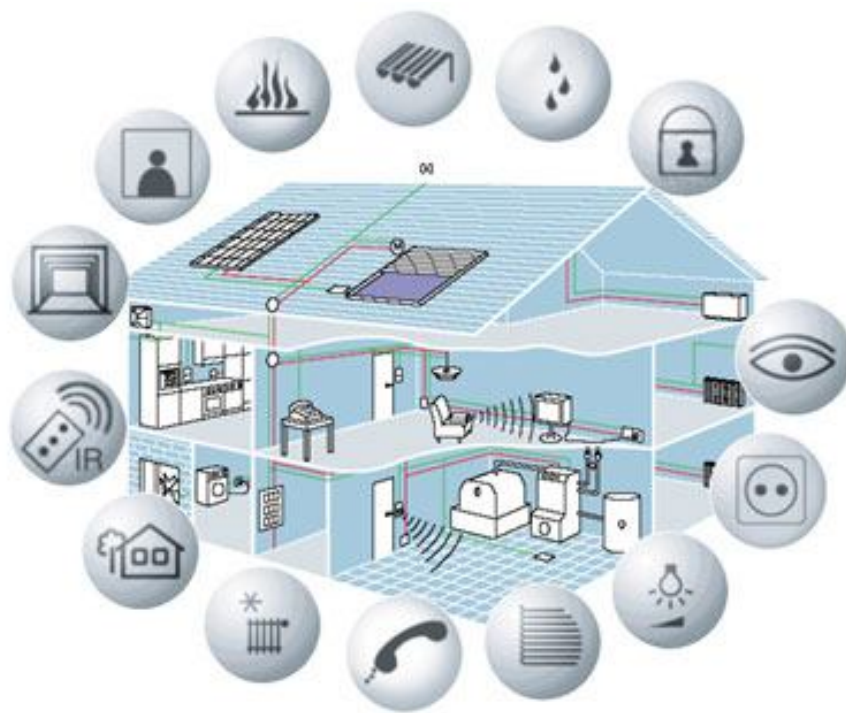


Domòtica per tothom: Configura l'automatització dels elements de la llar



Treball de recerca en l'àmbit d'investigació tecnològica

Autor: Carles Llobet Pons

Curs acadèmic: 2º Batxillerat A



Índex

1. Introducció.....	4
1.1 Introducció a la domòtica	4
1.2 Objectius	4
1.2.1 Aprenentatge	4
1.2.2 Guia	5
1.2.3 Com ho farem?	5
1.2.4 Què necessito?	6
1.2.5 Què necessitarà el lector per automatitzar la seva llar?	6
2. Història del projecte.....	7
2.1 Transcurs del projecte	7
2.1.1 Idea inicial	7
2.1.2 Hipòtesi	8
2.1.3 Aprenentatge	8
.....	9
2.1.4 Exposició del progrés al tutor.....	10
2.1.5 Reunió amb el tutor.....	10
2.1.6 Entrevista	11
2.1.7 Connexió entre Arduino i Processing	14
2.1.8 Programar actuadors	14
2.1.9 Motors	15
2.1.10 Alarma.....	16
2.1.11 Calibrant els termistors.....	20
.....	21
.....	21
2.2 Problemes i dificultats comuns.....	23
2.2.1 Solucions	23



3. Guia per a la realització	24
3.1 Programació	24
3.1.1 Plànol	24
3.1.2 Botons	25
3.1.3 Funcions dels botons	26
3.2 Connexió a l'Arduino	28
4. Conclusions	30
4.1 Opinió del Treball	30
4.2 Objectius assolits	30
4.3 Possibles millores	31
4.3.1 Ampliació dels elements	31
4.3.1 Ampliació dels controladors	31
5. Annex	32
5.1 Elements utilitzats	32
5.1.1 Per a la maqueta	32
5.1.2 Programes	35
5.1.3 Plànol	36
.....	36
5.2 Com hem connectat els elements	37
5.3 Programa	44
6. Bibliografia	69
6.1 Llibres	70
6.1.1 En format PDF	70
6.1.2 En format paper	70
6.2 Pàgines web	71
6.3 Agraïments als assessors	71



1. Introducció

1.1 Introducció a la domòtica

Una llar amb instal·lació domòtica és un habitacle en el qual els diversos elements de control de la casa, en comptes de tenir comandaments independents, convergeixen en un **control central** que serà l'encarregat de decidir com ha d'actuar cada element de la instal·lació. La manera de **programar** aquest control central dependrà del sistema domòtic que nosaltres fem servir.

N'hi ha de molts tipus; uns són més **intuitius** a l'hora de programar-los i uns altres tenen una mica més de complexitat. També existeixen alguns sistemes que estan més **estandarditzats** i altres que utilitzen un protocol propi o **propietari** (no estàndard). Cap sistema és millor que un altre, però òbviament, sempre és preferible un sistema creat únicament per al teu habitacle, que no un sistema estàndard.

1.2 Objectius

En aquesta memòria crearem un control central de la nostra llar a través d'un programa en Processing, que ens permeti controlar des de la il·luminació de les habitacions, com el sistema antiincendis, fins als sistemes de reg o de control de temperatura de la casa, d'una manera còmoda i senzilla a través de qualsevol sistema capaç d'executar un programa amb Java (des del típic Windows o Mac OS fins a Linux o variants d'aquest)

1.2.1 Aprenentatge

A través d'aquesta guia volem ser capaços d'entendre què és una casa domòtica i com funciona: Saber com funciona l'algorisme del programa, quins dispositius permeten comunicar l'ordinador amb la casa, etc,... per a ser capaços de complir el segon objectiu d'aquest projecte.



1.2.2 Guia

L'Objectiu d'aquest projecte és que tothom pugui tenir a l'abast aquest recurs de la domòtica sense tenir un nivell alt de vida econòmica. Ho faré a través d'una guia amb la qual qualsevol persona podrà transformar la seva llar en un habitacle domòtic perfectament funcional, reduint en gran mesura el preu al fer-ho ell mateix.

Aquests projecte està enfocat sobretot a persones que no poden assumir el cost d'una instal·lació domòtica però que necessitin aquest tipus d'instal·lació ja sigui perquè tenen una discapacitat, o perquè tenen una edat que ja els hi dificulta el moviment. Tot i així qualsevol persona amb ganes pot aconseguir realitzar aquest projecte

1.2.3 Com ho farem?

Realitzarem un programa amb Processing que mostri els plànols d'una casa indicant quins llums estan encesos o apagats, quines persianes estan obertes o tancades, la temperatura de la casa i fins i tot alarmes d'incendi i de seguretat permetent controlar-los; i tot, a través d'una senzilla connexió USB a una placa Arduino.

En el cas del nostre treball de recerca, al no poder fer-ho en una casa real, construirem una maqueta d'una casa model amb la qual representarem el funcionament real d'una casa (adaptant-nos al nostre pressupost, utilitzant elements que ja disposem, la majoria reciclats). Un cop fet això realitzarem una guia amb la qual qualsevol lector pugui adaptar aquest programa per a la seva pròpia llar. Així qualsevol podrà tenir automatitzada la seva casa a preu de cost, només pagant el cablejat de la casa i els motors de les persianes, ja que el programa, el dispositiu de control, i les connexions dels actuadors/sensors al dispositiu de control, les farà ell mateix.



1.2.4 Què necessito?

Coneixements

Per a aquest projecte necessitaré coneixements bàsics d'arquitectura i disseny per a fer els plànols de la casa (la maqueta representativa i el plànol d'aquesta); coneixements avançats de programació en la plataforma gràfica Processing (que treballa amb llenguatge Java) per crear un programa mutable i adaptable a qualsevol lector que segueixi la guia, i coneixements mitjans en electrònica per a connectar-ho tot a l'ordinador correctament i explicar com fer-ho de manera senzilla.

Materials

Aquest projecte té infinitat de possibilitats, i adaptarem la maqueta representativa amb els materials dels que disposem en aquell moment, tant per pressupost com per viabilitat dins d'una maqueta de mida reduïda. Suposem que aquests materials seran: Cartró-ploma, teles per les persianes, cablejat divers, led's, sensors de temperatura i de infrarojos, i d'altres similars. Explicarem els materials que finalment hem pogut utilitzar en els annexos.

1.2.5 Què necessitarà el lector per automatitzar la seva llar?

Coneixements

La guia estarà enfocada a que el lector, si ja té els plànols de la seva llar, no necessiti cap tipus de coneixement a part dels que se li ensenyaran a la Guia de realització.

Materials

Si el lector vol aplicar el projecte a la seva pròpia llar, només necessitarà els actuadors i sensors que vulgui incorporar, una placa Arduino, relés per a les llums, i metres suficients de cablejat. Són tot coses que si no sap com instal·lar, qualsevol lampista pot fer-li-ho a preu de cost, reduint el cost de l'automatització a només la factura d'aquest lampista. A les conclusions, a "objectius assolits" hi haurà una referència del preu que hem aconseguit reduir.



2. Història del projecte

2.1 *Transcurs del projecte*

2.1.1 Idea inicial

El principi d'un treball de recerca és sempre complicat. Tot i tenir una idea aproximadament del que es vol fer, mai saps per on començar. En el meu cas, tenia clar que havia de fer alguna cosa relacionada amb la domòtica, ja que tant l'arquitectura com, sobretot, la programació i la electrònica, m'apassionaven.

A més del fet de que la domòtica és un recurs poc explorat i molt útil, sobretot per a persones minusvàlides, que pot facilitar moltíssim la vida quotidiana del dia a dia. Com no era viable construir una casa domòtica real, vaig decidir fer una maqueta i així poder fer una demostració de la utilitat en qualsevol presentació, en persona.

El problema era que no sabia exactament què podia demostrar, ja que volia que el meu projecte tingués alguna utilitat, servís d'alguna cosa.

El meu tutor em va suggerir fer servir el llenguatge de programació Processing en comptes de JDK, que era massa complex, o Visual Basic, al qual li faltaven varies funcions. Així que a principis d'estiu i a través d'un seguit de llibres en anglès, sobretot de "Learning Processing" de Daniel Shiffman, vaig començar a aprendre aquest llenguatge de programació.



