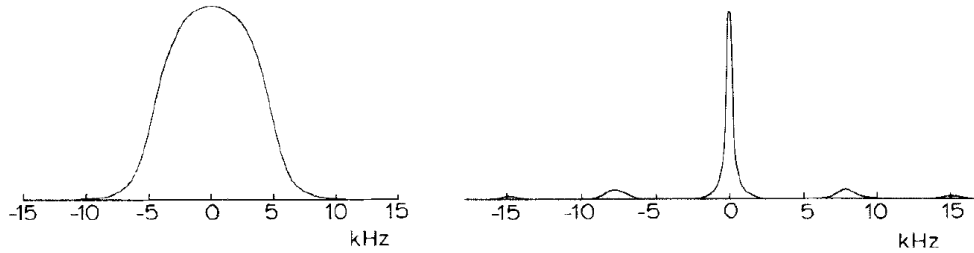
Hamiltondar hau biraketa angeluaren menpekoa da, eta $H \approx 0$ da angelu honek $54,74^\circ$ hartzen dituenean, hau da, Angelu Magikoan. Honez gain, erro-

tazioaren ondorioz eremu magnetikoaren zein laginaren ez-homogeneotasunen batezbestekoak zerora eramaten dira.

Teknika honekin bereizmena hobetzeko, ezinbestekoa da laginaren errotazio abiadura (Ω) lagina geldi dagoenean duen maiztasun espektroaren banda zabalera baino askoz handiagoa izatea [3]:

$$\Omega \gg \Delta w \quad (1.2)$$

MAS bidez lortutako bereizmen hobekuntzaren adibide bat 1.4 Irudian adierazten da:



1.4 Irudia: ^{27}Al lagin baten EMN espektroa [2]. (a) Lagina geldi dagoenean. (b) Laginak angelu magikoarekiko biratzen duenean 7,7 kHz-tan.

Hala ere, (1.2) ekuazioak adierazitako baldintza dela eta, teknika hau ez da erabilgarria alor guztietan, adibidez medikuntzan, lagina abiadura handiz biratu behar delako.

1.2.2 *Magic Angle Field Spinning* (MAFS)

Arazo hau konpontzeko, *Magic Angle Field Spinning* (MAFS) metodoa garatu zen [4], zeinean laginak biratu beharrean, eremu magnetikoaren norabidea biratuko den laginarekiko $54,74^\circ$ -tan [5]. Modu honetan, aurreko kasuan bezala, espektroen bereizmenaren hobekuntza bat lortzen da, eremuaren ez-homogeneotasunean, desplazamendu kimiko anisotropoan eta elkarrekintza dipolarrean eragin berdina baitu [6], baina lagina geldi mantentzen denaren abantailarekin. MAS kasuan bezala, eremu magnetikoaren errotazio abiadurak (w_1)

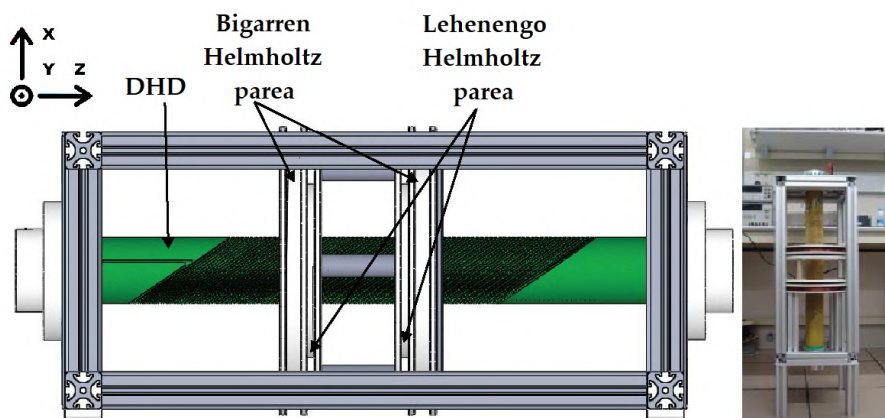
$$w_1 \gg \Delta w \quad (1.3)$$

baldintza bete behar du elkarrekintza anisotropoen batezbestekoa zero izateko Angelu Magikoan.

2. Kapituluia

Eremu magnetikoaren norabidea biratzeko sistema

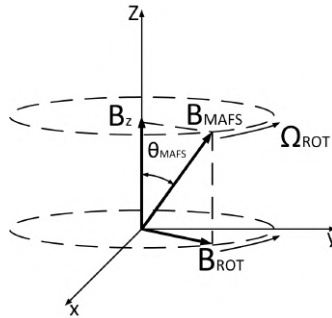
MAFS teknika inplementatzeko modu bat, *Design, Construction, and Characterization of a Magic Angle Field Spinning RF Magnet* [7] lanean azaltzen da. Lan honetan, eremu magnetikoak ardatzarekiko $54,74^\circ$ -tan biratzeko 2.1 Irudian adierazitako sistema erabiltzen da.



2.1 Irudia: MAFS teknika erabiltzeko sistemaren prototipoa eta amaierako muntaia [7].

Sistema honetan B_{MAFS} (z ardatzarekiko Angelu Magikoko distantzia angeluarrera biratuko duen eremu magnetikoa) sortzeko Helmholtz-en harilak eta kontzentrikoki kokatutako bi elektroiman edo DHD erabiltzen dira, B_z eremua eta xy planoan biratzen duen B_{ROT} eremua sortzeko, hurrenez hurren, hurrengo azpiataletan azalduko den modura. Sorturiko B_z eta B_{ROT} eremu birakariaren balio egokientzat, z ardatzetik $54,74^\circ$ -ra biratuko duen B_{MAFS} lor daiteke (ikusi

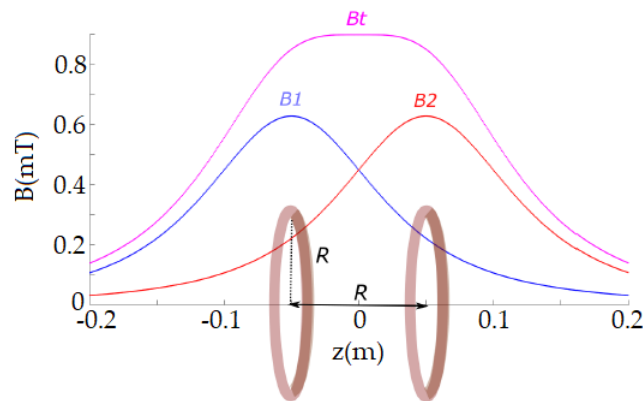
2.2 Irudia), MAFS teknikaren oinarria dena.



2.2 Irudia: B_{MAFS} sortzeko beharrezko diren eremu magnetikoen eskema.

2.1 Luzetarako eremu magnetikoa sortzeko harilak

Aurrez aipatu bezala, MAFS-rako beharrezko den B_z eremu uniformearen lortzeko Helmholtz-en harilen sistema [8] erabili da. Helmholtz-en harilen sistema bat diametro eta itzuli kopuru berdineko bi haril zirkularrez osatua dago, ardatz komun batekin eta beraien erradioaren luzerako R distantzia batera aldentuta, 2.3 Irudian ikus daitekeen bezala.



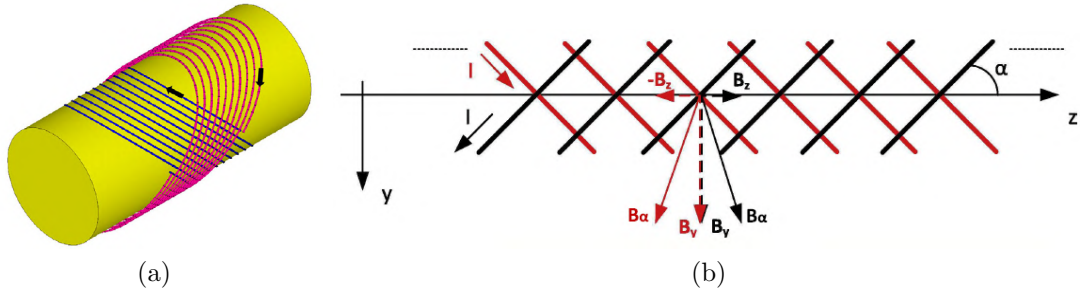
2.3 Irudia: Helmholtz harilen eskema. Harila bakoitzak sorturiko eremua (B_1 eta B_2) z ardatzean eta eremu totala (B_T).

Modu honetan, bi harilek sorturiko eremu magnetikoak batuz, sistemaren zentroan homogeneotasun askoko eremu magnetiko bat lortzen da.

2.2 Errotazio eremu magnetikoa sortzeko harilak

Eremu magnetikoaren errotazio osagaiak (B_{ROT}) B_Z eremuarekiko perpendikularra den planoan biratu behar du homogeneouski. Honetarako, helize bikoitzeko dipoloak (DHD, *Double Helix Dipole*) deritzenak erabiltzen dira: modu inklinatu batean bildutako harilen sistemak (ikusi 2.4a Irudia).

Elektroiman bat harilkatzen bada buelta bakoitza α angeluko inklinazioarekin elektroimanaren ardatzarekiko, buelta bakoitzari dagokion eremu bektoreak bi osagai izango ditu: bata elektroimanaren ardatzaren z norabidean eta bestea ardatzarekiko perpendikularra den norabidean (y norabidean, osagai dipolarra deritzona).



2.4 Irudia: (a) DHD baten eskema orokorra: aurkako norabidean harilkatutako bi elektroimanez osatua [9]. (b) DHD bateko bi harilek sortzen dituzten eremu magnetikoak.

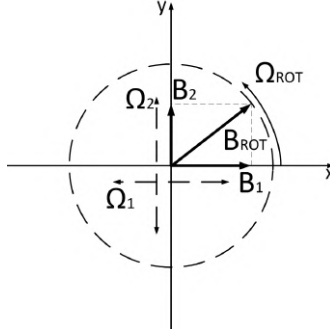
Mota honetako bi elektroiman aurkako inklinazio angeluekin eta aurkako korronte fluxu noranzkoarekin baditugu, bata bestearekin kontzentrikoki gainezartzean (ikus 2.4a Irudia) z norabideko eremuak ezabatu egiten dira eta osagai dipolarrak (B_y) batu egiten dira, 2.4b Irudian ikusten den bezala. Modu honetan, helize bikoitzeko dipoloa lortzen da, harilen sistemaren luzetarako ardatzarekiko (z) perpendikularra den norabideko eremu magnetiko bat sortuko duena (kasu honetan y norabidean).

DHDak luzera aski handia badu ($l \gg d$, d harilen buelten arteko tartea izanik), eremuaren magnitudea honakoa da:

$$B_y = \frac{\mu_0 I_0}{d} \cos \alpha \quad (2.1)$$

B_{ROT} eremu birakaria lortzeko, mota honetako bi elektroiman kontzentriko erabili dira. Hau da, bi DHD, bakoitza bi haril geruzaz osatua, zeharkako bi eremu sortuz (ikus 2.5 Irudiko B_1 eta B_2).

DHD hauek elikatzen dituzten seinale sinusoidalak funtzio sorgailu baten bidez sortzen dira, zeinak anplitude berdineko baina 90° -ko desfaseko bi seinale ematen dituen. Biratze abiaduraren maiztasuna (Ω_{Rot}) DHDak elikatzeke erabiltzen diren Ω_1 eta Ω_2 maiztasunen berdina izango da (elikadura maiztasun berdinean egingo da bi DHD-enzat).



2.5 Irudia: Bi DHD-ko sistema batek sorturiko eremuak: batek B_1 sortzen du eta besteak B_2 . Bien baturatik B_{ROT} lortzen da.

