



MIKROKONTROLADORE BATEAN OINARRITUTAKO ETXE DOMOTIKOA

1. MEMORIA

IKASLEAREN DATUAK

IZENA: ADRIAN

ABIZENAK: RUIZ CORPAS

SIN.:

DATA: 26/01/2016

ZUZENDARIAREN DATUAK

IZENA: OSKAR

ABIZENAK: CASQUERO OYARZABAL

SAILA: SISTEMEN INGENIARITZA ETA AUTOMATIKA

SIN.:

DATA: 26/01/2016



AURKIBIDEA

1. SARRERA.....	1
2. PROIEKTUAREN HELBURUA.....	3
3. PROIEKTUAREN DESKRIBAPENA.....	5
3.1 Deskribapen orokorra.....	5
3.2. Alternatibak.....	7
3.2.1. Mikrokontroladoreak.....	7
3.2.1.1. Arduino.....	7
3.2.2.2. Rasberry PI.....	10
3.2.3.3. Intel Edison.....	12
3.2.2. Sergurtasun sentsoreak.....	13
3.2.2.1. Ultrasoinu sentsoreak.....	13
3.2.2.2. infragorri sentsoreak.....	13
3.2.3. Erloju digitalak.....	14
3.3. Aukeraketa eta irizpideak.....	15
3.5. Aukeraketaren tabla.....	17
3.6. Deskribapen zehatza.....	17
4. PROIEKTUAREN DISEINUA.....	23
4.1. Sarrera.....	23
4.2. Muntaia.....	25
4.2.1. Hasierako maketa.....	25
4.2.2. Maketa finala.....	27
4.2.1. Muntaiaren izandako arazoak.....	33
5. SENTSOREAK.....	34
5.1. PIR.....	34
5.1.1. Deskribapena.....	34
5.1.2. Helburua proiektuan.....	34
5.1.3. Funtzionamendua.....	35
5.1.4. Eskema eta konexioak.....	36
5.2. Kamera.....	36
5.2.1. Deskribapena.....	36
5.2.2. Helburua proiektuan.....	37
5.2.3. Funtzionamendua.....	37



5.2.4. Eskema eta konexioak	38
5.3. TFT pantaila	38
5.3.1. Deskribapena	38
5.3.2. Helburua proiektuan	39
5.3.3. Funtzionamendua.....	40
5.3.4. Eskema eta konexioak	41
5.4. RTC	42
5.4.1. Deskribapena	42
5.4.2. Helburua proiektuan	42
5.4.3. Funtzionamendua.....	43
5.4.4. Eskema eta konexioak	43
5.5. LDR	44
5.5.1. Deskribapena	44
5.5.2. Helburua proiektuan	44
5.5.3. Funtzionamendua.....	45
5.5.4. Eskema eta konexioak	45
5.6. Hatz-aztarna detektagailua.....	46
5.6.1. Deskribapena	46
5.6.2. Helburua proiektuan	46
5.6.3. Funtzionamendua.....	47
5.6.4. Eskema eta konexioak	48
5.7. RFID	48
5.7.1. Deskribapena	48
5.7.2. Helburua proiektuan	49
5.7.3. Funtzionamendua.....	49
5.7.4. Eskema eta konexioak	51
6. KONPONENTEAK.....	52
6.1. Erresistentziak	52
6.1.1. Deskribapena	52
6.1.2. Helburua proiektuan	52
6.1.4. Eskema eta konexioak	53
6.2. Transistorea	53
6.2.1. Deskribapena	53
6.2.2. Helburua proiektuan	54
6.2.3. Funtzionamendua.....	54



6.2.4. Eskema eta konexioak	56
6.3. Sarraila.....	56
6.3.1. Deskribapena	56
6.3.2. Helburua proiektuan	57
6.3.3. Funtzionamendua.....	57
6.3.4. Eskema eta konexioak	58
6.4. Serbomotorra	58
6.4.1. Deskribapena	58
6.4.2. Helburua proiektuan	59
6.4.3. Funtzionamendua.....	59
6.4.4. Eskema eta konexioak	59
6.5. Motorra	60
6.5.1. Deskribapena	60
6.5.2. Helburua proiektuan	61
6.5.3. Funtzionamendua.....	61
6.5.4. Eskema eta konexioak	61
6.6. motorraren driver-a (H zubia).....	62
6.6.1. Deskribapena	62
6.6.2. Helburua proiektuan	63
6.6.3. Funtzionamendua.....	63
6.6.4. Eskema eta konexioak	64
6.7. mikroSD txartela.....	64
6.8. tentsio maila bihurtzailua.....	65
6.9. Transformadorea.....	66
7. HARDWARE	70
8. SOFTWARE.....	72
8.1. SOLID EGDE.....	72
8.2. FRITZING	74
8.3. DESIGNSPARK.....	75
8.4. cura	79
9. PROGRAMAZIOA.....	82
9.1. SARRERA.....	82
9.2. Programaren adibide grafikoak.....	82
9.2.1. Graficet:	82
• Logela:.....	82



• sarrera:.....	83
• Egongela.....	84
9.2.2. Prograna nagusia.....	85
9.3. Liburutegiak.....	86
9.3.1. Liburutegien esanahia.....	86
9.3.2. LIBURUTEGIEN INSTALAZIOA	86
9.3.3. LIBURUTEGIEN ERAIKUNTZA	87
9.4. DISPOSITIBOEN PROGRAMAZIOA	91
9.4.1. PIR.....	91
9.4.2. KAMERA.....	93
8.4.3. TFT PANTAILA	95
9.4.4. RTC	98
8.4.5. LDR	99
8.4.6. HATZ-AZTARNA DETEKTAGAILUA	99
8.4.7. RFID	100
8.4.8. SARRAILA	101
8.4.9. SERBOMOTORRA	101
8.4.10. MOTORRA	102
8.4.11. SD	103
10. BIBLIOGRAFIA.....	104
10.1. kONTSULTATUTAKO WEB ORRIAK.....	104



IRUDIEN AURKIBIDEA

1.1 IRUDIA: ETXE DOMOTIKOA.....	1
2.1 IRUDIA: KONPETENTZIAK.	4
3.1 IRUDIA: ETXEAREN BLOKE DIAGRAMA 1.....	6
3.2 IRUDIA: ARDUINO UNO.	8
3.3 IRUDIA: FREADUINO UNO.	9
3.4 IRUDIA: ARDUINO MEGA.....	9
3.5 IRUDIA: RASPBERRY PI A+.....	10
3.6 IRUDIA: RASPBERRY PI B+.....	11
3.7 IRUDIA: RASPBERRY PI 2B.....	12
3.8 IRUDIA: INTEL EDISON.....	12
3.9 IRUDIA: HC SR04.....	13
3.10 IRUDIA: PIR SENTSOREA.....	14
3.11 IRUDIA: DS1307.....	14
3.12 IRUDIA: RTC TINY.....	15
3.9 IRUDIA: ETXEAREN BLOKE DIAGRAMA 2.....	18
3.10 IRUDIA: SARRERA KONTROLA.....	19
3.11 IRUDIA: LOGELA KONTROLA.....	20
3.12 IRUDIA: EGONGELA KONTROLA.....	21
3.13 IRUDIA: TFT PANTAILAREN KONFIGURAZIOA.....	22
4.1 IRUDIA: SARRERA BARNEALDEA.....	23
4.2 IRUDIA: LOGELA.....	24
4.3 IRUDIA: EGONGELA BARNEALDEA.....	24
4.4 IRUDIA: KONTROL KONTROLA.....	25
4.5 IRUDIA: LANTEGIA.....	26
4.6 IRUDIA: MAKETAREN OINARRIA PLANOAREKIN.....	26
4.7 IRUDIA: HASIERAKO MAKETA.....	27
4.8 IRUDIA: MAKETAREN KONPONBIDEA.....	27
4.9 IRUDIA: SARRERAREN ATALA BARNEALDEAN.....	28
4.10 IRUDIA: SARRERAREN KANPOALDEA.....	28
4.11 IRUDIA: LOGELA.....	29
4.12 IRUDIA: PERTSIANEN PIEZAK ETA MATERIALA.....	30
4.13 IRUDIA: MOTORRA LOTURA.....	30
4.14 IRUDIA: EGONGELAREN KANPOALDEA.....	31
4.15 IRUDIA: KONEXIO EREMUA.....	31



4.16 IRUDIA: MAKETA OSOA	32
4.17 IRUDIA: MOTOR LOTURA TXARRA.....	33
5.1 IRUDIA: PIR	34
5.2 IRUDIA: PULL-UP ESKEMA	35
5.3 IRUDIA: PIR ESKEMA.....	36
5.4 IRUDIA: PIR ESKEMA.....	36
5.5 IRUDIA: KAMERA ESKEMA.....	38
5.6 IRUDIA: TFT PANTAILA.....	39
5.7 IRUDIA: TFT PANTAILA ETA ARDUINO MEGA	41
5. 8 IRUDIA: RTC	42
5.9 IRUDIA: TFT PANTAILA.....	43
5.10 IRUDIA: LDR.....	44
5.11 IRUDIA: LDR.....	44
5.12 IRUDIA: LDR KONEXIOAK.....	45
5.13 IRUDIA: HATZ-AZTARNA DETEKTAGAILUA	46
5.14 IRUDIA: RFID.....	48
5.15 IRUDIA: MIFARE SISTEMA.....	50
5.16 IRUDIA: MIFARE SISTEMA.....	51
6.1 IRUDIA: ERRESISTENTZIAREN IKURRA	52
6.2 IRUDIA: 470Ω-EKO ERRESISTENTZIA.....	53
6.3 IRUDIA: 220Ω-EKO ERRESISTENTZIA.....	53
6.4 IRUDIA: NPN TRANSISTOREA.....	54
6.5 IRUDIA: TRANSISTOREAREN IKURRA	54
6.6 IRUDIA: TRANSISTOREAREN ZIRKUITUA.....	55
6.8 IRUDIA: TRANSISTOREAREN ESKEMA.....	56
6.9 IRUDIA: SARRAILA.....	57
6.10 IRUDIA: SARRAILAREN ESKEMA.....	58
6.11 IRUDIA: SERBOMOTORRAREN ESKEMA	60
6.12 IRUDIA: MOTORRA	60
6.13 IRUDIA: MOTORRAREN ESKEMA	62
6.14 IRUDIA: MOTORRAREN ESKEMA	62
6.15 IRUDIA: H ZUBIA.....	63
6.16 IRUDIA: ZIRKUITU INTEGRATUAREN ESKEMA.....	64
6.17 IRUDIA: SD TXARTELA.....	65
6.18 IRUDIA: BREAKOUT.....	65
6.19 IRUDIA: BREAKOUT + HATZ-AZTARNA DETEKTAGAILUA.....	66



6.20 IRUDIA: TRANSFORMADOREA.....	66
6.21 IRUDIA: SEINALE FROGAK.....	67
6.22 IRUDIA: SEKUNDARIOAREN SEINALEA.....	68
6.23 IRUDIA: KONDENTSADOREAREN OSTEAN EMATEN DUEN SEINALEA.....	68
6.24 IRUDIA: ERREGULATZAILEAREN IRTEERA.....	69
7.1 IRUDIA: ARDUINO MEGA.....	70
8.1 IRUDIA: HASIERAKO PANTAILA.....	72
8.2 IRUDIA: HASIERAKO PANTAILA.....	73
8.3 IRUDIA: PIEZA 3D.....	73
8.4 IRUDIA: FORMATUA.....	74
8.5 IRUDIA: FRITZING.....	74
8.6 IRUDIA: PROTOBOARD.....	75
8.7 IRUDIA: DESIGNSPARK ESKEMA.....	76
8.8 IRUDIA: DESIGNSPARK ZIRKUITUA.....	76
8.9 IRUDIA: DESIGNSPARK.....	77
8.10 IRUDIA: ATAKATZEN.....	77
8.11 IRUDIA: 1 FROGA.....	78
8.12 IRUDIA: 2 FROGA.....	78
8.13 IRUDIA: AZKENENGO FROGA.....	78
8.14 IRUDIA: DESIGNSPARK.....	79
8.15 IRUDIA: CURA BI PIEZA.....	79
8.16 IRUDIA: CURA KAPAK.....	80
8.16 IRUDIA: CURA KAPAK.....	80
8.18 IRUDIA: 3D INPRIMAGAILUA.....	81
9.1 IRUDIA: LOGELAREN GRAFCET-A.....	83
9.2 IRUDIA: SARRERAREN GRAFCET-A.....	83
9.3 IRUDIA: EGONGELAREN GRAFCET-A.....	84
9.4 IRUDIA: LIBURUTEGIEN INSTALAKUNTZA.....	85
9.5 IRUDIA: LIBURUTEGIEN INSTALAKUNTZA.....	86
9.6 IRUDIA: LIBURUTEGIEN INSTALAKUNTZA.....	88
9.7 IRUDIA: ALDERDI PRIBATUA.....	89
9.8 IRUDIA: FITXATEGIA.....	90
9.9 IRUDIA: PROGRAMA NAGUSIA.....	90



TAULEN AURKIBIDEA

1. TABLA: KONTROLADOREAREN AUKERAKETA	17
2. TABLA: KONTROLADOREAREN AUKERAKETA	17
3. TABLA: KONTROLADOREAREN AUKERAKETA.	17
5.1 TABLA: RFID MEMORIAREN EGITURA	48
5.2. TABLA: RFID ATZIPEN BAIMENAK ADIBIDE BAT	50
9.1. TABLA: TFT OPERAZIOAK	96
9.2. TABLA: TFT FUNTZIOAK	97
9.3. TABLA: RTC FUNTZIOAK.....	98
9.4. TABLA: RTC FUNTZIOAK.....	100



1. SARRERA

Gaur egun, domotikak indar handia dauka teknologiaren arloan. Domotika etxe edo eraikinen sistema automatizatzeko gai den teknologiarri deritzo. Zenbait zerbitzu gehitu dezake: energiaren kudeaketa, segurtasuna, erosotasuna, komunikazioa... etxe barrutik edo kanpotik kontrolatu ahal dena.



1.1 Irudia: Etxe domotikoa.

Aterata: <https://es.wikipedia.org/wiki/Dom%C3%B3tica#/media/File:CITIB-56.jpg>

Proiektu honetan etxe domotiko bat eraiki eta kontrolatu egingo da. Etxe horrek sarreraren kontrola, logelaren segurtasuna eta egongelaren pertsianaren kontrolaz arduratuko da.

Etxearen maketa hiru ataletan banatuko da: sarrera, logela eta egongela. Sarreran etxera sartzen denaren kontrola eramango da, logelan sarkin bat sartzekotan segurtasun sistema baten bitartez kontrolatuko da eta egongelan pertsianaren kontrola gauzatuko da. Honetaz gain, arduino TFT pantailaren kontrolaz arduratuko da. TFT pantailaren bitartez etxearen atal guztiak manipulatu ahal dira.

Proiektuaren elementu nagusia Arduino izango da eta honen bidez etxearen atal guztien kontrola gauzatuko da. Gure kasuan, Arduino MEGA erabiliko da.



Beraz, bukaeran etxe domotiko baten maketa izango dugu eta automatizatutako atalak eta TFT pantaila direla medio, sistemak lan egingo du programatuta dagoen moduan.



2. PROIEKTUAREN HELBURUA

Proiektu honen helburua etxe domotiko baten maketa bat eraikitzea eta dagokion sistema ezartzea mikrokontrolagailu bat erabiliz. Etxea sortuz doan heinean, agertuko diren arazoa eta eragozpen guztiak aztertuko dira.

Etxe hau programatuta dauden atal guztiei dagokien erantzuna emateko gai izan beharko da. Jasotako egoera edo aginduaren arabera, sistemak egoera edo agindu hori identifikatu eta eman beharreko erantzuna emango du. Adibidez, sarkin bat logelara sartzen bada, sistema segurtasun protokoloa aktibatzekeo gai izan beharko da.

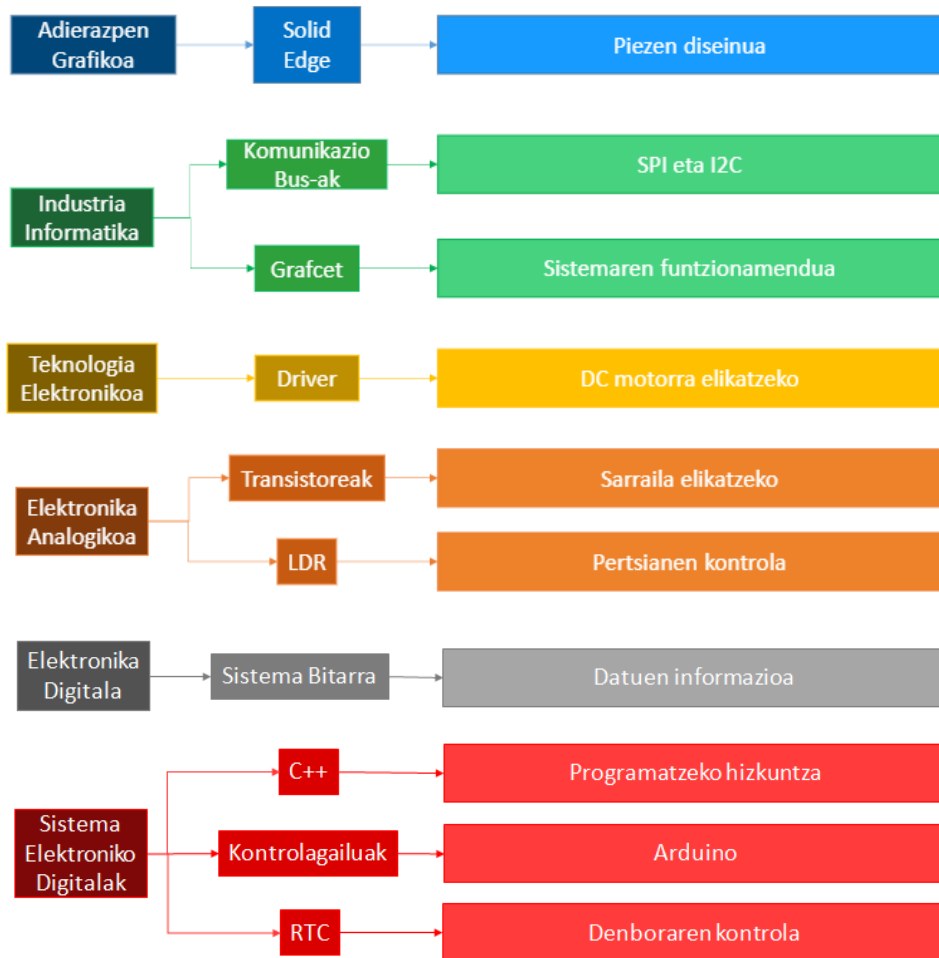
Helburu orokorraz gain, proiektuaren bidez graduan zehar eskuratutako kompetentzia espezifiko eta zeharkakoak praktikan jarri nahi da.

Horien artean, honako hauek:

- Dispositibo elektronikoaren programazioari loturiko kompetentziak, kasu honetan mikrokontrolagailua.
- Dispositibo elektronikoaren muntaia eta diseinuari loturiko kompetentziak, kasu honetan prototipo bat eraikiz praktikan jarriko direnak.
- Maketa eraikitzekeo egin beharreko planoak eta piezak.
- Proiektu baten idatzizko eta ahozko aurkezpenarekin loturiko kompetentziak. Hauek oso garrantzitsuak eta ezinbestekoak dira ideiak era ulergarri eta argi batean adierazteko, bai proiektuaren dokumentazioan, bai proiektuaren defentsan.



Hurrengo irudian, gradu amaierako lanaren zehar egindako zereginak, graduan zehar eskuratutako ikasgai eta konpetentziekin erlazionatzen dira:



2.1 Irudia: Konpetentziak.



3. PROIEKTUAREN DESKRIBAPENA

3.1 DESKRIBAPEN OROKORRA

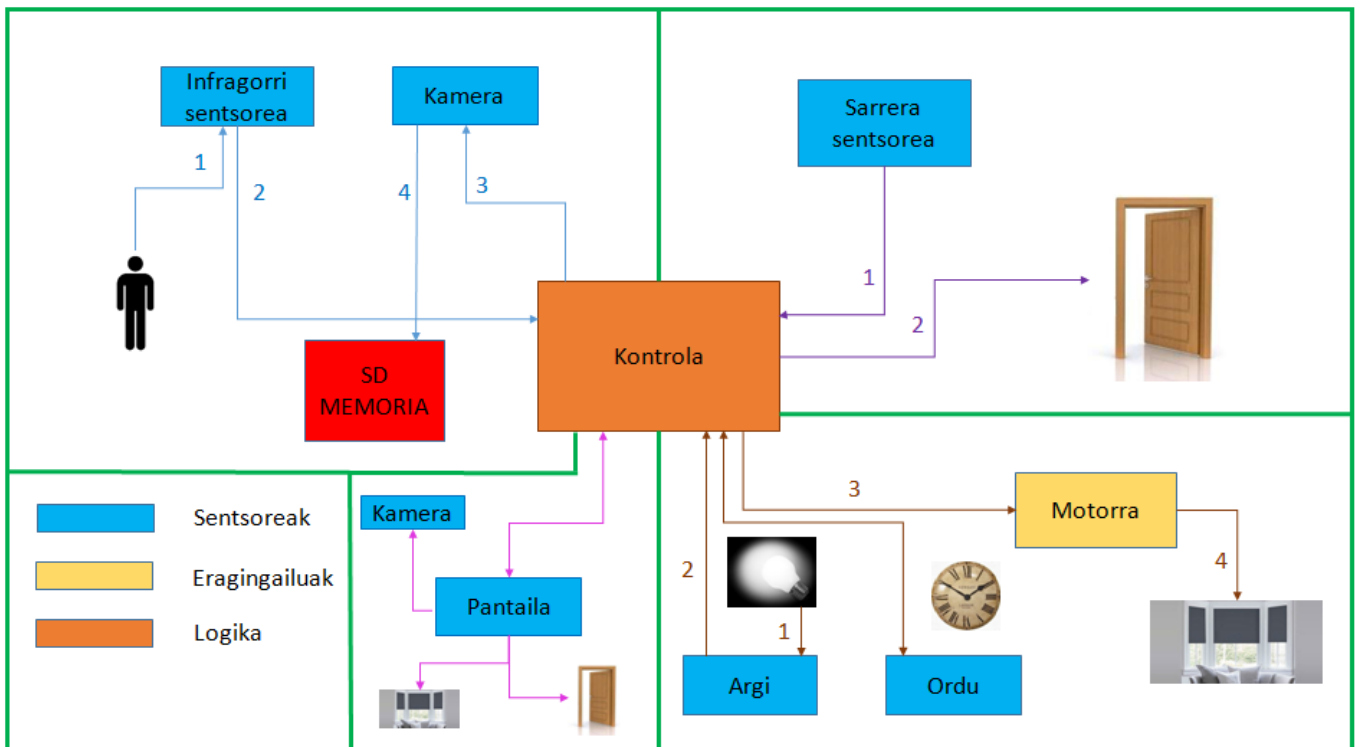
Proiektu honek automatizatuta dauden sistema guztiak kontrolatzeko ahalmena izango du. Honez gain, pantaila bat kontrolatuko du. Pantaila, etxea eskaintzen dituen sistema guztiez aparte beste egoera espezifiko batzuk ere kontrolatzeko gai izan beharko da.

Etxe domotikoa lau zatitan bananduta egongo da 3.1 irudian ikusten den bezala:

- **Sarrera:** Zati honetan etxera sartzeko modua kontrolatuko da. Horretarako sarrera kontrolatzen duen sentsore bat erabiliko da. Sentsorearen irteera balioduna bada, sarrerako atea ireki egin beharko da eta pantailan nor sartu den eta zein ordutan sartu denaren informazioa gordeko da SD txartelean. Sentsorearen irteera baliogabea bada, pantailan sarrera baliogabea dela adieraziko da. Hiru bider sarrera baliogabea denean, pantailan teklatu bat agertuko da pasahitz bat sartzeko. Pasahitza oker sartzen denean etxearen segurtasun protokoloa aktibatuko da. Aldiz, pasahitza zuzen sartzen bada, berriro hasierako egoerara itzuliko da sistema.
- **Logela:** Hemen infragorri sentsorea eta kamera kokaturik daude. Sarkin bat logelan sartzen denean, infragorri sentsoreak sarrera horren berri ematen dio mikro-kontrolagailuari etendura baten bitartez eta gertaera horrek kamerari argazkia ateratzeko beharrezko diren aginduak bidali egiten dizkio eta pantailak argazkia ateratzen ari deneko mezua adierazten du. Argazkia ateratzeak denbora bat irauten du. Denbora horretan zehar, mikroak ezin du ezer gehiago egin, kamera kudeatzen ari baita (argazkia jasotzen). Mikroak argazkia jasotzen amaitzen duenean, SD txartelean gordeko du.
- **Egongela:** Pertsianen kontrola eramango da. Pertsiana hiru egoeraren menpe igo edo jaitsi egingo da motor baten laguntzaz. Argiaren, orduaren edo pantailan dagoen botoiaren arabera. Urtaroaren arabera, pertsiana kontrolatzen duen ordutegia aldatuko da. Pertsiana jaitsita badago , sistemak ezin izango du berriro jaisteko agindua bidali. Berdin gertatuko da igota baldin badago eta igotzeko agindua jasoz gero.



- **Kontrol kontsola:** Azkenengo zatian pantaila bat egongo da eta bi funtzio beteko ditu: etxe osoaren kontrola eta beste zatien dispositiboen egoerak adieraziko ditu etxean gertatzen dena uneoro ikusi ahal izateko.



3.1 Irudia: Etxearen bloke diagrama 1



3.2. ALTERNATIBAK.

3.2.1. MIKROKONTROLADOREAK

3.2.1.1. ARDUINO

Arduino, erabilerrazak eta malguak diren hardware eta softwaretan oinarritutako prototipoak sortzeko elektronika irekiko plataforma da. Erabiltzaile kopuru handiago bati elektronika eskuragarriago egiteko, bai maila ekonomikoan eta bai konplexutasun mailan, Arduino jaio zen. Alde batetik, hezkuntzarako pentsatu zen, ikasle guztiek materiala eskuratzeko aukera izan zezaten. Bestetik, elektronika munduan hasi nahi zuen jendearentzat inolako ezagutzarik gabe. Arduino bi printzipiotan oinarritzen da:

- Wiring-en oinarritutako Arduino Programming Language. Mikrokontroladoreetarako kode irekiko programazio esparrua da, software multiplataforma idazten ahalbidetzen duena konektatutako dispositiboak, objektu interaktiboak, eremuak edo esperientzia fisikoak kontrolatzeko. Ideia kode-lerro batzuk idaztea izango da eta osagai elektronikoak konektatzea.
- Processing-en oinarritutako garapen eremua, Arduino Development Environment. Irudiak, animazioak eta elkarrekintzak sortu nahi dituzten pertsonentzako kode irekiko programazio lengoia da. Software zirriborro koaderno gisa erabiltzeko garatuta dago eta ikusmen testuinguru batean ordenagailu programazioa irakasten du.

Aipatu bezala, Arduino hardware libreko ingurumenean oinarritzen da. Honi esker edozein pertsonak proiektu bat sortzeko aukera izango du osagaiak erosiz edo zuzenean plaka erosiz, inolako lizentziarik gabe. Mikroprozesadorea programatzeko beharrezko softwarea dohainik deskargatu daiteke Arduinoren web orrialdean.

Arduino mikro batez eta sarrera/irteera ataka jakin batzuek, analogiko zein digitalak, horniturik dagoen plaka bat da. Horri guztiari esker, aldakortasuna eta kostu baxua bertute bat izatea ahalbidetzen da.

Arduinok beste abantaila bat ere aurkezten digu: multiplataforma izatea. Honi esker, programa anitzekin komunikatzeko gai da. Garapen eremua ondorengo sistema operatiboetan deskargatu daiteke: Windows, Mac eta Linux. Android dispositiboekin komunikatu daiteke.

Arduino plaka ezberdinak daude:

- Arduino UNO: Arduino UNO Atmega328 mikroprozesadorean oinarritutako plaka elektronikoa da. 14 sarrera/irteera digital ditu (horietako 6 PWM irteera bezala erabili daitezke), 6 sarrera analogiko, 16MHz-ko durundatzaile zeramikoa, USB konexioa, elikadura konektorea, ICSP goiburua eta berrasieratze botoia. Mikrokontrolagailua funtzionamenduan jartzeko nahikoa da USB kablearen bidez



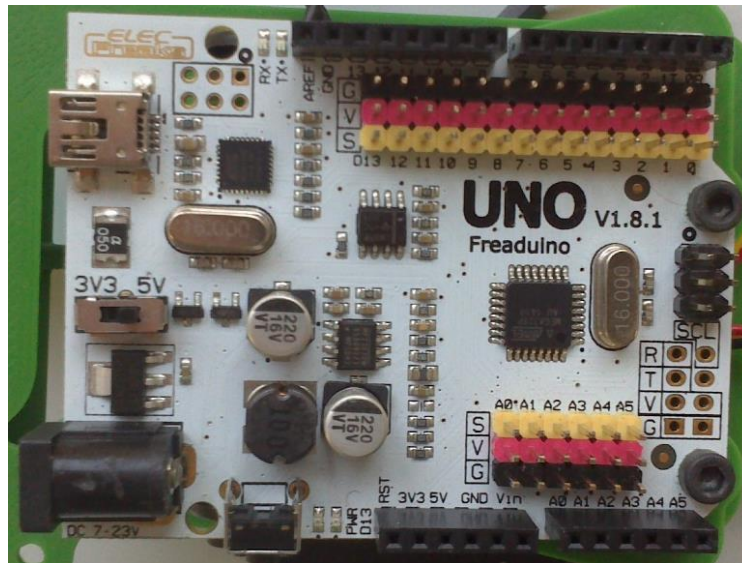
ordenagailu batera konektatzea edo AC-DC egokitzaile edo bateria baten bidez elikatzea.