2.1.2 Hipòtesi

Però com enfocar el treball? Quina hipòtesi podia preguntar-me? Que volia descobrir? Doncs la veritat això no ho vaig saber fins ben bé a la meitat del treball, però no em preocupava. Tenia clar el que volia, i sabia que trobaria aquesta hipòtesi.

Finalment vaig decidir que volia fer una guia amb la qual qualsevol persona pogués fer-se la seva pròpia casa domòtica reduint així el cost en casi dos terços del cost original, ja que amb l'actual crisi la domòtica està obsoleta per culpa dels excessius preus. Poc a poc l'idea va anar evolucionant i finalment quan vaig tenir una entrevista amb Roberto Novo, com veurem més tard, i va informar-me dels preus reals d'una instal·lació mitjana vaig decidir finalment que aquesta seria la hipòtesi o recerca del meu treball.

2.1.3 Aprentatge

Al principi no entenia ni un borrall, però després d'uns quants exercicis i pràctiques vaig començar a entendre'n el concepte. Vaig començar a estructurar l'algorisme del programa que faria servir per a la maqueta, i vaig començar a escriure codi amb les funcions que necessitava per crear una finestra amb el plànol inserit, i per a col·locar els botons, dels quals vaig tenir que buscar un a un les coordenades en píxels i escriure-les en una taula de dades a l'Excel:

Posició Botons		x	y	width	height
Botó 1	Persiana Saló	80	130	35	440
Botó 2	Llum Saló	172	298	40	40
Botó 3	Llum Rebedor	333	448	40	40
Botó 4	Llum Menjador	515	420	40	40
Botó 5	Llum Cuina	495	152	40	40
Botó 6	Llums Passadís	860	100	40	40
Botó 7	Llum nens 1	1071	487	40	40
Botó 8	Llum Bany fills	991	486	40	40
Botó 9	Llum nens 2	907	486	40	40
Botó 10	Llum pares	810	486	40	40
Botó 11	Llum Bany pares	679	486	40	40



Un cop vaig tenir la taula d'excel amb les coordenades dels botons al plànol, només havia de crear una variable booleana, ja que els botons només poden estar en dos estats (1 o 0, encès o apagat). Després vam crear unes variables "x1", "y1", "w1" i "h1" per entendre millor el programa i no confondre coordenades d'uns botons amb altres.

Finalment en el "void draw" només faltava definir la funció del botó tal com podem veure a la figura 1.

```
import processing.serial.*; //llibreria Port sèrie
import cc.arduino.*; //llibreria per comunicar amb Arduino

boolean button1= false; //declarar variable booleana del botó

int x1 = 80; //declarar posicions del botó
int y1 = 130;
int w1 = 35;
int h1 = 220;

void draw (){
  if button1(){
    "funció que volem que faci"
  } else{
    "funció mentre no és clicat"
  }
}
```

Figura 1

Amb això ja teníem pràcticament l'estructura base del que vindria a ser molt per sobre el nostre programa, ara només faltaria anar poc a poc expandint, ampliant, i perfeccionant el programa sobre aquesta estructura base fins arribar a complir els objectius que ens havíem marcat prèviament en l'apartat de la introducció.



2.1.4 Exposició del progrés al tutor

Un cop vaig aconseguir obrir el plànol i posar-hi botons just a on tocava, per seguir avançant necessitava concretar quina placa faria servir. Vaig posar-me en contacte amb el meu tutor, explicant que havia fet fins al moment, i quina placa podria fer servir. Va dir-me que busques informació sobre la placa Arduino UNO, que era la que segurament faríem servir, i vam concertar una cita una setmana abans de començar el curs.

A més de buscar informació sobre la placa esmentada, vaig buscar informació general sobre la domòtica, i vaig trobar una cosa ben interessant. Al buscar informació per internet sobre les cases domòtiques, només trobava pàgines on donaven opció a calcular un pressupost o explicacions de què era una casa domòtica, però no trobava ni preus ni explicacions de com es podia automatitzar una casa pel teu propi compte. Aquí va ser quan vaig començar a intuir cap a on podia enfocar el meu treball ja que si una instal·lació domòtica era de preus elevats, i sobretot ara amb l'actual crisi, la producció d'instal·lacions domòtiques es reduiria enormement, i ens privaria d'una comoditat que qualsevol desitja a la seva llar.

2.1.5 Reunió amb el tutor

Com havíem quedat, una setmana abans de començar les classes, vam tenir una reunió en Joan Alòs, l'Elisabet Alsina i jo. En aquesta, a part d'ensenyar els progressos que ja havia fet, vam estar valorant totes les possibilitats que teníem per poder aplicar les funcions que es farien en una instal·lació domòtica corrent a la nostra maqueta.

Un cop decidides les funcions que faríem, vam fer una llista amb els Actuadors i Sensors que necessitàvem, i les corresponents sortides i entrades de la placa que necessariem. Per no quedar-nos justos, vam decidir substituir la placa que havíem comentat, la Arduino UNO de 14 sortides/entrades, per la Arduino Mega 2560 amb 54 sortides/entrades digitals i 15 analògiques (Fig.2)



Figura 2



2.1.6 Entrevista



Figura 3

Després de la reunió amb el tutor vaig adonar-me que només ens havíem fet una idea per sobre del que faríem, i estava començant a aprendre a programar en Processing sense saber si realment seria factible la idea. A més, encara no sabia tampoc ben bé quin era el propòsit que volia aconseguir a través del meu projecte, per això, vaig buscar algú que em pogués assessorar.

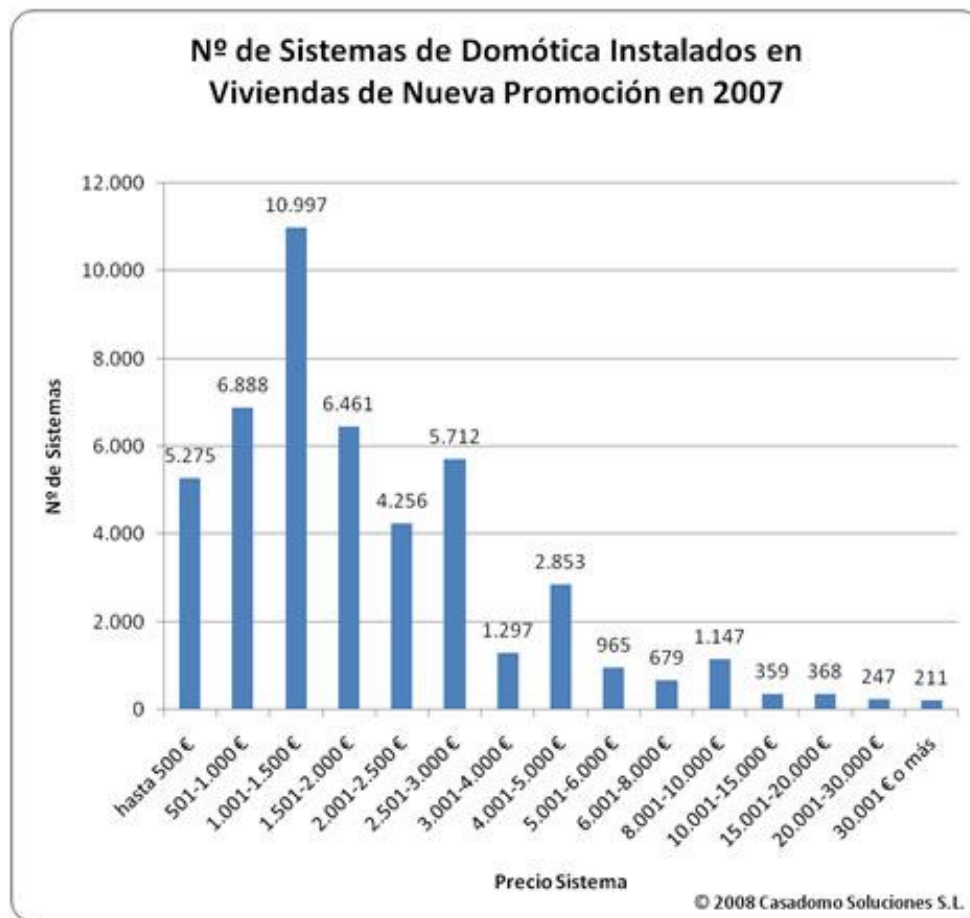
El primer que vaig fer és buscar les empreses més properes a Vilassar de Mar, i vaig decidir fer una visita a algunes d'aquestes. Empreses de Barcelona com Engtel o Equinsa, estaven tancades o no em proporcionaven gaire informació, però finalment vaig trobar el que buscava. Una empresa anomenada I-DO (integració domòtica), va atendre'm amablement tot i estar tancada des de feia uns quants anys.

L'ex-propietari Roberto Novo (Fig.3), enginyer en Telecomunicacions, va accedir a la meua entrevista inclús va deixar-me accedir a la seva pròpia casa per ensenyar-me les instal·lacions que ell mateix havia integrat a casa seva. Quan li vaig preguntar per els preus d'aquestes instal·lacions em va dir que oscil·len molt depenent de què es vulgui instal·lar. El preu d'una casa normal d'uns 120 m² amb funcions simples com il·luminació, temperatura, alarma de seguretat i antiincendis, persianes i sensors de presència oscil·la entre els 4000€ i 5000€ en total, comptant la instal·lació, la mà d'obra, i els controladors i dispositius de la casa. Hi ha instal·lacions molt més cares simplement per la grandària de la casa, o la quantitat d'elements a instal·lar i per funcions com diferents modes personalitzats de il·luminació i temperatura.

Les instal·lacions completes van dels 3000€ cap amunt pel que les instal·lacions més sol·licitades o no son completes o bé no garanteixen fiabilitat i estan fetes amb poca qualitat, fet que posa en risc la seguretat de les nostres vides i que poden produir des de curtcircuits que provoquen incendis fins a accidents propis d'una pel·lícula d'Alfred Hitchcock com deixar-nos tancats a casa sense possibilitat de sortir-ne.