2.3 B_{ROT} intentsitatearen homogeneotasuna

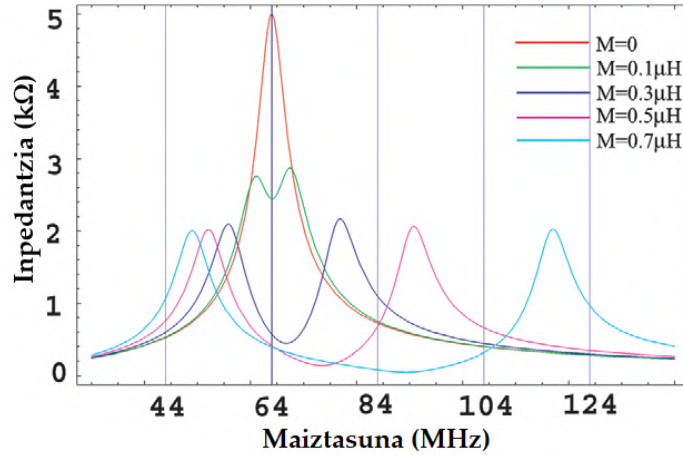
2.5 Irudian ikus daitekeen bezala, B_{ROT} guztiz zirkularra izateko B_1 eta B_2 magnitudez berdinak izan behar dira, eta honez gain eremuen norabideak perpendikularrak izan behar dira bata bestearekiko. Baldintza hauek bete ezean ez-homogeneotasunak agertuko dira B_{ROT} -en intentsitatean. Ez-homogeneotasun hauen ondorioz, MAFS teknika erabili ahal izateko beharrezko den B_{MAFS} eremuak ere ez-homogeneotasunak izango ditu, eta honek EMN teknikaren bidez lortutako emaitzak txarrak izatea eragingo du. Ezinbestekoa da, beraz, B_{ROT} ahalik eta homogeneoena izatea.

B_{ROT} -en intentsitatearen homogeneotasunak duen garrantzia ikusirik, GRAL honen helburu nagusia, [7] lanean eraikitako B_1 eta B_2 eremuak sortzen dituzten bi DHD-entzako lerrokatze sistema bat diseinatzea izango da, modu honetan B_1 eta B_2 perpendikularrak izateko eta B_{ROT} -en (eta ondorioz B_{MAFS} -en) intentsitatearen homogeneotasuna lortzeko.

Lerrokatze sistema diseinatu ostean, B_1 eta B_2 eremuak perpendikularrak noiz diren aztertu behar da eta horretarako *Coupling and Decoupling Theory and Its Application to the MRI Phased Array* [10] artikuluan deskribatutako metodoa erabiliko da. Artikulu honetan teorikoki frogatzen da independenteki sintonizatuta dauden bi elektroimanek aldeberean seinalea jasotzean, elektroiman hauetako bakoitzaren inpedantzia espektroa banatu egiten dela bi elektroimanen

arteko induktantziaren (M) ondorioz (elektroimanak akoplatu egiten direla esaten da). Honen adibide bat 2.6 Irudian adierazi da.

2.6 Irudian elektroiman baten inpedantzia espektroa adierazi da bigarren elektroiman batekin duen induktantziaren (M -ren) balio ezberdinentzat. Lehenengo elektroimana independenteki sintonizatuta dago 64 MHz-ko erresonantzia maiztasunean inpedantzia $50 \text{ k}\Omega$ izateko (elkarren arteko induktantziarik ez dagoeneko, $M = 0$, kasuaren parekoa da). Ikus daitekeen modura, elektroimanen arteko induktantzia handituz joan ahala erresonantzia maiztasunen maximoen arteko distantzia zabaltzen da.



2.6 Irudia: Elektroiman baten inpedantzia espektroa beste elektroiman batekiko duen elkarren arteko induktantzia (M) balio ezberdinentzat. M handitu ahala, bi erresonantzien arteko tartea handiagoa dela ikus daiteke. [10]

GRAL honetan erabiliko den MAFS teknikorako sistemaren kasuan, elkarrekiko printzipiotik ondorioztatzen da bi DHD-en arteko induktantzia (M) minimoa izango dela B_1 eta B_2 perpendikularrak direnean; maximoa izango da, berriz, paraleloak badira. Hau kontuan hartuta, [10] artikuluko metodoa erabili daiteke: DHD-en arteko posizio erlatiboa aldatzen den bitartean DHD-aren inpedantzia-espektroa aztertuko da, erresonantzia maiztasun bakarreko ($M = 0$) egoera lortu arte. Hau lortuta, DHD-en arteko induktantzia minimoa dela ondorioztatu daiteke, eta elkarrekikotasun printzipiotik, B_1 eta B_2 perpendikularrak direla. Modu honetan, B_{ROT} -en intentsitatearen homogeneousutasuna lortuko da DHD-en lerrokatzen zuzenaren bidez.

3. Kapituluia

B_{ROT} homogeneous lortzeko lerrokatze sistema

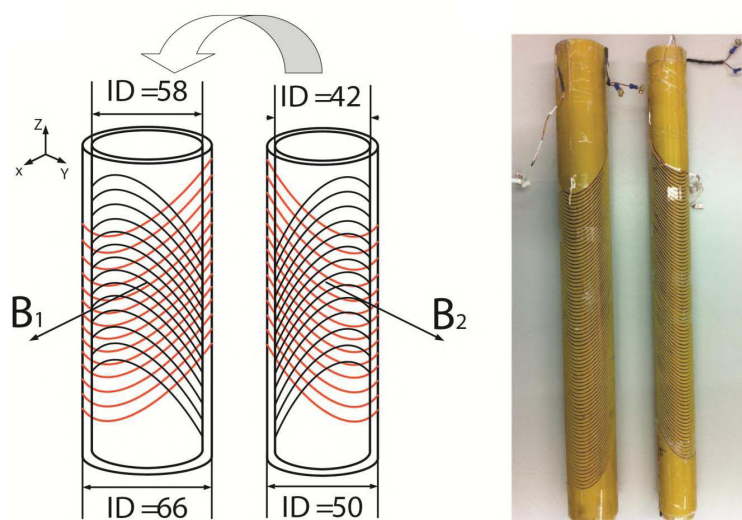
Atal honetan B_{ROT} eremuaren intentsitatearen homogeneousasuna lortzeko eraiki-ko den sistemaren diseinua azalduko da, eremu hau sortzen duten bi elektroimanen lerrokatze zuzenaren bidez. [7] laneko elektroimanen (DHD-en) lerrokatzea lortzeko, bi DHD-etaz gain, motor bat eta kodetzaile bat erabiliko dira. Honez gain, DHD-en eta motorraren eta kodetzailearen arteko loturak ahalbidetuko dituen plataforma bat diseinatuko da.

3.1 Elektroimanak

B_{ROT} eremu magnetikoa sortzeko bi DHD elektroimanez (3.1 Irudian adieraziak) osatutako sistema erabiltzen da, 2.2 atalean azaldu bezala. 3.1 Irudian ikus daitekeen modura diametro txikieneko DHDa diametro handikoaren barnean kokatuko da kontzentrikoki.

3.2 Motorra

Sistema honetan, DHD-en arteko posizio erlatiboa aldatzeko, DHD bat finko mantenduko da, eta bestea biraraziko da. Hau lortzeko, urratsez urratsezko motor bat erabiliko da: motorrak barneko DHD-a biratuko du, kanpoko finko mantentzen den bitartean. Honetarako, NEMA-17 pausokako motor bipolarra erabili da (ikusi 3.2a Irudia). Motor honen ezaugarrien artean honakoak ditugu: $0,067^\circ$ angeluko pausua, 1,7 A-ko korrante maximoa eta 1,7 Ω -ko erresistentzia bobinan; gainera, korrante zuzenean 12 V-ko tentsioa erabiltzea gomendatzen da.



3.1 Irudia: Erabiliko diren bi DHD elektroimanak eta beraien neurriak.

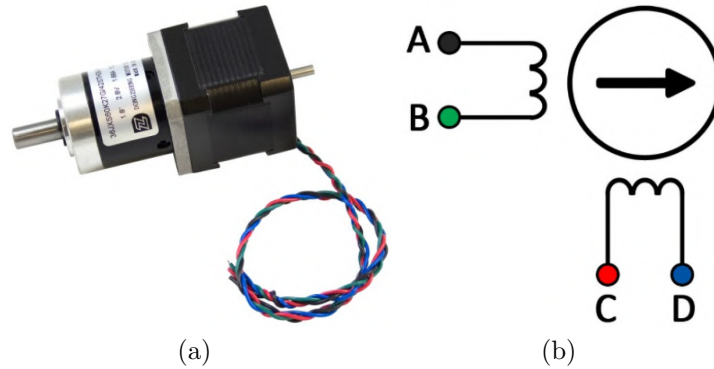
Erabiliko den NEMA-17 motorrak, gainerako urratsez urratseko motorrek bezala, lan egiteko modu ezberdinak ditu: pausu osoka, erdipausoka edo mikropausoka. Kasu honetan, pausu osoka motorrak ematen duen bereizmena $0,067^\circ$ denez eta hau lanaren garapen egokirako nahikoa denez, motorrak pausu osoka lan egitea erabaki da (ikusi 3.3 Irudia).

Motorrak pausu osoka lan egin ahal izateko 3.2b Irudiko A, B, C eta D sarreerek jaso beharreko seinaleen sekuentziak 3.1 Taulan adierazi dira. Sekuentzia hauen bidez motorraren biraketaren norabidea kontrolatu ahal izango da: erlojuaren orratzen norabidean (EON) edo erlojuaren orratzen aurkako norabidean (EOAN).

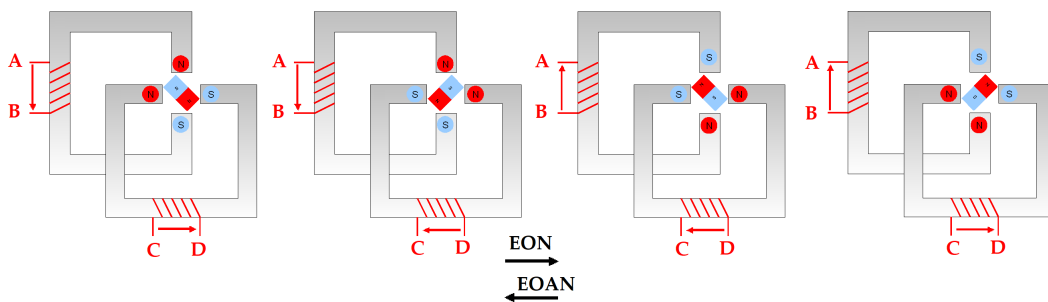
Pausua	EON				EOAN			
	A	B	C	D	A	B	C	D
0	+	-	+	-	+	-	+	-
1	+	-	-	+	-	+	+	-
2	-	+	-	+	-	+	-	+
3	-	+	+	-	+	-	-	+

3.1 Taula: Motorrak erloju orratzen norabidean (EON) edo erloju orratzen aurkako norabidean (EOAN) biratzeko jaso beharreko seinaleak.