1.2 Irudia: Arduino UNO.

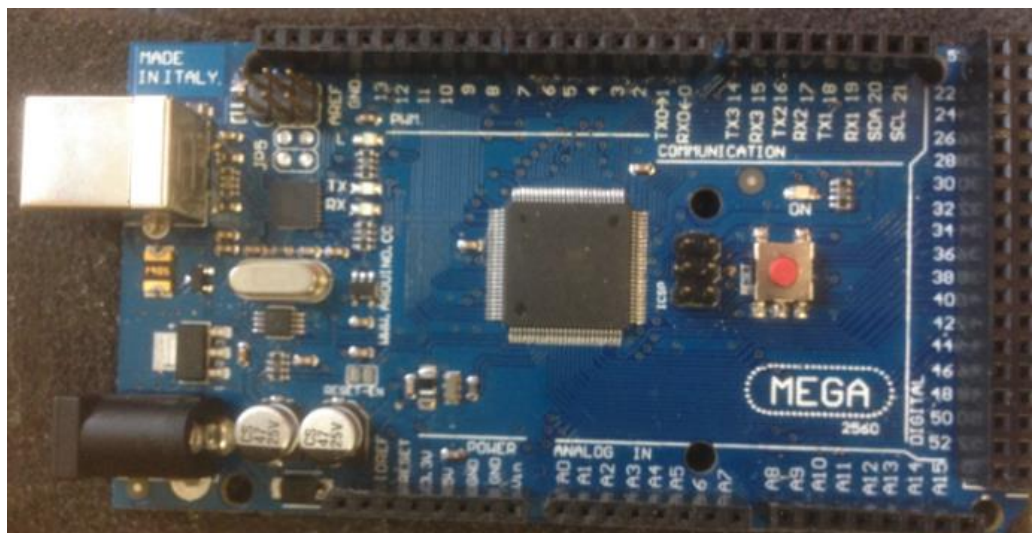
- Freaduino UNO: Arduino UNO bezala Atmega328 mikroprozesadorean oinarritutako plaka elektronikoa da. 14 sarrera/irteera digital ditu (horietako 6 PWM irteera bezala erabili daitezke), 6 sarrera analogiko, 16MHz-ko durundatzaile zeramikoa, USB konexioa, elikadura konektorea, ICSP goiburua eta berrasieratze botoia. Mikrokontrolagailua funtzionamendua jartzeko nahikoa da USB kablearen bidez ordenagailu batera konektatzea edo AC-DC egokitzaile edo bateria baten bidez elikatzea.

Freaduino UNO Arduino IDE-a eta babesak 100%-ekin bateragarri da. Plaka honen abantaila garrantzitsuenak kanpoko konexioak egiteko dituen headers dira, +Vcc eta GND gehitzen dituztenak. Headers hauek zein pin-ei dagokien adierazita daude eta oso lagungarriak dira serbomotor eta sentsoreak konektatzeko. Beste abantaila bat bere tentsio aukeraketa da. Arduino txartel guztiek 3,3v-eko irteerarekin lan egiten dute, baina mikrokontroladoreak beti 5Vcc-ra lan egiten du. Freaduino UNO-k tentsio aukeraketa duenez, mikrokontroladorea 3,3Vcc-ra lan egitea alda dezake. Hau erabilgarria izan daiteke erabili nahi diren dispositiboak tentsio honetan lan egiten badute.



2.3 Irudia: Freduino UNO.

- Arduino MEGA: Arduino MEGA ATmega2560 mikroprozesadorean oinarritutako plaka elektronikoa da. Host interface USB konexioa dauka Android txip MAX3421e dispositiboetan konektatzeko. 54 pin I/O (14 PWM-rekin), 16 sarrera analogikoak, 4 portu UARTs (hardware bidez), 16 MHz-ko kristal bat, USB konektore bat, Jack elikadura konektore bat, ICSP-rentzako pinak eta berrabiarazi botoi bat. Atmega8U2 txip batean oinarritutako bihurtutako Série/USB integratuta dauka.



3.3 Irudia: Arduino Mega.



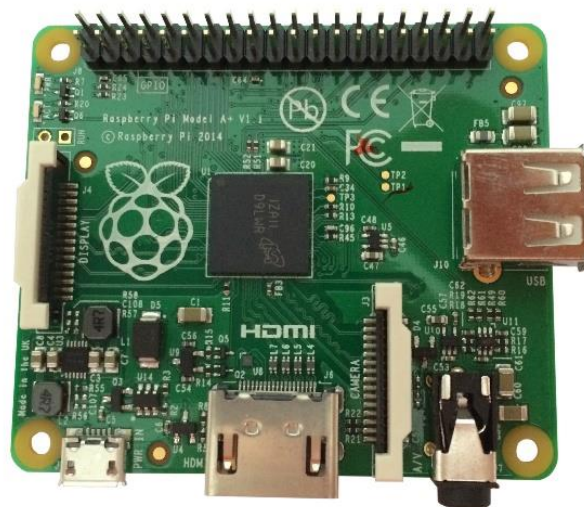
Tamaina txikiagoa duten plakak eremu edo toki txikiagotan sartzeko egokiagoak dira; baina, gure kasuan lekua sobera dago eta ez ditugu erabiliko.

3.2.2.2. RASBERRY PI

Raspberry pi fundazioak Erresuma Batuan garatutako kostu baxuko plaka konputagailua (SBC) da. Helburu nagusia konputagailuen zientzia ikastetxeetan irakastea da.

Diseinuak System-on-a-chip Broadcom BCM2835 du, honek ARM1176JZF-S prozesadore zentrala (CPU) du 700 MHz-etan (firmwareak “Turbo” modua du eta erabiltzaileak 1GHz-eko overlock-a egin dezake ziurtasuna galdu gabe), Video Core IV prozesadore grafikoa (GPU) eta 512 MB-eko RAM memoria nahiz eta merkaturatzean 256 MB ziren. Diseinuak ez du disko gogorrik ezta egoera solidoko unitaterik, SD txartel bat erabiltzen du biltegitate iraunkorrerako. Ez du elikadura iturririk ezta karkasarik.

- A+: Raspberry Pi A+ plaka ordenagailu txiki bat integratua eduki dezake: prozesadore Broadcom BCM2835 SoC Full HD, GPU Co-prozesadore multimedia Dual Core VideoCore IV, 256MB-ko SDRAM eta 700MHz-ko memoria RAM, mikroSD baten bitartez biltegitatzen da, USB portu bat dauka eta HDMI irteera, 40 pin GPIO ditu, eta bere energia kontsumoa 600 mA-koa da.

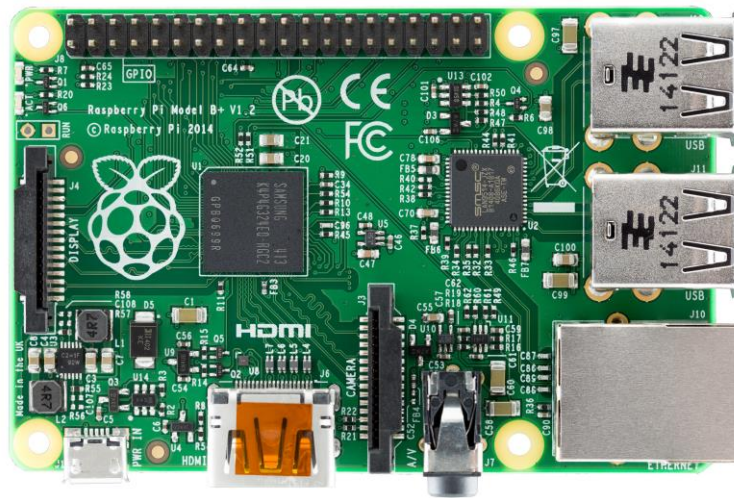


3.5 Irudia: Raspberry Pi A+.



Aterata: https://upload.wikimedia.org/wikipedia/commons/3/30/Raspberry_Pi_A%2B.jpg

- B+: Raspberry Pi B+ plaka Broadcom BCM2835 (CPU, GPU y SDRAM) prozesadorea dauka, CPU ARM1176JZF-S a 700 MHz-ekoa, GPU Broadcom VideoCore IV, 512 MB-ko memoria RAM (GPU-arekin partekatuta), 4 konexio USB 2.0, 3.5 mm-ko audio mini Jack irteera, audio/vídeo HDMI irteera bat, USB mikro bat, LAN 10/100 sarera konekta daiteke eta WiFi-ra, mikro SD txartelak sartzeko lekua.



3.6 Irudia: Raspberry Pi B+.

Aterata: https://upload.wikimedia.org/wikipedia/commons/6/6f/Raspberry_Pi_B%2B_top.jpg

- Raspberry Pi 2 B: Raspberry Pi 2 B ordenagailu plaka Raspberry Pi Foundation-en azkenengo aurrerapena da.

ARM A7 prozesadoreari esker, GNU/LINUX distribuzioak erabil daitezke, Windows 10 barne. Gainera Modelo B+ren formatu berdina dauka. Bere prozesadorea Quad-Core Cortex A7 a 900MHz da, 1GB-ko RAM memoriarekin, 4 portu USB 2.0 ditu, 40 pin GPIO, HDMI irteera, Ethernet portu bat, audio/mic konbo bat, interfaze kamera (CSI), interfaze pantaila (DSI), MIKROSD txartela eta 3D grafikoko nukleoa.



3.7 Irudia: Raspberry Pi 2B.

Aterata:

[https://upload.wikimedia.org/wikipedia/commons/3/31/Raspberry_Pi_2_Model_B_v1.1_top_new_\(bg_cut_out\).jp](https://upload.wikimedia.org/wikipedia/commons/3/31/Raspberry_Pi_2_Model_B_v1.1_top_new_(bg_cut_out).jp)

3.2.3.3. INTEL EDISON

Intel Edison-en erdian 22 nm-tako SoC Intel Atom bat dago, bi nukleo dituen 500MHz eko CPU-a eta 100MHz-eko MCU bat. Honek produktuaren datuak berriro gorde eta prozesatzeko ahalmena dauka, MCU-arekin kontsumo baxuan eta datuak CPU-ra bidaltzen bere azterketarako. Edison 1GB-ko memoria, 4GB-ko biltegia, WIFI eta Bluetooth 4.0 komunikazioetarako. 40 GPIO onartzen ditu konfigurazio aukera askorekin.

Plataforma berri bat da, ez dauka erabiltzaile euskarri handirik ezta sistema periferiko handirik ere.



3.8 Irudia: Intel Edison.

Aterata: https://c2.staticflickr.com/8/7493/15314330503_3715218321.jpg



3.2.2. SERGURTASUN SENTSOAREAK

Segurtasunaren arloan dispositibo asko daude. Dispositiboak funtzio desberdinak betetzeko eginak daude. Lekuaren arabera segurtasun dispositibo bat egokiagoa izango da beste bat baino.

3.2.2.1. ULTRASOINU SENTSOAREAK

Sentsore hauek 8 metrotara dauden objektuak detektatzeko gai dira ultrasoinuen bitartez. Sentsoreak ultrasoinu inpultsuak emititzen ditu. Hauek objektu batean islatzen dira, sentsoreak eragindako oihartzuna jasotzen du eta seinale elektrikoetan transformatzen ditu sentsorea prozesatzeko. Airean bakarrik lan egin dezakete eta edozein itxura, eremu edo material ezberdinen objektuak detekta ditzake.

- **HC SR04:** Kostu baxuko ultrasoinu sentsore bat da, non objektua detektatu eta zein distantziatara dagoen kalkula dezake.



3.9 Irudia: HC SR04

Aterata:

https://upload.wikimedia.org/wikipedia/commons/2/20/HC_SR04_Ultrasonic_sensor_1480322_3_4_HDR_Enhancer.jpg

3.2.2.2. INFRAGORRI SENTSOAREAK

Seinale infragorriak detektatzeko gai dira. Segurtasun sistemetan oso erabiliak dira alarma moduan lan egiteko.



- **PIR:** bere ikusmen eremuan kokatutako objektuetako argi infragorri erradiatua neurtzen duen sentsore elektronikoa da. PIR-a sentsoreek ez dute "beroa" antzematen edo neurtzen objektu batek igorritako erradiazio infragorria baizik, desberdina dena, baina sarritan objektuaren tenperaturarekin elkartuta dagoena.
-

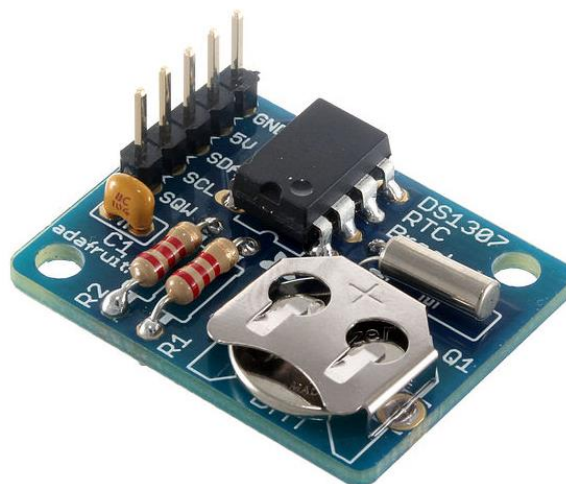


3.10 Irudia: PIR sentsorea

3.2.3. ERLOJU DIGITALAK

Denbora errealean funtzionatzen duen erlojua da uneoro ordua kontatzen duena. Sarritan RTC-ek ordezeko elikadura iturri bat dute; beraz denbora neurtzen jarrai dezakete elikadura nagusiko iturria itzalita dagoen bitartean.

- **DS1307:** Erloju digitala da. I2C komunikazioa erabiltzen du. Prezio handiko erlojua digitala da. I2C komunikazio busa erabiltzen du.



3.11 Irudia: DS1307



- **RTC Tiny:** DS1307 erloju digitalren oso antzekoa da. Ezberdintasun bakarra kostuan eta prezisioan dago. Honek kostu era prezisio txikiagoa dauka.



3.12 Irudia: RTC TINY

3.3. AUKERAKETA ETA IRIZPIDEAK

Ondorengo atalean proiektua eraikitzeke erabiliko diren dispositiboaren parametroak eta aukerak aztertuko dira, gure proiekturako egokiena erabili ahal izateko.

Irisgarritasuna

Produktura izan dezakegun irisgarritasuna kontuan izan behar da. Zenbat eta irisgarriagoa izan, orduan eta errazago eta azkarrago eskuratuko dugu nahi dugun produktuan. Azken hau oso garrantzitsua da proiektu bat gauzatzeko orduan denbora murrizteko.

Kostua

Kostu faktorea kontuan izan behar da, aukeraketa egokiak aurrezki ekonomiko garrantzitsua ekar dezakelako.

Malgutasuna

Diseinuan aldaketak egitea baimendu behar du sistemak. Hau da, aplikazioak behar izanez gero, handitze moduluak erabiliko dira aplikazio berdinetarako. Dispositiboa ordezkatzeko aukera ere egon behar da. Honek lan aurrezpen handia ekarriko du.

Abiadura

Atal honek plakaren prozesamendu abiadurari egiten dio erreferentzia. Honek gure aplikazioa arinagoa edo motelagoa izatera behartuko du.



Kontsumoa

Frogak egiteko erabiliko diren plaken kontsumoa kontuan izan beharreko puntu garrantzitsua da. Plaka hauek pilekin edo sare elektrikora konektatutako elikatzaile batekin lan egiten dute eta aurrekontua garestitu dezakete.

Potentzia

Kalkulu ahalmena edo potentzia oso garrantzitsua da. Izan ere, denbora tarte berdinean eragiketa gehiago edo instrukzio konplexuagoak gauzatuko direlako.

Funtzionalitatea

Proiektuan zehaztutako helburu bakoitza era egokian garatzeko izan behar dira aukeratuak elementuak. Etorkizunean garatuko diren aplikazioetarako eraginkorra izatea ere positiboa da.

Sentikortasuna

Sentikortasun maila, hau da, prozesuan zehar agertzen diren arazoen aurrean emango duen erantzuna, adibidez: interferentziekiko oso sentikorra izatea, materialen hauskortasuna dela eta apurtzea eta abar. Honek kostu gehigarriak ekar ditzake.

Ezaguera

Garapen eta diseinurako faktore oso garrantzitsua, aurretiaz ikasitako edo lan egindako guztia izango da. Abantaila handia izango da ezagutza jakinak izatea kostuak aurrezteko. Komunikazioak Dispositiboak PC-arekin edo beste periferiko batzuekin komunikatzeko duen ahalmena komunikazio protokoloak jarraituz.

Irisgarritasuna = IRI

Kostua = KOS

Malgutasuna = MAL

Abiadura = ABI

Kontsumoa = KON

Potentzia = POT

Funtzionalitatea = FUN

Sentikortasuna = SEN

Ezaguera = EZA

Komunikazioak = KOM

Maneiagarritasuna = MAN

I/O erantzuna = ERA

Pultsu zabaleraren araberrako modulazioa = PWM



3.5. AUKERAKETAREN TABLA

- **Mikrokontroladorea:**

1. Tabla: Kontroladorearen aukeraketa

	IRI	MAL	POT	KOS	FUN	SEN	EZA	ABI	TOTAL
Pisua (%)	20	10	5	20	10	5	20	10	100
Arduino MEGA 2560	10	8	9	8	6	6	8	8	820
Freaduino UNO	8	6	6	6	8	8	8	8	680
Arduino UNO	8	6	6	6	7	7	8	8	650
Raspberry Pi A+	7	8	5	7	7	6	3	3	635
Raspberry Pi B+	7	8	5	7	7	7	3	3	645
Raspberry Pi 2B	7	7	4	6	8	8	3	3	605
Intel Edisson	7	5	4	6	9	9	3	3	575

- **Detekzio sentsorea:**

2. Tabla: Kontroladorearen aukeraketa

	KOS	FUN	POT	SEN	TOTAL
Pisua (%)	30	20	15	35	100
Ultrasoinu sentsorea	7	8	8	9	720
PIR sentsorea	8	10	8	8	880

- **Erloju sentsorea:**

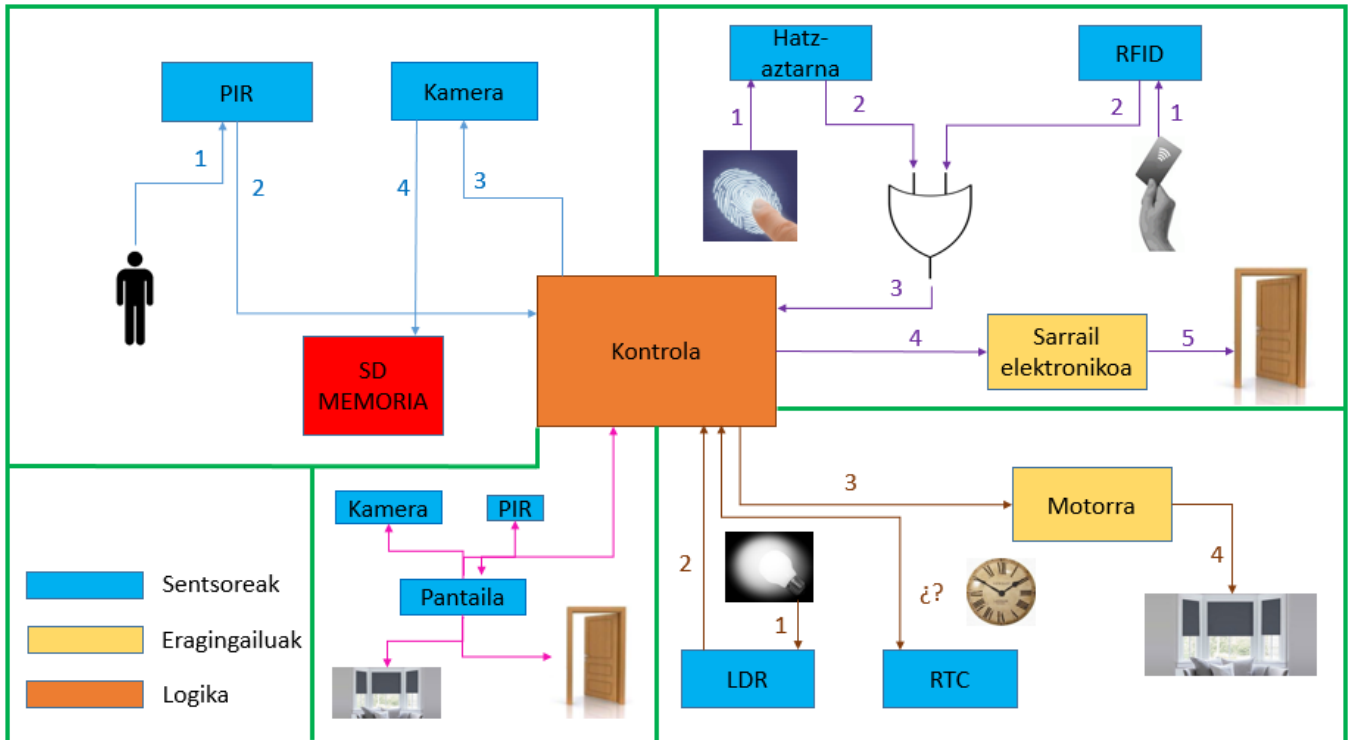
3. Tabla: Kontroladorearen aukeraketa.

	KOS	FUN	TOTAL
Pisua (%)	50	50	100
DS1307	4	9	680
RTC TINY	8	8	800

3.6. DESKRIBAPEN ZEHATZA

Proiektu honek automatizatuta dauden sistema guztiak kontrolatzeko ahalmena izango du. Gainera, TFT pantaila batek kontrolatuko du. TFT pantaila, etxeak eskaintzen dituen sistema guztiez aparte beste egoera espezifiko batzuk ere kontrolatzeko gai izan beharko da.

Etxe domotikoa lau zatitan bananduta egongo da 3.9 irudian ikusten den bezala:

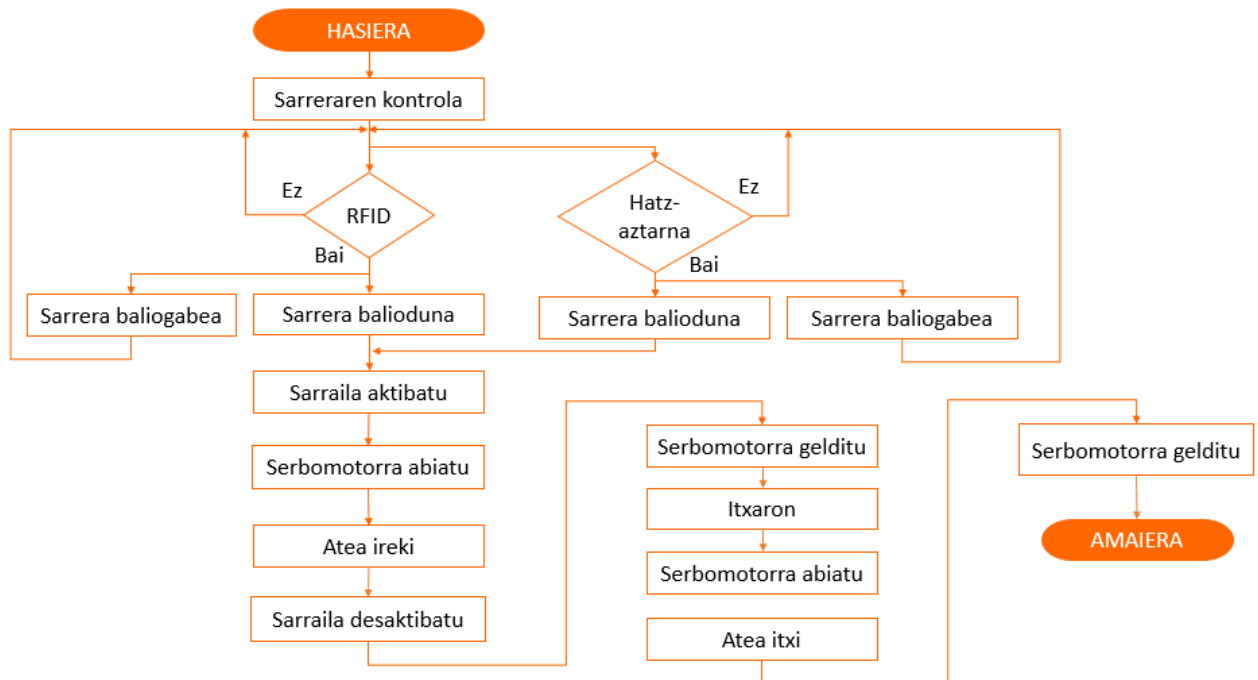


3.9 Irudia: Etxearen bloke diagrama 2

- Sarrera:** Zati honetan etxera sartzeko modua kontrolatuko da 3.10 irudian ikusiko den bezala. Horretarako sarrera kontrolatzen duen RFID sentsorea eta Hatz-detektagailu sentsorea erabiliko dira. Hiru egoera eman al dira:
 - Sentsorearen irteera balioduna bada, sarrerako atean Arduinok zenbait agindu bidaliko ditu. Sarraila aktibatzeke agindua bidaliko du sarrailaren kisketa sartuz. Kisketa sartuta dagoenean serbomotorra aktibatuko du dagokion posiziora heldu arte eta heltzen denean sarraila desaktibatu egingo da kisketa ateratzen. Hau guztia gertatzen den bitartean TFT pantailan nor sartu den eta RTC batekin zein ordutan sartu denaren informazioa gordeko da SD txartelean.
 - Sentsorearen irteera baliogabea bada, TFT pantailan sarrera baliogabea dela adieraziko da.

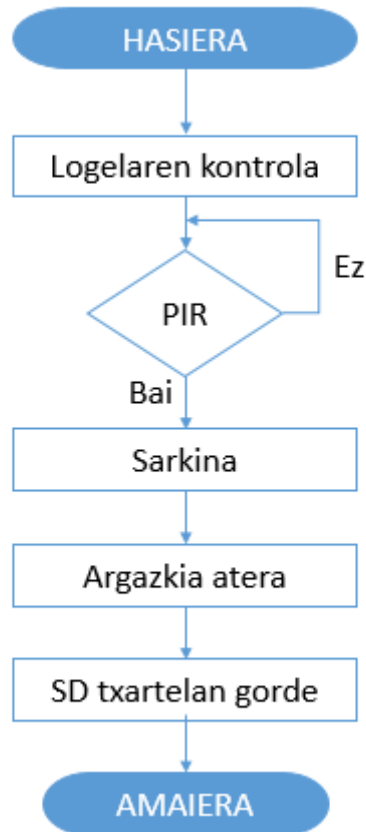


3. Hiru bider sarrera baliogabea denean, TFT pantailan teklatu bat adieraziko da pasahitz bat sartzeko. Pasahitza okerra sartzen denean etxearen segurtasun protokoloa aktibatuko da, hau da, TFT pantailan alarma egoera adieraziko da. Aldiz, pasahitza egokia sartzen bada, berriro hasierako egoerara itzuliko da sistema.