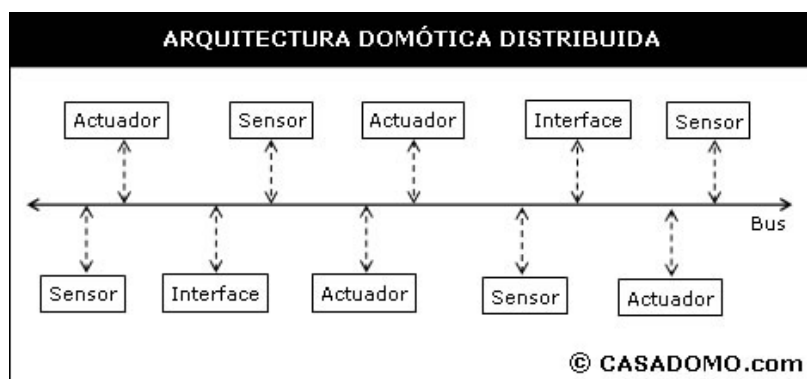


En el següent gràfic extret d'un estudi realitzat al 2008 (4^a pàgina web de l'annex) veiem la quantitat d'instal·lacions a Espanya en funció del preu d'aquestes, es pot observar com a partir dels 3000€, que és quan comencen a considerar-se instal·lacions completes, la quantitat d'habitacles construïts disminueix dràsticament. La majoria d'instal·lacions són d'entre 1000 i 1500 €, són instal·lacions incompletes o massa senzilles.

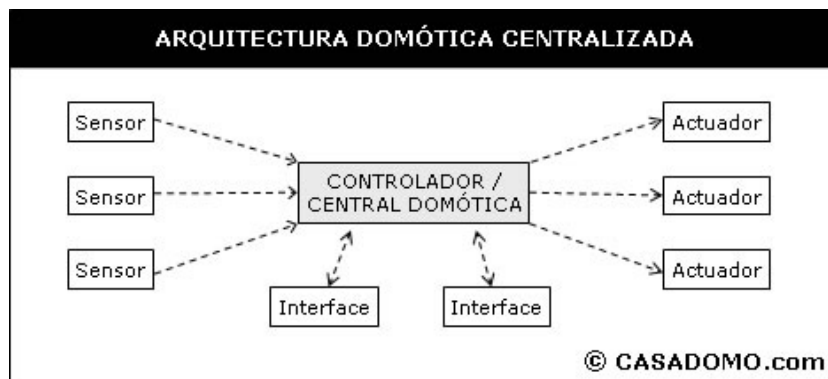


Gràfic 1

També em va explicar les diferents maneres de fer el cablejat de la casa. Una funciona per BUS, el qual es un sol cable que va per tota la casa i tu el punxes allà on vols per posar un dispositiu. Aquest mode es el més car, però per a futures possibles reformes o ampliacions, és molt còmode. Aquí en veiem un esquema extret de la 5^a pàgina web de l'annex.



L'altre es per Centralització: A partir d'un dispositiu central com ara seria el nostre ordinador o placa Arduino Mega, es tiren tots els cables necessaris (esquema de la 5^a pàgina web de l'annex).



Hi ha altres sistemes com el PLC un sistema d'autòmats programables basats en relés de Siemens o Omron, per exemple, que solen ser molt senzills de connectar i de veure on pot fallar algun cable, però la instal·lació s'encareix, òbviament.

Em va recomanar utilitzar un programa autòmat a l'hora de programar l'Arduino, si puc, ja que a l'hora d'adaptar-ho a la realitat és menys viable tenir un ordinador encès tant de temps sense descans, sense ser un ordinador industrial o amb un sistema de refrigeració específic.



2.1.7 Connexió entre Arduino i Processing

Després vaig veure que per introduir les variables dels actuadors (motors, llums, timbre) i sensors (de presència i de temperatura) havia de fer que el meu programa escrit en Processing que creava la interfície gràfica i la placa Arduino que enviaria o rebria els senyals dels actuadors i sensors es comunicessin i vaig descobrir que es podia fer de dues maneres.

En el primer mètode, que m'havien proposat inicialment els meus tutors, havia de escriure un programa en Arduino que llegís les dades de la placa i les enviés al port sèrie i que llegís les dades del port sèrie i les enviés a la placa, i després fer el meu programa en Processing que es comunicés de igual manera amb el port sèrie.

Però trobava molt farragós aquest mètode, i vaig trobar un altre mètode que utilitzava una llibreria especial per a Processing que incloïa diverses funcions de la programació Arduino a Processing, fent el programa més difícil de programar però més fàcil d'estructurar, fet que em convenia perquè fos més fàcil de modificar per al lector.

2.1.8 Programar actuadors

Un cop sabia com connectar Arduino i Processing, només faltava definir les funcions de tots els botons en funció de l'actuador al que corresponien, per exemple el botó del qual les coordenades corresponen en el pla al de la llum del menjador, ha d'enviar una senyal a la sortida on hem connectat aquella llum, i mostrar en el plànol de l'ordinador que aquella llum és encesa.

Exemple:

```
if (button4) {  
    stroke(52,216,111);  
    arduino.digitalWrite(PinLed1, Arduino.HIGH);  
}  
else {  
    stroke(255,0,0);  
    arduino.digitalWrite(PinLed1, Arduino.LOW);  
}  
noFill();  
rect(x4,y4,w4,h4);
```



A partir d'allà, ja podia programar totes les llums, però òbviament vaig anar fent petites modificacions conforme avançava, com per exemple les variables de text que vam utilitzar per escriure el valor del termistor a la interfície del programa o el sensor de presència i el timbre, encara que l'estructura bàsica del programa ja estava finalitzada.

2.1.9 Motors

Després de tenir els llums de la casa programats, vaig decidir programar les persianes. Era el segon element més fàcil de programar ja que no havia de rebre senyals de cap sensor com en els següents elements que anava a incorporar. Tot i així, al contrari que les llums, les persianes se'm van resistir molt.

Amb el material de que disposàvem, el que volíem fer era adaptar la persiana del saló de la maqueta amb un motor senzill (DC) i els motors del pis de dalt fer-los amb els motors Stepper que ens permetien regular l'angle de gir i la velocitat d'aquest.

Motor Stepper

Per programar aquest motors (Fig.4), després d'estar estancat varies tardes provant de programar-ho, vaig adonar-me que necessitava incloure una llibreria del programa Arduino a Processing, i que Processing no la tenia. Per això hauria de canviar tot el programa de nou, i faria molt més complicat al lector de la guia modificar el programa al seu gust. A més tampoc ens ajudava gaire controlar l'angle de gir ja que el que volem no és que s'obri i es tanqui sinó tenir total llibertat d'obrir i tancar la mesura que es vulgui. Per això vam decidir fer totes les persianes de la casa amb els mateixos motors, els de corrent continu.



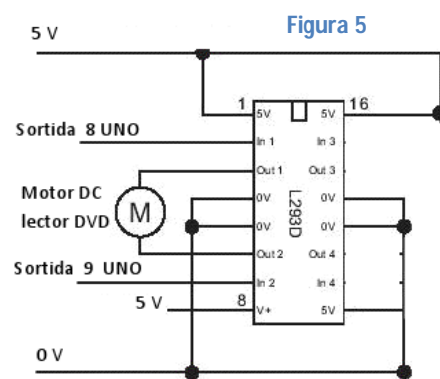
Figura 4



Motor DC

Per fer que el nostre motor funcionés tant de pujada com de baixada, havíem de realitzar un pont H, un tipus de circuit que permet canviar la polaritat d'un motor sense haver de desconnectar-lo i invertir el sentit dels cables a través d'un senyal que li pots enviar.

Per sort, a l'aula disposàvem d'uns circuits ja preparats per a l'ús de dos ponts H, els anomenats L293D, amb els quals podíem invertir la direcció del nostre motor. Vam agafar l'esquema de connexions (fig.5), de la pàgina web de Tecnologia del meu institut, situada als annexos (URL número 10 de la bibliografia) per a saber com connectar aquest circuit integrat. Al connectar i fer les proves vaig tenir alguns errors però els vaig solucionar al dia següent connectant 3 de les potes al positiu i la d'abaix a la dreta al GND (terra).



2.1.10 Alarma

Alarma de seguretat

Aquesta part del programa ja era més elaborada ja que havíem de rebre el valor d'un sensor d'infrarojos que detectes que si un cop activada l'alarma passava qualsevol cosa per davant del sensor, s'activés un timbre que no deixés de sonar fins desactivar l'alarma.

Al principi vaig intentar fer que si activàvem el botó alarma, ens enviés a llegir un "void loop" (tros de codi que es llegeix repetidament com el "void draw") del qual no sortiríem fins que es desactives l'alarma. Però no trobava cap funció a Processing que et permetés enviar a llegir una línia de codi diferent.

Finalment vaig optar per fer una subrutina, concretament un "if-then-else" dins d'un altre.

A l'hora de provar-lo ens donava alguns problemes: primer no funcionava el botó i vaig revisar totes les coordenades del botó, i ho vam solucionar(fig.6), i després no funcionaven els sensors correctament, vam provar amb uns diferents.

```
if (button1) { //botó timbre
  stroke(52,216,111); //pinta de verd el botó
  arduino.digitalWrite(PinEmisor, Arduino.HIGH); //encén l'IR emisor
  val_IR=arduino.digitalRead(PinReceptor); //llegeix l'IR receptor
  if (val_IR==1){ //si el receptor dona positiu
    arduino.digitalWrite(PinTimbre, Arduino.HIGH); //encen el timbre
  }
  } //si no dona positiu no facis res
else { //si el botó no esta clicat
  stroke(255,0,0); //pinta de vermell el botó
  arduino.digitalWrite(PinTimbre, Arduino.LOW); //apaga el timbre
  arduino.digitalWrite(PinEmisor, Arduino.LOW); //apaga l'emisor d'IR
}
noFill();|
rect(x11,y11,w11,h11);
}
```

Figura 6

Els primers sensors que vam fer servir eren dos DIY¹ (fig.7) que permetien de manera senzilla connectar els sensors a la placa Arduino sense tenir que afegir cap resistència ni cap tipus de element entre la placa i els sensors.