Sekuentzia hauek kontrolatu ahal izateko bi H-zubi (L298N gailua) erabiliko dira. H-zubi bat motor elektriko bat bi norabidetan (erlojuaren orratzen norabidean eta aurkakoan) biratzea ahalbidetzen duen gailua da, motorretik igarotzen



3.2 Irudia: (a) NEMA-17 motorra. (b) Motorra elikatzeko sistema bipolarra.



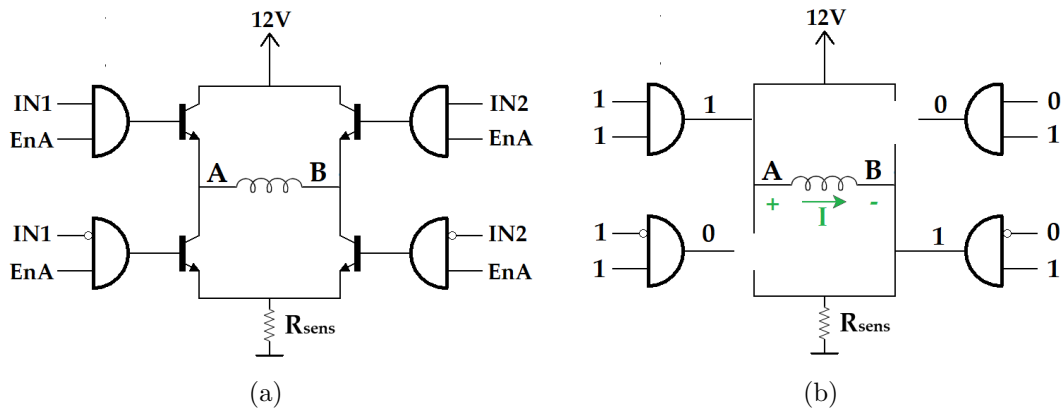
3.3 Irudia: Pausu osoka biratzen duen urratsez urratsezko motor baten sekuentzia.

den korrontearen norabidea aldatzen baitu. Kasu honetan motorra bipolarra denez, bi polo hauek elikatzeko bi H-zubi beharko dira (H-zubi bat AB harilarentzat eta bestea CD harilarentzat).

H-zubi batek bere sarreren konbinazio bakoitzarentzat irteera ezberdinak ematen ditu, motor baten harilen polarizazioaren aldaketa ahalbidetuz. L298N-aren H-zubi baten eskema orokorra 3.4a Irudian adierazi da.

3.4a Irudian ikusten den modura, H-zubi baten harilean zeharreko korrontearen norabidea kontrolatzeko bi seinale behar dira: IN1 eta IN2. Bestetik, EnA seinaleak, harilean zehar korrontea igarotzen den edo ez zehazten du: EnA=1 baldin bada korrontea igaroko da, baina EnA=0 baldin bada ez. Azkenik, R_{sens} erresistentzia erabiltzen da hariletik igarotzen den korrontearen intentsitatea kontrolatu ahal izateko.

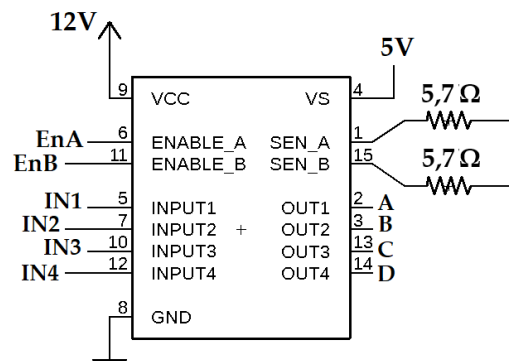
3.4b Irudiko kasuan, adibidez, ENABLE seinalea (EnA) 1-era mantentzen da (motorra martxan) eta IN1=1 eta IN2=0 izanik, korrontea A-tik B-ra joatea lortzen da. Korrontea B-tik A-ra joatea nahi izango balitz IN1 eta IN2-ren balioak trukatu beharko lirateke. Modu honetan, sarrera ezberdinen bidez motorra modu



3.4 Irudia: (a) L298N-aren lehenengo H-zubiaren eskema. (b) $EnA=1$, $IN1=1$ eta $IN2=0$ egoeraren eskema.

ezberdinetan elikatzea lortzen da, 3.1 Taulako sekuentzien kontrola ahalbidetuz.

H-zubi baten funtzionamendua ikusita, bi H-zubi uztartzen dituen gailuaren, L298N-aren, konexioak egin daitezke, 3.5 Irudian adierazi den modura. Irudi honetan, V_{CC} motorraren elikadura-tentsioa (gure kasuan 12 V) da eta V_S L298N-ak bere logikarako beharrezko duen tentsioa. Bestetik, lehenengo H-zubiaren funtzionamendua EnA , $IN1$ eta $IN2$ -k kontrolatuko dute, eta hauen bidez lortutako korrontea A eta B irteeren artetik igaroko da; azkenik, SEN_A -n konektatutako erresistentziaren bidez (3.4 Irudiko R_{sens} izango litzatekeena) korrontearen magnitudea kontrolatzen da. Gainerako sarrera eta irteerak modu berdinean bigarren H-zubia kontrolatzeko erabiltzen dira.



3.5 Irudia: L298N-aren (bi H-zubiren) loturak.

L298N hau izango da, beraz, motorraren mugimendua zehaztuko duen tresna. Hala ere, jasoko dituen sarrera seinaleak (EnA , EnB , $IN1$, $IN2$, $IN3$, $IN4$)

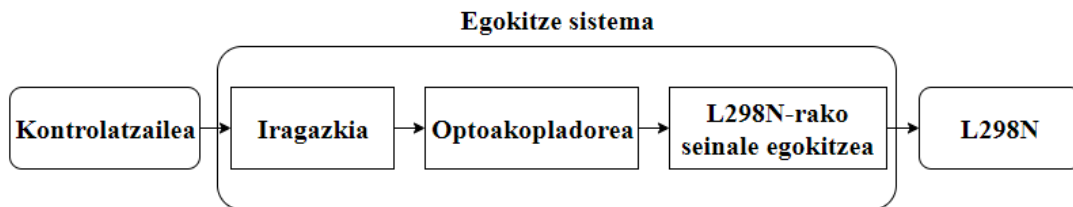
kontrolagailu batetik (gure kasuan Arduinoa) jaso beharko ditu. Kontrolatzailearen bidez bidalitako seinaleen eta L298N-ak jasotzen dituen seinaleen artean, beraz, egokitze sistema bat diseinatu behar da.

3.2.1 Kontrolatzailetik seinalea bidaltzea

Hasteko, kontrolatzailetik L298N-ra seinale bat bidaltzerako orduan ager daitezkeen arazoak aztertu dira:

- Seinaleen *high-low* eta *low-high* trantsizioetan errebotek ager daitezke.
- Motorra elikatzeko erabiltzen den korrontea ampere baten ordenakoa izan behar da. Zirkuituaren erroreren baten ondorioz korronte hau kontrolatzaileira iritsiz gero, azken hau hauts lezake.
- L298N-aren sarrerek (EnA, EnB, IN1, IN2, IN3, IN4) sarrera tentsio eta sarrera korronte baldintza jakin batzuk izan behar dituzte; hauek bete ezean gailua hautsi daiteke.

Arazo hauetako bakoitzari aurre egiteko iragazki bat, optoakopladore eta L298N-rako seinaleen egokitze sistema bat erabiliko dira, hurrenez hurren. Egokitze sistemaren diagrama 3.6 Irudian ikus daiteke.

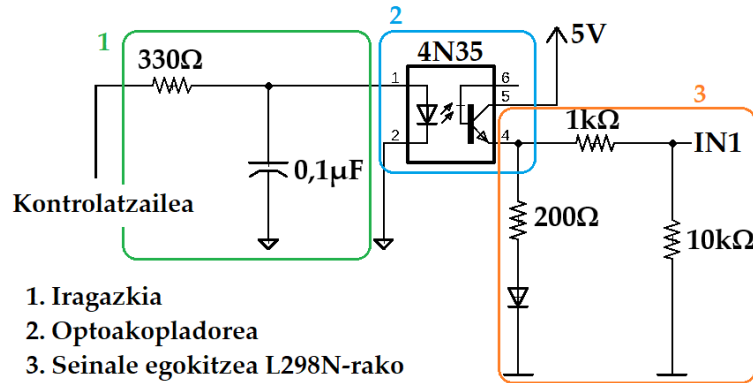


3.6 Irudia: Seinale baten egokitze sistemaren bloke diagrama.

Lehenengo, beraz, kontrolatzailetik bidaltzen den seinalea ($5\text{ V} \equiv 1$ edo $0\text{ V} \equiv 0$) 3.7 Irudian adierazitako RC iragazkitik igarotzen da, seinaleek 5 V -tik 0 V -rako edo 0 V -tik 5 V -rako trantsizioetan izan ditzaketen errebotek gutxitzeko.

Behin seinalea iragazita dagoela, 4N35 optoakopladoretik igarotzen da. Optoakopladore bat elektrikoki isolaturik dauden zirkuituen artean seinaleak argi moduan igortzen duen gailua da. Optoakopladorearen sarreran seinale bat aplikatzean diodoak argia igortzen du eta fototransistoreak argi hau detektatzean irteeran seinale elektriko bat ezartzen du.

Aurrez esan bezala, motorrak funtzionatu ahal izateko korronte handiak behar ditu, eta korronte hauek kontrolatzaileira iritsiz gero hau hausteko posibilitatea



3.7 Irudia: (a) Arduinoko seinala L298N-ra eramateko zirkuitua, 4N35 optoakopladore baten bidez bi zati ezberdinak bereiziz.

dago. Hau saihesteko, optoakopladorearen ezaugarrien bidez zirkuituan bi zati bereiziko dira, bata kontrolagailuari lotutako zati digitala eta bestea motorrari lotutako zati analogikoa. Modu honetan, beraz, zati analogikoko korrante handiak zati digitalera ez pasatzea lortzen da.

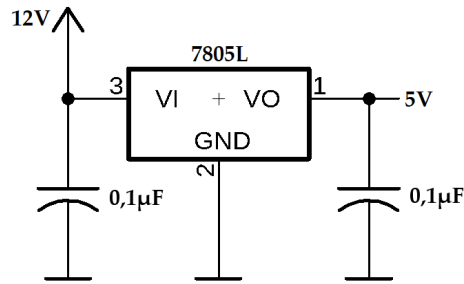
4N35a erabiltzeaz gain, bi zatien bereizmena ahalik eta hobereena izateko, zati bakoitzeko lurrek ezberdinak izan behar dute, modu honetan isolamendu osoa lortuz. Hau horrela, optoakopladorearen bidez zati digitalean dagoen seinala zati analogikora igarotzea lortzen da, 3.7 Irudiko 4 puntuan 1 puntuan dagoen seinala bera (altua edo baxua) lortuz.

Behin hau izanik, egokitze sistemaren azken urratsa L298N-ren sarrerak duten ezaugarrietara egokitzea izango da. Seinala altua (1) denean, sarrerako korrante maximoa $100 \mu A$ izan behar da eta sarrerako tentsio ($2, 3V \rightarrow 5V$) tartean egon behar du; seinala baxua (0) bada, ordea, korrante maximoa $-10 \mu A$ izango da eta sarrera tentsioa ($-0, 3V \rightarrow 1, 5V$) tartean egon beharko da.

3.7 Irudian adierazitako zirkuitua sei aldiz errepikatu da zirkuitu osoan zehar, L298N-aren sei sarrerentzat: EnA, EnB, IN1, IN2, IN3 eta IN4. Modu honetan, kontrolatzailetik L298N-rako seinalen transferentzia egokia dela ziurtatzen da.

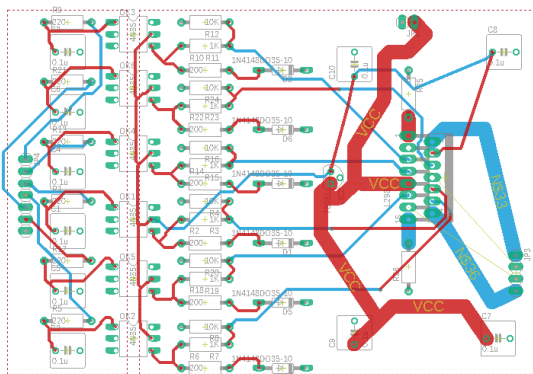
Zirkuitu elektronikoarekin amaitzeko, 3.5 eta 3.7 Irudietan ikus daiteke bi erreferentzia tentsio behar direla: motorra elikatzeko $12V$ eta L298N-ren eta 4N35-aren funtzionamendurako $5V$ tentsioak. Proiektu honetan erabiliko den tentsio-iturri bakarra $12V$ -ko iturria izango denez, $5V$ balio egonkorreko tentsioa nola lortu ikusiko da.