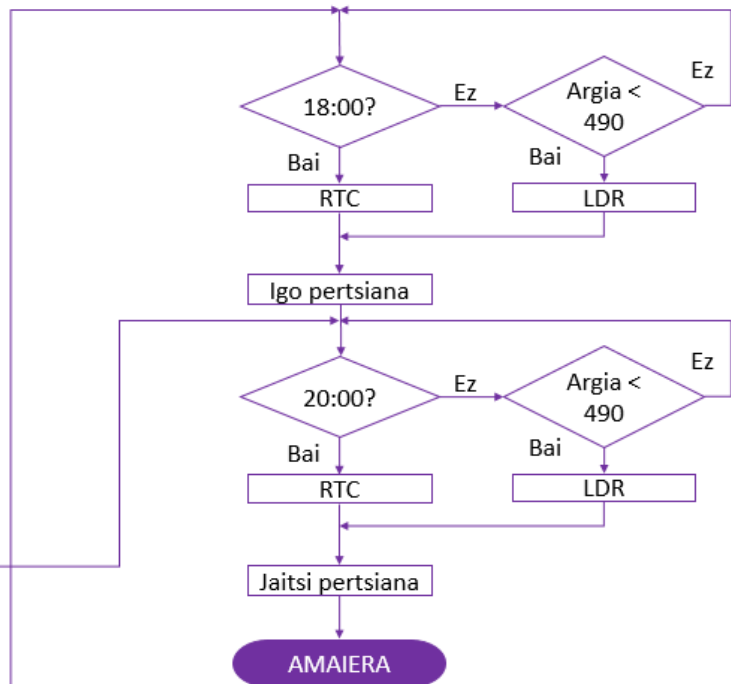
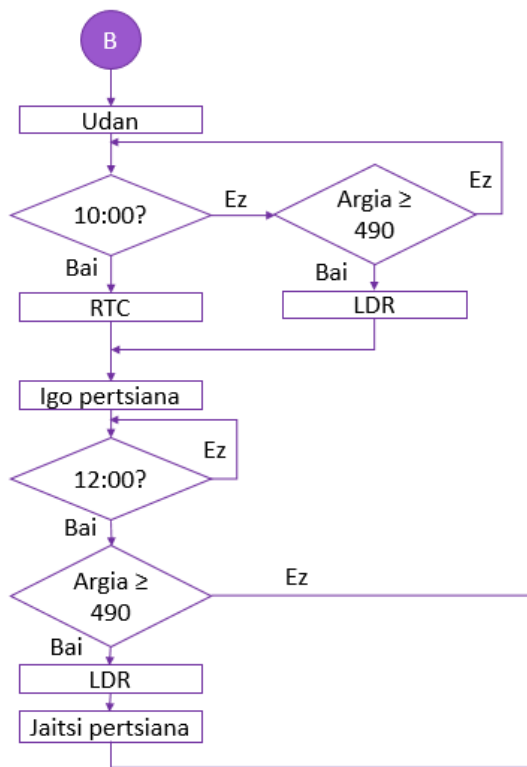
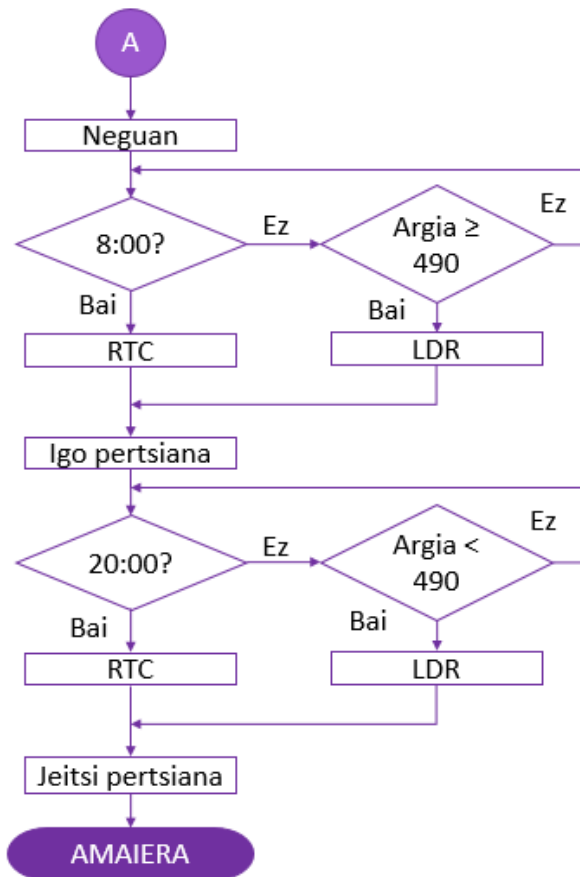
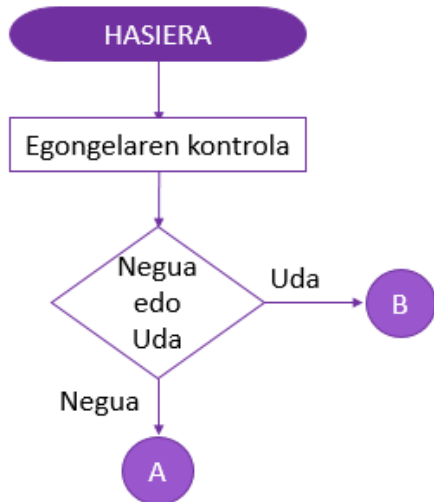
3.10 Irudia: Sarrera kontrola

- **Logela:** Hemen PIR sentsorea eta LinkSprite kamera kokaturik daude 3.11 irudian ikusiko den bezala. Bata bestearen diagonalean kokatuta. Sarkin bat logelan sartzen denean, PIR sentsoreak sarrera horren berri ematen dio Arduinori etendura baten bitartez eta gertaera honek LinkSprite kamerari argazkia ateratzeko beharrezko diren aginduak bidali egiten dizkio eta TFT pantailan argazkia ateratzen ari deneko mezua adierazten du. Argazkia ateratzeak denbora bat irauten du. Denbora horretan zehar, Arduinok ezin du ezer gehiago egin, LinkSprite kamera kudeatzen ari baita (argazkia jasotzen). Arduinoak argazkia jasotzen amaitzen duenean, SD txartelean gordeko du.



3.11 Irudia: Logela kontrola

- **Egongela:** Azkeneko zatian pertsianen kontrola eramango da. Pertsiana hiru egoeraren menpe igo edo jaitsiko da motor baten laguntzaz. Argiaren (LDR), orduaren (RTC) edo TFT pantailan dauden konfigurazio botoiaren arabera. Urtaoaren arabera pertsiana kontrolatzen duen ordutegia aldatuko da. Pertsiana jaitsita badago , sistema ezin izango du berriro jaisteko agindua bidali. Berdin gertatuko da igota baldin badago eta igotzeko agindua jasotzen badu.

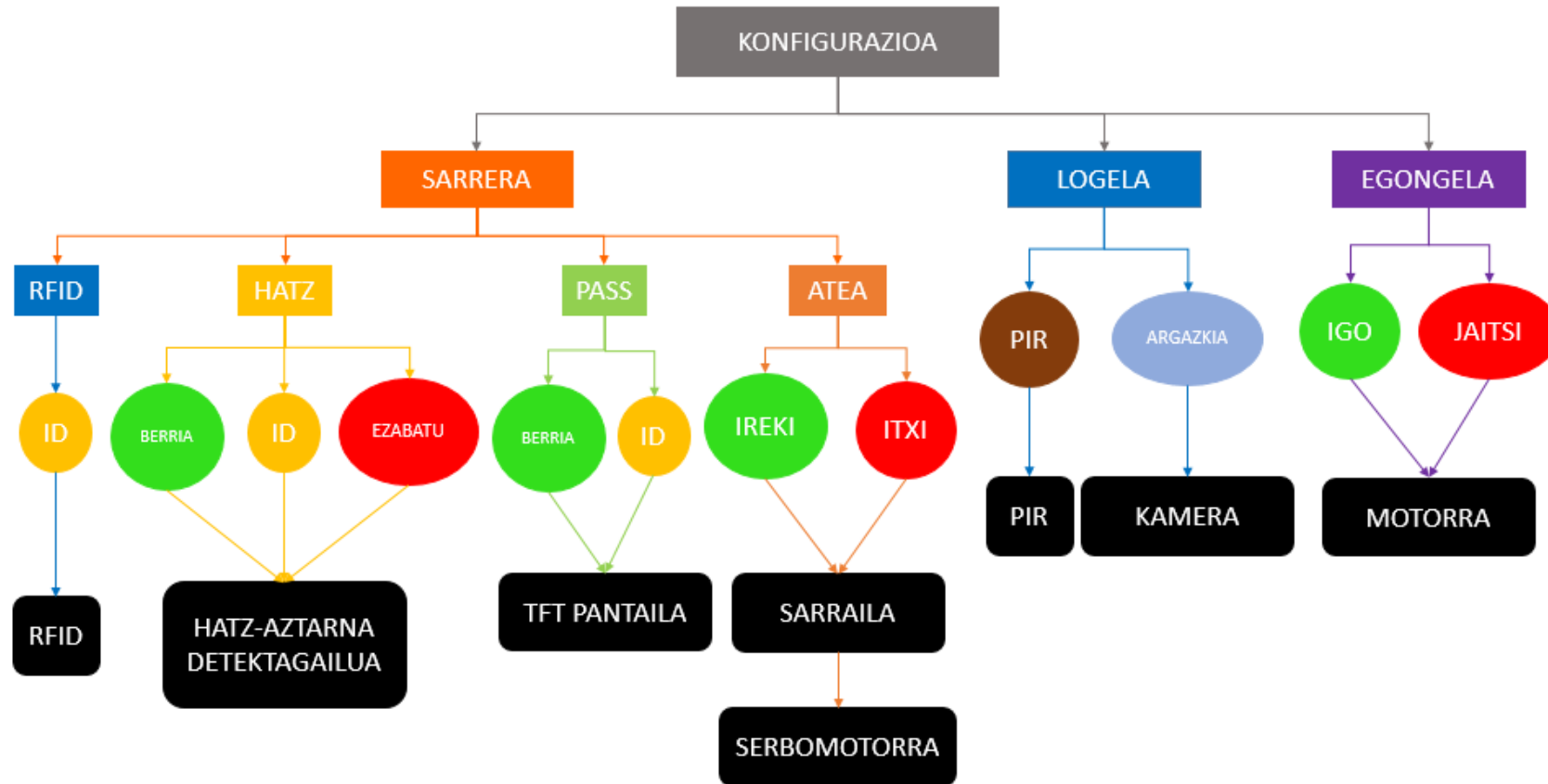


3.12 Irudia: Egongela kontrola



MIKROKONTROLADORE BATEAN OINARRITUTAKO ETXE DOMOTIKOA

- TFT pantaila etxe domotikoaren kontrol kotsola izango da. Etxearen egoera guztiak adieraziko dira pantailaren bitartez. Dispositibo gehienak manipulatzeko ahalmena dauka. Pantailaren konfigurazioa eta manipulatutako dispositiboak 3.13 irudian ikusiko dira:



3.13 Irudia: TFT pantailaren konfigurazioa



4. PROIEKTUAREN DISEINUA

4.1. SARRERA

Proiektu honetan burutuko den Arduino plaka erabiltzerakoan arazo desberdinak aztertuko dira. Proiektu honetan zehazki Arduino MEGA 2560 plaka erabiliko da. Horretarako adibide praktiko bat garatuko da, non proiektuaren garapenean agertzen diren eragozpenak eta traba guztiak gainditu diren. Adibide praktiko hau etxe domotiko baten maketa izango da. Etxe honek, funtzionaltasun desberdineko dispositibo elektroniko ugari dauzka. Hauek guztiak beharrezkoak izango dira helburua betetzeko eta eskuratutako ezaguerak erakusten dituen etxea garatzeko.

Erabiliko den elementu nagusia Arduino MEGA 2560 plaka izango da. Bertan sortutako programa grabatuko da eta honek berari konektatuta dauden gainontzeko dispositiboetara pasako die. Proiektua lau zati garrantzitsuetan zatituta dago. Sarrera, logela, egongela eta kontrol kotsola.

Sarreran, atearen kanpoko aldean bi identifikazio sentzore daude. Etxera sartzeko beharrezko identifikazioa eraman beharko da. Sarrerako identifikazio prozesua hiru bider okerra bada, pantailan pasahitz bat sartzeko aukera emango da. Atea ireki edo ixteko sarraila eta serbomotor baten bitartez kontrolatuko da. Dispositibo guzti hauek Arduino MEGA 2560 plaka batean konektatuta eta kontrolatuta egongo dira.



4.1 Irudia: Sarrera barnealdea



Logelan, bi sentsore erabiliko dira logelaren segurtasuna bermatzeko. Segurtasun sentsorea logelako izkina baten goialdean kokatuta egongo da eta kamera segurtasun sentsorearen kontrako partean jarriko da. Logelara sartzeko atea eskuz ireki egingo da. Bi dispositibo hauek Arduino MEGA 2560 plaka batean konektatuta eta kontrolatuta egongo dira.



4.2 Irudia: Logela

Egongelan, lau dispositibo erabiliko dira pertsianaren kontrola ahalbidetzeko. Argi sentsorea pertsiana kokatuta dagoen paretaren kanpoaldean kokatuta egongo da, hau da, kalera begira. Erloju sentsorea etxearen konexioen lekuan egongo da pantailarekin batera. Azkenik, motorra pertsianaren giak eta tanborrarekin konektatuko da pertsiana igo edo jaitsi dadin. Dispositibo guzti hauek Arduino MEGA 2560 plaka batean konektatuta eta kontrolatuta egongo dira.



4.3 Irudia: Egongela barnealdea



Kontrol kotsolan, TFT pantaila Arduino MEGA-ra konektatuta egongo da dorre bat izango balitz bezala. Hemen pantailaren lan guztia ikusi ahal izando da. Alde batetik, egoerak adieraziko ditu eta bestetik, kontrol kotsola bat izango da etxeko dispositibo gehienak kontrolatu ahal izango dituena.



4.4 Irudia: Kontrol kotsola

4.2. MUNTAIA

4.2.1. HASIERAKO MAKETA

Hasteko etxe baten maketa egitea planteatu da. Behin, etxeak izango duen egitura zehaztuta egonda, maketaren eraikuntzan oinarrituko gara. Etxea lehen aipatu bezala hiru ataletan bananduko da. Hiru atalak beharrezko konexioak konektatzeko leku bat egin behar da etxetik kanpo.

Maketa egurrezko xaflekin eraikiko da. Horretarako lantegi apropos bat behar da 4.4 Irudian ikusten den bezala. Lantegi horretan, egurrezko xafiak mozteko eta etxea eraikitzeke behar den materiala dago.



4.5 Irudia: Lantegia

Hasierako maketan etxearen prototipo bat eraikitzeko. Horretarako etxearen diseinu bat egingo da. 4.5 Irudian ikusten den bezala, etxearen oinarrian diseinuaren plano dago.



4.6 Irudia: Maketaren oinarria planoarekin

Behin plano eginda oinarrian, etxearen hormak egin eta lotu behar dira. 4.6 Irudian sarrerako horma eta egongelako horma kokatzea falta da oraindik pareta horietan aldaketa batzuk egin beharko direlako.



4.7 Irudia: Hasierako maketa

Maketa eraiki ostean, sentsoreen posizioa zehaztu beharra dago. Hemen, arazo batekin topatu ginen. Logelako segurtasun sentsorearekin eta egongelako pertsianaren diseinuarekin. Segurtasun sentsorea pareta hain baxua zenez, bere detekzio eremua etxe osoa detektatzen zuen eta hori ez da bere helburua. Pertsianaren diseinuan, berriz, pertsianaren tamaina oso txikia geratuko zen.

Bi arazo hauen konponbidea pareten altuera handitzea da. Horretarako maketa aldatu behar da.

4.2.2. MAKETA FINALA

Hasierako maketa dituen arazoaren konponbidea lortuko da. Hori lortzeko pareten altuera bikoitza izango da hurrengo irudian ikusten den bezala.



4.8 Irudia: Maketaren konponbidea



Behin hasierako maketaren arazoak konponduta, berriro sentsoreen posizioak zehaztuko dira maketa finala ondo dagoela konprobatzeko.

Hurrengo pausua etxearen atalak dituen zatiak eraiki edo dagozkion aldaketak egitea da.

Sarreraren ate bat egin behar da. Atea ireki eta ixteko sarraila eta serbomotor bat erabiliko da eta sarrerako paretaren barnealdean kokatuta egongo dira 4.8 irudian ikusten den bezala.



4.9 Irudia: Sarreraren atala barnealdean

Sarreraren kanpoaldean RFID eta hatz-aztarna detektagailua kokatuta egongo da 4.9 irudian ikusten den bezala.



4.10 Irudia: Sarreraren kanpoaldea



Logelan, logelara sartzeko ata bat egingo da. Atea eskuz ireki egingo da eta bi dispositibo egongo dira. PIR dispositiboa atearen paretako izkinaren goialdean kokatuko da eta Kamera PIR sentsoarearen kontrako aldean kokatuta egongo da.



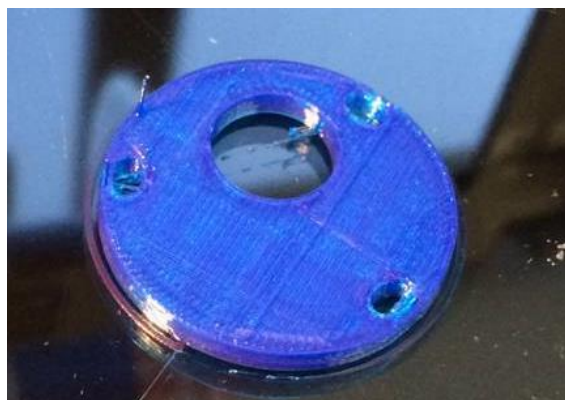
4.11 Irudia: Logela

Egongela hiru ataletatik zatirik zailena da, eraikitzerako orduan. Atal honetan, pertsiana bat egin behar da. 3D inprimagailu batekin egingo dira pertsiana egiteko behar diren piezak egiteko. Lehendabizi pertsianaren pieza guztien diseinua egin behar da. Diseinatutako piezak: laminak, loturak, tanborra eta gidak. Kobre harila erabiliko da laminak eta loturak lotzeko. 4.11 irudian pertsiana egiteko piezak eta material guztia ikusten dira pertsianaren gidak izan ezik.



4.12 Irudia: Pertsianen piezak eta materiala

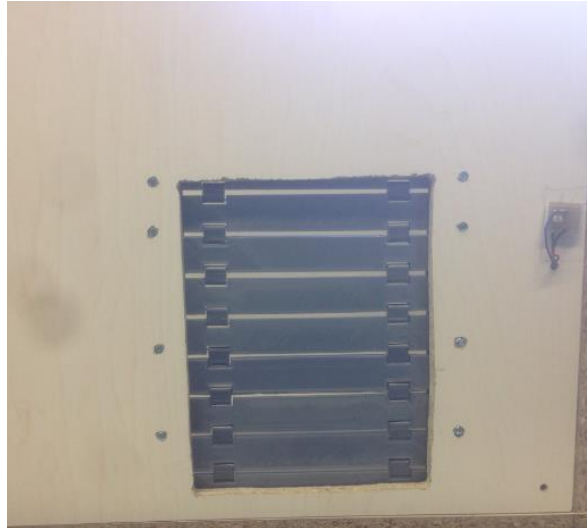
Pieza eta material guztia edukita pertsiana egongelan kokatu behar da. Behin pertsiana horman kokatuta eta pertsianarentzako leiho bat egingo da pertsiana etxe kanpotik ikusteko, motorra pertsianaren gida batera eta tanborrera lotu behar dugu. Horretarako beste lotura pieza bat egin behar da 3D inprimagailuarekin. 4.13 irudian ikusiko da lotura pieza.



4.13 Irudia: Motorra lotura



Pertsianaren osagai guztiak ondo kokatuta daudenean, LDR-aren kokapena zehaztu behar da. LDR-a pertsianaren paretaren kanpoaldean kokatuta egongo da eta horrela atal hau bukatutzat emango da.



4.14 Irudia: Egongelaren kanpoaldea

Bukatzeko dispositibo guztiak elikatzeko zirkuitu integratu bat garatu da. Zirkuitu integratua 3 irteera tentsio edukiko ditu dispositibo guztiak dagokion tentsioarekin elikatzeko. Arduino etxeak duen konexio eremuan kokatuta egongo da kontrol pantailarekin batera.



4.15 Irudia: Konexio eremua



Atal guztiak bukatuta daudenean, elkartu eta dispositibo bakoitzaren konexioak egin behar dira. 4.16 irudian maketa bere osotasunean ikusiko da.



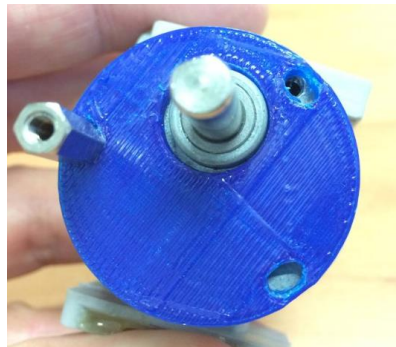
4.16 Irudia: Maketa osoa



4.2.1. MUNTAIAN IZANDAKO ARAZOAK

Muntaia egiterako orduan, muntaia guztien moduan zenbait arazo agerrarazi dira:

- Aurreko atalean aipatu den bezala, hasierako maketan paretaren tamainarekin arazoak izan ditugu.
- 3D inprimagailua inprimatutako piezekin. Inprimatutako pieza ez zegoen bat, lortu nahi zenaren piezarekin. Motorraren lotura piezan, gidetan, laminen tamainan... Hurrengo irudian arazoaren adibide bat ikusiko da.



4.17 Irudia: Motor lotura txarra

Irudian ikusten den bezala ez dira motorrak dituen hiru zuloak ikusten, bi bakarrik ikusten dira.

- Etxe domotikoaren kableatuan. Hainbat dispositibo daudenez, kableatua egiterakoan beste dispositiboekin interferentziak ez izateko leku espezifiko batzuetatik pasatu behar izan dira dagokion lekura konektatu arte.



5. SENTSOREAK

5.1. PIR

5.1.1. DESKRIBAPENA

PIR (infragorri sentsore pasiboa) sentsore elektroniko bat da, non eremu espezifiko batean argi infragorriak neurtzen duena. Argi geratu dadin, sentsore honen funtzionamendua ez dela tenperaturaren detekzioa edo neurketa, baizik eta, objektu batek emititzen duen erradiazio infragorria. Beste objektuen energia detektatzeko eginak daude. Horrexegatik sentsore pasiboak dira. Gehien erabiltzen den arloa alarma sistemetan da.



5.1 Irudia: PIR

5.1.2. HELBURUA PROIEKTUAN

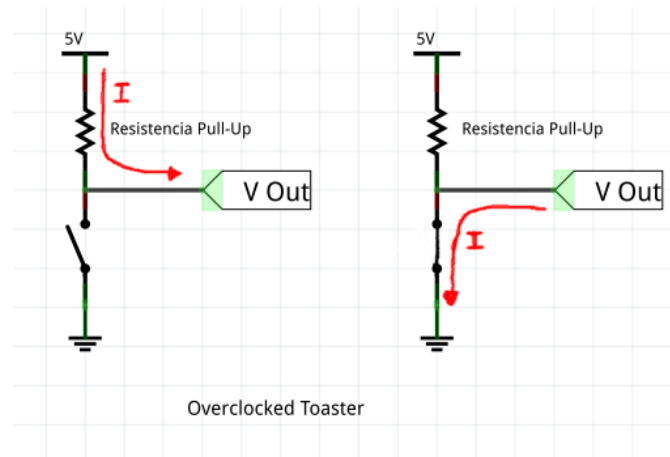
Proiektuan sentsore honek edukiko duen helburua, etxearen logela bat segurtasuna edukitzea izango da. Alarma sistema baten funtzioa edukiko du. Norbait logelara sartuko balitz, sentsorea aktibatuko litzateke seinale bat bidaliz alarma erantzun bat bezala. Logelako izkina baten goialdean kokatuta egongo da, logelaren eremu guztia detektatuz.

Sentsorea 24 ordu lanean egongo da egunero, hau da, denbora osoan.



5.1.3. FUNTZIONAMENDUA

Programatzerako orduan, sarrera digital moduan adieraziko da. Bi egoera edukiko ditu HIGH edo LOW eta honen kontrola ziurragoa izateko Pull-Up erresistentzia erabiliko da. Pull-Up erabiliko da Pull-Down erabili ordeztu sistema ziurrago delako. Sistema honek etengailua irekita dagoenean intentsitatea elikadura iturritik V_{out} -ra doa HIGH egoeran eta etengailua aktibatzen denean intentsitatea lurrera (GND) joango da $V_{out}=0$ balioarekin.



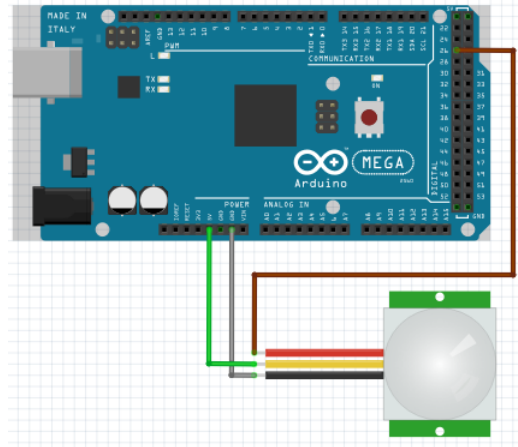
5.2 Irudia: Pull-Up eskema

Uneoro "itxaron" egoeran egongo da (HIGH) sarkin bat sartzen den momentura arte. Une horretan, erradiazioa aldaketa nabaritu du eta egoeraz aldatuko da alarma erantzun bat ematen kontrolagailuari, gure kasuan arduinori. Arduino sentsorearen seinalea jaso eta egin beharreko kudeaketa edo prozedura martxan jarriko du. Kontrolagailua "itxaron" egoeran egon ez dadin uneoro, etendura baten bidez saihestuko da arazoa. Kontrolagailua beste prozesu edo agindu bidali edo jaso dezake etendurari esker. Sarrera digital bat denez, dagokion etendura erabiliko da.



5.1.4. ESKEMA ETA KONEXIOAK

Arduino MEGA-ko 18 pinean (Marroia) , 5 V (Berdea) eta lurrera (Grisa).



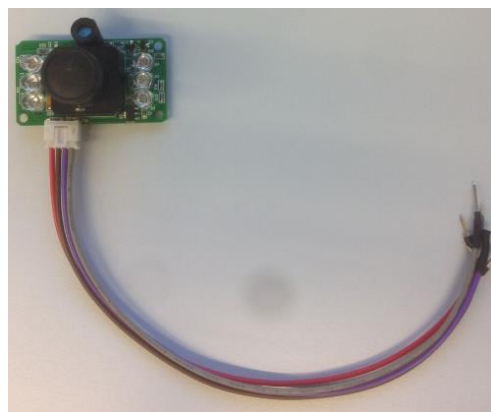
5.3 Irudia: PIR eskema

5.2. KAMERA

5.2.1. DESKRIBAPENA

Kalitate altuko argazkiak ateratzeko gai den dispositiboa da. Bere portu seriea erabiliz argazkia bidaltzen du. Ateratako argazkiak JPEG formatuan itzultzen ditu UART-aren bitartez.

Elikatzerako orduan gomendatzen da 5v, zeren eta, 3 v-ekin elikatzen bada bere abiadura motelagoa izango da.



5.4 Irudia: PIR eskema



5.2.2. HELBURUA PROIEKTUAN

Proiektuan sentsore honek edukiko duen helburua, etxearen logela bat segurtasuna edukitzea izango da. Seinale baten edo pantailan “argazkia” botoiaren menpe aktibatuko da, gure kasuan PIR sentsorearen seinalearen menpe.

Norbait logelara sartuko balitz, PIR sentsorea aktibatuko litzateke seinale bat kamerara bidaliz eta honek argazkia aterako luke. Logelako atearen aurkako izkina baten goialdean kokatuta egongo da, logelaren eremu guztia enfokatuz. Sentsorea seinale bat jasotzen duenean lanean egongo da, hau da, beste sentsore edo egoeraren menpe dago.

Segurtasun sentsore bat izango da. Logelan kokatuta egongo da eta seinale baten edo pantailan “argazkia” botoiaren menpe aktibatuko da, gure kasuan PIR sentsorearen seinalearen menpe. Kamera uneoro seinalea itxaroten egongo da eta seinalea heltzen zaionean argazki bat aterako da memoria txartelean argazkia gordez. Honekin lortu nahi dena da, sarkin bat logelara sartzekotan argazki bat ateratzea eta horrela sarkina aurkitzeko edo nor den jakiteko aukera edukitzea logelaren segurtasuna handituz.

5.2.3. FUNTZIONAMENDUA

Kamera programatzerakoan, lehen aipatu den bezala serie konexioaren bidez konektatuko da. Dispositibo honek Arduino kontroladorea seinale bat bidaltzen daionean argazki bat aterako du. Kamera ateratako argazkia “.jpg” formatuan ateratzen ditu. Proiektu honetarako erabilitako pantailak ezin duenez “.jpg” fitxategiak irakurri, mikroSD txartelean gordeko dira ateratako argazki guztiak.

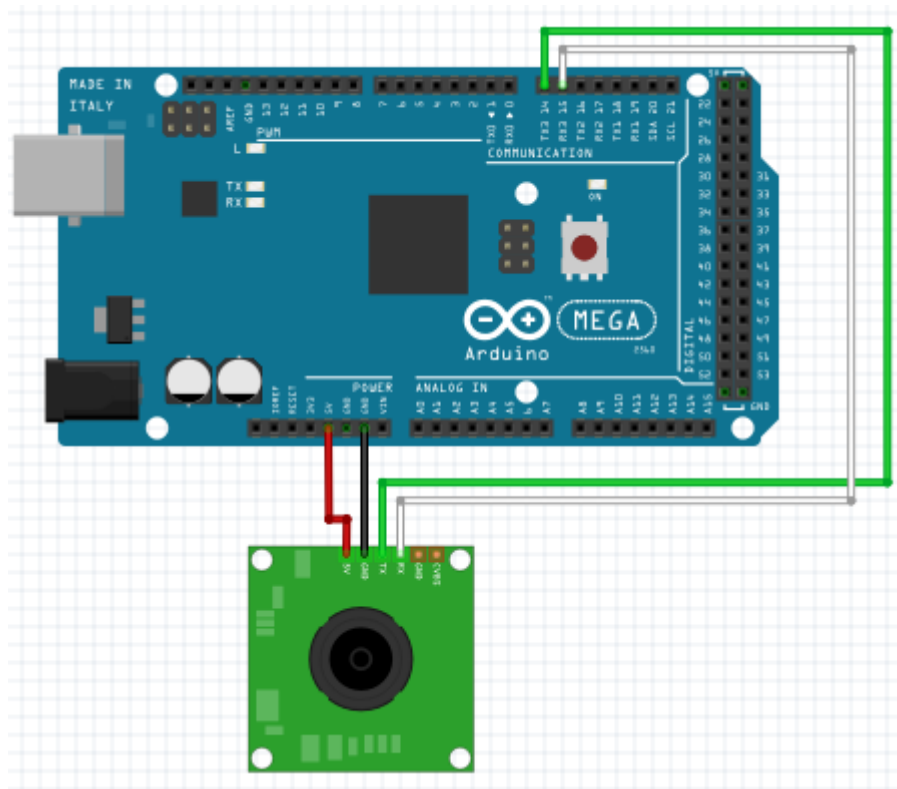
Kamera bi egoeraren menpe egongo da uneoro. PIR sentsorea egoeraz aldatzen bada Arduinok argazkia ateratzeko agindua bidaliko dio eta bestea kontrol kontsolan argazkia ateratzeko aukerarekin nahi diren argazkiak aterako dira.