Figura 7

¹ DIY (Do It Yourself): Sensors integrats llestos per funcionar

Però al final per un error del DIY emissor vam haver de substituir-lo per un emissor d'infrarojos no integrat (el TSUS4400) havent de posar nosaltres les connexions com en el circuit extret de la pàgina web de tecnologia de l'IES Vilatzara (fig.8):

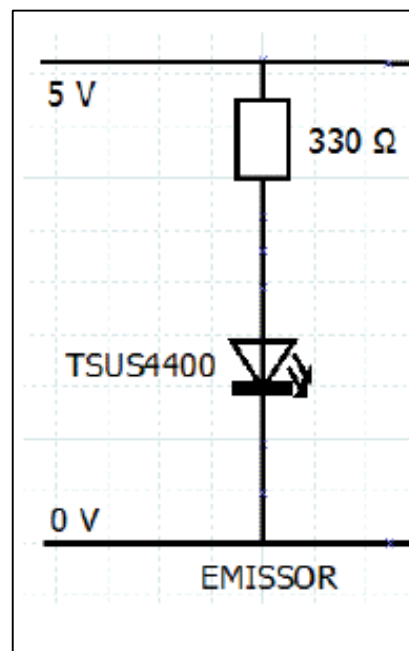
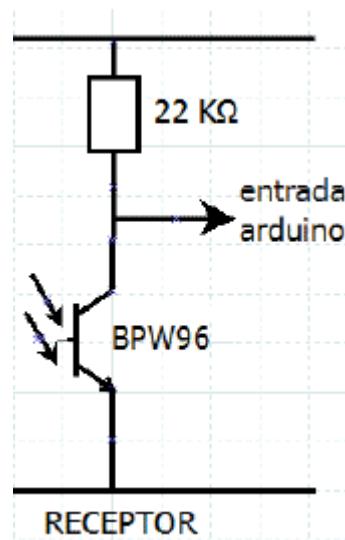


Figura 8

Vam haver d'afegir la resistència de 330Ω de la Figura 8. Ara l'emissor funcionava (veiem com s'encenia amb una càmera) però seguia fallant pel que vam canviar el DIY receptor per un receptor no integrat (el BPW96) amb una resistència de 22000 ohms, tal com es pot observar en el seu datasheet:





Alarma antiincendis

Tot i així no ens va acabar de funcionar per culpa de les poques possibilitats de funcions que ens dóna la connexió de Processing i Arduino a través del Firmata (però m'interessava més que em donés problemes a mi, que no pas al usuari de la guia a l'hora de modificar el programa).

Per tant, vaig decidir canviar l'alarma de seguretat per una alarma d'incendis aprofitant el termistor que explicaré com vam calibrar en el següent apartat.

Aquesta nova alarma ens avisarà a la interfície quan la casa pugi a temperatures extremes per prevenir incendis, encendrà una alarma (bronzidor a la maqueta) i activaria els sistemes de reg, tot i que no podem mullar la nostra maqueta. En segon cas, ens avisarà si la casa disminueix massa de temperatura a la interfície, i activaria la calefacció de la casa. A la següent figura podem veure l'algorisme:

```
if (button1) {          //botó alarma antiincendis
  stroke(52,216,111);   //pinta de verd el botó
  val_termistor2= arduino.analogRead(PinTermistor);
}
else {                 //si el botó no esta clicat
  stroke(255,0,0);     //pinta de vermell el botó
  arduino.digitalWrite(PinTimbre, Arduino.LOW); //apaga el timbre
}
noFill();
rect(x11,y11,w11,h11);
  if (val_termistor2<520){ //si el receptor dona positiu
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura alta: Activant sistemes antiincendis", width/2-25, 60);
    arduino.digitalWrite(PinTimbre, Arduino.HIGH); //encen el timbre
  }
  else{
    arduino.digitalWrite(PinTimbre,Arduino.LOW);
  }

  if (val_termistor2>600){
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura baixa: Activant calefacció", width/2-25, 60);
  }
  else{
  }
}
```



2.1.11 Calibrant els termistors

KY-13

A falta de coneixements i/o manuals o llibres on poder cercar com programar un termistor, vaig acabar optant pel mètode prova-error. Vaig començar a modificar les línies de codi allà on m'apareixien errors, pensant-hi una estona en que podia funcionar, i provant-ho.

Després d'hores fent prova-error re-escrivint aquella part del programa, finalment vaig aconseguir eliminar tots els errors i fer-lo funcionar.

Però això òbviament, no significa que faci el que tu vols que faci. Quan vaig provar el termistor vaig al·lucinar amb el valor que em donava (550). Ara, com ja no era error de programació sinó del termistor en sí, vaig buscar informació sobre els termistors.

Els termistors (Fig.9) són uns aparells que generen diferents resistències en funció de la temperatura a la que estan; però no varien segons una funció lineal, sinó en funció d'una asíptota, que és el que ens va portar més mal de caps.

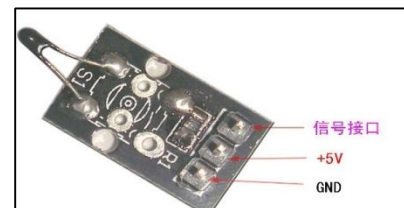


Figura 9

Vaig demanar ajuda a un treballador de la companyia Picaxe (Joan Pellicer Moreno) via e-mail i em va respondre que per a transformar la resistència a una temperatura, primer hem de trobar la temperatura característica del material del termistor (β) a través els següents càlculs:

$$\beta = \frac{\ln \frac{R_1}{R_2}}{\frac{1}{T_1} - \frac{1}{T_2}}$$

on:

- R1 i R2 són les resistències que posa el termistor a les temperatures T1 i T2 respectivament
- T1 i T2 són dos temperatures ja conegudes que em escollit per a calcular el valor,



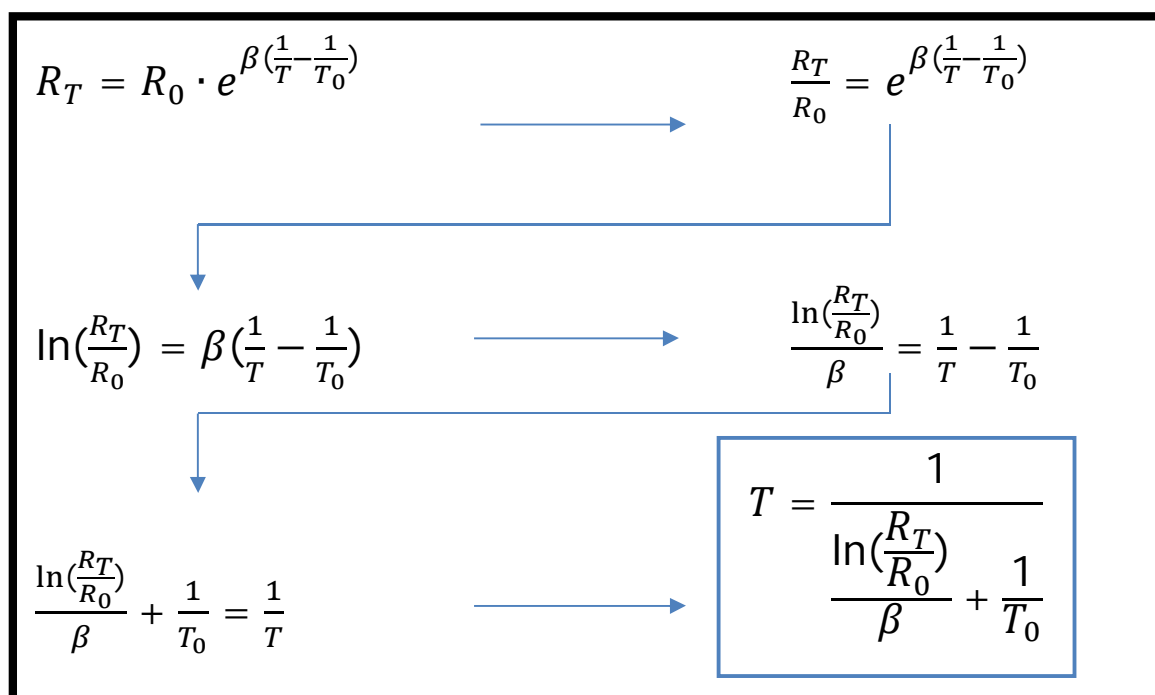
Un cop hem trobat la β (que ens va donar "2,93315") haurem d'aplicar la següent equació per a transformar la resistència obtinguda en la temperatura real:

$$R_T = R_0 e^{\beta \left(\frac{1}{T} - \frac{1}{T_0} \right)}$$

on:

- R_T és la resistència que ens dóna el termistor a la temperatura T i R_0 la resistència a la temperatura T_0 .
- T és la temperatura a la que està el termistor i T_0 una temperatura de referència qualsevol

Però clar, el que nosaltres busquem no és el valor de resistència en funció de la temperatura a que el posem, sinó que busquem la temperatura equivalent al valor de resistència que ens indica el termistor. Per a això vam demanar ajuda al físic i professor de matemàtiques Jaume Serra per a assegurar que es pogués aïllar sense problemes i no errar a l'hora de treballar amb logaritmes i nombre d'euler. Un cop aïllada la temperatura només havíem de definir la temperatura del programa en base a la formula resultant. Al requadre de sota podem observar com vam aïllar la temperatura de la formula perquè ens quedés en funció d'aquesta:



Dallas DS18B20

Al provar d'aplicar la fórmula de la temperatura de l'anterior termistor, ens aproximava el valor del logaritme neperià a 0, pel què ens donava una temperatura 0. Després de varies proves, vam decidir provar amb aquest altre termistor què ja havia funcionat amb altres membres del grup.