$12V$ -ko iturri batetik $5V$ egonkor lortu ahal izateko, 3.8 Irudiko zirkuitua eraiki da. Zirkuitu honetan 7805L tentsio erregulatzailea erabiltzen da $5V$ -ko tentsio konstante bat sortzeko. Hau izanik, 4N35aren elikadurarako zein L298N-aren logikarako beharrezko izango den $5V$ -ko iturri bat lortzen da.

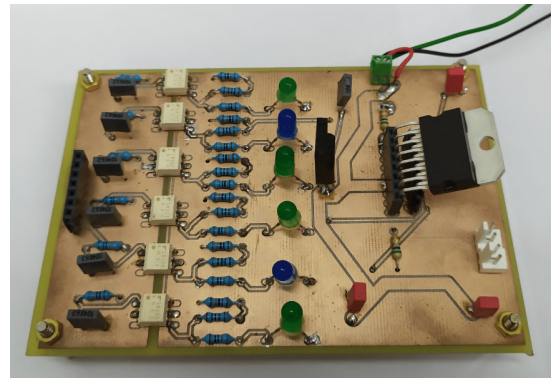


3.8 Irudia: 12 V tentsio-iturritik 5 V lortzeko zirkuitua.

Zirkuituaren diseinua erabaki ondoren, frogak egin ahal izateko, *protoboard* batean muntatu da. Behin *protoboard*-ean zirkuituaren funtzionamendua egokia dela ikusi ondoren, *Eagle* programa erabiliz PCB bat diseinatu da (ikusi 3.9a Irudia) eta ondoren fresatzeko makina baten bidez zirkuitu inprimatuko plaka egin da (ikusi 3.9b Irudia). 3.9a Irudian ikus daitekeen modura, zirkuitua lerro etenez mugatutako bi zati ezberdinetan banatua dago: aurrez aipatu diren bi lur ezberdinak dira hauek. Gainera, PCBa egiterako orduan kontuan hartu da motorra zeharkatzen duen korrontea handia izan daitekeela: hau konpontzeko, korronte hauek zeharkatzen dituzten bideak 3 mm zabal egin dira eta gainerakoak 0,5 mm zabal. Azkenik, plakako bideak egiteko prozesua sinplifikatzeko bideak plakaren goiko (gorriz) eta beheko (urdinez) aldeetan, bietan, egitea erabaki da.



(a)



(b)

3.9 Irudia: (a) *Eagle* programaren bidez egindako PCBaren diseinua: gorriz plakaren goiko aurpegia eta urdinez azpikoa. (b) Proiektuan erabilitako plaka.

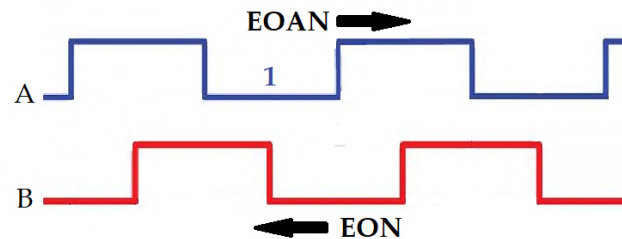
Honekin, motorraren funtzionamendurako beharrezko den elektronika amaitutzat jo daiteke.

3.3 Kodetzailea

Motorraren bidez DHD-ari mugimendua transferitzea posible izango da, beraz, 3.2 ataleko elektronikaren eta 3.4 atalean ikusiko den motorra eta DHD-aren loturaren bidez. Hala ere, posible da erroreren baten ondorioz motorrak eman beharreko pausuren bat salto egitea edo mugimendu transferentzia guztiz perfektua ez izatea. Horregatik, DHD-aren mugimenduaren zenbaterainokoaren informazio erreala izateko kodetzaile (encoder) bat erabili beharra dago.

Proiektu honetan, E6B2-C kodetzaile inkrementala erabiliko da. Encoder honek 1024 pulsu birako bereizmena du, hau da, $0,352^\circ$ bereizteko gai izango da. Datu hau eta motorrak $0,067^\circ$ -ko bereizmena duela jakinda, plataforma egi-terako orduan kontuan izan behar da biratzen duen sistemaren (DHD-aren) eta kodetzailearen artean gutxienez 5:1 balioko transmisio erlazioa duen engranaje sistema bat garatu beharko dela, 3.4 atalean ikusiko den modura.

Bestetik, kontuan izan behar da erabiliko den kodetzailea inkrementala izango dela. Honek, gainerako kodetzaile inkrementalek bezala, bi arazo nagusi izango ditu: mugimenduaren noranzkoa eta posizio absolutua ezin dira jakin. Orokorrean, noranzkoa jakiteko bi buru irakurle erabiltzen dira 90° desfasatutako irteerak ematen dituztenak (ikus 3.10 Irudia); modu honetan, noranzkoa jakiteko seinalea lehenago nork ematen duen jakin behar da soilik.



3.10 Irudia: Kodetzaileak ematen dituen 90° desfasatutako bi seinaleak.

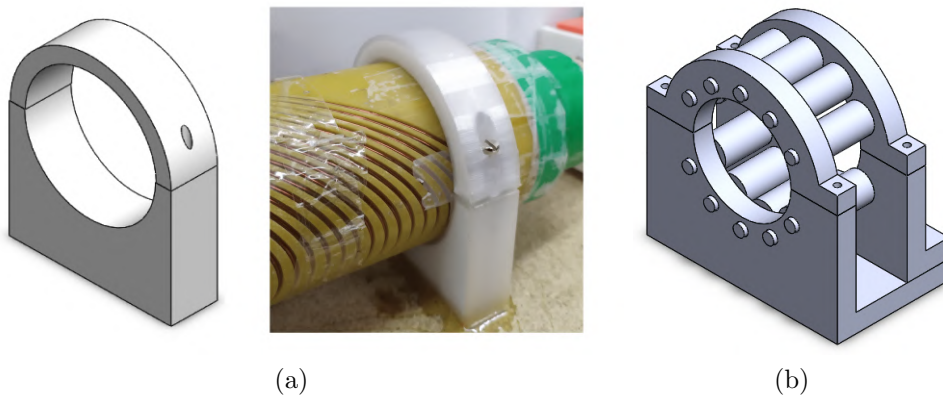
Posizio absolutua ez jakitearen arazoari dagokionez, proiektu honetan ez da kontuan izango, izan ere DHD-a bereziki zein puntutan kokatuta dagoen ez da garrantzitsua; garrantzitsua izango dena ordenagailutik biratzeko eskatzen zaion angelua mugitzea izango baita.

3.4 Plataforma eta motorra eta kodetzailearen loturak

Elektroimanen eustea eta beraien posizionamendu mikrometrikoa ahalbidetuko duen sistema birakari bat lortu ahal izateko plataforma *SolidWorks* programaren

bidez diseinatu da. Hau egiterako orduan kontuan izan da diseinu aldetik diametro txikiko DHDa biratzea diametro handikoa biratzea baino errazagoa izango dela. Beraz, DHD txikia biratzea erabaki da, DHD handia finko mantentzen den bitartean.

Hasteko, elektroiman bakoitzarentzako euskarriak diseinatu dira, 3.11 Irudian ikusten den bezala. Kontuan hartu da DHD txikia biratu behar dela, honetarako bere euskarrian arrabolak jarriz. Hala ere, diseinu hau burutu arren, praktikan ikusi da beharrezkoa ez dela, DHD txikia motorreko loturaren euskarriaren (aurrerago ikusiko dena) laguntzarekin soilik gai baita biratu beharreko z ardatzaren inguruan biratzeko.

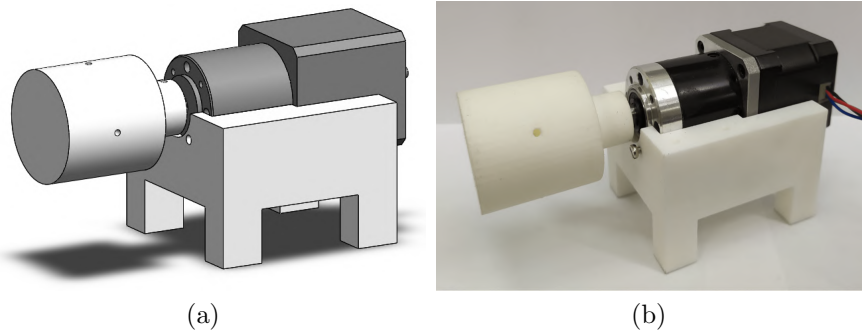


3.11 Irudia: (a) Kanpoko zilindroarentzat euskarriaren *SolidWorks*-eko diseinua eta inprimatutako euskarria. (b) Barneko zilindroarentzat euskarriaren diseinua *SolidWorks*-en (ez da erabili amaierako muntaian).

Euskarri batekin, beraz, kanpoko zilindroa finko mantentzea eta barnekoa biratzea ahalbidetzen da. Biraketa hau motorraren bidez transferitu behar zaio zilindroari, eta hau lortzeko, 3.12 Irudian ageri den mekanismoa diseinatu da.

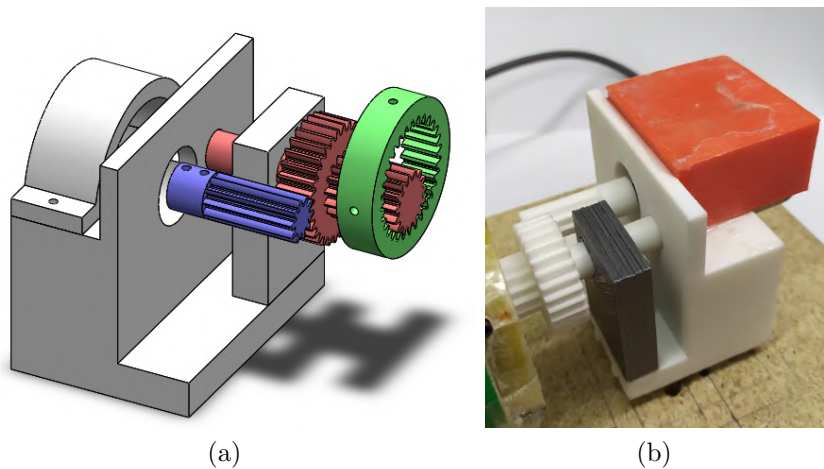
Mekanismo honetan bi zati nagusi bereizten dira: motorra finko mantentzeko euskarria eta mugimendua zilindrora transmitituko duen pieza. Pieza hau bi zatitan bana daiteke: zilindro txikia eta zilindro handia. Zilindro txikian, motorraren ardatza sartzeko zuloa eta ardatz hau piezari lotzeko bi torlojuentzako zuloak egin dira. Piezako zilindro handiak, ordea, bere perimetroan zehar lau zulo ditu, sistemaren zilindro txikira lotzeko erabiliko direnak.