5.2.4. ESKEMA ETA KONEXIOAK

Kamera lau irteera ditu:

- kable berdea (arduinoren TX3).
- kable zuria (arduinoren RX3).
- kable beltza (GND) .
- kable gorria (5v-etara).



5.5 Irudia: Kamera eskema

5.3. TFT PANTAILA

5.3.1. DESKRIBAPENA

Erabilera askoko TFT pantaila bat da. Irudiak adierazteko, ukipen-pantaila bat edo bi gauzak egiteko gai da. Kontrolagailu askorekin erabili daiteke. Arduino, Seeduino eta Arduino Mega-ren kasuetan konexioa zuzena da.



SPI komunikazioa erabiltzen du, horrela ez da abiadura galtzen datuak transmitzerakoan. SD txartel modulu bat integratuta dauka bere ezkutuan edozein informazio gorde ahal izateko.



5.6 Irudia: TFT pantaila

5.3.2. HELBURUA PROIEKTUAN

Elementu garrantzitsuenetarikoa da. Bi funtzio beteko ditu: Egoeren adierazpena eta etxearen kontrola ukipenaren bidez.

Egoeren adierazpena duen abantaila, guk automatizatu ditugun atalak bere funtzioak betetzen ari diren jakiteko gai izango gara akzioaren lekuan egon gabe. Adb: sarreran bagaude eta pertsianak jaisten badira, guk pantailatik ikusi dezakegu pertsianak jaisten ari direla, ez gara egongelara joan behar ikusteko zelan pertsianak jaisten diren.

Etxearen kontrola ukipenaren bidez duen abantaila, automatizatutako prozedurak edozein momentuetan aldatu edo pultsatu ditzakegula. Adb: Argazki bat atera nahi badiogu logelara pantailaren dagokion botoia pultsatuz gero argazkia aterako da. Horrela kamera edozein momentuan argazkia ateratzeko aukera edukiko du eta ez bakarrik sarkin bat sartzen bada. Denbora osoan martxan egongo da bai egoerak adierazten edo etxearen kontrolean.



5.3.3. FUNTZIONAMENDUA

Lehen aipatu den bezala, bi funtzio beteko ditu: Egoeren adierazpena eta etxearen kontrola ukipenaren bidez.

Hainbat egoeren berri adieraziko dira dispositibo honen bidez:

- Sarreraren kontrolean:
 - Balioduna adierazi.
 - Baliogabea adierazi.
 - Sarbide hiru aldiz erratzekotan, segurtasun teklatura adierazi.
 - Pasahitza okerra bada, Alarma adierazi.
 - Atea ireki adierazi.
 - Atea itxi adierazi.
- Logela kontrolean:
 - Sarkin bat sartzekotan, Argazkia ateratzen dagoela adierazi.
- Egongela kontrolean:
 - Pertsiana igo adierazi.
 - Pertsiana jaitsi adierazi.

Dispositibo honekin etxearen egoera kontrolatu ahalko da edozein momentuan ukipenaren bitartez:

- Sarreraren kontrolean:
 - Hatz-aztarnan: Identifikatu, berri bat sartu edo hatz bat ezabatu.
 - RFID: Identifikatu.
 - Atea: Ireki edo ixteko aukera.
 - Pasahitza: Identifikatu edo berri bat jarri.
- Logela kontrolean:
 - Argazkia bat atera.
- Egongela kontrolean:
 - Pertsiana igo.
 - Pertsiana jaitsi.

Esan beharra dago ukipenaren bitartezko kontrolaren barruan beste zenbait egoera berri adieraziko direla:

- Pasahitza:
 - Pasahitza egokia
 - Pasahitza okerra
 - Pasahitza berria
- Hatz-aztarna:
 - Hatz-a “gorde” dauden mezuak.
 - Hatz-a “identifikatu” dauden mezuak.
 - Hatz-a “ezabatu” dauden mezuak.
- RFID:
 - Txartela “identifikatu” dauden mezuak.



- Atea:
 - o Irekitzen edo ixten mezuak.

5.3.4. ESKEMA ETA KONEXIOAK

TFT pantailak arduinorekin “dorre” konexioan konektatzen da, hadu da, pataila arduinoren gainean konektatuko da. Egin beharreko konexioak:

- Arduinoren 6 pina TFT_DC
- Arduinoren 5 pina TFT_CS
- A0 ukipen pantaila
- A1 ukipen pantaila
- A2 ukipen pantaila
- A3 ukipen pantaila



5.7 Irudia: TFT pantaila eta Arduino MEGA



5.4. RTC

5.4.1. DESKRIBAPENA

Real time clock edo denbora errealeko erlojua ordua adieraziko duen sentsorea izango da. Orduaren datuak eta helbideak seriean bidaltzen dira komunikazio bus I2C baten bitartez. Bi sentsuetan informazioa bidaltzeko aukera dauka, hau da, jaso eta datuak bidaltzeko aukera. Erloju honek segundoak, minutuak, orduak, eguna, hilabetea eta urtearen informazioa ematen du. Bi eratan jazo daiteke ordua 24orduko formatuan ala 12 orduko formatuan AM/PM. Bere kontsumoa erreserban oso baxua da, 500nA. Elikatze akatsak detektatzeko gai da eta detektatuz gero, bere barneko pilarekin elikatuko litzateke ordua eta dataren informazioa ez galtzeko.



5. 8 Irudia: RTC

5.4.2. HELBURUA PROIEKTUAN

Etxe domotikoaren ezinbesteko atal bat segurtasuna da eta segurtasuna handitzeko orduaren ezagutza behar da uneoro. Horregatik uneoro ordua jakiteko eta adierazteko sentsore bat erabiliko da.

Sarreraren kontrola eta segurtasuna hobetzeko. Etxearen jabeak zein orduan sartzen diren jakiteko. Sarkin bat sartzekotan, ordua eta data adierazteko.

Beste aldetik, orduaren ezagutza beste adar batzuetarako ezinbestekoa da: Zein estazioan gauden jakiteko: udan edo neguan; eta pertsianen automatizazio prozeduran orduaren menpe pertsianaren kontrola eramango da.

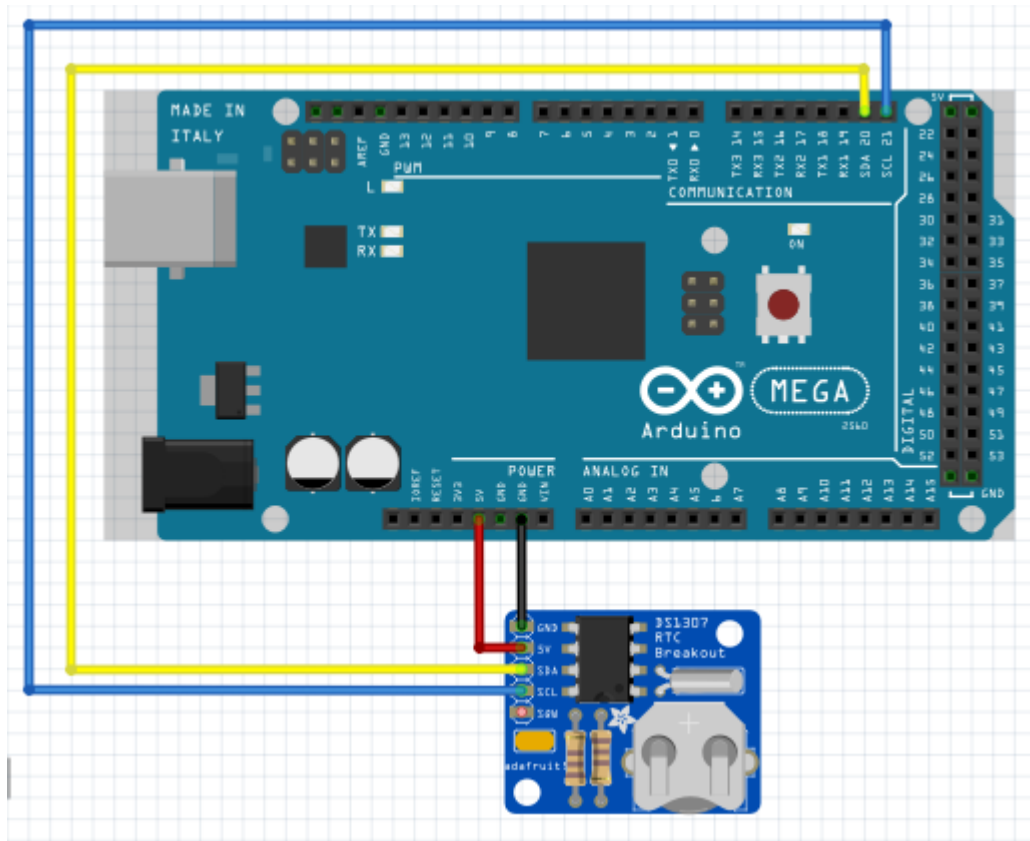
5.4.3. FUNTZIONAMENDUA

Erlojua 24 orduko formatuan ezarri da(PM).

Lehendabizi, programa hasieran jarri nahi dugun data idatzi beharko dugu eta zehaztutako datatik erlojua martxan jarriko da. Ezarritako datarekin, uda edo negua den adieraziko da programan. Estazioaren arabera, pertsiaren kontrol desberdina gauzatuko da:

- **Negua:** Urritik – Ekainera arte
- **Uda:** Ekainetik-Urrira arte

5.4.4. ESKEMA ETA KONEXIOAK



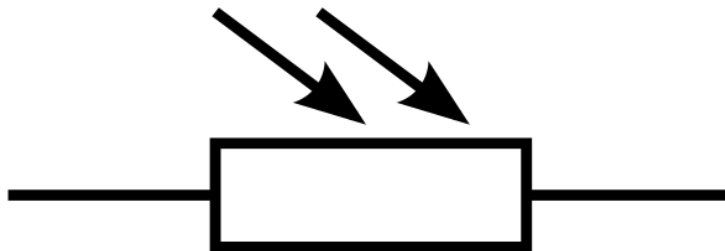
5.9 Irudia: TFT pantaila



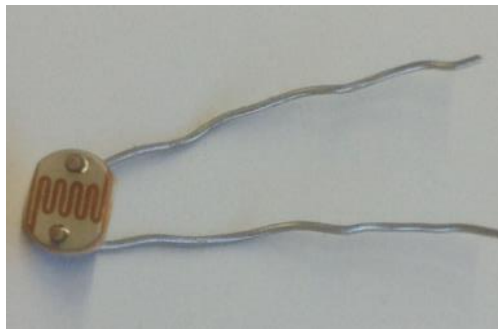
5.5. LDR

5.5.1. DESKRIBAPENA

Osagai elektriko bat da, non bere erresistentzia murrizten joango da argiaren intentsitatea handitzen denean. LDR baten erresistentzia elektriko oso baxua da argi asko dagoenean eta oso altua iluntasuna dagoenean.



5.10 Irudia: LDR



5.11 Irudia: LDR

5.5.2. HELBURUA PROIEKTUAN

Etxearen atal batetarako konponente garrantzitsua izango da. Kasu honetan egongelaz hitz egiten ari gara. Egongelan pertsianaren kontrolerako oso lagungarria izango da, zeren eta, kaletik egongelara sartzen den argiaren arabera pertsiana igo edo jaitsi egingo da automatikoki. LDR-a uneoro argia neurtzen egongo da horrela programa automatiko bat lortuko dugu.

5.5.3. FUNTZIONAMENDUA

Hasteko, LDR-aren parametroak frogatu behar dira, gure kontrolean balio egokiak ezartzeko. Bi balio zehaztuko dira: Iluntze eta Argitasun balioak.

Orduan, bi egoera eman al dira:

1. Iluntze balioaren berdina edo txikiagoa bada, pertsiana dagokion agindua jasoko du.
2. Argitasun balioaren berdina edo txikiagoa bada, pertsiana dagokion agindua jasoko du.

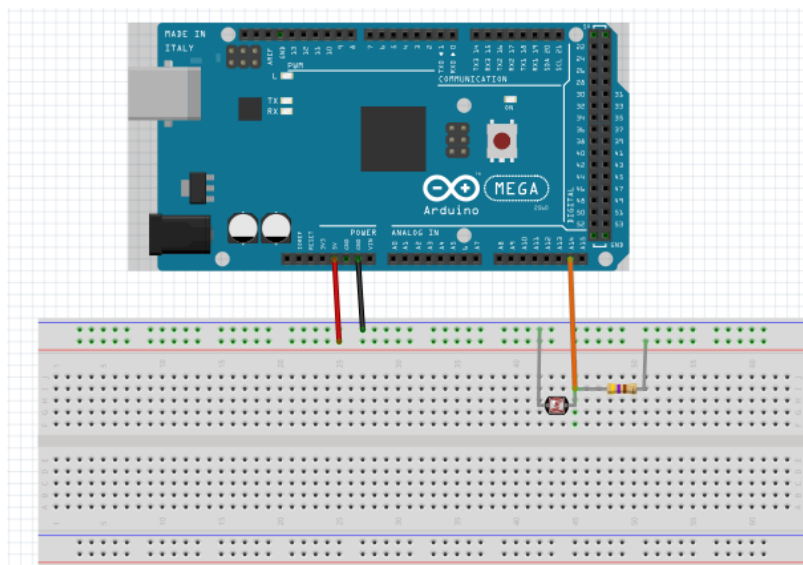
RTC-an zehaztutako dataren arabera (negua edo uda)era batetan edo bestetan konfiguratuko da programa.

Bloke diagrama batekin azalduko da, egongelaren deskripzioan agertuko dena.

5.5.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak:

- LDR-aren hanka bat GND.
- LDR-aren beste hanka 470 ohm-eko erresistentziaren hanka batera.
- Erresistentziaren beste hanka 5v.
- A14 arduinoren pina LDR eta erresistentziaren tartean konektatuko da.



5.12 Irudia: LDR konexioak



5.6. HATZ-AZTARNA DETEKTAGAILUA

5.6.1. DESKRIBAPENA

Hatz-aztarnak detektatzeko dispositibo bat da. Sistema biometriko dispositibo guztiak bezala ukipena irakurtzeko gai dira. Moduluaren barnean hatz-aztarna algoritmo bat eta sentsore optiko bat dauka eta horren bitartez hatzak gorde edo identifikatu ditzake. Komunikazio ataka hardware seriean (TX/RX) egiten da 5v-eko elikadurarekin.

Datu basean irakurri, idatzi edo ezabatu egin daiteke, hau da, hatz-aztarnak gordetzeko, identifikatzeko edo ezabatzeko ahalmena dauka.

Dispositibo honek, pertsona baten berezko ezaugarriren bat neurtzeko, kodetzeko, konparatzeko, biltzeko, transmititzeko eta/edo ezagutzeko, zehaztasuneko eta fidagarritasuneko gradu batekin gai da. Erabilitako ezaugarri fisikoak desberdinak izan daitezke, oinatz digitaletatik, ahotseko oinatz bezala, eskuaren geometria... Gure kasuan, hatz-aztarnarekin etxera sartuko gara.



5.13 Irudia: Hatz-aztarna detektagailua

5.6.2. HELBURUA PROIEKTUAN

Etxera sartzeko prozedura seguruago bat ematea. Hatz-aztarna ezin denez galdu edo kopiatu, giltzak baino seguruagoa da. Arduraren aldean, ezin da galdu giltzen moduan edo ahaztu pasahitzen moduan.



Atea zabaltzeko ukipenaren bitartez egindako prozedura bat ematea. Prozedura honetan baliiodun jendea baliogabeko jendearekin ezberdintzeko gai izango da. Baliiodun jendea sartzera, pertsonari buruzko informazioa gordetzea, adibidez: Pertsonaren izena irakurri berari egokitutako agurra pantailan adieraziz.

5.6.3. FUNTZIONAMENDUA

Sarreraren kontrolean funtzio garrantzitsu bat beteko du. Norbait etxera sartu nahi baldin bada hatz-aztarna jarritz, bi irteera egongo dira: Baliioduna edo Baliogabea.

- Baliioduna: pertsona ezagun bat da. Hatz hori erregistratua dagoela hatz-aztarnaren erregistroan.
- Baliogabea: pertsona ezezagun bat da. Ez dago erregistratua.

Dispositibo honek hatzak irakurri, idatzi edo ezabatu dezake:

- Gordetzeko prozedura (enroll): 200 hatz-aztarna gorde dezake. Hatz-aztarna bat gordeta geratzeko datu basean, hiru aldiz enroll funtzioa gauzatuko da. Enroll funtzioan gorde nahi den hatza jarri behar da denbora tarte txiki batean. Hatza gorde eta berriro bi bider gehiago hatz berdina eskatuko da. Hiru irakurketak positiboak badira, hatza gordeko da eta prozesu honetan irakurketa bat okerra bada berriro prozesu berdina hasi beharko da.
- Identifikatze prozedura (identify): Datu basean dauden hatz-aztarnekin konparatu egingo da eta gordeta badago adieraziko da zein hatz-aztarna den.
- Ezabatzeko prozedura (delete ID/ALL): Hatz-aztarna bakarra edo guztiak ezabatzeko aukera ematen da.

Hatz-aztarnaren liburutegian gordeta egongo dira familiaren hatzak. Orduan, gonbidatuaren bat etxean epe txiki edo handi bat geratu nahi ez gero, bere hatza gordetzeko aukera emango da pantailaren bitartez.



5.6.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak 5.13 irudian ikusten den bezala:

1. Kable beltza: RX.
2. Kable zuria: TX.
3. Kable zuria: GND.
4. Kable zuria: 5V.

5.7. RFID

5.7.1. DESKRIBAPENA

Egin beharreko konexioak: Datuak biltegitratzeko eta berreskuratzekeo sistema bat da, non, txartelak, etiketak edo tags RFID dispositiboak erabiltzen dituen.



5.14 Irudia: RFID

Txartelaren memoria egitura hiru zatitan banatzen da: sektoreak, datu eta trailer blokeak.

5.1 Tabla: RFID memoriaren egitura

		Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Sektor1	Bloke 7	KEYA_S1 (6 bytes)						ACCESS_S1 (3 bytes)			KEYB_S1 (6 bytes)						
	Bloke 6																
	Bloke 5																
	Bloke 4																
Sektor 0	Bloke 3	KEYA_S0 (6 bytes)						ACCESS_S0 (3 bytes)			KEYB_S0 (6 bytes)						
	Bloke 2																
	Bloke 1																
	Bloke 0	NUID						Manufacturer Data									



Txartel batek hainbat sektore ditu. Sektore bakoitzak 4 bloke eta bloke bakoitza 16 byte ditu. Beraz, 1Kbyte-eko txartel batek 16 sektore ditu.

Sektore baten lehenengo hiru blokeak datu blokeak dira (0. sektoreko 0. blokea izan ezik).

Sektore baten azkenengo blokeari "sector trailer" deitzen zaio. Bere edukia:

- Bi gako: KEYA eta KEYB.
- Sektore osatzen duten bloke bakoitzaren atzipen baimenak.

Etxeko sarreraren atea zabaltzeko ukipen gabeko prozedura bat ematea. Baliiodun sarkinak baliogabeko sarkinetatik ezberdintzea. Txartelaren jabeari buruzko informazioa gordetzea, adibidez: Pertsonaren izena irakurri berari egokitutako agurra pantailan agerraraziz.

Erosotasun handiagoa eskaintzen du.

5.7.2. HELBURUA PROIEKTUAN

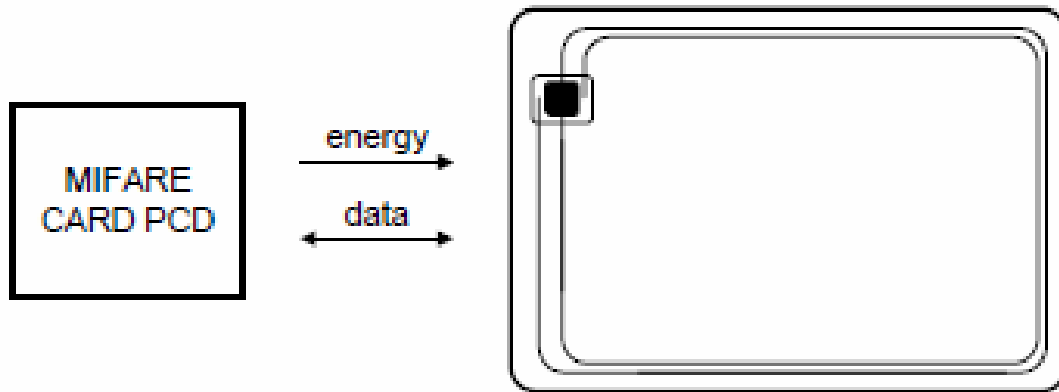
Etxeko sarreraren atea zabaltzeko ukipen gabeko prozedura bat ematea. Baliiodun sarkinak baliogabeko sarkinetatik ezberdintzea. Txartelaren jabeari buruzko informazioa gordetzea, adibidez: Pertsonaren izena irakurri berari egokitutako agurra pantailan agerraraziz.

Erosotasun handiagoa eskaintzen du.

5.7.3. FUNTZIONAMENDUA

Egin beharreko konexioak:

Txartelak haririk gabeko memoria ez hegakor batek bezala funtzionatzen du. Elikadura irakurleak sortzen duen eremu magnetikotik dator.



5.15 Irudia: MIFARE Sistema

Txartel bat RFID-aren eremura hurbiltzen denean, honek erantzun bat emango du. Bi erantzun eman dezake: Baliioduna edo Baliogabea.

- Baliioduna: pertsona ezagun bat da. Hatz hori erregistratua dagoela hatz-aztarnaren erregistroan.
- Baliogabea: pertsona ezezagun bat da. Ez dago erregistratua.

Txartelaren atzipen baimenak aldatzeko adibide batekin azalduko da. Demagun 1. sektoreko gakoak eta atzipen baimenak aldatu nahi direla:

- KEYA_S1 {0xAA, 0xAA, 0xAA, 0xAA, 0xAA, 0xAA}
- KEYB_S1 {0xBB, 0xBB, 0xBB, 0xBB, 0xBB, 0xBB}
- Sektorearen atzipen baimenak {0x78, 0x77, 0x88}, hau da:

7. blokea: C1 ₇ =0, C2 ₇ =1, C3 ₇ =1	} Beraz, 1. sektoreko datu blokeak:	<ul style="list-style-type: none"> • KEYA edo KEYB gakoekin irakurri daitezke • KEYB gakoarekin bakarrik idatzi daitezke 	} Beraz, 1. sektoreko:	<ul style="list-style-type: none"> • KEYA gakoa ezin daiteke inoiz irakurri, baina KEYB gakoa erabilita aldatu daiteke. • KEYB gakoa ezin daiteke inoiz irakurri, baina KEYB gakoa erabilita aldatu daiteke. • Atzipen baimenak KEYA edo KEYB gakoak erabilita irakurri daitezke, baina KEYB gakoarekin bakarrik aldatu daitezke.
6. blokea: C1 ₆ =1, C2 ₆ =0, C3 ₆ =0				
5. blokea: C1 ₅ =1, C2 ₅ =0, C3 ₅ =0				
4. blokea: C1 ₄ =1, C2 ₄ =0, C3 ₄ =0				

Txartelean gordeko dena:

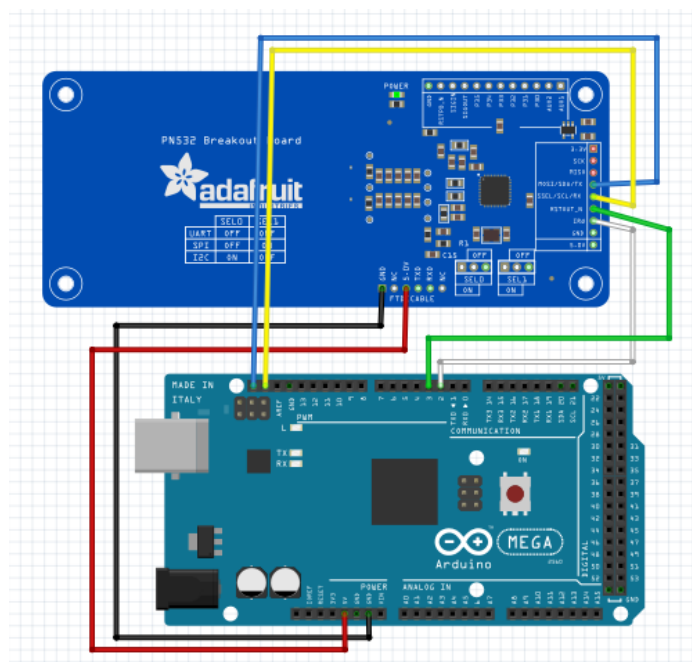
5.2. Tabla: RFID atzipen baimenak adibide bat

	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8	Byte 9	Byte 10	Byte 11	Byte 12	Byte 13	Byte 14	Byte 15
Sektor 1	Bloke 7	0xAA	0xAA	0xAA	0xAA	0xAA	0xAA	0x78	0x77	0x88						
	Bloke 6															
	Bloke 5															
	Bloke 4															
Sektor 0	Bloke 3	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0xFF	0x07	0x80		0xFF	0xFF	0xFF	0xFF	0xFF
	Bloke 2															
	Bloke 1															
	Bloke 0	NUID					Manufacturer Data									

5.7.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak:

- Kable beltza: GND
- Kable gorria: 5V
- Kable horia: SDA
- Kable urdina: SCL
- Kable berdea: PN532_RESET
- Kable zuria: PN532_IRQ



5.16 Irudia: MIFARE Sistema



6. KONPONENTEAK

6.1. ERRESISTENTZIAK

6.1.1. DESKRIBAPENA

Erresistentzia osagai elektriko pasibo bat da. Erresistentzia ideal baten erresistentzia elektrikoa ez da aldatzen nahiz eta berari ezarritako potentzial diferentzia edo bera zeharkatzen duen korrante elektrikoa aldatu.

Ekoizten diren erresistentziak ez dira idealak eta horrela tenperatura edo ingurugiroaren beste ezaugarri batzuen aldaketek erresistentzia elektrikoaren aldaketa egite dute, halere nahigabeko aldaketa hori ahal den txikiena izatea lortu nahi izaten dute erresistentzia ekoizleek.

Erresistentziak finkoak edo aldagarriak izan daitezke, aldagarriei potenziometroak edo erreostatoak ere deritze eta euren erresistentzia elektrikoa aldatzeko mugimendu mekanikoa egin behar da, adibidez biraketa eskuz edo bihurkinez eginda.

Erresistentziak zirkuitu elektrikoetan eta zirkuitu integratuen barnean egoten dira.



6.1 Irudia: Erresistentziaren ikurra

Aterata: <https://pixabay.com/es/variable-resistencia-resistencias-36565/>

6.1.2. HELBURUA PROIEKTUAN

Sistemaren zenbait dispositiboek lagunduko die haien funtzioa betetzeko.

Hiru dispositiboetan konektatuta egongo dira:



- PIR sentsorean Pull-Up sisteman lan egiteko. Arduinoren barne erresistentzia erabiltzen da.
- LDR sentsorearen determinatu diren balioak lortzeko. 470Ω -eko erresistentzia erabiliko da.



6.2 Irudia: 470Ω -eko erresistentzia

- Transistorean kommutadore bezala lan egiteko. 220Ω -eko erresistentzia erabiliko da.



6.3 Irudia: 220Ω -eko erresistentzia

6.1.4. ESKEMA ETA KONEXIOAK

5.12 eta 6.7 irudien eskema berdina da.

6.2. TRANSISTOREA

6.2.1. DESKRIBAPENA

Solidoa den material erdieroalez (Silizioz, batik bat) eginda daude. Hiru zati dauzka: Batek eramaileak igortzen ditu (igorlea), beste batek jaso egiten ditu (kolektorea) eta hirugarrenak, beste bien artekoak, eroaleen jarioa kontrolatzen du (basea).



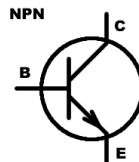
Transistorea amplifikatzaile moduan erabili daiteke. Gainera, amplifikatzaile guztiak bezala, osziladore moduan ere erabili daiteke, baita zuzentzaile eta on-off kommutadore moduan.



6.4 Irudia: NPN transistorea

6.2.2. HELBURUA PROIEKTUAN

Proiektu honetarako, NPN transistorea erabiliko da sarreran dagoen konponente bat kontrolatzeko. Amplifikatzaile bezala erabili da, zeren eta, Arduinok 5vmax bidaltzeko ahalmena bakarrik duelako eta sarraila 8v-eko elikadura behar du aktibatzeko.