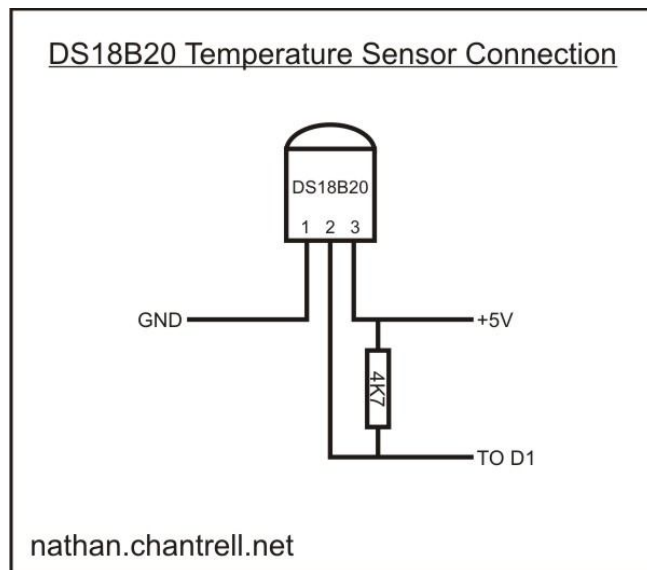
Figura 10

Vam connectar-lo tal com es mostra en el seu Datasheet (fig.10) i com expliquem en els annexos en l'apartat "5.2 Com hem connectat els elements". A l'hora de provar-lo ens donava un valor 1021 però aquest ni tan sols reaccionava a la calor o al fred.

Vam modificar el programa i provar-lo amb programes de prova extrets

de exemples per internet (8^a i 9^a pàgina web de l'annex), però amb aquests exemples ni tan sols ens reconeixia el termistor com a element connectat, pel que vam decidir tornar a buscar una solució per al termistor anterior, que almenys reaccionava al calor i només havíem de transformar la resistència en temperatura.

Per a fer-ho vam optar per, en comptes de fer servir enrevessades fórmules per transformar el valor, fer una taula de condicions que definís cada grau entre dos valors de resistència. Així marcava la temperatura a la què correspondria aquell valor. Es un mètode molt més farragós per calibrar, però segur.





2.2 Problemes i dificultats comuns

Al començar a programar van començar a sorgir, com sempre acostuma a passar al programar, problemes que no tenien aparent explicació. Hi havia problemes que solucionava en un segon, i altres que estancaven el projecte setmanes senceres. Aquí us els explicarem per tal que no entropesseu amb la mateixa pedra, amb la seva solució adjunta corresponent al mateix número:

1. El primer problema que va aparèixer va ser a l'hora d'adjuntar el plànol com a imatge al programa. Em donava error sempre que ho intentava, i tot i revisar la ruta de la imatge mil cops, no aconseguia trobar l'error.
2. Podeu trobar també molts errors com "missing expected ;" o "doesn't expect some function".
3. A l'hora de programar el valor del termistor, si només li dieu "analogRead(___)" us donarà el valor de la resistència segons aquella temperatura.

2.2.1 Solucions

1. Finalment vaig descobrir que la imatge l'has de transportar a la carpeta d'instal·lació de Processing, o a la carpeta on es guarda l'aplicació de Processing que exportes.
2. Aquest problema s'arregla normalment buscant una línia inconclusa en la que ens haguem deixat el signe de puntuació punt i coma.
3. És important preguntar per el valor Beta del vostre sensor de temperatura allà on el compreu, que és el que us ajudarà a través de l' equació de la temperatura de l'apartat "2.1.11 Calibrant els termistors", a transformar el valor de resistència en un valor de temperatura en graus Centígrads. Sinó, haureu de trobar-lo com he hagut de fer a través de diverses proves a diferents temperatures.



3. Guia per a la realització

Aquí és on es centra el gruix de la guia per a automatitzar la teva pròpia casa. Mentre realitzeu el primer apartat de programació, per aprofitar el temps, seria convenient que els lampistes comencessin a cablejar i a fer les regates necessàries.

3.1 Programació

Per a realitzar el projecte d'automatització a la vostra llar, només haureu de seguir tres senzills passos que s'expliquen a continuació

3.1.1 Plànol

El primer que hem de fer és afegir el nostre plànol a la carpeta on hem guardat el programa, el qual ens podem descarregar de: www.domoticaperlallar.blogspot.com.

Un cop substituïm la imatge que hi ha (la del plànol de la maqueta) per la de la nostra casa, només hem de canviar la posició dels botons que teníem del plànol de la maqueta a la del nou plànol. Per a això primer ens hem de descarregar el processing de <http://www.processing.org/download/> per poder editar el programa.

Al obrir el programa el primer que hem de fer és a anar al apartat "void setup" (fig.11) i substituir la mida de la imatge (on posa "size(1276,696)") i el nom d'aquesta (on posa ("Planol.jpg")) per les mides en píxels del vostre plànol (apareixen a les propietats de la imatge) i el nom del arxiu on heu guardat la foto del vostre plànol.

Figura 11

```
void setup() {  
  
    size(1276,696); //obrim un recuadre del tamany del planol  
    img= loadImage("Planol.jpg"); //insertem la imatge del planol i
```

Un cop haguem fet això, al clicar el botó de play s'executarà un programa on s'obrirà el plànol de la vostra casa, però amb els mateixos botons que teníem en el plànol anterior. El que s'ha de fer ara és substituir la posició d'aquests botons per la dels nous botons. És una feina laboriosa però gens complicada, com veureu a continuació.



3.1.2 Botons

Per agilitzar al màxim i modificar el mínim de coses possibles, a l'hora de canviar la posició de cada botó heu de tenir en compte el número de llums, motors o botons connectats a un sensor que tinc. Així si col·loqueu les coordenades de les vostres llums on jo tinc una llum, no haureu de canviar la funció d'aquell botó. Veureu com canviar les funcions al següent apartat.

Per canviar la posició dels botons haurem d'obrir la imatge en Paint i col·locar el ratolí al vèrtex superior esquerre de cada botó i apuntar els valors que ens donen que correspondran a les variables "x", "y" del nostre programa. Ara hem de calcular l'amplada i l'alçada dels botons des d'aquests botons fins a on volem que arribin.

(Agafem el valor "x" del costat dret del botó i el restem al valor de "x" que el teníem, i el mateix amb el costat inferior i la "y", i ens donaran els valors "w" (width, amplada) i "h" ("height", alçada).

Un cop fem això amb tots els botons (podem escriure-ho en una taula de dades com Excel per estructurar-ho millor, si voleu podeu agafar com a exemple la del meu projecte, que està penjada aquí:

www.domoticaperlallar.blogspot.com

haurem de substituir els valors de la taula que hem fet en el programa.

Per fer-ho anirem al apartat on definim les variables de cada botó (fig.12) i substituïrem els valors.

```
int x1 = 80; //Botó Persiana Saló Pujar
int y1 = 130;
int w1 = 35;
int h1 = 220;

int x2=80; //Botó Persiana Saló Baixar
int y2=350;
int w2=35;
int h2=220;

int x3 = 172; //Botó Temperatura casa
int y3 = 298;
int w3 = 40;
int h3 = 40;

int x4= 515; //Botó Llum Menjador
int y4= 420;
int w4= 40;
int h4= 40;

int x5= 1039; //Botó Persiana nens 1 pujar
int y5= 586;
int w5= 96;
int h5= 20;

int x6= 879; //Botó Persiana nens 2 pujar
int y6= 586;
int w6= 96;
```

Figura 12



El número que segueix cada variable "x", "y", "w" o "h" simbolitza el número que hem assignat a cada botó (1, 2, 3...)

També podeu editar els comentaris que apareixen en gris per aclarir-vos millor a l'hora d'entendre el programa, i posar noms a cada botó en el vostre plànol.

3.1.3 Funcions dels botons

Tot i haver intentat col·locar les llums en el número de botó on jo tenia les llums o les persianes en el número de botó on jo tenia les persianes, sempre hi haurà botons als quals haureu de canviar la funció. Aquest punt és el més crític i allà on podem tenir més errors.

Si necessiteu afegir un botó al programa, heu de tenir en compte TOTES les parts del programa, des de definir la variable, com definir les seves coordenades fins a definir què succeirà quan el premeu.

Primer hem de definir el nou botó en l'estat d'apagat, com a variable booleana, ja que només pot estar encès o apagat. Les coordenades del botó ja les haurem estipulat en el pas anterior. Si volem afegir un altre botó (11, per

```
boolean boton1 = false; /*declarem la variable dels botons i
                           fem que comencin desactivats*/
boolean boton2 = false;
boolean boton3 = false;
boolean boton4 = false;
boolean boton5 = false;
boolean boton6 = false;
boolean boton7 = false;
boolean boton8 = false;
boolean boton9 = false;
boolean boton10 = false;

int x1 = 80; //Botó Persiana Saló Pujar
int y1 = 130;
int w1 = 35;
int h1 = 220;
```

Figura 13

exemple), haurem de definir unes noves coordenades ("x11", "y11", "w11" i "h11").

Això ho farem en la primera part del programa (fig.13). Un cop fet això, el programa ja reconeix el boto com a una variable amb unes coordenades a la interfície. Ara només falta explicar-li que al clicar a sobre canviarem l'estat d'aquest (d'apagat a encès i a la inversa), i definir què volem que faci tant quan està encès com quan està apagat.



Per activar/desactivar al prémer el ratolí a sobre, haurem de copiar una de les línies de l'última part del programa, el "void mousePressed", i substituir el valor de tots els números que correspondrien al número del botó anterior per el del nostre nou botó, com al següent exemple:

```
void mousePressed() {  
  if (mouseX > x1 && mouseX < x1 + w1 && mouseY > y1 && mouseY < y1 + h1 &&  
mousePressed){  
  button1 = !button1;  
  }  
  if (mouseX > x11 && mouseX < x11 + w11 && mouseY > y11 && mouseY < y11 + h11 &&  
mousePressed){  
  button11 = !button11;  
  }  
}
```

I per últim haurem d'assignar la funció que desitgem que faci, la qual només hem de copiar d'un dels exemples que jo he posat (llum, motor DC, motor Stepper, sensor de temperatura o sensor de presència) de la part central del programa, i substituir el número de botó pel del nostre nou botó.