Honez gero, sistemari mugimendua transferituko dion mekanismoa sortua dago, baina oraindik kodetzailearentzat lotura bat egon behar da. Lotura honetan motorrak (eta ondorioz, DHDak) egindako bira bakoitzeko kodetzaileak gutxienez bost bira eman beharko ditu (arrazoia 3.3 atalean); hau da, gutxieneko 5:1 erlazioa egon behar da bien artean. Hau lortzeko, 3.13a Irudiko mekanismoa



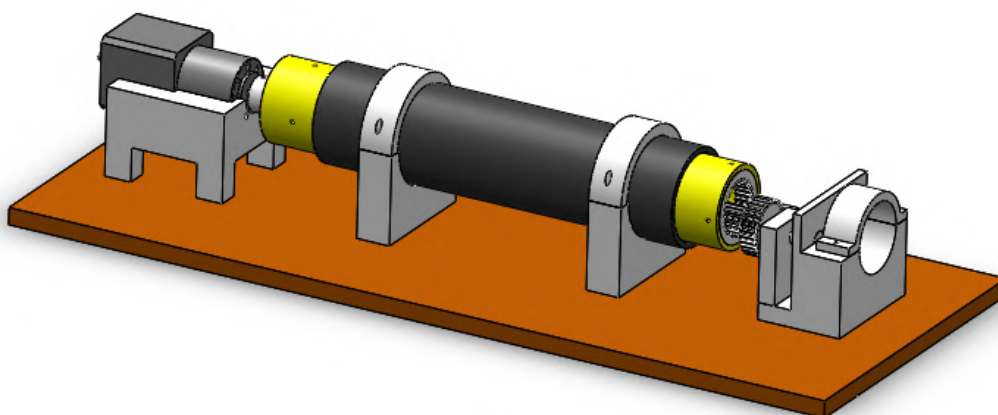
3.12 Irudia: Motorreko biraketa zilindrora transmititzeko mekanismoaren diseinua (a) *SolidWorks*-en. (b) Muntaian erabiliko den pieza.

diseinatu da. Engranaje sistema honetan, kodetzailearen ardatza pieza urdinari lotzen zaio. Pieza urdin eta gorriaren artean 3:1 eta pieza gorriaren eta berdearen artean 2:1 transmisio erlazioak daudenez, mekanismo osoan guztira 6:1 erlazioa lortzen da, modu honetan gutxieneko 5:1 baino handiagoa izatearen baldintza betetzen duen engranaje sistema lortzen da. Azkenik, pieza berdea bere perimetroan zehar eginiko lau zuloren bidez zilindro txikiarekin lotu daiteke.

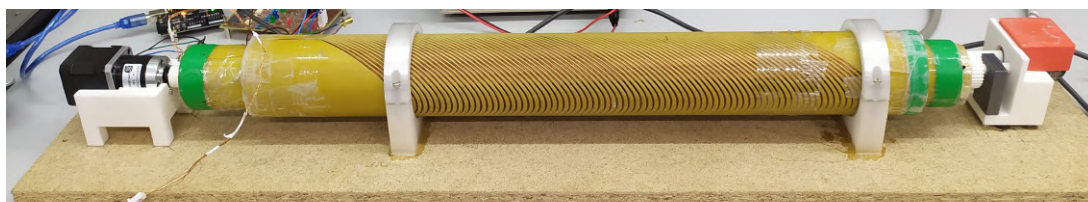


3.13 Irudia: Zilindroaren mugimendua kodetzailera transmititzeko engranaje sistemaren diseinua. (a) *SolidWorks*-en. (b) Muntaian erabiliko den pieza.

SolidWorks-en diseinatutako piezak 3D inprimagailu baten bidez inprimatuz, plataforma osoa eraikitzea lortu da (ikusi 3.14 Irudia). Modu honetan, proiektu honetarako beharrezkoa den plataforma sortua dago eta DHD-aren mugimendua kontrolatuko duen sistemarekin has daiteke.



(a)



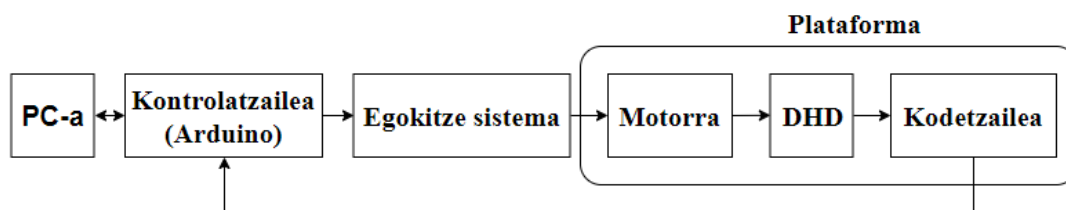
(b)

3.14 Irudia: (a) *SolidWorks* bidez diseinatutako plataforma. Zilindro beltzak kanpoko DHDa eta zilindro horiak barneko DHDa adierazten dituzte. (b) Proiektuan erabiliko den plataforma.

4. Kapituluia

Sistemaren kontrola

Behin plataforma osoa eta kontrolatzailearen seinaleak egokitzeko sistema eginda daudela, DHD-aren mugimendua kontrolatuko duen sistema diseinatu behar da. Honetarako, 4.1 Irudiko bloke diagraman azaldutako eskema erabili da.



4.1 Irudia: Kontrol sistemaren bloke diagrama.

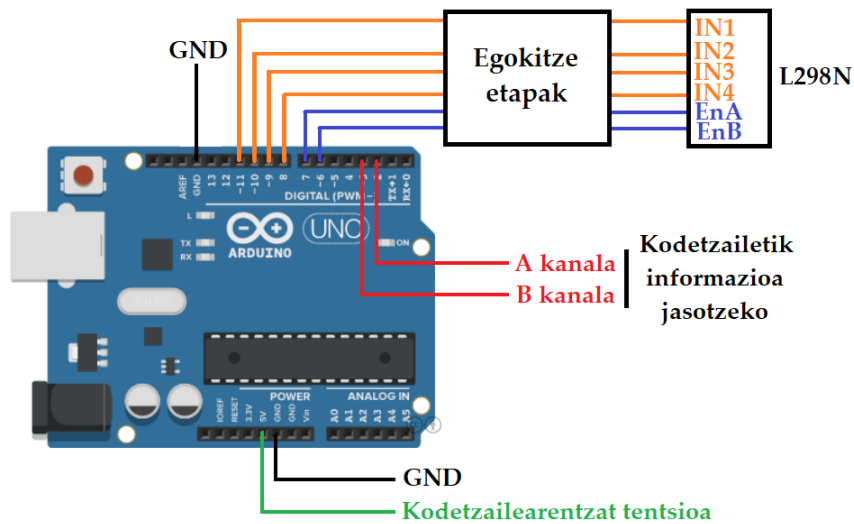
Hasteko, ordenagailutik *Serial* transmisioaren bidez DHD-ak biratu beharreko angelu kopuruaren informazioa bidaliko zaio kontrolatzaileari (gure kasuan, Arduino plakari). Kontrolagailuak, biratu beharreko angelua jakinik, biraketa hau lortu ahal izateko seinale sekuentziak (3.1 Taulan ikusiak) bidaliko dizkio L298N gailuari, 3.2.1 atalean ikusitako egokitze sistematik igaroz.

L298N-ak jasotako seinaleen bidez motorra mugiaraztea lortuko da, eta modu honetan, 3.4 atalean diseinatutako plataformaren bidez, DHD-a eta kodetzailearen biraketa ere burutuko da. Kodetzailearen bidez lortutako informazioa Arduino plakari bidaliko zaio, eta *Serial* komunikazioaren bidez, ordenagailuan ikusi ahal izango da.

Modu honetan, DHD-a biratzea (motorraren bidez) eta benetan zenbat biratu den irakurtzea (kodetzailearen bidez) lortuko da. Hala ere, gure kasuan posizio-kontrol zehatz bat izatea nahi denez, Arduinoko programan PID kontrolagailu bat inplementatuko da, posizio zehaztasuna handitzeko. PID kontrolagailu honek erreferentzia bezala ordenagailutik jasotako seinalea izango du, eta hau

kodetzailetik jasotakoarekin alderatuz, posizio zehaztasuna ahalbidetuko duen kontrol seinalea bidaliko dio motorrari.

Programaketa aztertzen hasi aurretik, erabiliko den Arduinoaren sarrera eta irteera PIN-en diagrama adierazi da 4.2 Irudian.



4.2 Irudia: Arduino konexioak.

Ikus daitekeen modura, Arduinoaren 11, 10, 9, 8, 7 eta 6 PIN-ak L298N-ko IN1, IN2, IN3, IN4, EnA eta EnB sarreretara joango dira, bakoitza bere egokitze etapatik pasa ondoren. Bestetik, kodetzailetik informazioa jaso ahal izateko, kodetzailerearen A kanala Arduinoaren 2 PIN-ean konektatuko da, eta B kanala 3 PIN-ean.

4.1 Programaketa

Plataforma guztia martxan jartzeko azkenengo pausua programaketa izango da. Erabiliko den kodea Arduinoan programatuko da, eta bi zati bereizgarri izango ditu: alde batetik motorraren funtzionamendua ahalbidetuko duen kodea eta bestetik kodetzailetik informazioa jasotzeko erabiliko den kodea. Bi hauek uztartuz plataforma martxan jarri eta mugimenduaren zenbaterainokoa neurtu ahal izango da. Hala ere, plataformaren higidura kontrolatzeko eta funtzionamendu hobe izateko PID kontrolagailu bat gehituko zaio. Modu honetan, hiru kodeak (motorra, kodetzailera eta PID-a) bateratuz plataformaren kontrol zehatza posible izango da.

4.1.1 Motorra abiaraztea

Motorra abiarazteko erabilitako kodea A.1 eranskinean adierazi da. Kode honetan, *norabidea* aldagaiaren zeinua ezarriz, biratu nahi den norabidea zehazten da (EON edo EOAN). Norabide hauei dagozkien pausuen sekuentziak (3.1 Taulan adieraziak) jarraituz, motorra norabide honetan etengabe biratzea lortzen da, berriro ere *norabidea* aldagaiaren zeinua aldatu arte.

Erloju orratzen norabidean (*norabidea* = "+") biratzea aukeratzen bada, adibidez, *pausua* aldagaia 0-tik 200-eraino (motorrak bira oso bat emateko behar duen pausu kopururaino) bere balioa unitate batez gehitzen joango da. Modu honetan, *motorraMugitu()* funtzioko *switch*-ean (0 → 1 → 2 → 3 → 0) sekuentzia jarraituko da, EON-ri dagokiona. *norabidea* = "-" baldin bada, ordea, *pausua* aldagaia 200-tik 0-raino txikitzen joango da eta *switch*-ean jarraituko den sekuentzia aurrekoaren alderantzizkoa izango da.

Kode honekin, motorra etengabe biratzea lortzen da. Hala ere, *norabidea* ez bada ez "+" ez "-", kodeak *gelditu()* funtziora jauzi egingo du, bi ENABLE seinaleak 0-ra eramanez eta ondorioz motorra geldiaraziz.

4.1.2 Kodetzailetik informazioa jasotzea

Proiektu honetan beharrezkoa denez uneoro DHD-a mugitutako gradu kopurua-ren informazioa izatea, kodetzailetik informazioa lortzeko kode bat idatzi beharra dago.

3.3 atalean ikusi den bezala, kodetzaile inkrementalek bi arazo izan ohi dituzte: posizio absolutua eta mugimenduaren noranzkoa ezin direla jakin. Proiektu honetarako lehenengo arazoak ez du garrantzirik izango. Garrantzitsua sistemak biratzen duen angelu kopurua neurtzea izango da.

Bigarren arazoari konponbide bat jarri behar zaio, kodetzailea zein norabidetan mugitzen den jakitea ezinbestekoa baita. Arazo hau konpontzeko eta kodetzailetik jasotako seinaleetatik informazioa jasotzeko kodea A.2 eranskinean adierazi da.

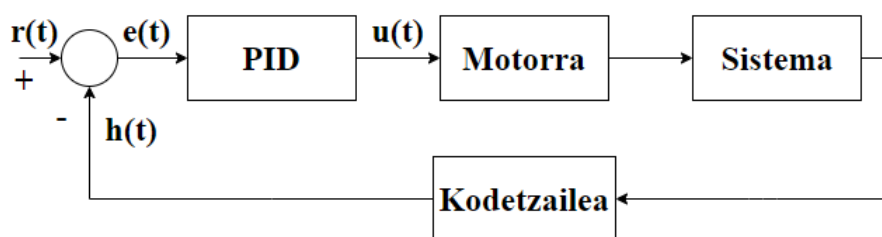
Kode honetako *encoder()* funtzioan, A kanalean 0-tik 1-erako trantsizioa baldin badago, B kanalaren egoera aztertzen da, eta hau jakinda *encoder0Pos* aldagaia aldatzen du, zeinak pausu kopurua adierazten duen. Hau ulertu ahal izateko, erreferentzia modura 3.10 Irudiko 1 puntua erabiliko da. Hasierako egoera 1 puntuan izanik, trantsizio bat baldin badago, B seinalea aztertzen du, eta hau 0 egoeran baldin badago kodetzailea EOAN mugitu dela ondorioztatzen du, *encoder0Pos* aldagaiari balio bat kenduz. B seinalea 1 egoeran baldin badago, ordea, EON mugitzen dela ondorioztatzen da eta *encoder0Pos*-i bat gehitzen dio.