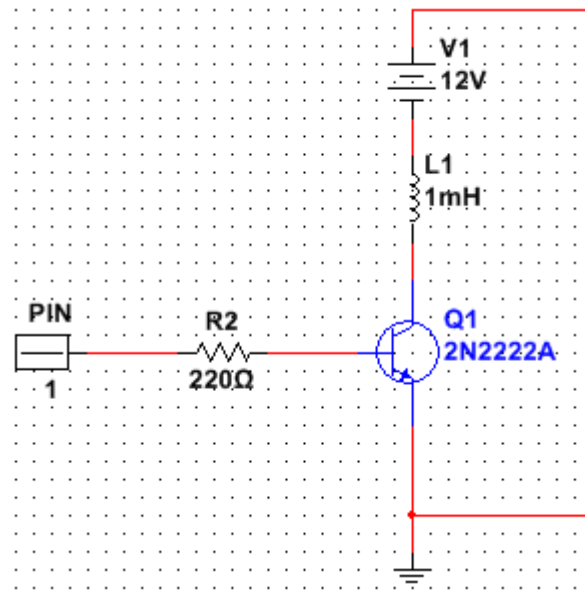


6.5 Irudia: Transistorearen ikurra

Atarata: https://commons.wikimedia.org/wiki/File:Transistor_NPN_symbol.png

6.2.3. FUNTZIONAMENDUA

PIN-etik seinalea jasotzen duenean transistorea saturatu egingo da eta kommutadore bezala lan egingo du. Kommutadore bezala lan egiten duenean 5.21 irudian gezi gorriaz adierazita dagoena, sarrailari 8V-ko tentsioa heltzea lortuko da



6.6 Irudia: Transistorearen zirkuitua

R_B erresistentziaren balio kalkulatzeko hurrengo formulak erabili behar dira:

$$I_C \leq \beta \cdot I_B$$

$$I_B \geq \frac{800}{150} = 6,3mA$$

$$5 - R_B \cdot I_B - V_{BE,sat} = 0$$

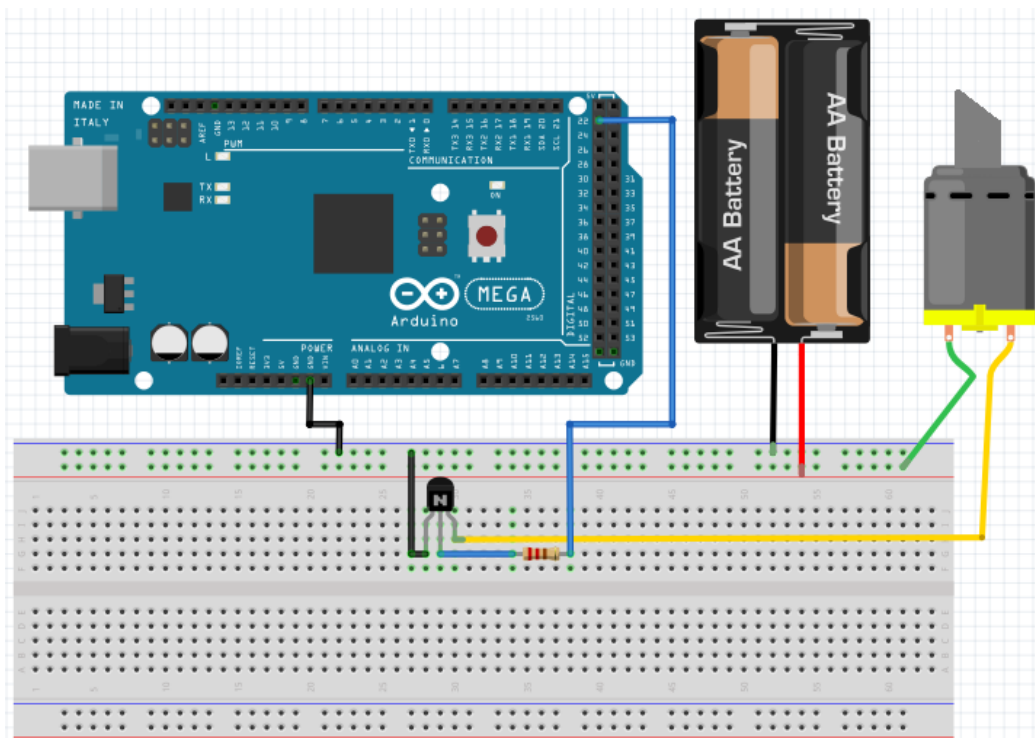
$$R_B \leq \frac{5 - V_{BE,sat}}{I_B} = \frac{5 - 0,8}{6,3 \cdot 10^{-3}} = 666,7 \Omega$$

Baldintza hori betetzen duten erresistentzia balio ezberdinak probatu ondoren, 220 ohmioko erresistentziak funtzionamendurik hoberena bermatzen duela ondorioztatu zen.

6.2.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak:

- Kable beltza: GND eta NPN-aren Kolektorera.
- Kable gorria: 8V
- Kable horia: NPN-aren emisorera
- Kable urdina: PIN 22-tik erresistentziara eta NPN-aren basera.
- Kable berdea: 8V



6.8 Irudia: Transistorearen eskema

6.3. SARRAILA

6.3.1. DESKRIBAPENA

Kisketa mota bat da. Tentsioa ematen zaionean elektroimanen bitartez sarrailaren kisketa sartu eta atera egiten da.



6.9 Irudia: Sarraila

6.3.2. HELBURUA PROIEKTUAN

Etxearen segurtasunerako osagai garrantzitsua izango da. Sarrerako atean segurtasun sistema bat izango da.

6.3.3. FUNTZIONAMENDUA

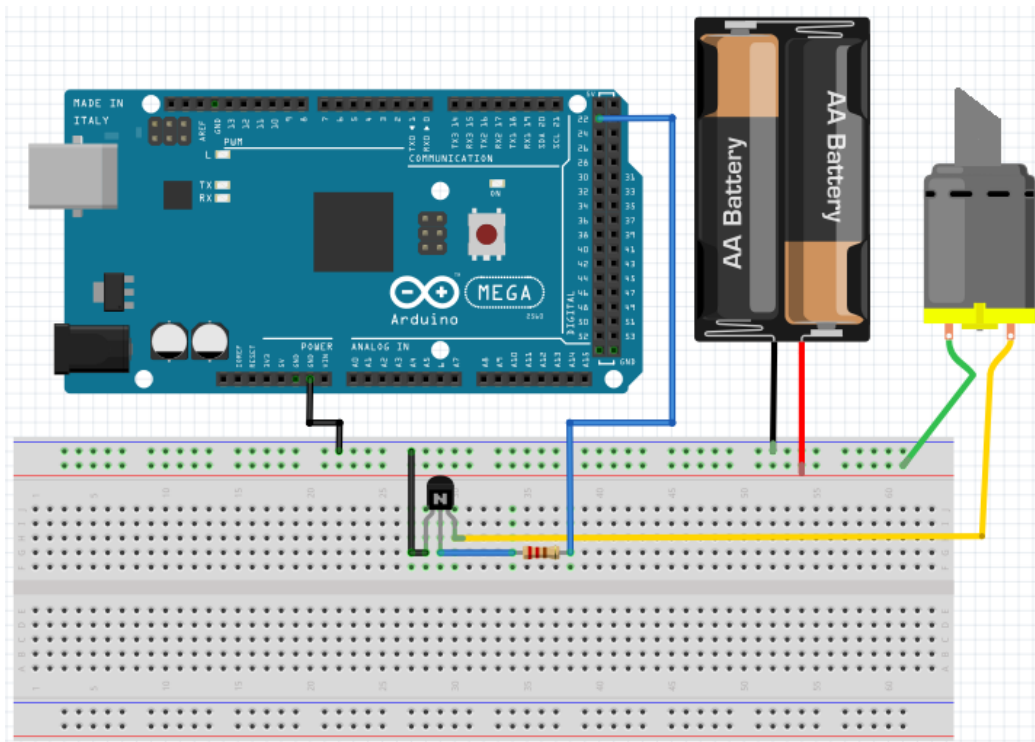
Etxearen sarrerako atearen atzealdean ezarrita egongo da eta atea ireki edo ixteko erabiliko da. Gure kasuan, 8 V-rekin elikatuko da ondorengo kontrola pasatu ondoren.

Ettxera sartzeko bi metodo daude: Hatz-aztarnaren bitartez edo txartelaren bitartez. Gure kontrolean gordeta dauden hatz-aztarna edo txartela bat bada, sarrailari tentsioa bidaliko zaio kisketa barrurantz sartuz eta horrela atea irekitzeko aukera emango da.

Pantailaren konfigurazioetan sartuz gero, atea ireki edo itxi ahalko da.

Atea bere "segurtasun-tartea" pasatu ondoren, tentsioa pasatzeari utziko zaio kisketa kanporatuz.

6.3.4. ESKEMA ETA KONEXIOAK



6.10 Irudia: Sarrailaren eskema

6.4. SERBOMOTORRA

6.4.1. DESKRIBAPENA

Serbomotorrak korrante zuzeneko motorren antzekoak dira, baina hauek euren operazio tarteko edozein posiziotan kokatzeko eta posizio horretan egonkor mantentzeko ahalmena dute. Motorrak eraldatzea posiblea da hauen mugimendu tarte mugatua ezabatzeko. Honen ondorioz motorren gaineko kontrola gutxituko da, baina indarra, abiadura eta inertzia baxua mantenduko dira. Honengatik maiz erabiltzen dira robotikan eta irratzi bidezko kontrolean.



6.4.2. HELBURUA PROIEKTUAN

Helburu garrantzitsuena sarrerako atea irekitzeko sistema automatizatu bat edukitzea da. Esfortzu fisikorik egin gabe atea ireki edo ixteko aukera izatea.

Sarreraren atalerako ezinbesteko aparatua izango da. Sarrerako atea ireki eta ixteko abiadura eta posizio kontrolatu baterako erabiliko da.

Atea irekitzeko hiru metodo daude: Hatz-aztarnaren bitartez , txartela edo pantailaren bitartez. Gure kontrolean gordeta dauden hatz-aztarna edo txartela bat bada, serbomotorrari tentsioa bidaliko zaio dagokion posiziora heldu arte eta horrela atea irekiko da. Pantailaren konfigurazioetan sartuz gero, atea ireki edo itxi ahalko da.

6.4.3. FUNTZIONAMENDUA

Helburuetan aipatu den bezala, bi egoeraren menpe egongo da. Sarreraren kontrolean hatz-aztarna edo RFID txartelaren bitartez etxera sartzeko aukera izango dugu sarrera balioduna bada. Sarrera balioduna denean, arduinoren bitartez sarraila aktibatuko da eta serbomotorrari posizio batera joateko agindua bidaliko zaio abiadura zehatz batean. Denbora tarte bat pasatu ostean atea ixteko agindua jasoko du hasierako posizioraino joaten abiadura kontrolatu batean. Beste egoeran, pantailaren kontrolean, bi botoi edukiko ditugu pantailaren aukeretan: Ireki eta Itxi. Ireki botoia sakatzuz gero, serbomotorra sarrera balioduna edukiko bagenu bezala funtzionatuko du eta Itxi botoia sakatuko bagenu atea ixteko agindua bidaliko da.

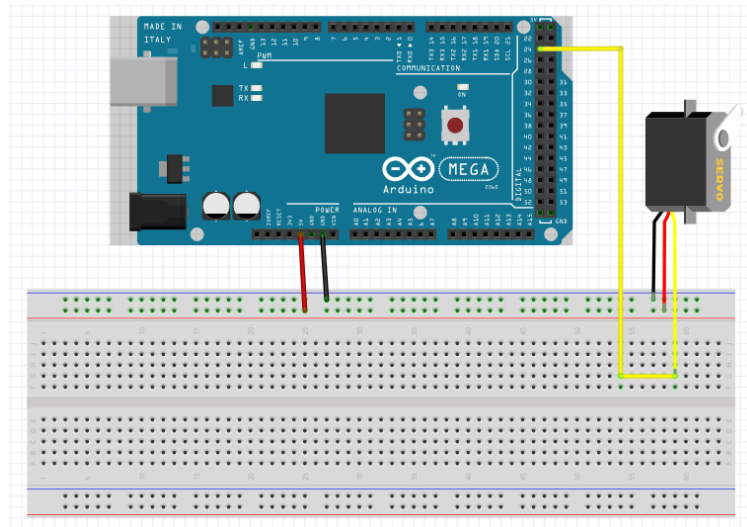
6.4.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak:

-Kable beltza: GND.

-Kable gorria: 5V.

-Kable horia: Arduinoren 24. Pin digitalera.

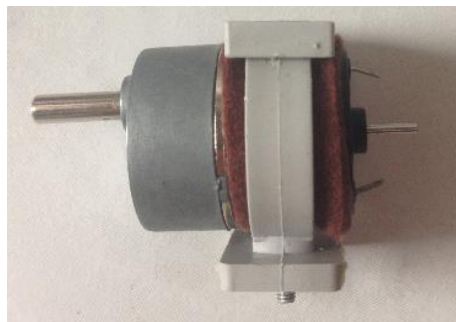


6.11 Irudia: Serbomotorraren eskema

6.5. MOTORRA

6.5.1. DESKRIBAPENA

Korronte zuzenaren energia, errotaziozko energia mekanikoan bihurtzen duen dispositiboa da DC motorra. Industria mailan gehien erabiltzen ziren motorrak ziren abiadura, pare eta posizioaren kontrol errazagatik. DC motorra bi alderdi nagusiz osatuta dago: estatorea eta errotorea. Estatoreak, orokorrean zilindrikoa denak, euskarri mekanikoa ematen dio aparatuari eta hutsa da erdigunean. Estatorean poloak aurki ditzakegu, iman iraunkorrek edo Burdinazko nukleo gainean kobrezko hariz harilkatuak izan daitezkeenak. Errotorea orokorrean zilindrikoa izaten da, harilkatua eta nukleoduna non korrontea bi eskuilen bidez iristen den. Gure kasuan, pertsianak igo eta jaisteko erabiliko dugu.



6.12 Irudia: Motorra



6.5.2. HELBURUA PROIEKTUAN

Egongelako kontrolaren atalean erabiliko da. Egongelako pertsiana igo eta jaisteko tresna izango da. Motorrarekin pertsianen kontrola automatizatzea lortu nahi da. Pertsianak motorrari agindua heltzen zaionean, igo edo jaitsi egingo dira inolako eskuzko indarririk gabe. Honekin pertsonen esfortzuak murriztuko dira.

6.5.3. FUNTZIONAMENDUA

Motorra martxan jartzeko hiru egoera gerta daitezke:

- Ezarritako ordua heltzea .
- Ezarritako argia edo iluntasuna egotea.
- Pantailan IGO edo JAITSI botoia sakatzerakoan.

Aipatutako egoera bat ematen bada, motorra martxan jarriko da dagokion norabidean pertsiana igoz ala jaitsiz.

6.5.4. ESKEMA ETA KONEXIOAK

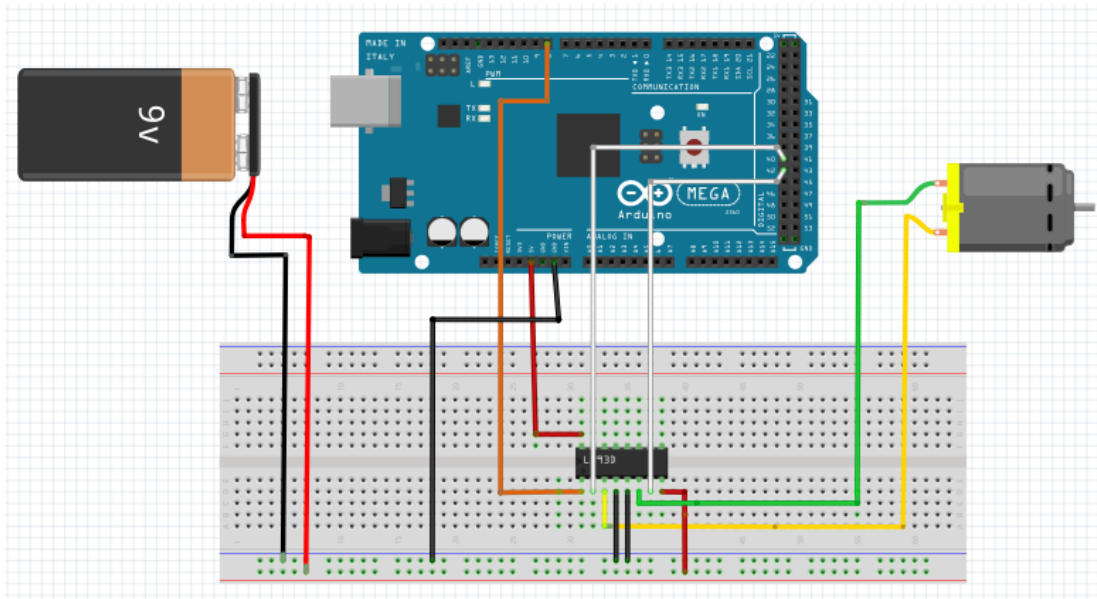
Egin beharreko konexioak:

-Kable beltza: GND

-Kable gorria: L293D-aren VS(9V)

-Kable berdea: L293D-aren OUTPUT2

-Kable horia: L293D-aren OUTPUT1



6.13 Irudia: Motorraren eskema

6.6. MOTORRAREN DRIVER-A (H ZUBIA)

6.6.1. DESKRIBAPENA

Driver honek lau zirkuitu dauzka bere barnean eta potentzia erdiko kargak (motor txikiak eta karga inдукtiboak) maneiatzea ahalbidetzen du. L293D zirkuitu integratuak karga inдукtiboek sortutako korrante alderantzizkoetatik babesteko diodoak dituzte. Banako zirkuituak era independentean erabil daitezke edozein motako kargak kontrolatzeko, motorren kasuan biraketa noranzkoa kontrolatzeko. Zirkuitu hauetako edozeinek H zubi erdia konfiguratu dezake.

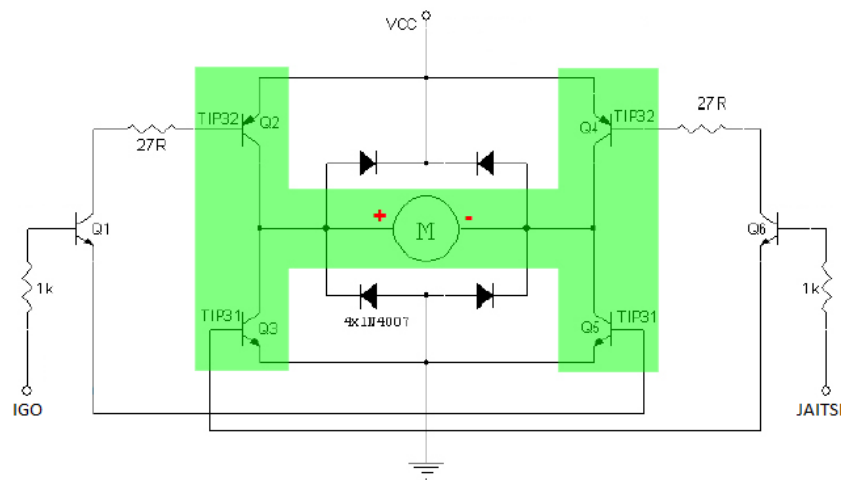


6.14 Irudia: Motorraren eskema

6.6.2. HELBURUA PROIEKTUAN

Pertsianaren motorra bi norantzatan biratu behar da. Orduan osagai hau erabiliko da pertsiana igo eta jaitsi dadin.

6.6.3. FUNTZIONAMENDUA



6.15 Irudia: H zubia

Funtzionamendua ondorengoa da:

Igo izena duen sarreran seinale positibo bat aplikatzen bada, Q1 transistorea asetu egingo da. Q1 transistoreko kolektoreko korronteak Q2 transistoreko basetik zirkulatuko du eta emisoreko korronteak aldiz, Q5 transistoretik. Honek motorraren borne positiboa Vcc-ra iristea ahalbidetzen du, Q2 transistorea asetua dagoelako eta borne negatiboa ere lurrera konektatuta dagoenez Q5 transistorea asetua dagoelako. Era honetan motorrak alde batera biratuko du.

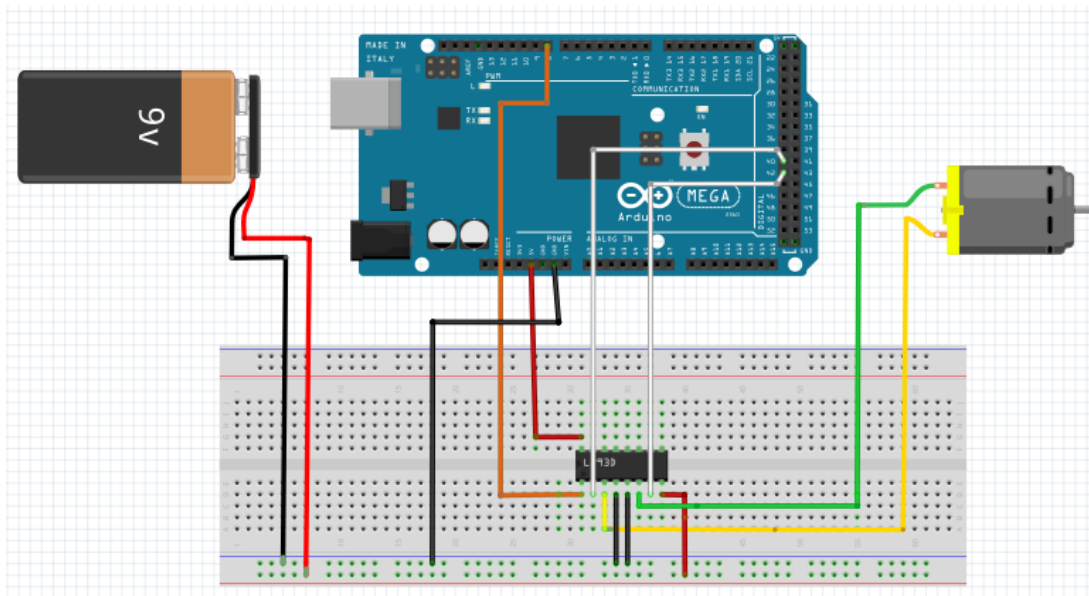
Jaitsi izena duen sarreran seinale positiboa aplikatzen badugu ordea, Q6 transistoreak eroango du korrontea, eta Q4 eta Q3 transistoreen basetatik igaroko da. Honela motorraren borne negatiboa Vcc-ra iritsiko da eta borna positiboa lurrera konektatuta geratuko da. Motorrak beste norabidean biratuko du oraingoan.



6.6.4. ESKEMA ETA KONEXIOAK

Egin beharreko konexioak:

- Kable beltza: GND
- Kable gorria: L293D-aren VS(9V) eta L293D-aren VSS(5V)
- Kable berdea: L293D-aren OUTPUT2
- Kable horia: L293D-aren OUTPUT1
- Kable zuria: L293D-aren INPUT1 (A) Arduinoren 40. pinera ETA INPUT2 (B) Arduinoren 42. pinera
- kable laranja: L293D-aren ENABLE-1 Arduinoren 8. pinera



6.16 Irudia: Zirkuitu integratuaren eskema

6.7. MIKROSD TXARTELA

Memoria ez hegazkordun flash motako memoria txartel bat da . Egoera solidoko biltegi familia bat da. Proiektuaren sarreran eta logelan segurtasuna kontrolaren datuak gordetzeaz arduratuko da. Etxera sartzen diren pertsonen kontrola (nor sartu eta zein ordutan) gordeko da



eta logelan sarkin bat sartzekotan argazkia eta orduaren datuak gordeko dira. MIKROSD txartela TFT-an konektatuko da.



6.17 Irudia: SD txartela

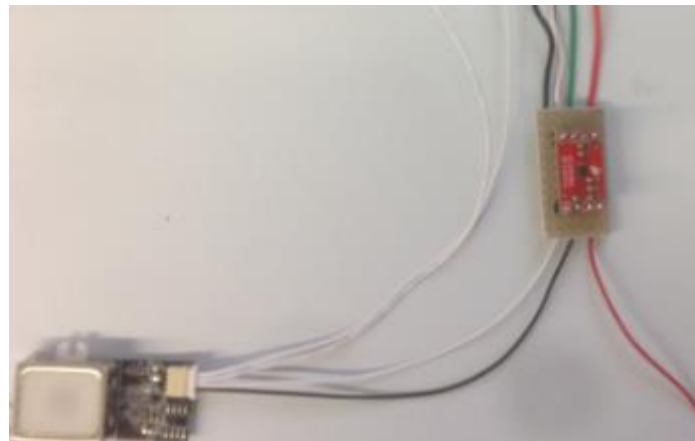
6.8. TENTSIO MAILA BIHURGAILUA

Tentsio maila jakin batetik beste tentsio maila jakin baterako bihurbeta egiten duen zirkuitu integratua da. Kasu batzuetan, alde batetik tentsio batera lan egiten du eta beste aldetik beste tentsio batekin lan egin dezake. Gure kasuan hatz-aztarna 3,3 v etan lan egiten du, orduan arduinok 5v-eko tentsioa bidaliz gero dispositiboa erretzeko probabilitate handia edukiko genuke. Horregatik, PCA9306 breakout dispositiboa erabiliko dugu.



6.18 Irudia: Breakout

I2Cbus eta SMBus erabiltzen du komunikazioetarako. Tentsio baxuko aldean 1.0V-3.6V ko tartean eta tentsio altuko aldean 1.8V-5.5V tartean. VREF1 tentsio baxua eta VRF2 tentsio altua da. Proiektu honetarako, hatz-aztarna detektagailuarekin konektatu egingo da.

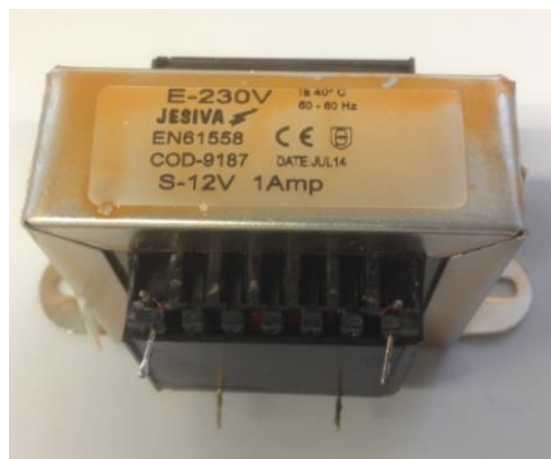


6.19 Irudia: Breakout + Hatz-aztarna detektagailua

6.9. TRANSFORMADOREA

Dispositibo elektriko bat da tentsioa handitu edo txikitu dezakena zirkuitu elektriko alferno batean, bere potentzia aldatu gabe.

Proiektu honetan, etxe domotikoa behar duen zirkuitu osoa elikatzeaz arduratuko da. Zirkuitu inprimatura konektatu egingo da eta zirkuitu inprimatutik 3 tentsio aterako dira : 5V, 8V eta 9V.



6.20 Irudia: Transformadorea

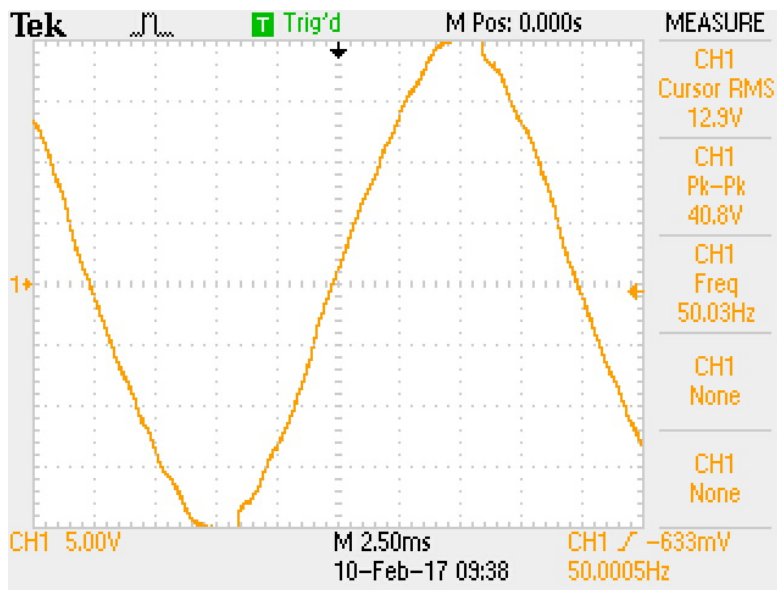
Transformadorea ondo funtzionatzen duen jakiteko osziloskopio baten bidez transformadoreak jasotzen dituen seinaleak ikusiko ditugu.



6.21 Irudia: Seinale frogak

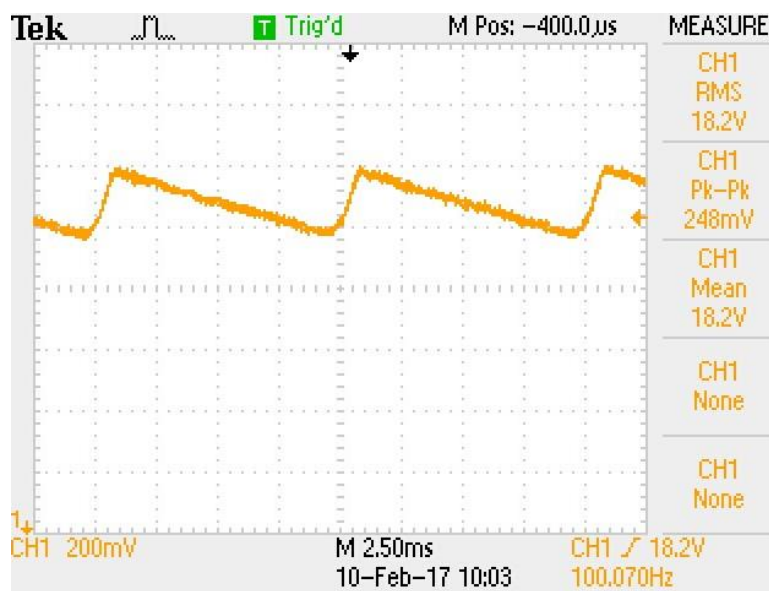
Hiru zirkuituaren hiru zatitan aztertuko da jasotako erantzuna: Sekundarioan , erregulatzailerearen irteeran eta kondentsadorearen ostean.