En l'exemple de la imatge, el botó 1 al encendre's, es posa de color verd (funció stroke), i activa la sortida "PinMotor31" a la que hem assignat un dels nostres motors.² Al apagar-se (else), es posa de color vermell i s'apaga la sortida. La funció "noFill" sobre el rectangle de les coordenades del botó la utilitzem per només pintar de color verd/vermell el requadre del botó, i no la part de dins.

```
void draw() {  
  background (0);  
  image(img,0,0);  
  
  if (button1) {  
    stroke(52,216,111);  
    arduino.digitalWrite (PinMotor31, Arduino.HIGH);  
  }  
  else {  
    stroke(255,0,0);  
    arduino.digitalWrite (PinMotor31, Arduino.LOW);  
  }  
  noFill();  
  rect(x1,y1,w1,h1);  
}
```

² Amb l'únic objectiu de facilitar la comprensió del programa i saber a quina sortida pertany cada botó. Es podria evitar i posar directament el número de la sortida o entrada desitjades

3.2 Connexió a l'Arduino

Ara ja hem finalitzat la programació, que era l'etapa més àrdua, i tenim un programa funcional amb botons que s'encenen i s'apaguen sobre el plànol de la nostra casa. A més el lampista ja haurà acabat amb la instal·lació de cablejat de la casa, i haurà connectat els actuadors (motors de persiana, llums, sistemes de reg) i us haurà deixat un munt de cables. L'únic que falta és connectar els cables de cada actuator o sensor a la seva sortida o entrada corresponent.

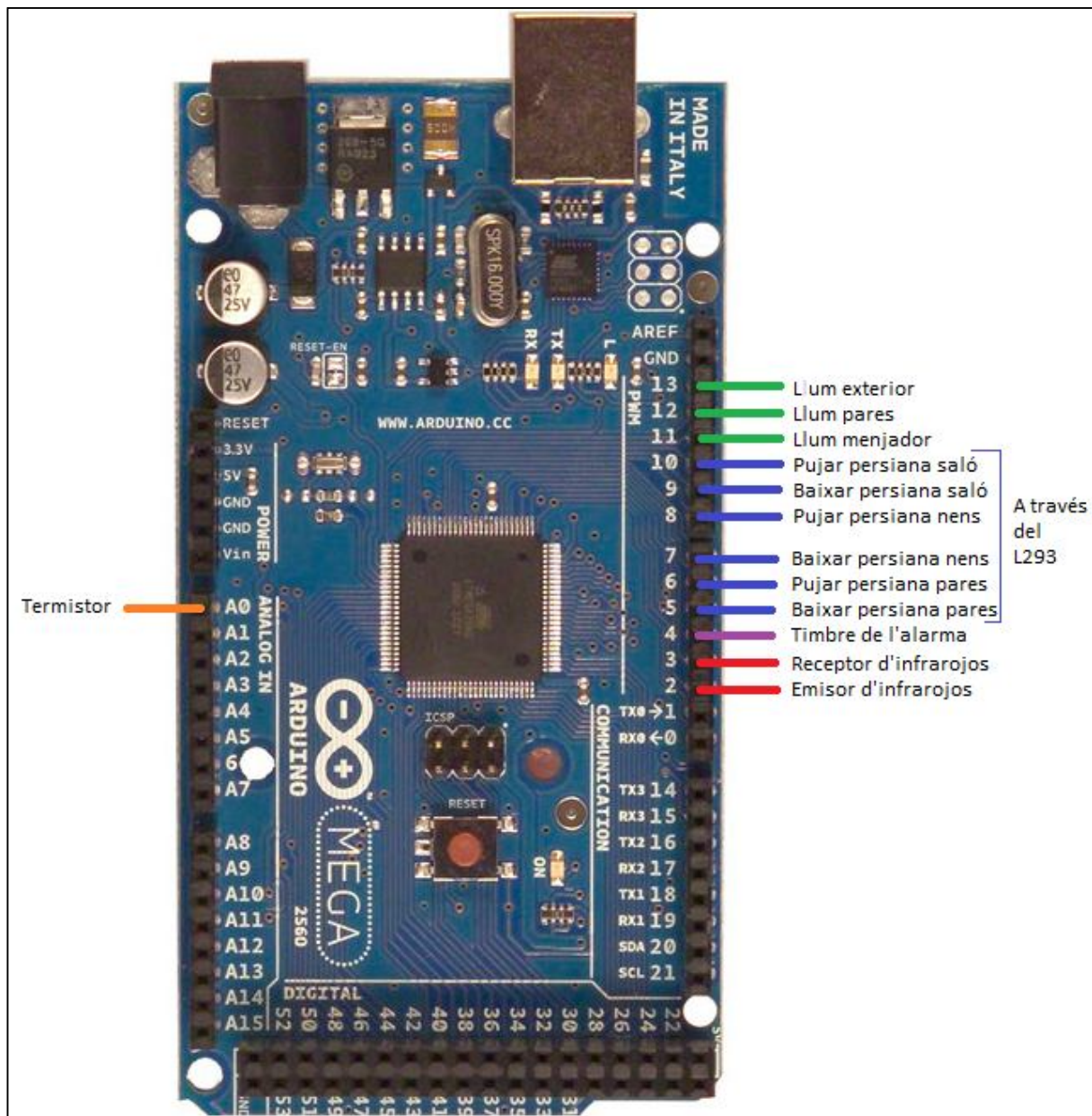


Figura 14



Per connectar correctament el cablejat que us posi el lampista heu de reconèixer on ha d'anar cada cosa: Si observem la figura 14, que és l'esquema de connexions que hem fet per a la nostra maqueta, veurem que l'apartat esquerre (Analog IN) és on es connecten els sensors (temperatura, presència humitat, ...). A la part dreta (PWM) i en el seu defecte si faltessin connectors, a la part de la inferior (Digital), és on connectarem les sortides (motors, llums,...). Seguint aquest sistema vosaltres haureu de connectar cada un dels cables a on li pertoqui depenent de la sortida que li hem assignat en el nostre programa.

Sempre haurem d'especificar en el nostre "void setup" si les sortides les estem tractant com a sortida o entrada, com a la figura 15, primer encenent la placa Arduino amb la funció "Arduino = new Arduino (this, Arduino.list()[0], 57600)" i posteriorment amb la funció "Arduino.pinMode" especificant cadascuna de les sortides com a "INPUT" o "OUTPUT".

```
arduino = new Arduino(this, Arduino.list()[0], 57600);  
arduino.pinMode(PinTermistor, Arduino.INPUT);  
arduino.pinMode(PinReceptor, Arduino.INPUT);  
arduino.pinMode(PinEmisor, Arduino.OUTPUT);  
arduino.pinMode(PinMotor11, Arduino.OUTPUT);  
arduino.pinMode(PinMotor12, Arduino.OUTPUT);
```

Figura 15



4. Conclusions

4.1 Opinió del Treball

Un treball així és una prova de maduresa que no només m'ha demostrat els coneixements que he adquirit al llarg del període estudiantí, sinó que també m'ha fet veure i desenvolupar la meua capacitat de moure'm i buscar de manera autònoma i autodidacta la informació que requeria per a realitzar el treball. Gràcies al treball he comprovat i millorat la capacitat de suportar la tensió i la responsabilitat d'haver de lliurar un treball sense algú que estigi constantment sobre teu explicant-te com fer les coses i guiant-te a cada pas fins finalitzar el teu treball.

Aquesta autonomia i responsabilitat del treball, fa realment difícils les decisions que has de prendre de com encaminaràs el teu treball, ja que d'aquestes decisions depèn el resultat final que obtindràs, i tu en tens la total responsabilitat. Personalment se'm va fer difícil escollir què voldria controlar a la meua casa, què era essencial i què no ho era, a l'hora de automatitzar una casa.

4.2 Objectius assolits

Tot i així hem pogut demostrar el que vam plantejar-nos al principi: Hem demostrat que la domòtica és cara perquè les empreses així ho volen, i que podem transformar la nostra llar en una casa domòtica nosaltres mateixos sense coneixements aparents, seguint aquesta guia. Hem aconseguit reduir el preu de l'automatització en aproximadament un 80% (considerant que la nostra instal·lació té la qualitat d'una instal·lació que comprada costaria uns 2500 €, i que l'hem reduït a 500€, el que ens podria costar un lampista pel cablejat sencer i motors de persiana, relés, etc...)



4.3 Possibles millores

4.3.1 Ampliació dels elements

Tot i haver assolit els nostres objectius, el projecte està obert a moltíssimes possibilitats. El podríem millorar augmentant el pressupost del treball per a posar més sensors (llums que s'activen amb presència o amb so) o més actuadors (portes automàtiques).

Tot i així les bases per a programar aquests nous sensors o actuadors serien les mateixes, pel que seguint aquesta guia podríem millorar el projecte fins a pràcticament on volguéssim, gràcies a les infinites possibilitats de programar amb llenguatge Java.

4.3.1 Ampliació dels controladors

A part d'ampliar l'exemple d'actuadors i sensors de la nostra llar, podem millorar la comoditat amb la que controlem la casa. Estudiant els llenguatges de programació d'Apple o Android, podem crear una aplicació similar a la nostra però per a plataformes mòbils com el telèfon o l'iPod.

Això obriria portes a possibilitats com enviar-te una notificació quan algú entri a casa sense identificar-se o poder visualitzar la teva casa amb càmeres des de qualsevol altre lloc del món a través d'internet, a més de poder controlar la casa des d'on vulguis.

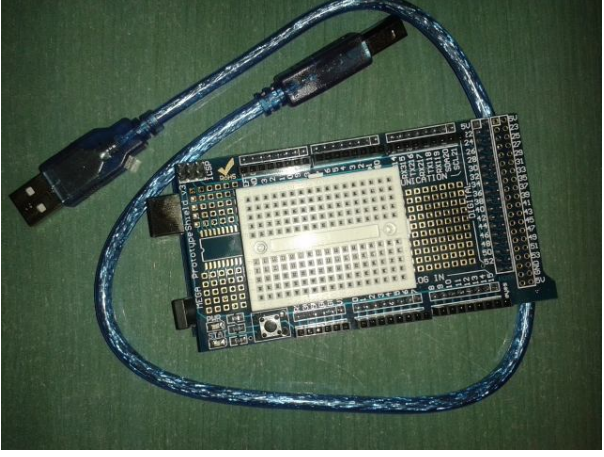

Les plaques Arduino que hem utilitzat tenen les portes obertes a tot el món d'internet a través d'una llibreria anomenada Wifi Shield que permet connectar la teva placa a internet de casa teva per enviar-te el que faci falta al mòbil en qualsevol moment i a qualsevol lloc on tinguis connexió.