Bestetik, ikus daitekeenez *encoder()* funtzioa ez dago *loop* barruan sartuta. *loop* baten barruan sartuta egongo balitz, pultsu aldaketaren bat galtzeko arris-

kua egongo litzateke pulstu aldaketa hau gertatzean Arduinoa *loop*-ean beste eragiketa batzuk egiten izango balitz. Hau saihesteko *attachInterrupt()* funtzioa erabiltzen da, zeinak A kanalean aldaketaren bat ikusiz gero zuzenean *encoder()* funtziora jauzi egingo duen.

4.1.3 PID kontrolagailua

Motorra abiarazteko eta kodetzailetik informazioa jasotzeko kodeak idatzi ondoren plataforma osoa martxan jar daiteke. Hala ere, sistemaren gainean posizio-kontrol zehatz bat izateko PID kontrolagailu bat diseinatu da begizta itxian (ikusi 4.3 Irudia).



4.3 Irudia: Erabilitako PID-aren bloke-diagrama.

Modu honetan, ordenagailutik jasotako $r(t)$ erreferentzia seinalea eta kodetzailetik jasotako $h(t)$ berrelikatze seinaletik $e(t)$ errore seinalea lortzen da. Errore seinale hau PID-an sartuz, motorra mugiaraziko duen $u(t)$ kontrol seinalea lortuko da:

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de}{dt}, \quad (4.1)$$

K_p , K_i eta K_d proportzionaltasun, integrazio eta deribazio konstanteak izanik, hurrenez hurren.

Motorra mugiaraztea, kodetzailetik informazioa jasotzea eta PID kontrolatzailea exekutatzen duen programaren kodea A.3 eranskinean adierazi da. Kodean, lehenik eta behin, *Serial*-etik mugitu nahi den angelua eta norabidea irakurtzen da: *irakurri1*-en mugitu beharreko angelua irakurtzen da eta *irakurri2*-n zein aldetara mugitu nahi den (EON 90° biratzeko “90+” idatzi beharko litzateke, adibidez). Behin erreferentzia balio hau dugula, kodetzailetik jasotako *posizio-Berria*-rekin alderatu daiteke eta *pid()* kontrol funtzioaren bidez zenbat mugitu behar den agintzen zaio motorrari.

pid() funtzio honetan (4.1) ekuazioko kalkuluak burutzen dira. Funtzioaren irteerako balioari balio maximo eta minimo bat ezartzen zaio, motorrari bidaliko

zaion $u(t)$ seinaleak balio absolutu handiegirik izan ez dezan. Funtzio honetako irteerarekin, beraz, motorrak zein norabide eta abiaduratan biratu zehaztuta geratzen da. Bi hauek jakinda, *motorraMugitu()* funtzioa (A.1 eranskinean definitua) erabili daiteke motorra mugitzeko eta *encoder()* funtzioa (A.2 eranskinean definitua) kodetzailearen posizioaren informazioa lortzeko.

Kode honen bidez, beraz, sistemari mugimendua transferitzea eta mugimendu honen zenbaterainokoaren neurketak egiteaz gain, PID kontrolatzaileak ematen duen posizio-kontrol zehatza lortzen da. PID kontrolatzaile honen bidez lortutako emaitzak hurrengo atalean aztertu dira.

5. Kapituluia

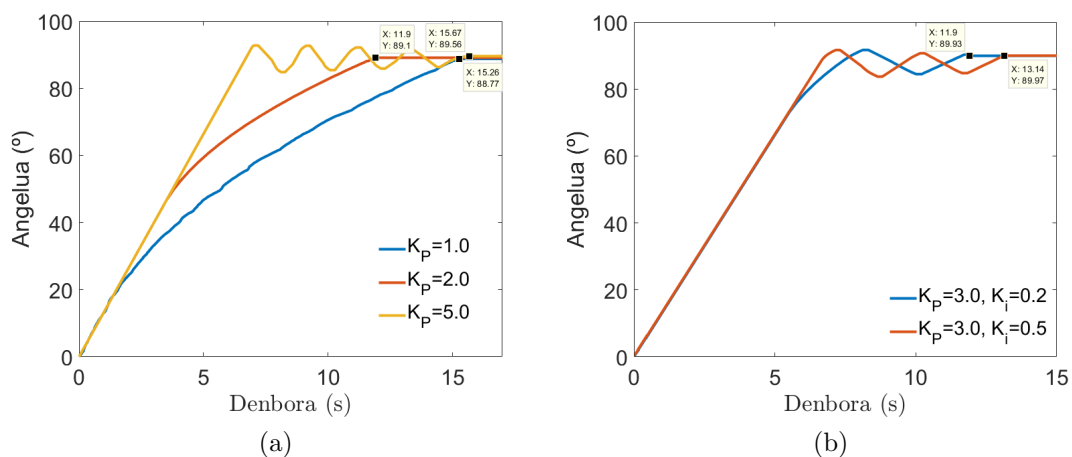
Emaitzak

Behin plataforma osoa muntatuta eta bera kontrolatzeko kodea idatzita dagoela, sistema osoaren funtzionamendua frogatu da.

5.1 PID kontrolatzailearen sintonizazioa

Lehenbizi, PID kontrolagailuaren erabilera ikusi da. Hau egiteko, A.3 eranskineko kodeko K_p , K_i eta K_d balioak eskuz aldatu dira, aldaketa bakoitzean sistemaren erantzuna aztertuz, nahi den erantzuna lortu arte.

Hasteko, kontrol proportzionala bakarrik erabili da, $K_i = 0$ eta $K_d = 0$ aukeratuz. K_p -ren balio ezberdinentzat lortutako balioak 5.1a Irudian adierazi dira 90° balioko sarrera seinale batentzat.



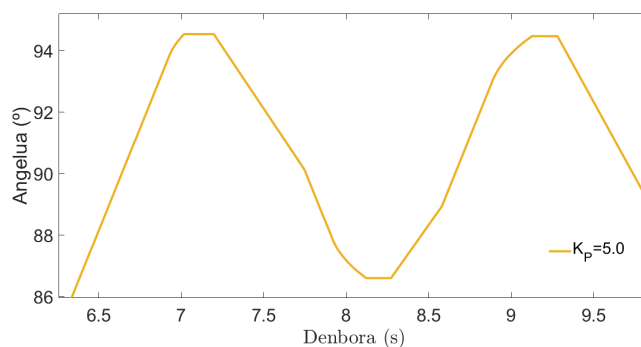
5.1 Irudia: (a) P kontrolagailuaren erantzunak. (b) PI kontrolagailuaren erantzunak.

5.1a Irudian ikus daitekeen modura, K_p irabazia handituz, seinalearen erantzunaren egonkorreko errorea txikitzen da (nahiz eta ezin daitekeen guztiz ezabatu) eta erantzuna azkartzen da. Hala ere, proportzionaltasun konstantearen balioa handituz ($K_p = 5$, adibidez) joan ahala, errorea txikitzen joan arren, erantzunean oszilakortasunak eta gainditzea agertzen dira.

Proiektu honetan, sistemaren abiadurak ez du garrantzirik izango, DHD-a azkarrago edo motelago birarazteak ez baitu ezberdintasunik sortuko. Hala ere, egonkorreko errorea ezabatu behar da, posizio-kontrol zehatz bat lortzea baita helburua. Honetarako, kontrol proportzionalari kontrol integrala gehituko zaio (PI kontrolatzailea), modu honetan egonkorreko errorea ezabatzea lortzen baita.

PI kontrolatzailearen bidez lortutako bi erantzun 5.1b Irudian adierazi dira. Ikus daitekeen modura, kontrol proportzionalari kontrol integrala gehituz egonkorreko errorea txikitzen da, gure sistemarentzat lortu nahi dugun erantzuna lortuz. Hau ikusirik, A.3 eranskineko kodean erabiliko diren balioak $K_p = 3.0$, $K_i = 0.2$ izango dira, modu honetan lortzen den erantzunak oszilakortasun gutxi eta egonkorreko errore txikia baitu, gure sistemarentzat beharrezko diren ezaugarriak.

Honez gain, sistemaren erantzuna aztertzerako orduan, DHD-ko mugimendua kodetzailerara transmititzerako momentuan arazoak sortzen direla ikusi da. Hau ikusteko, 5.1a Irudiko $K_p = 5.0$ erantzunaren handipen bat egin da. 5.2 Irudian ikusten den bezala, motorrak norabidez aldatzen duenean, kodetzailea denbora tarte batez posizio berdinean geratzen da. Hau da, DHD sistema mugitzen da baina kodetzaileak ez du informazio hau jasotzen.



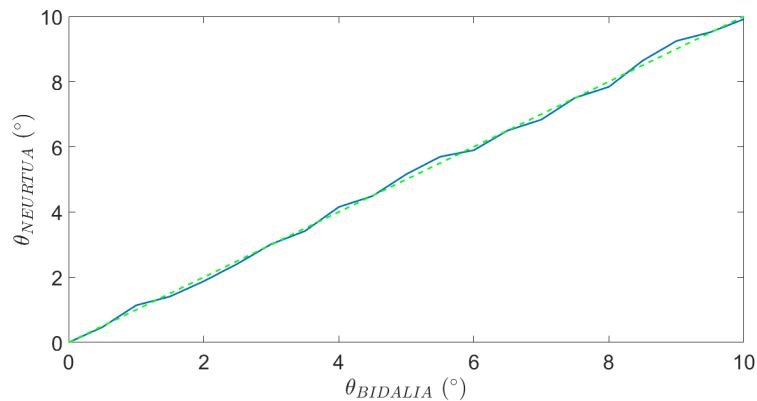
5.2 Irudia: Seinalearen erantzuna handituta.

Efektu honen arrazoia 3.13 Irudian adierazitako engranaje sistemak duen lasaiera da. Engranaje hauen diseinua egiterako orduan hortzen arteko tartea behar bezalakoa ez izategatik DHD-aren eta kodetzailearen arteko mugimendu transmisioa ez da perfektua, informazioa galtzen da. Arazoa konpontzeko hortzen diseinu berri bat egin beharko litzateke; proiektu honetan, ordea, lasaieraren

eragina kontuan hartuko da soilik, diseinu berririk egin gabe.

5.2 Sistemaren mugimenduaren karakterizazioa

Lasaiera honen eragina ikusi ahal izateko, sistemaren mugimenduaren karakterizazio bat egin da. Honetarako, 5.3 Irudian adierazi da ordenagailutik bidalitako seinalearen eta neurtutakoaren arteko erlazioa 0° - 10° tartean.



5.3 Irudia: Bidalitako seinalearen eta neurtutakoaren arteko erlazioa (urdinez) 0° - 10° . Kolore berdez adierazi da $\theta_{BIDALIA} = \theta_{NEURTUA}$ lerroa.

5.3 Irudiak adierazitako lerroan, desbiderapen maximoa $0,246^\circ$ izan da. Hau da, engranaje sistemako lasaieraren ondorioz, hasiera batean lortu beharko litzatekeen bereizmenaren ($0,067^\circ$) ia laukoitza lortzen da. Errore hau neurketak egiterako orduan kontuan hartuko den arren, onena engranaje sistemaren diseinu berri bat egitea izango litzateke.

Dena den, errore hau kontuan hartuta proiektuarekin aurrera jarraitzea posible da, hurrengo atalean ikusiko den modura.