- **Sekundarioan:** Seinale sinusoidal bat agertuko da oraindik ez delako diodo zubiatiak pasatu. RMS, Pk-Pk eta frekuentzia adieraziko dira eta hauen balioak ondo daudela konprobatu egin dira.



6.22 Irudia: Sekundarioaren seinalea

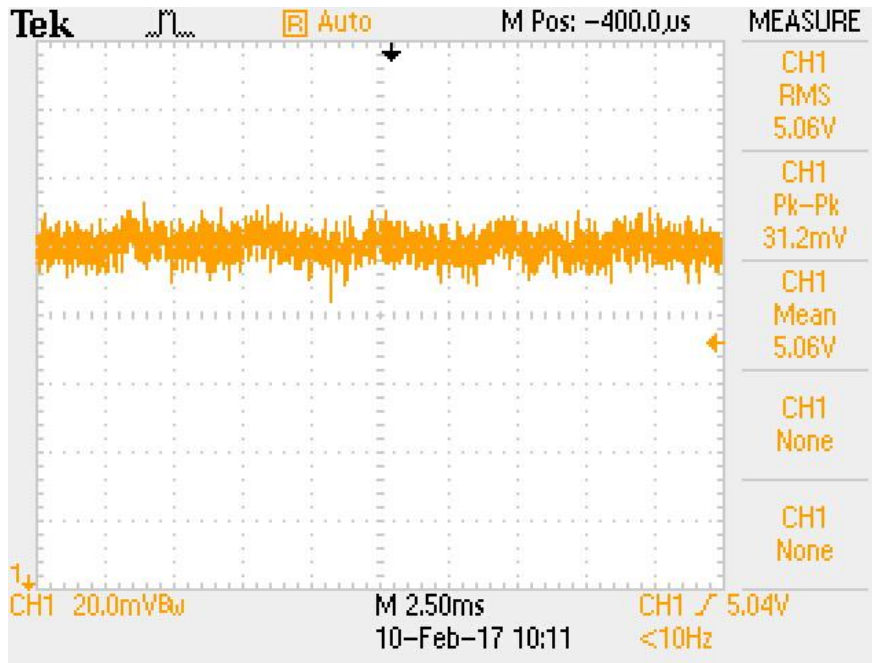
- **Kondentsadorearen ostean:** Seinale zuzena aterako, zeren ete, erreguladore batzuk erabiliko ditugu elikatze tentsioa egonkortzeko. Bere balioa 18,2V-eko da eta bere tentsio gorabehera apurraren balioa 248mv-ekoa da.



6.23 Irudia: Kondentsadorearen ostean ematen duen seinalea



- **Erregulatzaillearen irteera:** Behar dugun elikaduraren seinalea da. Seinalea asko handituz gero, bere tentsio gorabehera apurra jaitsi dela ikusi dezakegu 248mV-etik 31,2mV-era. Azkeneko balio honek mespretxagarria da 5v elikadura batentzako.

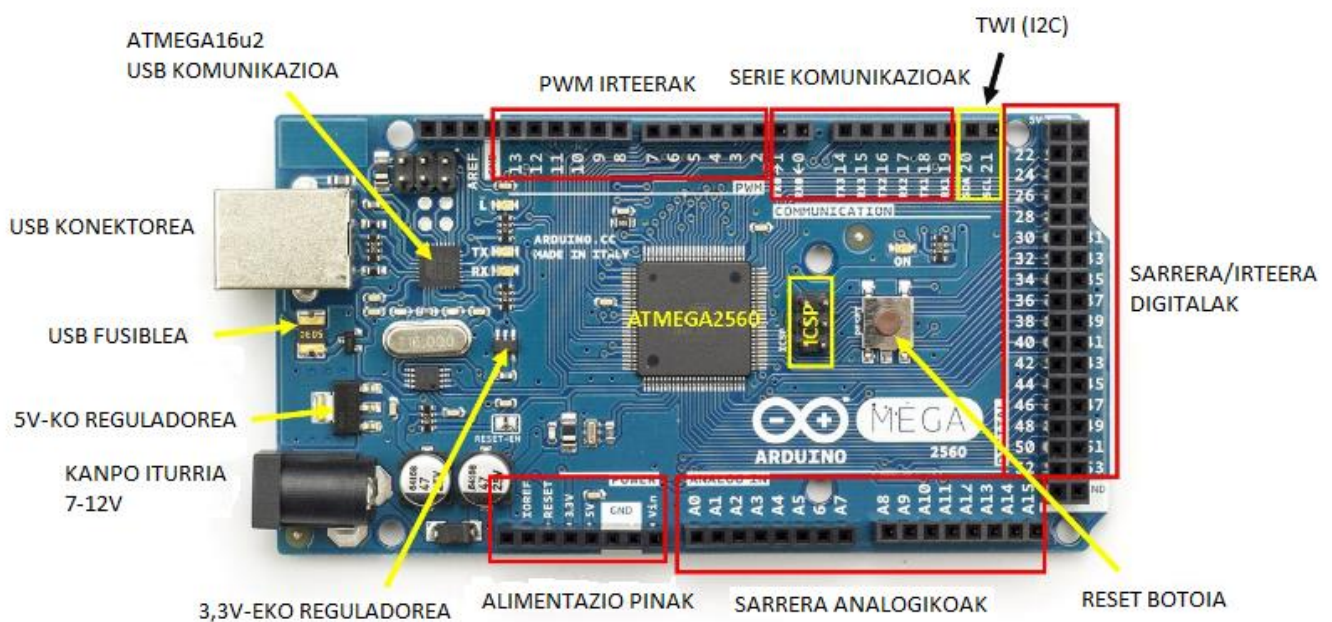


6.24 Irudia: Erregulatzaillearen irteera

7. HARDWARE

Aurreko atalean aipatu den bezala, erabiliko den hardware-a Arduino izango da, Arduino MEGA zehazki. Gama bereko gainontzeko Aduino guztiak bezala, garapen ingurune eta mikroprozesadore batean oinarritzen den plataforma askea da.

Hardware askea denez, bai diseinua bai distribuzioa, edozein eratako proiektu gauzatzeko erabili daiteke inolako lizentziarik gabe. Hau dela eta, oso aukera egokia da ikasleentzat edota edozein motako proiektu egiteko prest dauden pertsona trebeentzako elektronika arloan.



7.1 Irudia: Arduino MEGA

Ezaugarriak:

- Eredua: MEGA 2560.
- Mikrokontroladorea: Atmega2560.
- Erabilpen tentsioa: 5V.
- Sarrera tentsioa (gomendatua) : 7-12V.
- Sarrera tentsioa (limitean): 6-20V.
- S/I pin digitalak: 54 (15 PWM irteeran).
- Sarrera analogikoko pinak: 16.
- I/O pinetako korrante zuzena: 20mA.



- 3.3V-eko pinaren CC korrontea: 50mA.
- Flash memoria: 256KB.
- SRAM: 8KB.
- EEPROM: 4KB.
- Erloju abiadura: 16MHz.
- LED_BUILTIN: 13.
- Luzera: 101.52mm.
- Zabalera: 53,3.
- Pisua: 37g.

Dispositibo guztiak plakara konektatuta badaude, kontuan izan behar da dispositibo guzti hauen kontsumo korrontea. Ordenagailuaren USB-a ez da nahikoa izango, orduan plaka sare elektrikora konektatuta egon beharko da zirkuito inpresu batean eta transformadore baten bitartez.

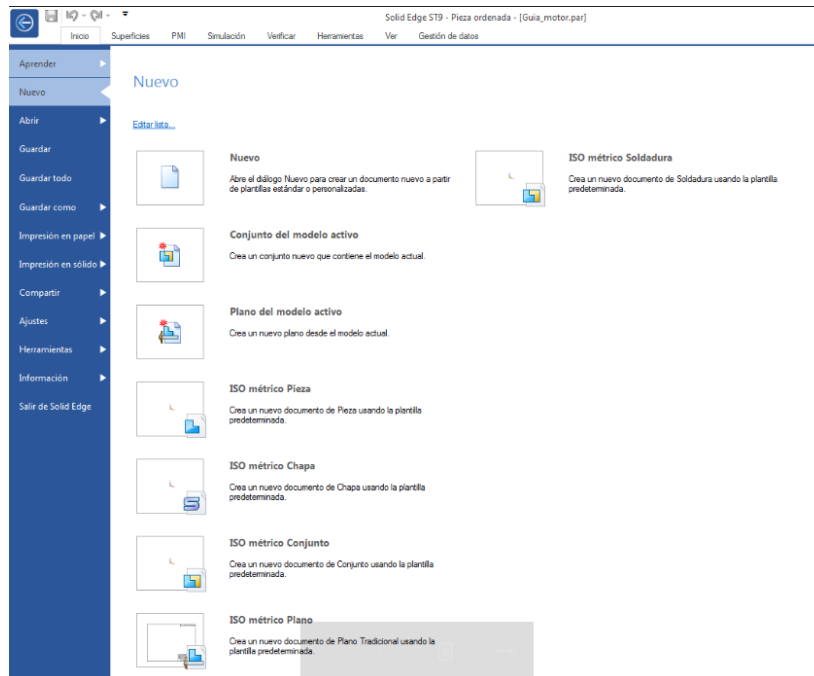
8. SOFTWARE

8.1. SOLID EGDE

Hiru dimentsiotan piezak egiteko gai den programa bat da. Piezen modelaketa, piezaren planoak, multzoak, soldadurak... egin daitezke.

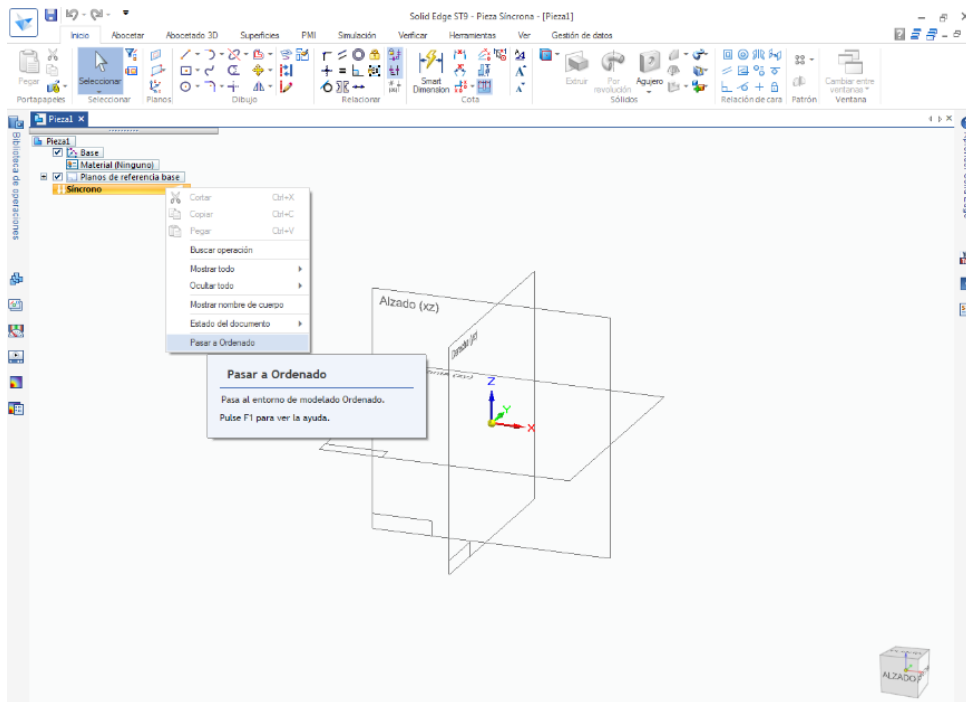
Programa pausuz pausu azalduko da:

1. Lehendabizi, Solid Edge programa exekutatu dugu. Programa irekita dagoenean, zein fitxategi mota eraiki nahi dugun aukeratu behar da, gure kasuan "ISO metrico Pieza" fitxategiak. Fitxategi horretan proiektuak behar dituen piezak egingo dira.



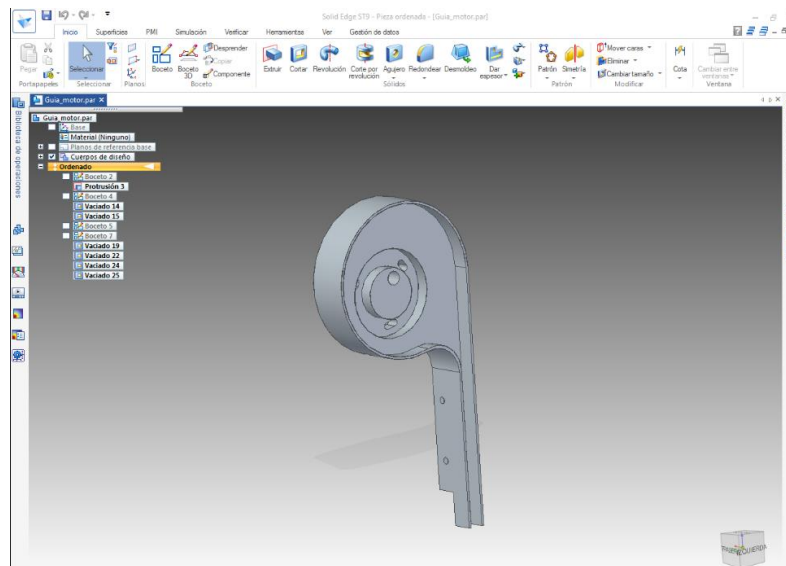
8.1 Irudia: Hasierako pantaila

2. Orain bi aldaketa egin behar ditugu, sistema ordenatura pasatu eta ardatzak agertzea klikatu.



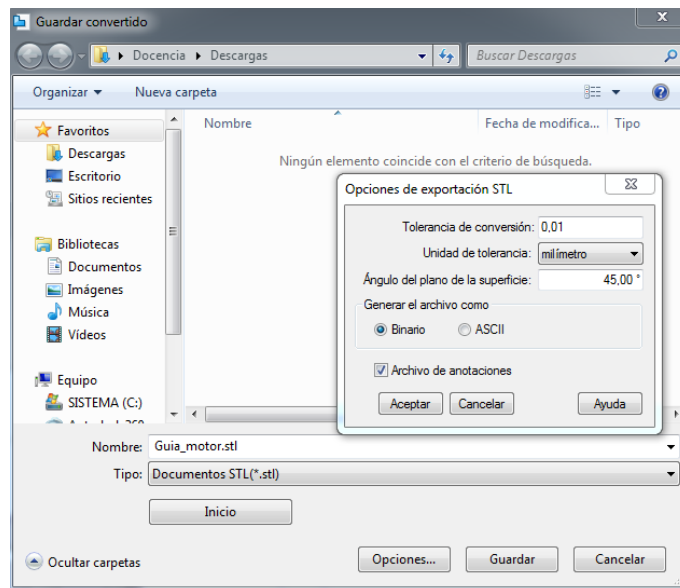
8.2 Irudia: Hasierako pantaila

3. Aurreko pausua egin ostean, piezaren bozetoarekin hasi gaitzke. Bozetoa eginda daukagunean, “extruir” operazioa egingo da piezari sakonera emateko. Sakonera eman ondoren, beste zenbait operazio egin behar ditugu piezaren arabera. Ondorengo irudian ikusi dezakegu emandako pausu guztiak.



8.3 Irudia: Pieza 3D

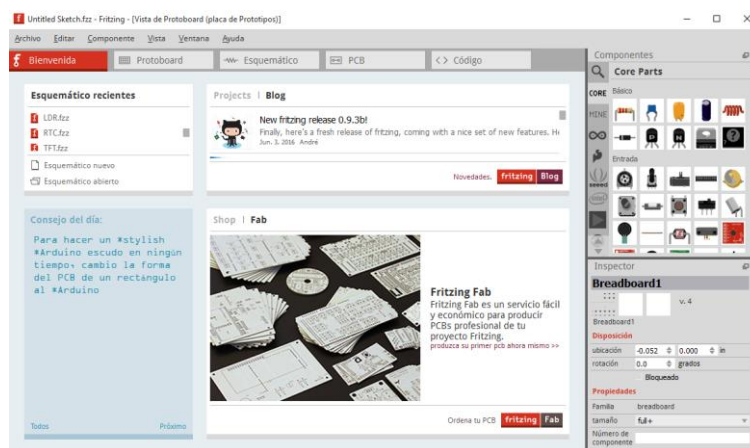
4. Bukatzeko, behar dugun pieza bukatuta dagoenean, fitxategi berezi batean “.stl” gorde behar da eta mm-tan 3D inprimagailua fitxategia irakurri ahal izateko.



8.4 Irudia: Formatua

8.2. FRITZING

Zirkuituaren diseinua garatzeko erabiliko den softwarea edo programa Fritzing izango da. Fritzing diseinu elektronika libreko automatizazio programa da. Fritzing, Arduino eta Processing-en printzipio pean sortu zen. Programa honi esker Arduinon oinarritutako prototipoak dokumentatu daitezke eta zirkuitu inprimatuen eskemak sortu, gerora fabrikatzeko nahi izanez gero.

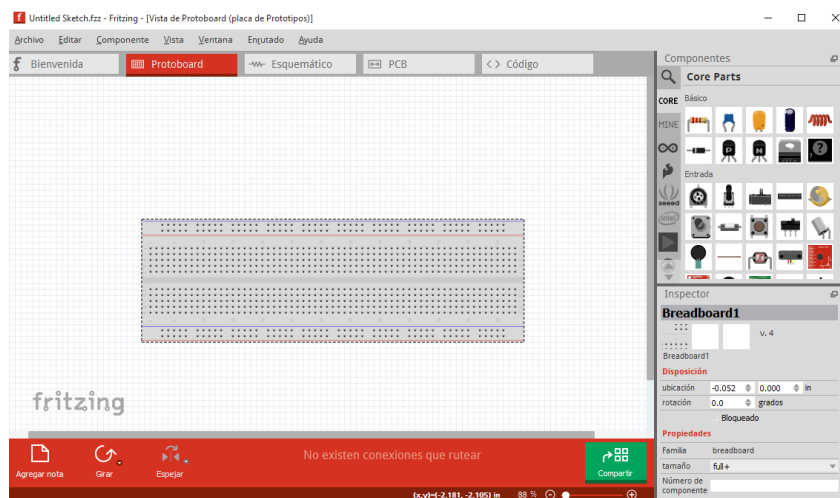


8.5 Irudia: Fritzing



Software honek ondorengo lan eremuak eskaintzen ditu 7.6 Irudian ikusten den bezala:

- Proiektuaren bista: eremu honetan zirkuitu birtuala diseinatu eta eraikitzen da. Hiru eratan egin daiteke: protoboard, PCB edo Eskema.
- Eskemaren leihoa: protoboard gainean eraikitzen den zirkuitua erakusten du. Oso praktikoa da zirkuituen ikurrak erabiltzen ohiturik dauden erabiltzaileentzako.
- PCB bista: zirkuitu inprimatuen produkzioa gauzatzeko, diseinua eta dokumentazioa egitea ahalbidetzen ditu. Beraz, zirkuitu inprimatuaren eskema ikustea posiblea litzateke norbaitek proiektu hori aurrera eraman nahi izango balu.

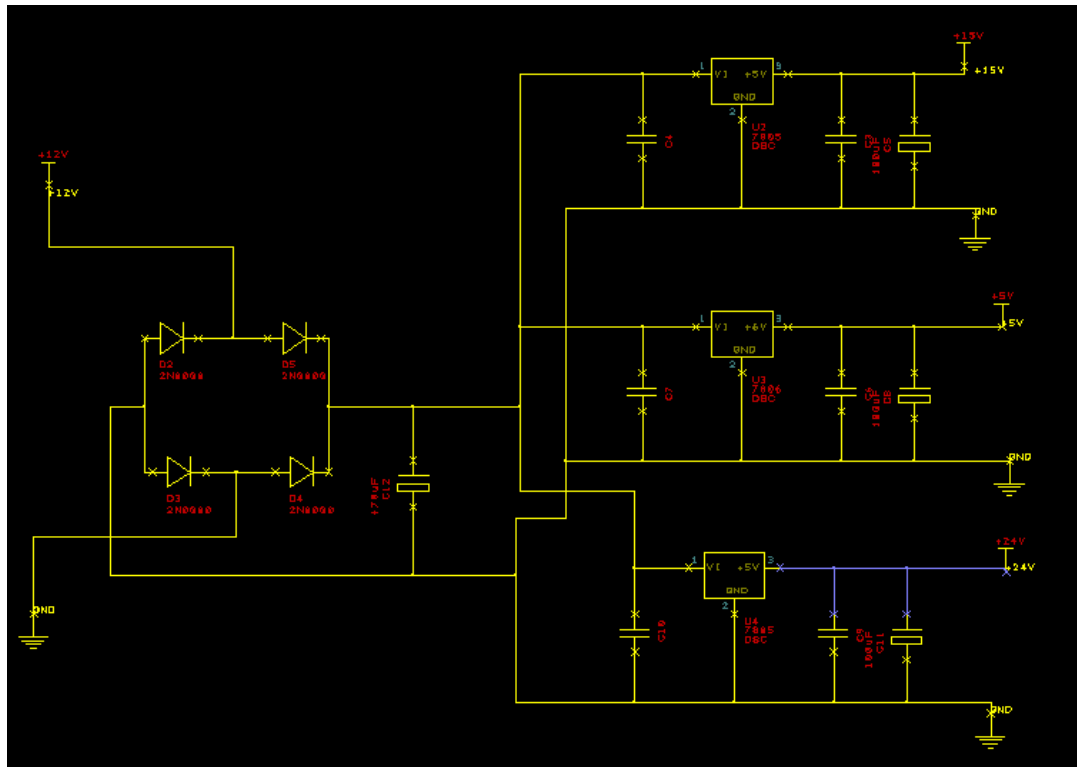


8.6 Irudia: Protoboard

8.3. DESIGNSPARK

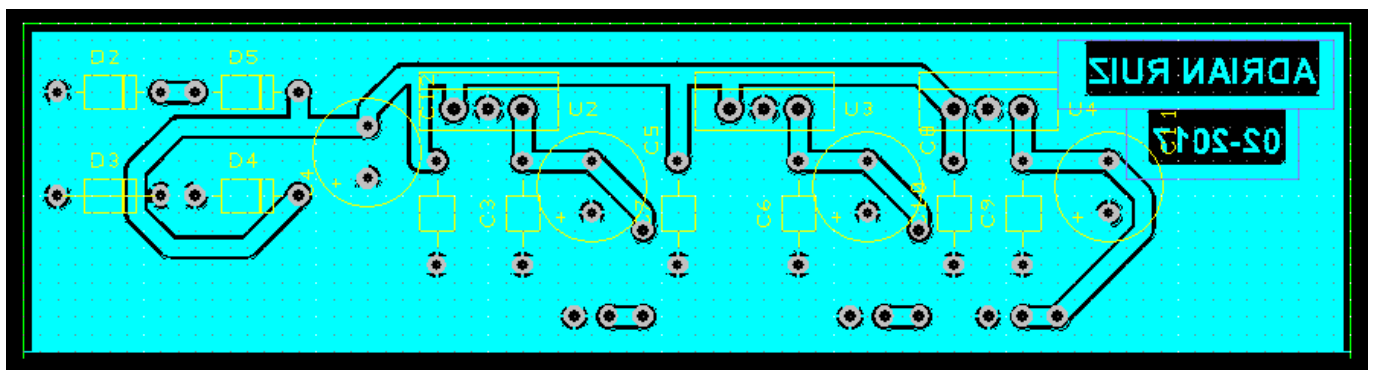
Etxea elikatzeko zirkuitua, konexioa elikadura iturriak erabili beharrean, era txukunago batean agertzeko, zirkuitu integratu batez aldatu da. Zirkuitu integratuak diseinatzeko erabili den programa DesignSpark izan da.

Programa honetan, konponente elektronikozko ezberdinak aukeratzeko aukera dago programaren barneko liburutegian. Hori esanda egin beharreko da, aukeratu behar diren konponente elektronikozko guztiak liburutegitik hautatu eta jarri nahi diren posizioetan kokatu dira. Konponente guztiak pantailan daudenean, elkarren arteko konexioa egiten hasiko da.



8.7 Irudia: DesignSpark eskema

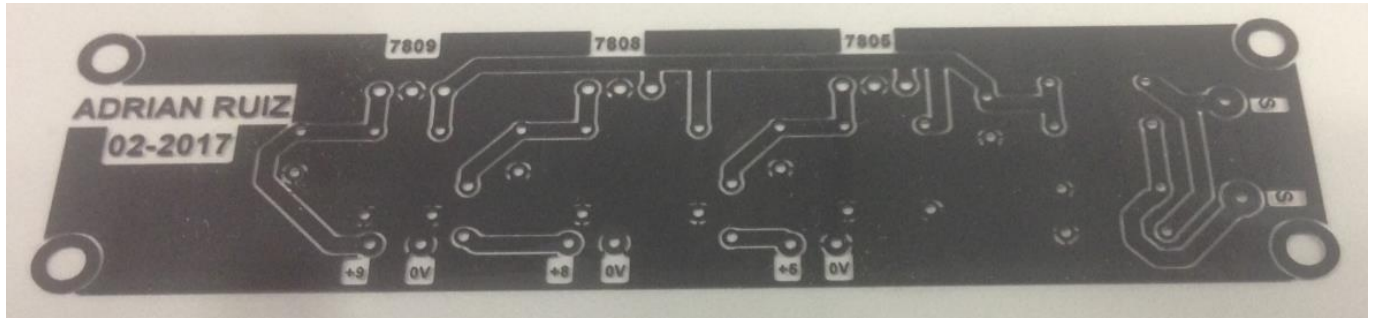
Konponente elektroniko guztien arteko konexioak eginda daudenean, DesignSpark programak ematen duen aukeretako bat da, kableen konexioak bera bakarrik ordenatzen dituela, bata bestearekin ez gurutzatzeko. Aukera honeri esker, konexio gehienak eginak egongo dira elkar gurutzatu barik. Egon daiteke konexioren bat egin gabe utzi izana eta horretarako bi aukera daude konexio horrek egiteko; bata, konponente elektronikoak posizioz aldatu eta ea horrela bide berriak aurkitu ditzakeen programak eta bestea, eskuz falta diren kableak konektatu bideak elkar gurutzatu barik.



8.8 Irudia: DesignSpark zirkuitua



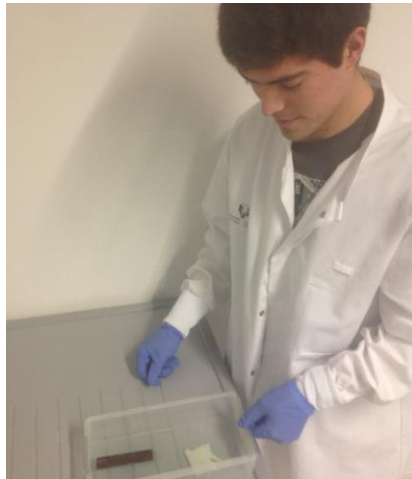
Diseinatutako zirkuitua, konponente elektroniko guztiekin eta egin beharreko konexio guztiekin egin baldin badago, inprimatzea bakarrik falta da. Zirkuitua azetato paperan inprimatzean da.



8.9 Irudia: DesignSpark

Azetato paperan inprimatu den zirkuitua, kobrezko plaka baten gainean jartzen da eta "insoladore" makinan sartu egiten da, zirkuitua plakan kopiaturik geratzeko. Lau minutu pasatzean "insoladore" makinatik kobrezko plaka ateratzen da eta kableen bideak marraztuta daudela ikusi beharko da.

Kobrezko plakaren zirkuitua hobeto ikusteko, errebelatzaile likidoan sartzen da fotolitoak isladatzeko. Denbora pixka bat pasatzean urarekin garbitzen da plaka eta atakatzaila den likido batean sartu egingo da kobrea kentzeko. Bideak kobre kolorea kentzen hasten denean erreakzioa egiten ari dela esan nahiko du eta likidotik atera egingo da, uretara pasatuz eta berriz ere garbituz.