5. Annex

5.1 Elements utilitzats

5.1.1 Per a la maqueta

<p>Arduino Mega amb cable USB</p>	
<p>Soldador d'estany, estany, suport i pasta per soldar</p>	



Pistola de silicona







Motors per les persianes i pont H
(L293D)



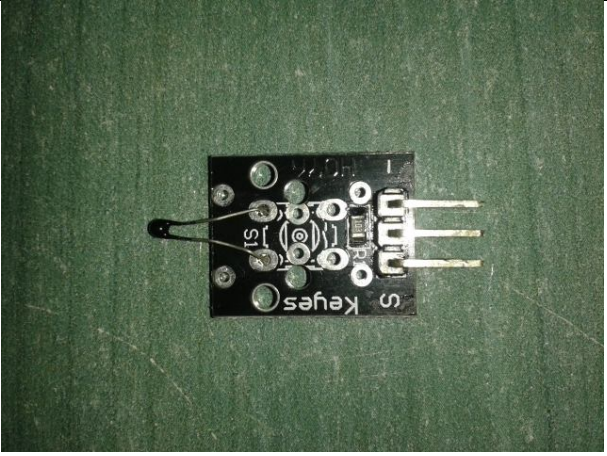
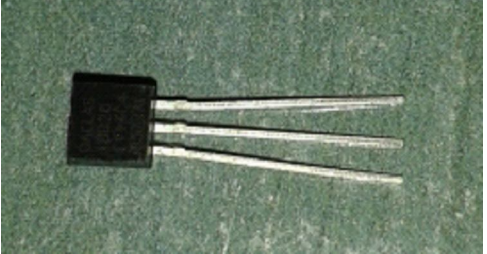
Timbre





<p>Resistències (330Ω i 220Ω)</p>	
<p>LED'S</p>	
<p>Emissor d'Infrarojos</p>	
<p>Receptor d'Infrarojos</p>	



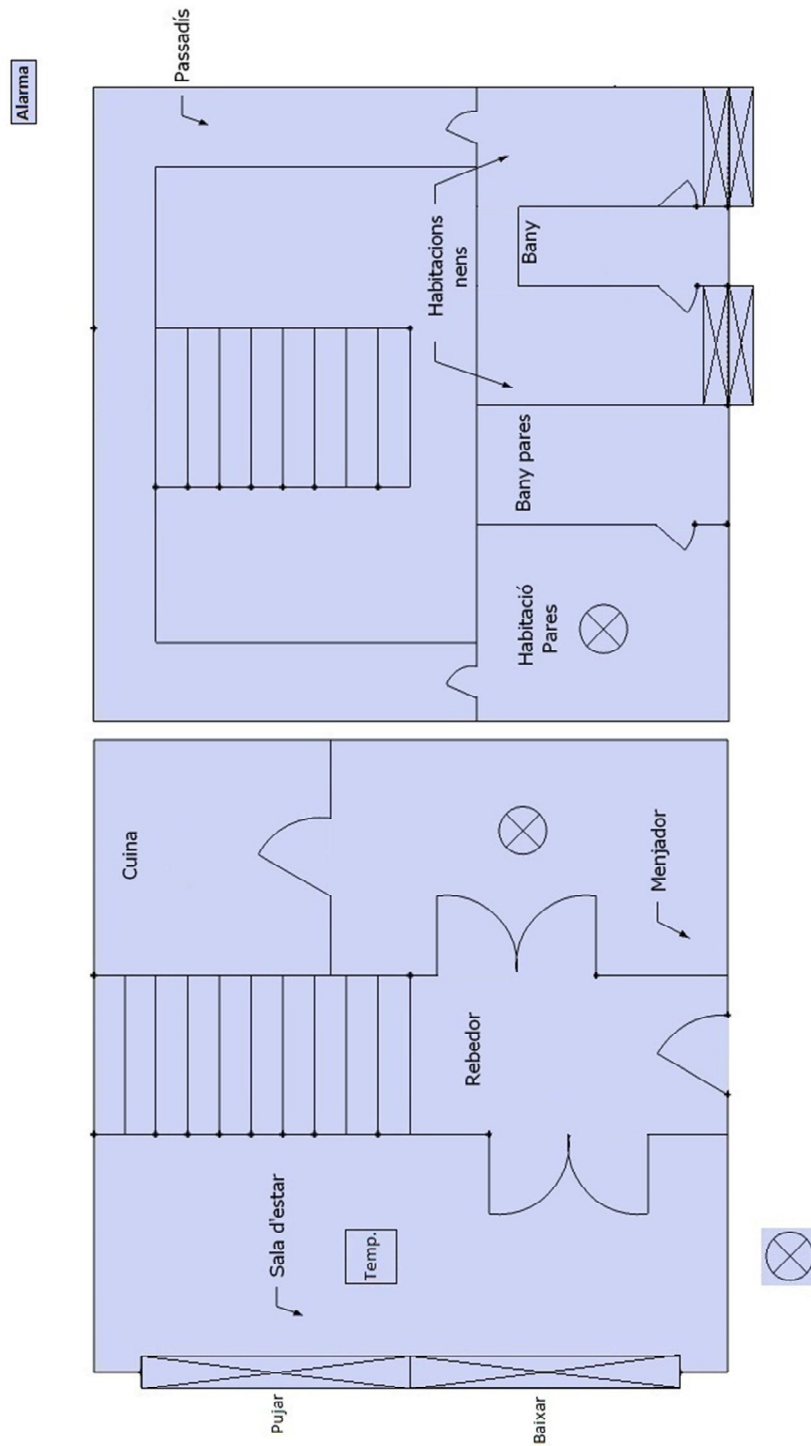
Termistor 1 (KY-13)	
Termistor 2 (DS18B20)	

5.1.2 Programes

Processing i Arduino instal·lats a l'ordinador	
---	--



5.1.3 Plànol





5.2 Maqueta

5.2.1 Maqueta abans d'automatitzar

A les següents fotografies veiem la maqueta just després de ser construïda amb cartró-ploma i cola termostable. És una casa qualsevol que inicialment no va ser construïda per a ser automatitzada, i no té cap preparació.

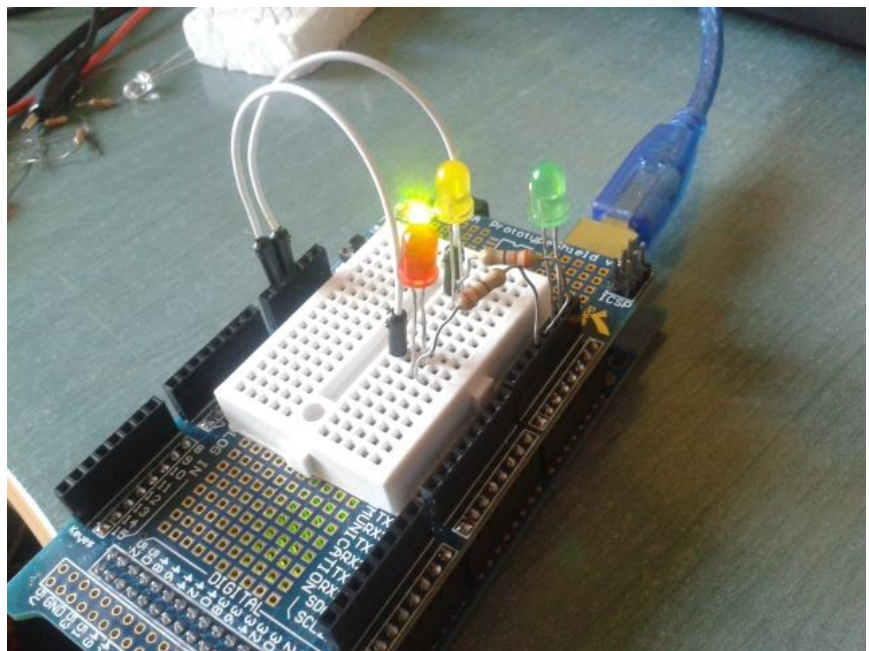
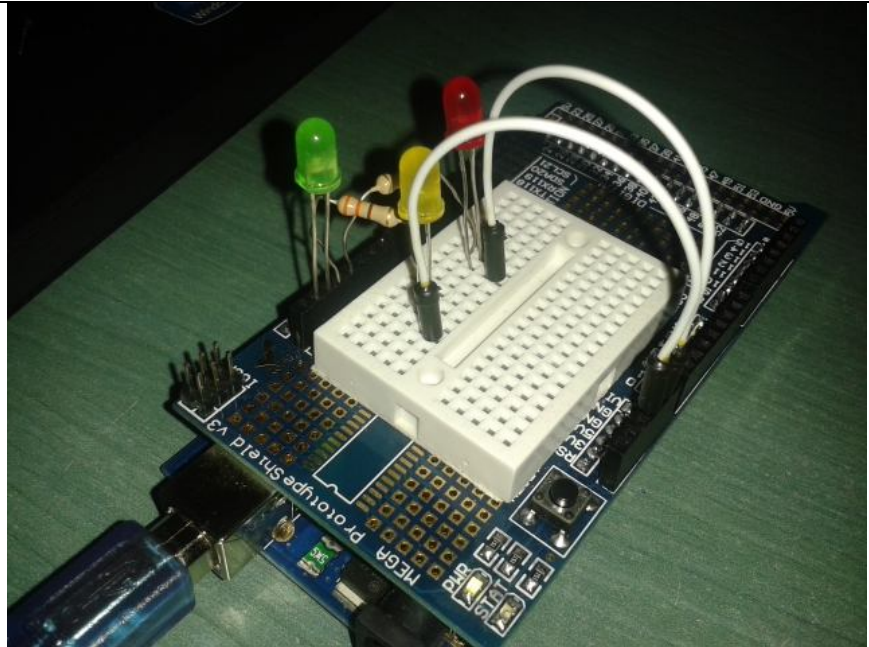