5.3 B_{ROT} -en homogeneotasunaren neurketa

Sistemaren PID-ko emaitzak ikusi eta sistemaren mugimenduaren karakterizazioa egin eta gero, 2.3 atalean azaldutako [10] artikuluko metodoaren bidez B_{ROT} -en intentsitatearen homogeneotasun maximoa lortzen den lerrokatze egoera lortuko da. Honetarako, ordenagailutik barneko DHD-ari gradu konkretu batzuk mugitzeko aginduko zaio, eta mugimendua amaitzean laborategiko sare-analizagailuaren bidez elektroimanen inpedantzia espektroa aztertuko da. Inpedantzia espektro honek erresonantzia maiztasun bakarra duen momentuan

(DHD-en akoplamendua minimoa denean, elkarren arteko inдукtantzia $M=0$), B_{ROT} homogenea dela ondorioztatu ahal izango da.

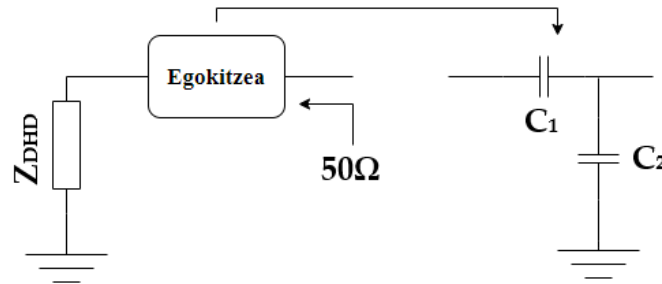
5.3.1 DHD-en sintonizazioa

[10] lanean adierazi bezala, M aztertu aurretik, DHD bakoitza independenteki sintonizatu beharra dago. Gure kasuan, DHD elektroimanen erresonantzia maiztasuna 16 kHz-tan izatea aukeratu da, eta maiztasun honetan elektroimanaren irteerako inpedantzia 50Ω . Erresonantzia maiztasunaren aukeraketa beste bat izan zitekeen (ez du garrantzirik 16kHz edo 50kHz aukeratzeak, adibidez), baina irteerako inpedantzia 50Ω izateak garrantzia du, potentzia transferentzia maximoaren teoriagatik. Teoria honen arabera, iturriaren inpedantzia (gure kasuan sare-analizagailuarena) finkatuta dagoela, potentzia transferentzia maximizatu-ko duen karga inpedantziaren balioak iturriaren inpedantziaren balio bera izan behar du. Sare-analizagailuak 50Ω -ko inpedantzia duenez, karga inpedantziek (DHD-renak) ere balio hau izan beharko dute.

Gure kasuan, 16 kHz-tan,

$$\begin{aligned} Z_{barneDHD} &= (1,038 + 19,26i)\Omega, \\ Z_{kanpoDHD} &= (1,116 + 34,23i)\Omega. \end{aligned}$$

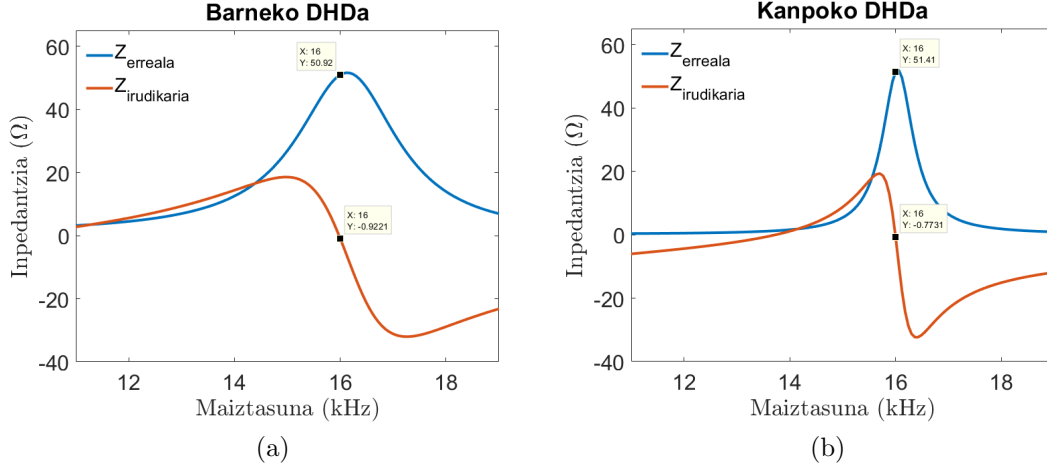
Beraz, potentzia transferentzia maximoa izateko inpedantziak egokitu beharra izan dugu 16 kHz-tan 50Ω balioa izateko, honetarako 5.4 Irudiko zirkuitua erabiliz.



5.4 Irudia: Inpedantziak egoitzeko zirkuitua.

Sintonizazioa independenteki egiteko, plataformatik DHD-ak atera dira, sintonizazioa egiterako orduan ez baitugu nahi DHD-en arteko inдукtantiaren efekturik nabaritzea. Hau horrela, DHD bakoitza indibidualki aztertu da sare-analizagailuan. Egoitze sistemako C_1 eta C_2 aldatuz, DHD bakoitzaren inpedantzia espektroek erresonantzia maiztasuna 16 kHz eta maiztasun honetan

irteerako inpedantzia 50Ω dituzten egoerak bilatu dira, 5.5 Irudian adierazten den modura.



5.5 Irudia: (a) Barneko DHDaren inpedantzia-espektroa. (b) Kanpoko DHDaren inpedantzia-espektroa.

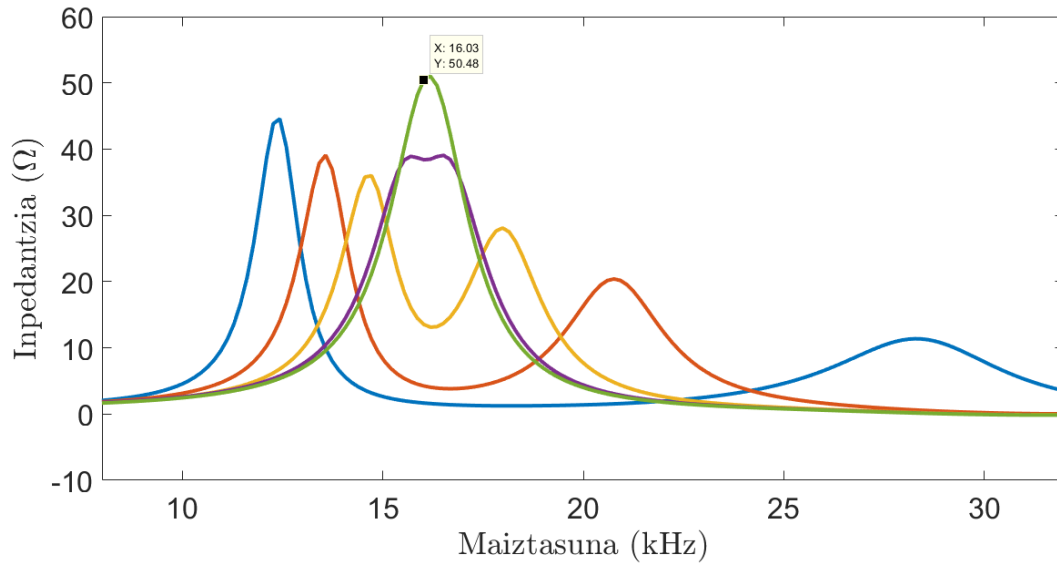
DHD bakoitzarentzat lortzen diren C_1 eta C_2 -ren balioak honakoak dira:

$$\begin{aligned} DHD_{barne} &\rightarrow C_1 = 439,5nF & C_2 = 1196,5nF \\ DHD_{kanpo} &\rightarrow C_1 = 1622nF & C_2 = 880,7nF \end{aligned}$$

Behin biak independenteki aztertu direla (5.5 Irudia), sintonizazioa egokia dela ondorioztatu daiteke, biak 50Ω baliotik gertu baitaude, 51Ω inguruan. Hau frogatu ondoren, bi DHDak berriro ere eraikitako plataforman kokatu dira eta elkarren arteko induktantziak inpedantzia-espektroan duen eragina aztertu da.

5.3.2 DHD-en lerrokatze zuzena

Behin DHDak sintonizatuta eta plataforman kokatuta daudela, [10] artikuluko metodoa erabil daiteke. Honetarako, barneko DHDaren inpedantzia-espektroa aztertu da sare-analizagailuan (kanpoko DHDarena aztertu izan balitz emaitza berdinak lortuko lirateke). Ordenagailutik DHD-a birarazteko angelu konkretu bat bidaltzen zaio sistemari, eta behin sistemak angelu konkretu hauek biratu dituenean sare-analizagailuaren bidez barneko DHD-aren inpedantzia espektroaren datuak hartzen dira. DHDa biratuz angelu ezberdinetara, kasu bakoitzerako inpedantzia-espektroa lortu da, 5.6 Irudian adierazten den bezala.



5.6 Irudia: Barneko DHDaren inpedantzia-espektroa bi DHD-ren lerrokatze egoera ezbedinentzat.

5.6 Irudian ikus daitekeen modura, hasiera batean proposatutako helburua bete da, lerro berdeak adierazitako inpedantzia-espektroan bi elektroimanen akoplamendua desagerrarazten baita (DHD-en arteko induktantzia $M \approx 0$). Beste modu batera esanda, B_{ROT} eremu magnetikoaren homogeneotasun maximoa grafika berdez adierazitako elektroimanen lerrokatzearekin lortuko da.

Honez gain, [10] artikuluan adierazitako beste efektua ere ikusten da 5.6 Irudian: bi elektroimanen elkarrekiko induktantzia (M) handitu ahalaz, inpedantzia-espektroan agertzen diren bi maximoen arteko distantzia handitu egiten da. Hau kontuan izanda, 5.6 Irudiko adierazpen urdinean M maximoa dela ondorioztatu daiteke eta adierazpen berdekoa, espero bezala, minimoa, $M \approx 0$.

Modu honetan, proiektuaren helburua betetzat eman daiteke, [10] artikuluko teoria jarraituz desakoplamendu maximoko egoera lortu baita, hau da, DHD-en arteko induktantzia minimoko egoera, eta honek B_{ROT} -en intentsitatearen homogeneotasuna inplikatzeko du.

6. Kapituluia

Ondorioak

Memoria honetan azaldutako garapenaren bidez, hasiera batean ezarritako helburua lortu da. Hau da, [7] lanean MAFS teknika erabiltzeko eraikitako sisteman erabiltzen diren DHD-en lerrokatze sistema sortu da eta modu honetan MAFS teknika aurrera eraman ahal izateko beharrezko den B_{ROT} eremu magnetikoaren intentsitatearen homogeneotasuna lortu da.

Hau lortu ahal izateko, zenbait puntu jorratu behar izan dira: plataforma sortzeko *SolidWorks* programan egindako diseinu mekanikoak, motorraren funtzionamendurako beharrezko den zirkuitu elektronikoaren analisia, diseinua eta inplementazioa, Arduino programaren bidez plataformara aginduak bidali eta informazioa jasotzeko programazioa eta PID kontrolagailua, besteak beste.

Hauek guztiak bateratuz lortutako sistemaren bidez B_{ROT} eremuaren homogeneotasuna lortu da. Gainera, bi DHD-en arteko induktantziaren portaera ikusi da elektroimanen posizionamendu ezberdinentzat, eta portaera hau teorian aurreikusitakoarekin bat datorrela ikusi da.

Hala ere, emaitzetan ikusi den modura, egin den diseinuak akats nabarmen bat du, erabilitako engranaje sistemak lasaiera bat baitu. Proiektuarekin aurrera jarraituko balitz, engranaje sistemaren diseinu berri bat burutu beharko litzateke, modu honetan posizioaren zehaztasuna hobetuko litzateke eta. Honez gain, interesgarria izango litzateke aztertutako posizio bakoitzean DHD-en arteko induktantzia zein den kalkulatzeko, honetarako [10] artikuluan azaldutako eredu matematikoa jarraituz.