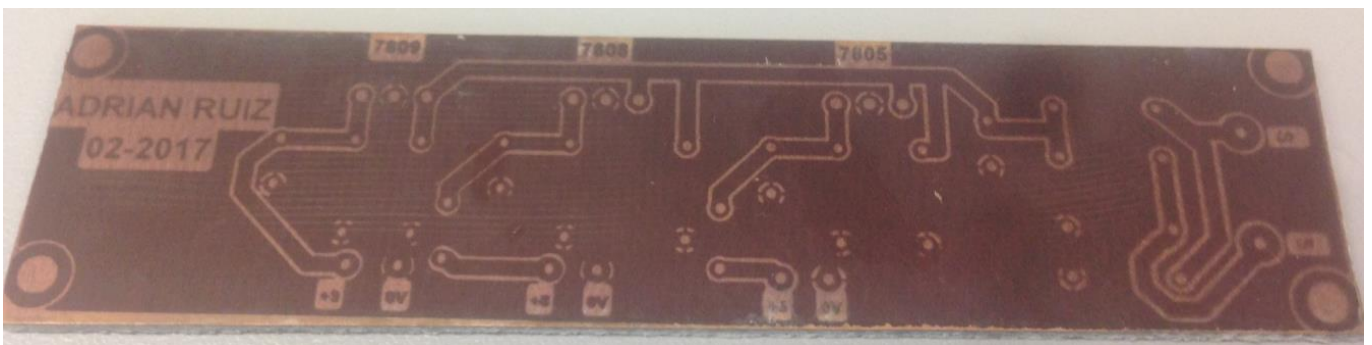


8.10 Irudia: Atakatzan

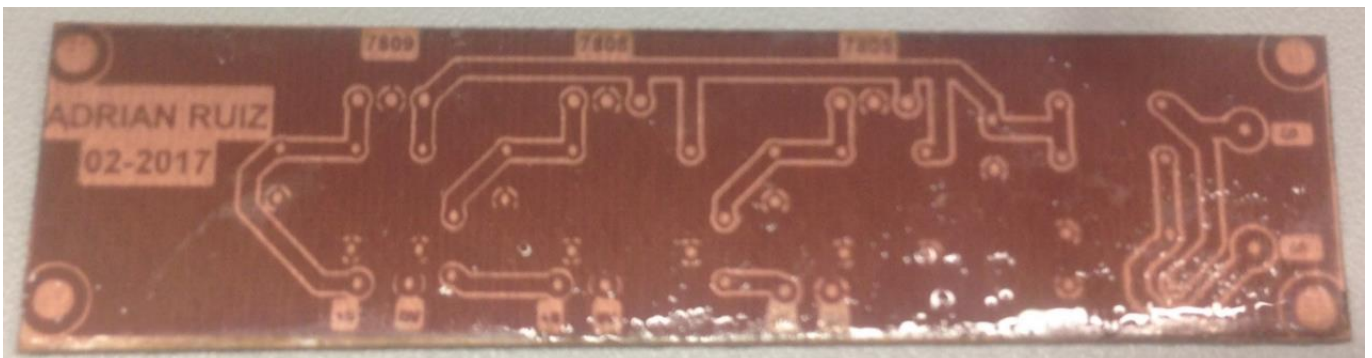
Plaka uretatik garbitzen denean eta lehortzen denean, berriz ere "insoladore" makinan 5 minutuz sartuko da. Denbora hori pasatzean berriz ere likido bereziarekin atakatu egingo zaio bideetako kobre kolorea kentzeko. Prozeduraren atal honetan zenbait arazo eduki nituen eta bi aldiz plaka gehiago egin behar izan ditut prozesua ondo atera den arte.



8.11 Irudia: 1 froga



8.12 Irudia: 2 frogak



8.13 Irudia: Azkenengo froga

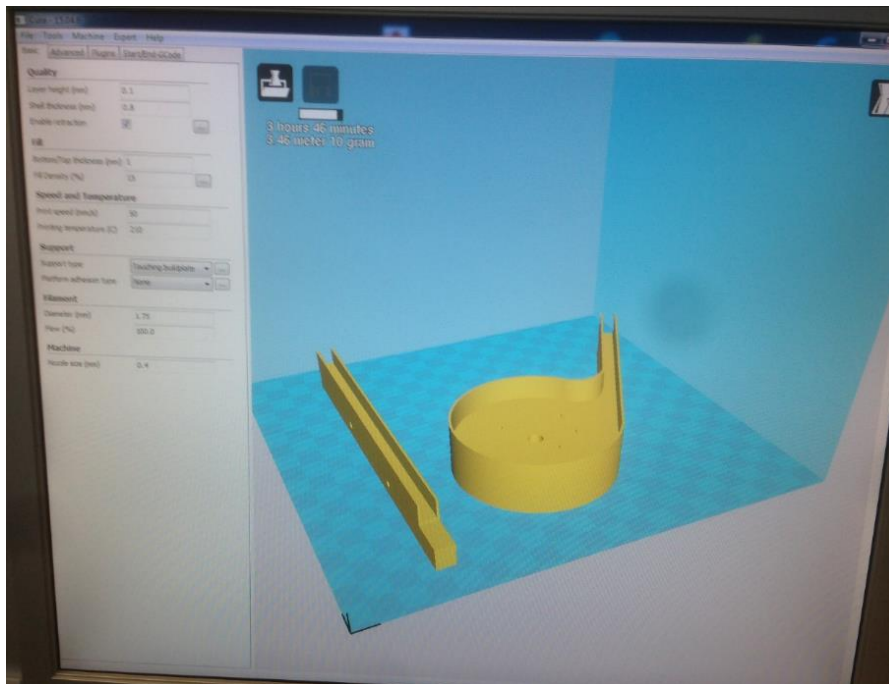
Plakaren zirkuitu integratua ondo ibiltzeko egindako pauso guztien ondoren falta den pausoa hurrengokoa da, konponente elektronikoko bakoitza plakan jartzea. Hori egiteko pin gehigarriak jarri behar dira plakan. Irudian ikusten den bezala zirkulu bakoitza, konponente bakoitzaren pinentzat dira eta zulatzeko makina erabilita, zuloak egingo dira pin gehigarriak jartzeko. Pin gehigarriak, plakari soldatuko zaizkio kontu handiz. Horrela plakan, konponente elektronikoak jartzea bakarrik faltako zen.



8.14 Irudia: DesignSpark

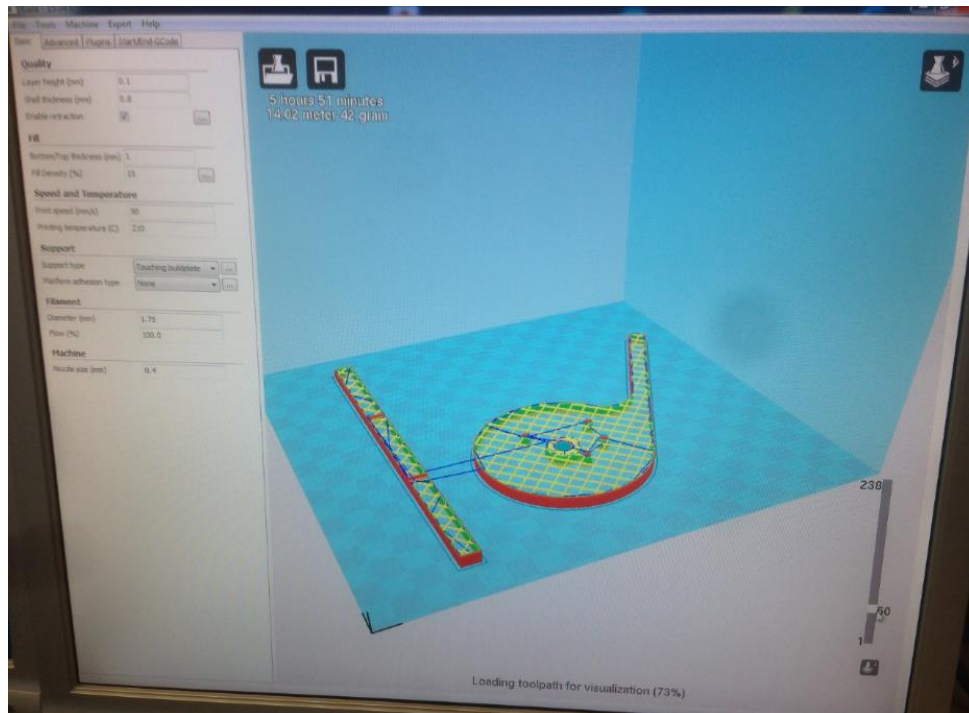
8.4. CURA

Programa honek egin nahi den pieza dagokion programarekin, gure kasuan Solid Edge, pieza solidoaren fitxategia aldatzeaz arduratzen da 3D inprimagailua irakurri ahal izateko. Prozesu honi “slicing” deitzen zaio.

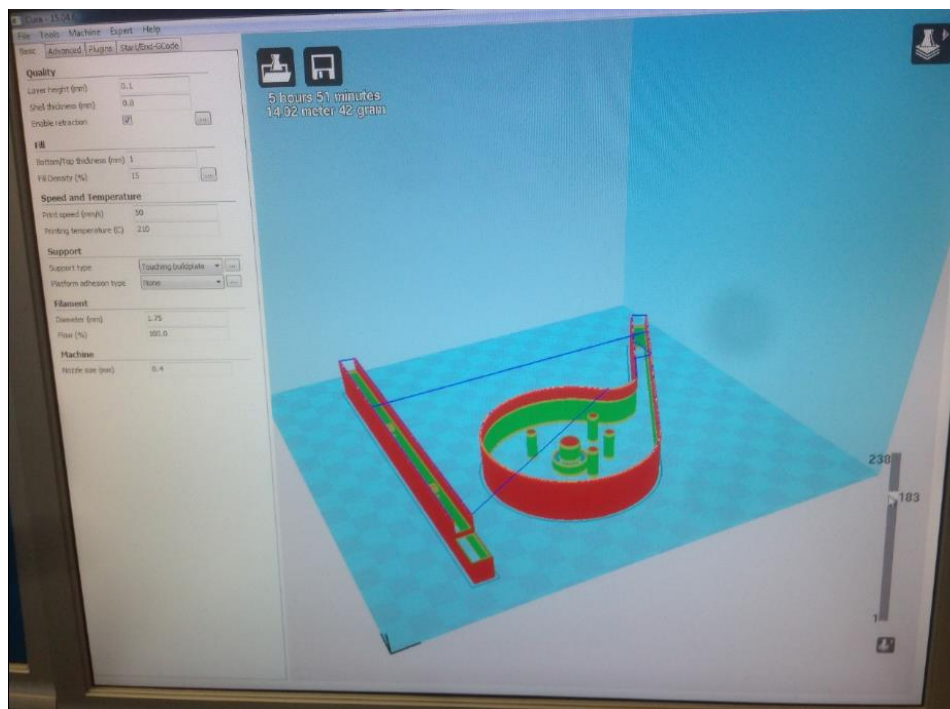


8.15 Irudia: CURA bi pieza

“Slicing” prozesua solidoa kapa ezberdinetan banatzen ditu inprimagailuaren fabrikazio teknologikoa adaptatzeko.



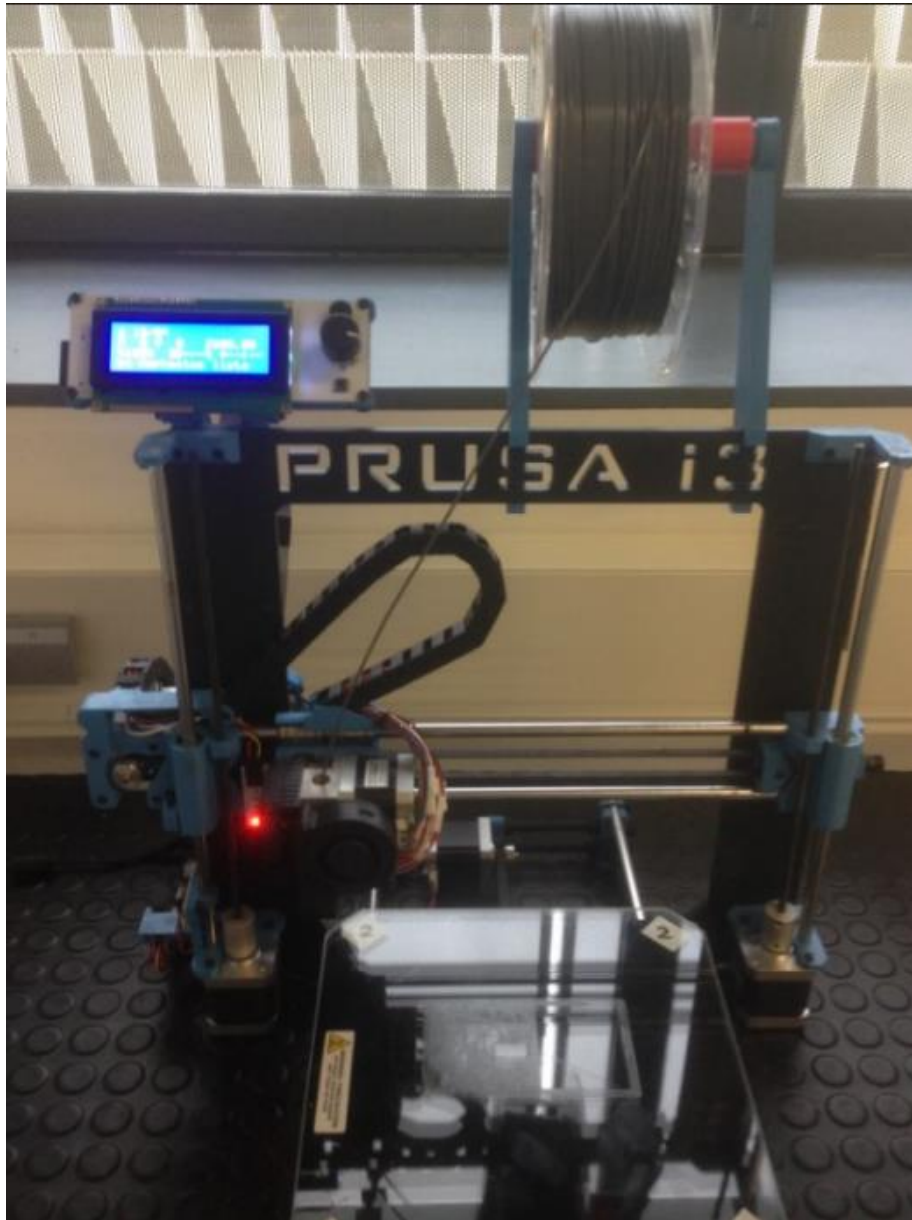
8.16 Irudia: CURA kapak



8.16 Irudia: CURA kapak



Aplikazioan kaparen altuera, inprimatzeko abiadura, piezaren dentsitatea eta beste parametro batzuk espezifikatu behar dira. Parametroak definituta daudenean dokumentu bat eraikiko da "g kode" formatuan inprimagailua irakurri dezakeen formatua delako.



8.18 Irudia: 3D inprimagailua



9. PROGRAMAZIOA

9.1. SARRERA

Programazio atalean Arduinon egindako kodea ulertzeko behar diren azalpenak emango dira. Proiektua lau atal garrantzitsuetan bananduta dago: sarrera, logela, egongela eta kontsola kontrola. Atal guztiak Arduino MEGA 2560 baten bitartez programatzen dira.

PROIEKTU HONEN INGURUAN GARATU DEN KODE GUZTIA GITHUB-EKO BILTEGI BATEAN GORDETA DAGO, BERE ESTEKA ONDORENGOA DELARIK:

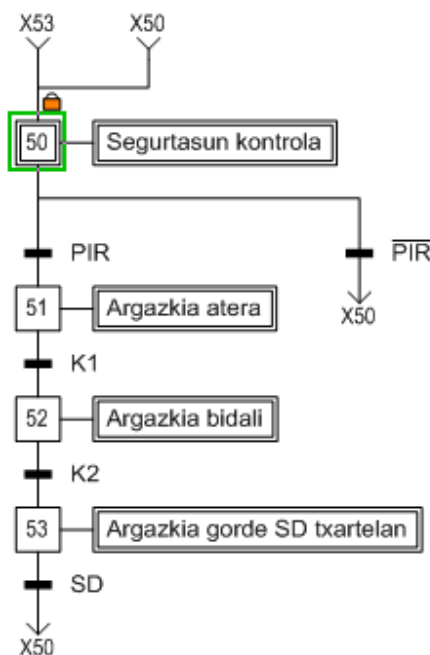
https://github.com/Adrian1808/Etxe_Domotikoa

9.2. PROGRAMAREN ADIBIDE GRAFIKOAK

Atal honetan, programa hobeto ulertzeko Grafcet baten bidez azalduko da. Grafcet-a sistema logikoak bere sarrera eta irteerekin azaltzeko era grafikoan erabiltzen da. Proiektu honek sarrera eta irteera asko dituenez, aproposa iruditu zait metodo honekin sistema islatzea.

9.2.1. GRAFCET:

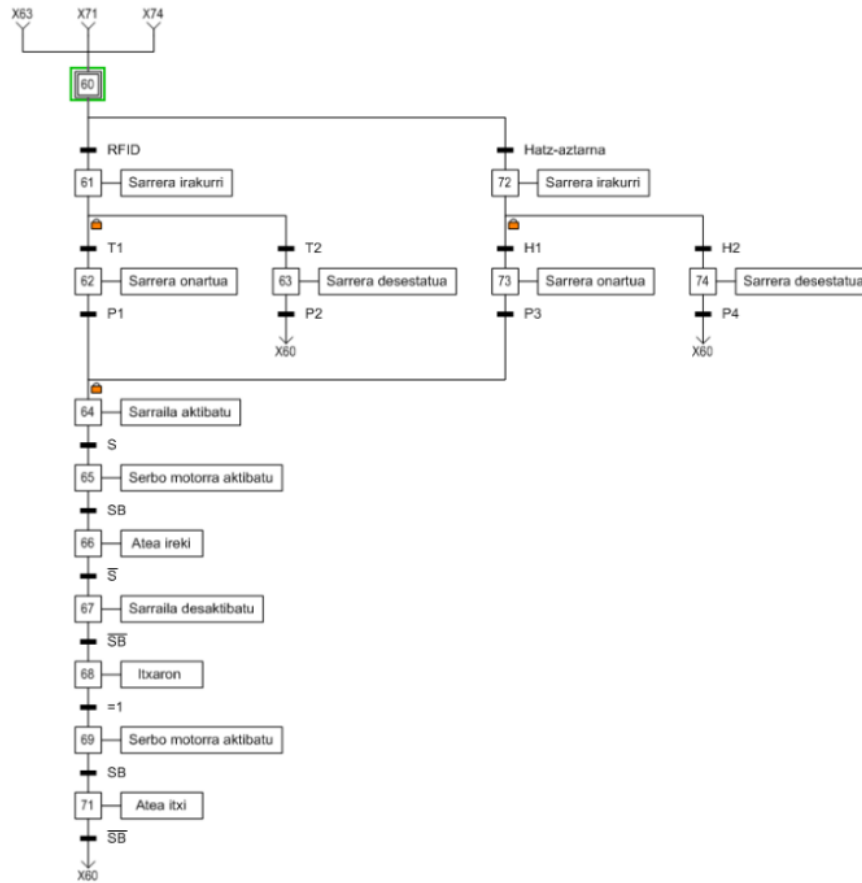
- LOGELA:





9.1 Irudia: Logelaren Grafcet-a

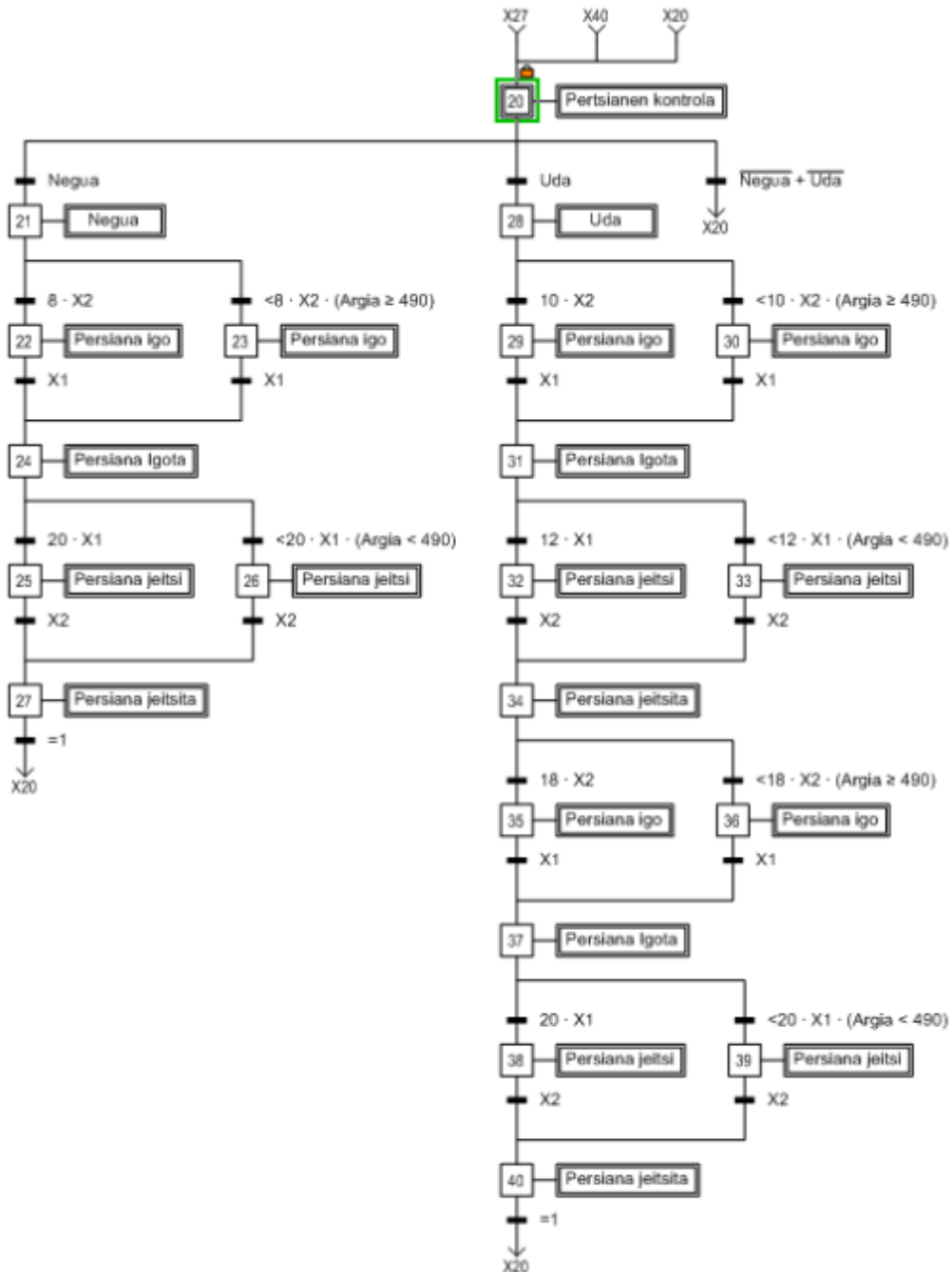
- **SARRERA:**



9.2 Irudia: Sarreraren Grafcet-a



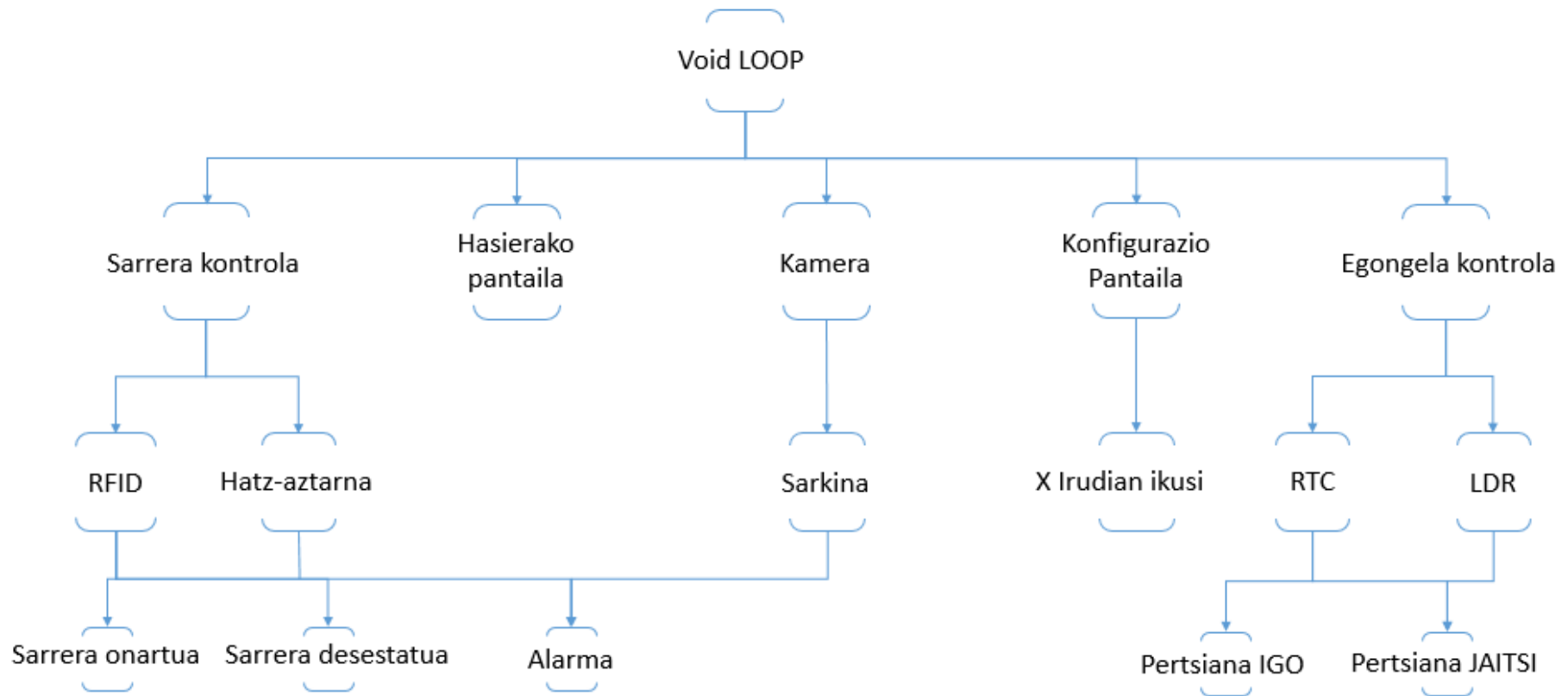
- EGONGELA



9.3 Irudia: Egongelaren Grafcet-a



9.2.2. PROGRAMA NAGUSIA



9.4 Irudia: Liburutegien instalakuntza



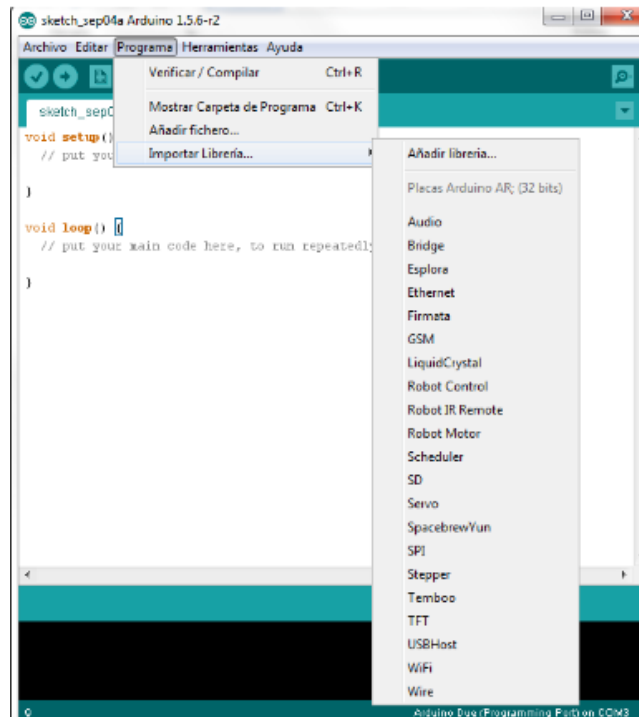
9.3. LIBURUTEGIAK

9.3.1. LIBURUTEGIEN ESANAHIA

Liburutegiak, programen funtzionaltasuna handitzea ahalbidetzen duten C edo C++ hizkuntzetan idatzitako artxiboak dira. Arduino-rako egindako programa batetik deituko den kode fitxategi bat da, zeregin zehatz bat egiteko deituko dena. Artxibo hauen helburu nagusia programatzailearen lana erraztea da, hau da, funtzio konplexuagoak sortzeko orduan, kode guztia idatzi behar ez izatea.

9.3.2. LIBURUTEGIEN INSTALAZIOA

Programa baten barnean liburutegi bat erabiltzeko, Programa > Importar Librería agindua jarraitu behar da.



9.5 Irudia: Liburutegien instalakuntza

Existitzen den liburutegi bat erabili nahi bada, Programa leihoan klik egin eta bertan Importar Librería aukeratu behar da. Bertan hainbat liburutegi egongo dira aukeratu ahal izateko. Honek # include zuzendaritza sartuko du liburutegi karpeta (.) goiburua duen



fitxategi bakoitzean. Zuzendaritza honek, liburutegian definitutako konstante eta funtzioak programan egon daitezten ahalbidetzen du.

Horrez gain, Arduino plakan kargatu edo konpilaketa burutuko den momentuan, liburutegiko kodearen eta programaren bateratzea agintzen du. Posiblea da liburutegiak sortzea edo hirugarren pertsona batek sorturiko liburutegiak erabiltzea. Proiektu honetan hori gertatzen da, etxe domotikoa kontrolatzeko erabiliko diren sentsoze batzuk liburutegi berezi bat behar du. Liburutegi hori lortzeko lehenik eta behin deskargatu egin behar da eta ondoren deskonprimatu. Ondoren ezaguna den karpeta batean izango dugu. Orokorrean bi artxibo daude, bata “.h” eta bestea “.cpp”. Ondoren Arduino karpeta irekiko da. Behin karpeta hau irekita dagoela, liburutegi karpetan sartu eta beharrezko liburutegia bertan itsatsiko da. Arduino IDE berriz abiaraziko da eta liburutegia kargatu dela ikus daiteke.

Programa batek liburutegi baten beharrik ez duenean, goiburuko `#include` zuzendaritza ezabatu eta liburutegia erabiltzeari utziko dio. Honen bidez liburutegiaren eta programaren bateratzea saihestuko da eta Arduino plakan erabilitako tokia murriztuko da.

9.3.3. LIBURUTEGIEN ERAIKUNTZA

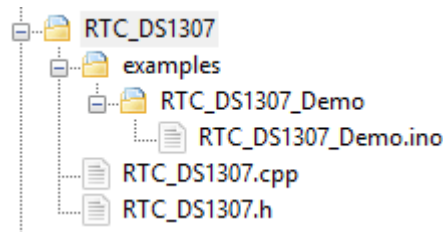
Arduinon bi motatako liburutegiak bereiztu behar dira:

1. Funtzio liburutegiak: liburutegi klasikoak, non fitxategi (.h) batean funtzioen prototipoak adierazten dira eta beste fitxategi batean (.cpp) funtzio horiek definitzen dira.
2. Klase liburutegiak: objektuak adierazteko liburutegiak dira, non fitxategi batean (.h) klasearen egitura adierazten da (datu eta metodo publiko eta pribatuak) eta beste fitxategi batean (.cpp) metodo horiek definitzen dira.

Atal honetan ez da funtzio liburutegiei buruzko programazio azalpenik emango, hori 1. mailako “Informatikaren oinarriak” ikasgaietan, programazio inperatiboa irakasten den ikasgaietan, alegia, ikasten baita. Aldiz, dokumentazio honetan klase liburutegiak zelan programatzen diren azaldu beharra dago, graduan ez baita objektuetara bideratutako programazioa ikasten.

KLASE LIBURUTEGIAK

Klase liburutegiak zelan programatzen diren azaltzeko, proiektuan programatutako liburutegi bate erabiliko da adibide bezela, DS1307 denbora errealeko erlojua kudeatzeko programatu den liburutegia, alegia. Liburutegi honek ondorengo egitura du:



9.6 Irudia: Liburutegien instalakuntza

- RTC_DS1307.h fitxategiak klasearen egitura adierazten du.
- RTC_DS1307.cpp fitxategiak klasean adierazita dauden metodoak definitzen ditu.
- RTC_DS1307_Demo.ino fitxategiak klasearen erabilera eredu adierazten du.

RTC_DS1307.h fitxategia

Irudian ikus daitekeen bezela, goiko mailan, fitxategia lau atalez osoturik dago:

1. **#ifndef RTC_DS1037_h** eta **#define RTC_DS1037_h** kode lerroek liburutegia behin bakarrik kargatu daitela bermatzen dute, liburutegia hainbat lekutan inportatu egiten bada ere.
2. **#include "Arduino.h"** kode lerroak Arduino kode muina inportatu egiten du, Arduino plataforman definitutako API-a (Application Programming Interface) erabiltzeilearentzako eskuragarri jarriaz.
3. **#define DS1307_I2C_ADDRESS B1101000** kode lerroak makro bat sortzeko balio du, hau da, balio jakin batekin erlazionatu nahi den string edo kate jakin bat; kasu honetan, **DS1307_I2C_ADDRESS** makroa liburutegian edo programa nagusian jartzen den bakoitzean, konpiladoreak **B1101000** balioarengatik ordezkaturiko du.
4. Klasearen definizioa. Klaseak bi alderdi ditu: publikoa eta pribatua.
 - a. Alderdi publikoa programa nagusian sortzen den objektu batek deitu ditzakeen datu eta metodoek osotzen dute. Metodo horien artean, objektua sortzerakoan klasea instantziatzeko beharrezkoa den eraikitzailea agertu behar da; kasu honetan: **RTC_DS1307()**
 - b. Alderdi pribatua RTC_DS1307.cpp fitxategian metodoak definitzerako orduan liburutegiaren barne erabilpenerako datuak edo metodoak adierazten ditu; kasu honetan: **bdc2dec()** eta **dec2bcd()**



```
#ifndef RTC_DS1037_h
#define RTC_DS1037_h

#include "Arduino.h"

#define DS1307_I2C_ADDRESS 0x68

class RTC_DS1307 {
public:
    RTC_DS1307();
    void getTime(byte* seconds, byte* minutes, byte* hours,
                byte* day,
                byte* date, byte* month, byte* year);
    void setTime(byte seconds, byte minutes, byte hours,
                byte day,
                byte date, byte month, byte year);
private:
    byte bcd2dec(byte bcd);
    byte dec2bcd(byte dec);
};
#endif
```

9.7 Irudia: Alderdi pribatua

RTC_DS1307.cpp fitxategia

Irudian ikus daitekeen bezela, fitxategi honek goiburu fitxategian (.h) adierazitako metodoak definitzen ditu, horretarako beharrezkoak diren liburutegiak inportatzen direlarik. Kasu honetan, objektu eraikitzailea hutsik dago, objektua sortzen denean ez baita daturik abiarazi behar; bestetik, **setTime()** metodo publikoak (programa nagusian erlojutik I²C protokolo bitartez denbora irakurtzeko erabiliko duguna) **dec2bcd()** metodo pribatua erabiltzen duela ikus daiteke.



```
#include "RTC_DS1307.h"
#include "Wire.h"

RTC_DS1307::RTC_DS1307() {
}

void RTC_DS1307::setTime(byte seconds, byte minutes, byte hours,
                          byte day,
                          byte date, byte month, byte year) {
  Wire.beginTransmission(DS1307_I2C_ADDRESS);
  Wire.write(0x00);
  Wire.write(dec2bcd(seconds));
  Wire.write(dec2bcd(minutes));
  Wire.write(dec2bcd(hours) & 0x3F);
  Wire.write(dec2bcd(day));
  Wire.write(dec2bcd(date));
  Wire.write(dec2bcd(month));
  Wire.write(dec2bcd(year));
  Wire.endTransmission();
}

byte RTC_DS1307::dec2bcd(byte dec) {
  byte units = dec%10;
  byte tens = (dec/10) << 4;

  return tens|units;
}
```

9.8 Irudia: Fitxategia

RTC_DS1307_Demo.ino

Fitxategia honek klasearen erabilera erakusten duen Arduino sketch edo programa nagusi baten eredua da, hau da, liburutegia inportatu (#include), objektu bat sortu (klasea instantziatu) eta, objektu horren inguruan, klasean definitutako metodoen erabilera erakusten duena.

```
#include <RTC_DS1307.h>
#include "Wire.h"

RTC_DS1307 rtc; // objektuaren sorrera
                // klasea instantziatu

void setup() {
  Serial.begin(115200);

  Wire.begin();

  rtc.setTime(50,10,15,5,22,6,79);
}
```

9.9 Irudia: Programa nagusia



9.4. DISPOSITIBOEN PROGRAMAZIOA

9.4.1. PIR

-Arduinoren kudeaketa:

KANPO ETENDURA:

Demagun sentsore baten seinale bat 1-ean jartzen denean detektatu nahi dugula. Zelan programatu dezakegu?

➤ ADIBIDEA:

```
Void setup () {  
    Serial.begin (9600);  
    pinMode (2, INPUT); // Demagun sentsorea Arduinoren 2. pinera konektatu dugula  
}  
Void loop () {  
    if(digitalRead(2)) {  
        Serial.println("Seinalea 1-ean jarri da");  
    }  
}
```

Programazio honek ekarriko dituen desabantailak azalduko dira:

- Gure programa aldaketa detektatzeaz arduratu behar da. Zer gertatzen da aldaketa *detekzioa egiteaz arduratzen den kodea exekutatzen ez denean* jazotzen bada?

```
void loop() {  
    KODEA // 2.- Demagun aldiune honetan seinalea 0-ra bueltatzen dela  
    if(digitalRead(2)) { // 3.- honek ez luke ezer detektatuko !!!  
        Serial.println("Seinalea 1-ean jarri da");  
        KODE GEHIAGO // 1.- Demagun aldiune honetan seinalea 1-ean jartzen dela  
    }  
}
```

- Ezin dira igoera/jaitsiera saihestak detektatu: RISING, FALLING.
- Arazo hauen adibide adierazgarria:



- Zelan kalkulatu PWM baten duty-cycle edo frekuentzia???

Orain arazoari konponbide bat azalduko da:

Kanpo-etendura → attachInterrupt()

```
Void setup () {  
  Serial.begin(9600);  
  pinMode(2, INPUT); //demagun sentsorea Arduinoren 2. pinera konektatu dugula  
  attachInterrupt(0, zerotik_batera, RISING); // int.0 --> pin 2  
}  
void loop() {  
  KODEA  
  KODE GEHIGO // Demagun aldiune honetan seinalea 1-ean jartzen dela.  
               // Programaren exekuzioa hemen gelditzen da eta ISR-ra salto  
               // egiten da.  
               // ISR-a exekutatzen amaitzen denean, puntu honetara itzultzen da  
               // programaren exekuzioarekin jarraitzeko.  
}  
// ISR (Interrupt Service Routine – Etendura Zerbitzu Errutina)  
Void zerotik_batera () {  
  Serial.println (“Seinalea 1-ean jarri da”);  
}
```

➤ **GURE KASUA:**

PIR pina sarrera bezala konfiguratu kanpo-etendura pin batean (18. pina). Hori Arduinon adierazteko oso erreza da:

```
pinMode(PIR, INPUT); // PIR pinean eta Arduinoren Pull-Up barne erresistentzia aktibatu
```



Kanpo-etendura pin horri etenduraren zerbitzu-errutina (ISR -Interrupt Service Routine-) esleitzen zaio:

```
attachInterrupt(5, argazkia_atera, FALLING); // int.5 --> pin 18
```

Lehenengo parametroa, etenduraren zenbakia izango da(5) .

Bigarren parametroa, etendura gertatzen denean deituko den ISR-a.

Azkenengo parametroa, etendura noiz aktibatu behar den definitzen du. Kasu honetan, FALLING erabiliko da HIGH egoeratik LOW egoerara doalako.

Azkenik, etendura berriro bere hasierako egoerara itzultzeko:

```
detachInterrupt(5); // int.5 --> pin 18
```

Idatzi behar dugu etendura desaktibatuzko. Gero egin beharreko aginduak eta hori bukatutakoan, berriro etendura aktibatu.

```
attachInterrupt(5, argazkia_atera, FALLING); // int.5 --> pin 18
```

9.4.2. KAMERA

-Arduinoren kudeaketa:

Hasteko kamerari dagozkion liburutegiak kargatu behar ditugu arduinon:

```
#include <LinkSpriteCamera.h> // kamera liburutegia
```

```
#include "LSC_funtzioak.h" // funtzio sinpleak dituen liburutegia
```

Behin liburutegia kargatuta kamerari serie portu bat definitu behar zaio:

```
LinkSpriteCamera camera(&Serial3); // gure kasuan 3 serie portua
```



“LSC_funtzioak.h” guk sortutako liburutegi bat da. Barruan hainbat funtzio simple daude kameraren liburutegi originala eredu hartuta:

Lehendabizi kamera abiarazi behar da.

LSC_abiarazi(&camera); // kamera abiarazi

```
void LSC_abiarazi(LinkSpriteCamera* camera) {
    Serial.println("CAMERA: Begin");
    camera->begin(38400); // kamera abiarazi
    Serial.print("CAMERA: Reset");
    camera->reset(); // kamera berrabiarazi
}
```

PIR sentsorea seinalea edo pantailaren “argazki” botoia sakatzerakoan erabilitako funtzioa:

LSC_argazkia_atera(&camera, &rtc); // argazkia ateratzeko funtzioa ordua gordez

```
void LSC_argazkia_atera(LinkSpriteCamera* camera, RTC_DS1307* rtc) {
    Serial.print("CAMERA: Take Picture");
    camera->takePicture(); // argazkia atera

    byte seconds, minutes, hours, day, date, month, year;
    rtc->getTime(&seconds, &minutes, &hours, &day, &date, &month, &year); // data irakurri
    String folder_name = String(year) + String("_") + String(month) + String("_") +
    String(date); //urtea, hilabetea eta eguna gorde
    String file_name = String(hours) + String("_") + String(minutes) + String("_") +
    String(seconds) + String(".jpg"); // odua, minutuak eta segunduak gorde
    Serial.print(folder_name); Serial.println(" "); Serial.println(file_name);

    if(!SD.exists(folder_name)){
        SD.mkdir(folder_name); // SD txartelean gorde
    }
}
```



```
File dataFile = SD.open(folder_name+String("/") + file_name, FILE_WRITE);

Serial.print("CAMERA: Read File Size");
unsigned int fileSize = camera->readFileSize(); // irakurri karpetaren tamaina
Serial.print(fileSize); Serial.println();

Serial.println("CAMERA: Read File Content");
byte chunk[CHUNK_SIZE];
unsigned int fileAddress = 0;
while(fileAddress < fileSize){
int cont = camera->readFileContent(chunk, fileAddress, CHUNK_SIZE); // edukia irakurri
    for(int i=0; i<cont; i++) {
        dataFile.write(chunk[i]);
    }
    fileAddress+=CHUNK_SIZE;
    Serial.print(".");
}
dataFile.close(); // datu karpeta utzi
Serial.println("");

Serial.print("CAMERA: Stop Taking Pictures");
camera->stopTakingPictures(); // utzi argazkiak ateratzeari
}
```

8.4.3. TFT PANTAILA

-Arduinoren kudeaketa:

Arduinon programatzerako orduan, hainbat liburutegi kargatu behar ditugu:

SPI komunikazio erabiltzen duenez bere liburutegia gehitu behar da:



```
#include <SPI.h>
```

TFT pantailaren liburutegi orokorrak irudiak adierazteko:

```
#include <Adafruit_GFX.h>
```

```
#include <Adafruit_ILI9341.h>
```

TFT pantailaren ukipen liburutegiak:

```
#include <SeeedTouchScreen.h>
```

Bukatzeko guk egindako ukipen eta adierazpen liburutegiak:

```
#include "TS_funtzioak.h"
```

```
#include "TFT_funtzioak.h"
```

Eman beharreko hurrengo pausua, definitu behar diren balioak definitu:

```
#define TFT_DC 6 // Adafruit_ILI9341 tft
```

```
#define TFT_CS 5 // Adafruit_ILI9341 tft
```

```
#define YP A2 // TouchScreen ts
```

```
#define XM A1 // TouchScreen ts
```

```
#define YM 54 // TouchScreen ts
```

```
#define XP 57 // TouchScreen ts
```

```
Adafruit_ILI9341 tft = Adafruit_ILI9341(TFT_CS, TFT_DC);
```

```
TouchScreen ts = TouchScreen(XP, YP, XM, YM);
```

TFT pantailan erabilitako operazioak:

9.1. Tabla: TFT operazioak

Operazioak	Deskribapena
Tft.fillScreen(ILI9341_COLOR);	Pantaila osoa kolore batez irudikatu
Tft.fillRect(X1, X2, Y1, Y2, ILI9341_COLOR);	Laukizuzen bat kolore batekin irudikatu
Tft.fillCircle(X,Y,Ø,(ILI9341_COLOR));	Zirkunferentzia bat kolore batekin irudikatu
Tft.setCursor(X,Y);	Ardatza puntu horretan zehaztu
Tft.setTextSize(N);	Testuaren tamaina
Tft.setTextColor(ILI9341_COLOR);	Kolore bat aukeratu testuarentzako
Tft.println("Testua");	Idatzi nahi den testua adierazi
Tft.setRotation(α);	Pantaila biratu



TFT pantaila-ren funtzioak azaltzeko, ondorengo taularekin adieraziko da:

9.2. Tabla: TFT funtzioak

Funtzioak	Deskribapena
TFT_abiarazi(&tft);	Dispositiboa abiarazteko
TFT_sarrera_baliogabea(tft);	Etxea jasotako sarrera ezezaguna da
TFT_sarrera_baliozkoa(tft, data);	Etxea jasotako sarrera ezaguna da
TFT_teklatua_marraztu(tft);	9 zenbaki dauzkan teklatua adierazi. Zenbaki bakoitza botoi bat izango da
TFT_zenbakia_berrestu();	Tekleatu den zenbakia adierazi
TFT_mezua();	Edozein mezu idazteko. Egoerak adierazteko pentsatua
TFT_konf_etxea();	Hasierako pantaila : SARRERA, LOGELA eta EGONGELA botoiak
TFT_sarrera();	Sarrera barruan dauden aukerak: HATZ, RFID, ATEA, PASS eta ITZULI botoiak
TFT_sarrera_RFID();	RFID barruan dauden aukerak: ID eta ITZULI
TFT_sarrera_FPS();	HATZ barruan dauden aukerak: BERRIA, ID , EZABATU eta ITZULI botoiak
TFT_ATEA();	ATEA barruan dauden aukerak: IREKI, ITXI eta ITZULI botoiak
TFT_sarrera_PASAHITZA();	PASAHITZA barruan dauden aukerak: BERRIA, ID eta ITZULI botoiak
TFT_egongela ();	Egongela barruan dauden aukerak: IGO, JAITSI eta ITZULI botoiak
TFT_logela();	Logela barruan dauden aukerak: ARGAZKIA eta ITZULI botoiak
TFT_itzuli();	Aurreko pantailara itzultzeko botoia
TFT_alarma();	Irudi baten bitartez alarma adieraziko da
TS_teklatu_kontrola();	Teklatuaren zein zenbaki ukitu den jakiteko
TS_tekla_gorde();	Ukitu den zenbakia gordetzeko
TS_Sarrera_Kontrola();	RFID , HATZ, ATEA edo PASAHITZA pultsadoreak ukitzeko aukera
TS_sarrera_RFID_Kontrola ();	ID eta ITZULI pultsadoreak ukitzeko aukera
TS_sarrera_FPS_Kontrola ();	BERRIA, ID , EZABATU eta ITZULI pultsadoreak ukitzeko aukera
TS_sarrera_Atea_Kontrola ();	IREKI eta ITXI pultsadoreak ukitzeko aukera
TS_Pasahitza_Kontrola ();	BERRIA, ID eta ITZULI aukerak ukitzeko aukera
TS_Egongela_Kontrola ();	IGO, JAITSI eta ITZULI pultsadoreak ukitzeko aukera

TS operazio erabiliak:

Lehenik irudikatu diren objektuekin lotura eduki behar duen objektu "irudikaria" zehaztu behar dugu, hau da, irudikatu dugun objektua gure atzamarrarekin ukituz gero pantaila ukipena detektatu behar du eta erantzun bat emango digu . Hurrengo adibidearekin hobeto ulertuko da:

Tft.fillRect(35,35,85,65,ILI9341_BLUE); // Laukizuzen hau edukita zelan zehaztu dezakegu laukizuzen "irudikaria" bat?



```
Point p = ts.getPoint(); // Ukipen puntuak gordetzeko  
p.x = map(p.x, TS_MINX, TS_MAXX, 0, 240); // Pantailaren X ardatzaren ukipen max  
p.y = map(p.y, TS_MINY, TS_MAXY, 0, 320); // Pantailaren Y ardatzaren ukipen max
```

```
if(p.x>140 && p.x< 205 && p.y>35 && p.y<120){} // Tarte horretan ukituz behar da
```

Puntuaren azalpena:

$p.x = 240 \text{ (max)} - 100(35+65) = 140$

$p.x = 240 \text{ (max)} - 35 = 205$

$p.y = 0 \text{ (min)} + 35 = 35$

$p.y = 0 \text{ (min)} + 120(35+85) = 120$

Tarte honetan ukitzen bada, dagokion aginduan exekutatu da.

9.4.4. RTC

-Arduinoren kudeaketa:

```
#include <RTC_DS1307.h> // RTC-aren liburutegia gehitu behar da.
```

```
RTC_DS1307 rtc;
```

RTC-aren funtzioak azaltzeko, ondorengo taularekin adieraziko da:

9.3. Tabla: RTC funtzioak

Funtzioak	Deskribapena
<code>rtc.setTime(seconds,minutes,hours,day,date,month,year);</code>	Dispositiboa abiarazteko
<code>rtc.getTime(&seconds, &minutes, &hours, &day, &date, &month, &year);</code>	Txartela irakurri eta begiratu erregistratua dagoen



Bestetik, data RTC-an ezartzeko bide jarrai daitezke: Alde batetik, data programan bertan kodifikatuta egon daiteke ("HARDCORE") edo erabiltzaileari sarrera estandarra erabiliz eskatu diezaioke.

8.4.5. LDR

-Arduinoren kudeaketa:

Konponente simple bat denez, ez da liburutegi bat gehitu behar. Arduino pin analogiko bat erreserbatuko dio LDR-ari. LDR-a nahi dugun tarteko balioekin funtziona dadin hurrengo kodea jarri behar da:

```
int LDR=14; // A14 pin analogikoa
int val=0; // LDR emandako balioa gordeko duen aldagaia
int final =0; // Ezarritako tartera heldu ostean geratu den emaitzaren aldagaia
```

Arduinoren bihurgailu analogiko digitala erabili behar da LDR-aren funtzionamendu zuzena izateko. Horretarako analogRead(pina); programan idatzi behar da.

8.4.6. HATZ-AZTARNA DETEKTAGAILUA

-Arduinoren kudeaketa:

```
#include <FPS_GT511C3.h> //berezko liburutegia
#include "FPS_funtzioak.h" //funtzioen liburutegia
```

```
FPS_GT511C3 fps(&Serial2); //serial2 erabiliko da komunikaziorako
```

Hatz-aztarnaren funtzioak azaltzeko, ondorengo taularekin adieraziko da:



9.4. Tabla: RTC funtzioak

Funtzioak	Describapena
FPS_abiarazi(nfc);	Dispositiboa abiarazteko
FPS_irakurri(nfc, data_aux);	Hatz berri bat erregistratu
FPS_Berria (tft, fps);	Hatza identifikatu
FPS_ID (tft, fps, f_printed_konf_sarrera);	Hatza identifikatu
FPS_Ezabatu (tft, fps, f_printed_konf_sarrera);	Hatza ezabatu

FPS_Berria (tft,fps); funtzioa dispositibo honen funtzio garrantzitsuena da. Funtzio honetan “deskribapen zehatza”-ren atalean azaltzen den prozesua berdina erabiltzen da.

8.4.7. RFID

Arduinoren kudeaketa:

```
#include <Adafruit_PN532.h> //berezko liburutegia
```

```
#include "RFID_funtzioak.h" //funtzioen liburutegia
```

```
#define PN532_IRQ 2 //arduinoren 2 pinean konektatu
```

```
#define PN532_RESET 3 // arduinoren 3 pinean konektatu
```

RFID-aren funtzioak azaltzeko, ondorengo taularekin adieraziko da:

Funtzioak	Deskribapena
RFID_abiarazi(nfc);	Dispositiboa abiarazteko
RFID_irakurri(nfc, data_aux);	Txartela irakurri eta begiratu erregistratua dagoen



8.4.8. SARRAILA

Arduinoren kudeaketa:

```
#define SARRAILA 22 // sarraila arduinoren 22 pinan zehaztu
```

```
digitalWrite(SARRAILA, HIGH); //sarraila aktibatu
```

```
digitalWrite(SARRAILA, LOW); //sarraila desaktibatu
```

8.4.9. SERBOMOTORRA

-Arduinoren kudeaketa:

Atearekin lehenengo probak egin zirenean, atea azkarregi itxi eta irekitzen zela nabaritu zen, hau serbomotorraren funtzionamendu azkarrak erangindako ondorioa delarik. Arazo hau konpondu nahian, software bitartez serbomotorra abiadura moteltzeko modurik ote dagoen aztertu zen. Azterketa honen emaitzak VarSpeedServo liburutegia erabiltzera eraman gintuen.

VarSpeedServo liburutegia gehitu eta serbomotorraren pina zehaztu behar da. Gure kasuan posizioa eta abiadura kontrolatu nahi da. Programan jarri behar den kodea :

```
#include <VarSpeedServo.h> //serbomotorraren liburutegia gehitu
```

```
#define SERBO 24 //serbomotorraren pina definitu
```

```
VarSpeedServo miServo; //serbomotorraren aldagai definitu
```

```
VOID SETUP () {
```

```
miServo.attach(24, 650, 2400); // 24. Pina eta bere parametroak
```

```
miServo.write(74); //hasierako posizioa
```

```
VOID LOOP () {
```

```
//Sarrera baliioduna bada
```



```
digitalWrite(22, HIGH); //sarraila aktibatu
miServo.slowmove(145,40); // serbomotorra posizio eta abiadura zehatz batera joango da
delay(500); //denbora tarte txiki bat serbomotorrari denbora emateko atearen segurtasunetik
irteteko
digitalWrite(22, LOW); //sarraila desaktibatu
//Denbora tarte itxaron
delay(5000); // denbora tarte luze bat etxera sartu den pertsonari denbora emateko etxera
sartzera
miServo.slowmove(74,82); // serbomotorra hasierako posiziora itzultzeko agindua
```

//Ireki botoia

```
digitalWrite(22, HIGH); //sarraila aktibatu
miServo.slowmove(145,40); // serbomotorra posizio eta abiadura zehatz batera joango da
delay(500); //denbora tarte txiki bat serbomotorrari denbora emateko atearen segurtasunetik
irteteko
digitalWrite(22, LOW); //sarraila desaktibatu
//Itxi botoia
miServo.slowmove(74,82); // serbomotorra hasierako posiziora itzultzeko agindua
```

8.4.10. MOTORRA

-Arduinoren kudeaketa:

```
#define MOTORRA_Enable 8 // motorraren seinale pina definitu
#define MOTORRA_A 40 // motorraren sentzu pina definitu
#define MOTORRA_B 42 // motorraren sentzu pina definitu
// Igotzeko pertsiana
digitalWrite (MOTORRA_A, HIGH); // motorra sentzu batean biratzeko
digitalWrite (MOTORRA_B, LOW); // motorra sentzu batean biratzeko
analogWrite (MOTORRA_Enable, 130); //PWM seinalea motorrari bidali
delay (150);
```



```
analogWrite (MOTORRA_Enable, 110); //PWM seinalea motorrari bidali
delay (250);
analogWrite (MOTORRA_Enable, 90); //PWM seinalea motorrari bidali
delay (510);
analogWrite (MOTORRA_Enable, 82); //PWM seinalea motorrari bidali
delay (410);
analogWrite (MOTORRA_Enable, 0); //PWM seinalea motorrari bidali
// Jaisteko pertsiana
digitalWrite (MOTORRA_A, LOW); // motorra beste sentzuan biratzeko
digitalWrite (MOTORRA_B, HIGH); // motorra beste sentzuan biratzeko
analogWrite(MOTORRA_Enable, 80); //PWM seinalea motorrari bidali
delay (100);
analogWrite(MOTORRA_Enable, 75); //PWM seinalea motorrari bidali
delay(200);
analogWrite (MOTORRA_Enable, 70); //PWM seinalea motorrari bidali
delay (330);
analogWrite (MOTORRA_Enable, 0); //PWM seinalea motorrari bidali
```

8.4.11. SD

Arduinoren kudeaketa:

```
#include <SD.h> //MICROSD txartelaren liburutegia
#define SD_CS 4 // Arduinoren 4. Pina erabili
if (SD.begin(SD_CS)) { // egiaztatzeko SD-a aurkitu den ala ez
  Serial.println("SD abiarazi da");
} else {
  Serial.println("SD card initialization failed");
  while(1);
}
```




10. BIBLIOGRAFIA

10.1. KONTSULTATUTAKO WEB ORRIAK

- Arduino:

<http://www.arduino.cc/es/>

<http://es.wikipedia.org/wiki/Arduino>

- Arduino Mega 2560:

http://store.arduino.cc/eu/index.php?main_page=product_info&cPath=11&products_id=196

- Arduino Uno:

http://store.arduino.cc/eu/index.php?main_page=product_info&cPath=11&products_id=195

- Raspberry Pi:

<http://www.raspberrypi.org/faqs>

http://es.wikipedia.org/wiki/Raspberry_Pi

- PIR PULL-UP RESISTENCIAS

<http://ovtoaster.com/resistencias-pulldown-y-pullup/>

- Interrupzioak arduinon

<https://www.arduino.cc/en/Reference/DetachInterrupt>

<https://www.arduino.cc/en/Reference/AttachInterrupt>



- LinkSprite camera

<https://www.sparkfun.com/products/retired/10061>

- Fritzing:

<http://fritzing.org/>

- RTC datasheet

<https://datasheets.maximintegrated.com/en/ds/DS1307.pdf>

- LDR

<https://es.wikipedia.org/wiki/Fotorresistor>

- Motorra

https://es.wikipedia.org/wiki/Motor_el%C3%A9ctrico

- Serbo

<https://es.wikipedia.org/wiki/Servomotor>

- TFT pantaila

<http://www.promotec.net/panel-tactil-tfts/>

- RFID datasheet

<http://category.alldatasheet.com/index.jsp?components=RFID>

<https://es.wikipedia.org/wiki/RFID>