Bibliografia

- [1] E.M. Purcell, H.C. Torrey eta R.V. Pound. Resonance absorption by nuclear magnetic moments in a solid. *Physical review*, 69(1-2):37, 1946.
- [2] E.R. Andrew. Magic angle spinning. *International Reviews in Physical Chemistry*, 1(2):195-224, 1981.
- [3] W.I. Goldburg eta M. Lee. Nuclear magnetic resonance line narrowing by a rotating RF field. *Phys. Rev. Lett.*, 11:255-258, 1963.
- [4] M. Lee eta W.I. Goldburg. Nuclear-magnetic-resonance line narrowing by a rotating rf field. *Physical Review*, 140(4A):A1261, 1965.
- [5] E.R. Andrew eta R.G. Eades. Possibilities for high-resolution nuclear magnetic resonance spectra of crystals. *Discussions of the Faraday Society*, 34:38-42, 1962.
- [6] C.A. Meriles, D. Sakellariou, A. Moule, M. Goldman, T.F. Budinger eta A. Pines. High-resolution nmr of static samples by rotation of the magnetic field. *Journal of Magnetic Resonance*, 169(1):13-18, 2004.
- [7] J. Alonso, B. Sisniega, I. Rodrigo, J. Perez, J.M. Collantes eta F. Plazaola. Design, Construction, and Characterization of a Magic Angle Field Spinning RF Magnet, *IEEE Transactions on Instrumentation and Measurement*, PP(99):1-10, 2018.
- [8] E.L. Bronaugh. Helmholtz coils for calibration of probes and sensors: limits of magnetic field accuracy and uniformity. *Electromagnetic Compatibility, 1995. Symposium Record, 1995 IEEE International Symposium*, 72-76. IEEE, 1995.
- [9] S. Farinon eta P. Fabbriatore. Refined modeling of superconducting double helical coils using finite element analyses. *Superconductor Science and Technology*, 25(6):065006, 2012.

- [10] R.F. Lee, R.O. Giaquinto eta C.J. Hardy. Coupling and Decoupling Theory and Its Application to the MRI Phased Array. *Magnetic Resonance in Medicine*, 48:203-213, 2002.

A. Eranskina

Arduino kodea

A.1 Motorra abiaraztea

```
//Arduinoaren irteerak definitu
int in1Pin = 11; //IN1
int in2Pin = 10; //IN2
int in3Pin = 9; //IN3
int in4Pin = 8; //IN4
int enableA = 7; //H1 zubia martxan
int enableB = 6; //H2 zubia martxan
//Pausu baten eta hurrengoaren arteko denbora tartea definitu
unsigned long aurrekoDenbora = 0;
long denboraTartea = 10; //Motorraren abiadura definitu
//Motorraren aldagaiak zehaztu
int pausua = 0; //Motorra zein pausutan dagoen
int pausuKopuruTotala = 200; //Motorrak bira osoa emateko behar
    dituen pausuak (360/1.8)
char norabidea; //Motorrak EON (zenbaki positiboa) edo EOAN
    (negatiboa) biratzen duen definitu

void setup() {
    pinMode(in1Pin,OUTPUT);
    pinMode(in2Pin,OUTPUT);
    pinMode(in3Pin,OUTPUT);
    pinMode(in4Pin,OUTPUT);
    pinMode(enableA,OUTPUT);
    pinMode(enableB,OUTPUT);
}

void loop() {
    norabidea = '+'; //Norabidea definitu
    unsigned long oraingoDenbora = millis(); //Abiatzetik igaro den
    denbora
```

```

if (oraingoDenbora - aurrekoDenbora >= denboraTartea){
  aurrekoDenbora = oraingoDenbora;
  //Norabidearen arabera, zentzu batean edo bestean biratu
  if (norabidea == '+'){ //Erloju orratzen norabidean
    pausua++;
    if (pausua == pausuKopuruTotala){
      pausua = 0;
    }
    motorraMugitu(pausua%4);
  }
  else if (norabidea == '-'){ //Erloju orratzen aurkakoa
    if (pausua == 0){
      pausua = pausuKopuruTotala;
    }
    pausua--;
    motorraMugitu(pausua%4);
  }
  else{ //Motorra gelditu
    gelditu();
  }
}
}

void motorraMugitu(int paso){
  digitalWrite(enableA ,HIGH);
  digitalWrite(enableB ,HIGH);
  switch(paso){
    case 0: //1010
      digitalWrite(in1Pin ,HIGH);
      digitalWrite(in2Pin ,LOW);
      digitalWrite(in3Pin ,HIGH);
      digitalWrite(in4Pin ,LOW);
      break;
    case 1: //1001
      digitalWrite(in1Pin ,HIGH);
      digitalWrite(in2Pin ,LOW);
      digitalWrite(in3Pin ,LOW);
      digitalWrite(in4Pin ,HIGH);
      break;
    case 2: //0101
      digitalWrite(in1Pin ,LOW);
      digitalWrite(in2Pin ,HIGH);
      digitalWrite(in3Pin ,LOW);
      digitalWrite(in4Pin ,HIGH);
      break;
    case 3: //0110
      digitalWrite(in1Pin ,LOW);
      digitalWrite(in2Pin ,HIGH);
      digitalWrite(in3Pin ,HIGH);

```



```

        digitalWrite(in4Pin,LOW);
        break;
    }
}

void gelditu(){
    digitalWrite(enableA,LOW);
    digitalWrite(enableB,LOW);
}

```

A.2 Kodetzailetik informazioa jasotzea

```

//Arduinoaren sarrerak definitu
int encoder0PinA = 2;
int encoder0PinB = 3;
//Encoderra mugitzen den ikusteko aldagaiak
volatile long encoder0Pos = 0;
int encoder0PinAAzkena = LOW;
int n;
int azkenPosizioa = 0;
int posizioBerria;

void setup() {
    pinMode(encoder0PinA,INPUT);
    pinMode(encoder0PinB,INPUT);
    Serial.begin(2000000);
    attachInterrupt(0, encoder, CHANGE); //0 erabiltzen da Arduinoaren
    2 PIN-a adierazteko (A kanala konektatuta dagoena)
}

void loop() {
    posizioBerria = encoder0Pos;
    if (posizioBerria != azkenPosizioa) {
        azkenPosizioa = posizioBerria;
        Serial.println(azkenPosizioa);
    }
}

void encoder(){
    n = digitalRead(encoder0PinA);
    if ((encoder0PinAAzkena == LOW) && (n == HIGH)) {
        if (digitalRead(encoder0PinB) == LOW) {
            encoder0Pos--;
        } else {
            encoder0Pos++;
        }
    }
}
encoder0PinAAzkena = n;

```

```
}

```

A.3 PID kontrolagailua

```
//Arduinoaren irteerak definitu
int in1Pin = 11;
int in2Pin = 10;
int in3Pin = 9;
int in4Pin = 8;
int enableA = 7;
int enableB = 6;
//Arduinoaren sarrerak definitu
int encoder0PinA = 2;
int encoder0PinB = 3;
//Serial-etik datuak irakurtzeko
float irakurri1; //Graduak
char irakurri2; //Norabidea, + edo -
//Encoderra mugitzen den ikusteko aldagaiak
volatile long encoder0Pos = 0;
int encoder0PinAAzkena = LOW;
int n;
int azkenPosizioa = 0;
int posizioBerria;
//Pausu baten eta hurrengoaren arteko denbora tartea definitzeko
aldagaiak
unsigned long aurrekoDenbora = 0;
unsigned long oraingoDenbora;
long denboraTartea = 0;
//Motorraren aldagaiak zehaztu
int pausua = 0;
int pausuKopuruTotala = 200;
int norabidea;
int norabideaAbs;
int emanBeharrekoPausuak;
int erreferentzia;
//PID-aren aldagaiak zehaztu
double Kp = 3.0; //Proporzionaltasun konstantea
double Ki = 0.2; //Integrazio konstantea
double Kd = 0.0; //Deribazio konstantea
int erroreBatura = 0;
int azkenErrorea = 0;
double norMax = 2500.0;
double norMin = -2500.0;

void setup() {
  pinMode(in1Pin,OUTPUT);
  pinMode(in2Pin,OUTPUT);
  pinMode(in3Pin,OUTPUT);

```

```

pinMode(in4Pin,OUTPUT);
pinMode(enableA,OUTPUT);
pinMode(enableB,OUTPUT);
pinMode(encoder0PinA,INPUT);
pinMode(encoder0PinB,INPUT);
Serial.begin(2000000);
attachInterrupt(0, encoder, CHANGE);
}

void loop() {
  posizioBerria = encoder0Pos;
  if (posizioBerria != azkenPosizioa) {
    azkenPosizioa = posizioBerria;
    Serial.println(posizioBerria);
  }
  //Ordenagailutik mugitu beharreko angeluaren balioa eta norabidea
  jaso eta kodetzailea mugitu beharreko erreferentzia puntua
  zehaztu
  oraingoDenbora = millis();
  if (oraingoDenbora - aurrekoDenbora >= denboraTartea){
    aurrekoDenbora = oraingoDenbora;
    if (Serial.available()>2){
      irakurri1 = Serial.parseFloat();
      irakurri2 = Serial.read();
      if (irakurri2 == '+') {
        emanBeharrekoPausuak=(int)((irakurri1)*1024.0*6/360.0);
      }
      if (irakurri2 == '-') {
        emanBeharrekoPausuak=-((int)((irakurri1)*1024.0*6/360.0);
      }
      erreferentzia=emanBeharrekoPausuak+azkenPosizioa;
    }
    //PID zatia
    norabidea = int(pid(posizioBerria,erreferentzia, Kp, Ki, Kd,
    norMax, norMin, denboraTartea));
    norabideaAbs = abs(norabidea);
    denboraTartea = map(norabideaAbs, 0, 2500, 20, 5);

    //Norabidearen arabera, zentzu batean edo bestean biratu
    if (norabidea > 0){ //EOAN
      pausua++;
      if (pausua == pausuKopuruTotala){
        pausua = 0;
      }
      motorraMugitu(pausua%4);
    }
    else if (norabidea < 0){ //EON
      if (pausua == 0){
        pausua = pausuKopuruTotala;

```

```

    }
    pausua--;
    motorraMugitu(pausua%4);
  }
  else{
    gelditu();    //Motorra geldiarazi
  }
}
}

double pid(int sarrera , int erreferentzia , double kp, double ki ,
  double kd, double maximo, double minimo, int laginketaDenbora){
  //sarrera—>non gauden; erreferentzia—>nora goazen
  //Laginketa denbora segundotan
  double laginketaDenboraS = ((double)laginketaDenbora)/1000;
  //Zati proportzionala
  double errorea = erreferentzia -sarrera;
  double pTerm = kp * errorea;
  //Zati integrala
  erroreBatura += errorea;
  if (erroreBatura > maximo) erroreBatura = maximo;
  else if (erroreBatura < minimo) erroreBatura = minimo;
  double iTerm = ki * laginketaDenboraS * erroreBatura;
  //Zati diferentziala
  int dError = errorea - azkenErrorea;
  double dTerm = kd * dError / laginketaDenboraS;
  //PID
  double kontSeinalea = pTerm + iTerm + dTerm;    //Kontrol seinalea
  azkenErrorea = errorea;
  if (kontSeinalea > maximo) kontSeinalea = maximo;
  else if (kontSeinalea < minimo) kontSeinalea = minimo;
  return kontSeinalea;
}

void gelditu(){
  digitalWrite(enableA ,LOW);
  digitalWrite(enableB ,LOW);
}

void motorraMugitu(int paso){//A.1 eranskinoko motorraMugitu()
  funtzioa
}

void encoder(){//A.2 eranskinoko encoder() funtzioa
}

```