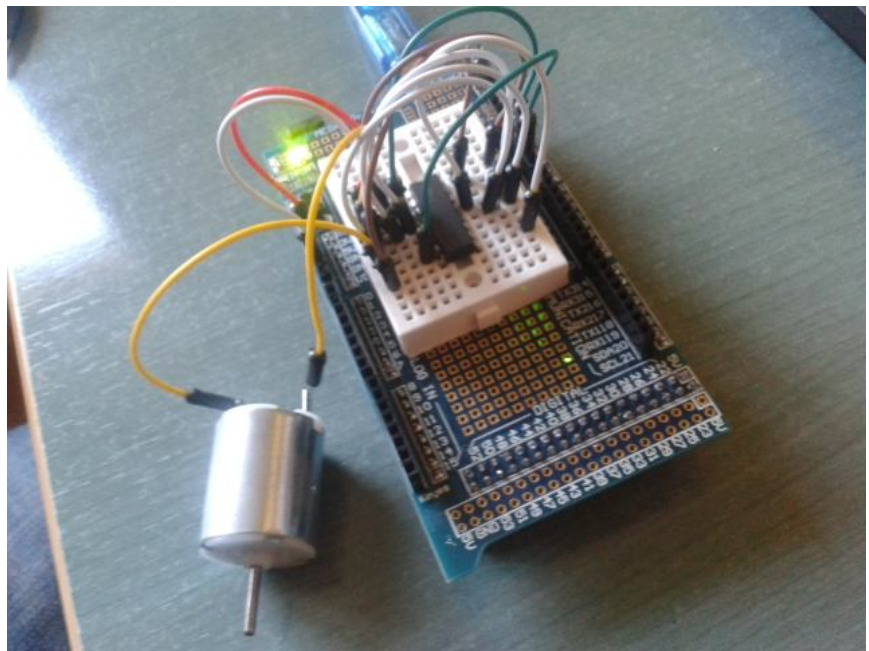
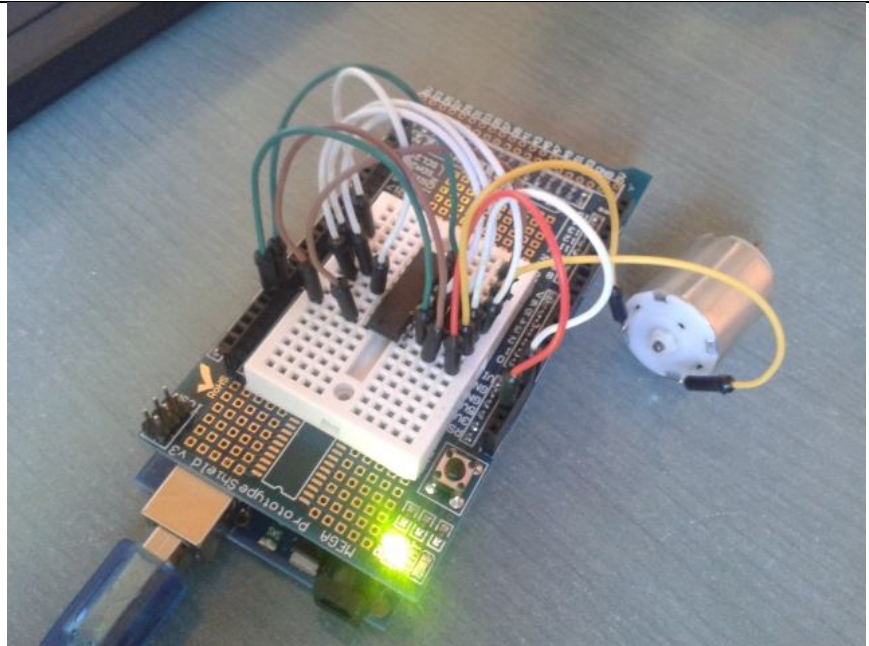
5.2.2 Com hem connectat els elements

LED: Per connectar els LED sempre ho hem de fer amb una resistència de 330Ω en sèrie. Per al nostre projecte hem aprofitat la resistència que incorpora la placa Arduino a la sortida 13 especial per a LED. Per als altres dos hem hagut de connectar la resistència entre l'ànode (pota llarga) i la sortida. Els càtodes (potes curtes) van tots connectats a la terra de l'arduino (GND).



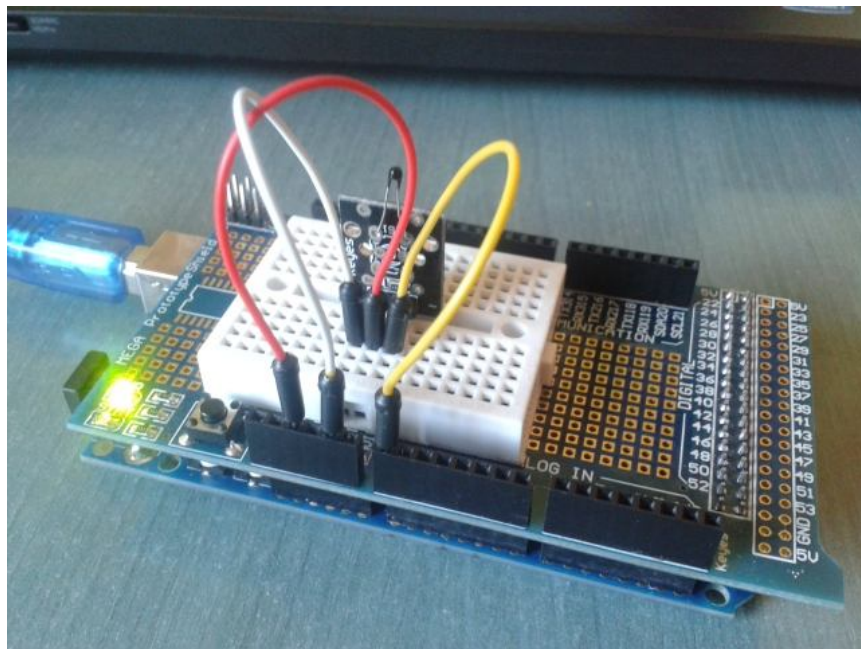
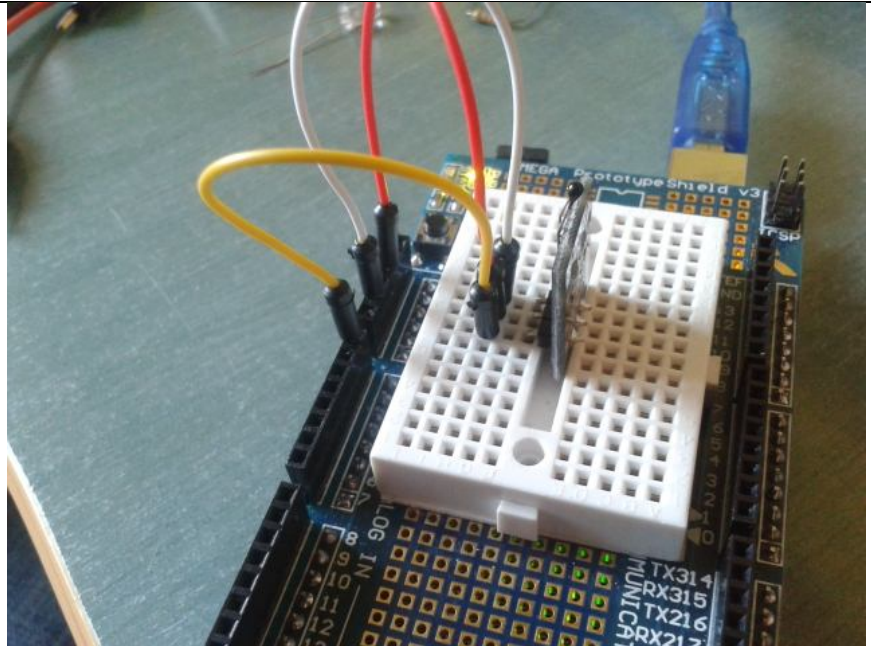


Motors DC: Hem connectat els motors al nostre L293D tal com es mostra en la figura 5 de la pàgina 17. Per a reduir la velocitat dels motors hem connectat el nostre pont H a 3v en comptes de a 5v, ja que sinó les persianes baixaven massa ràpides i podien portar problemes.



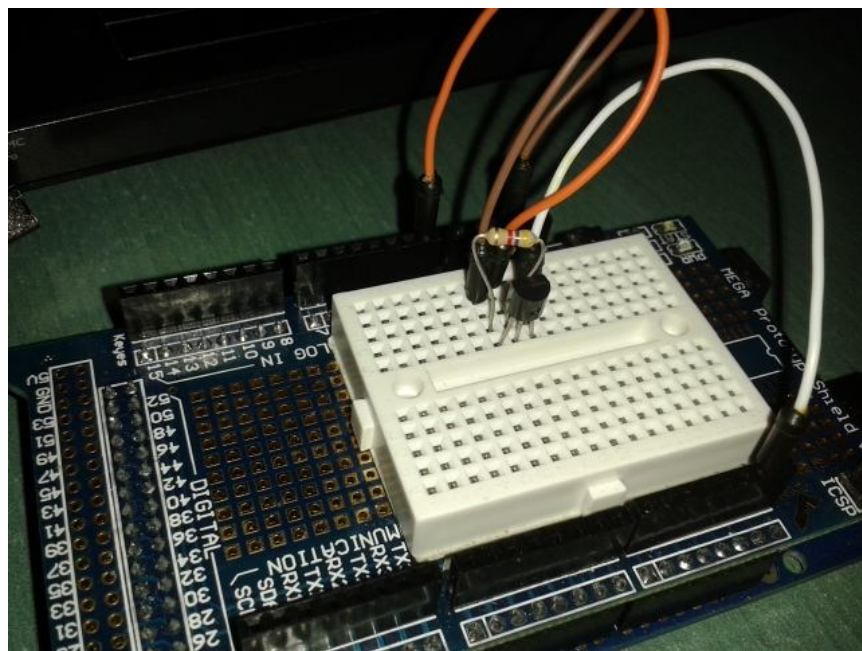
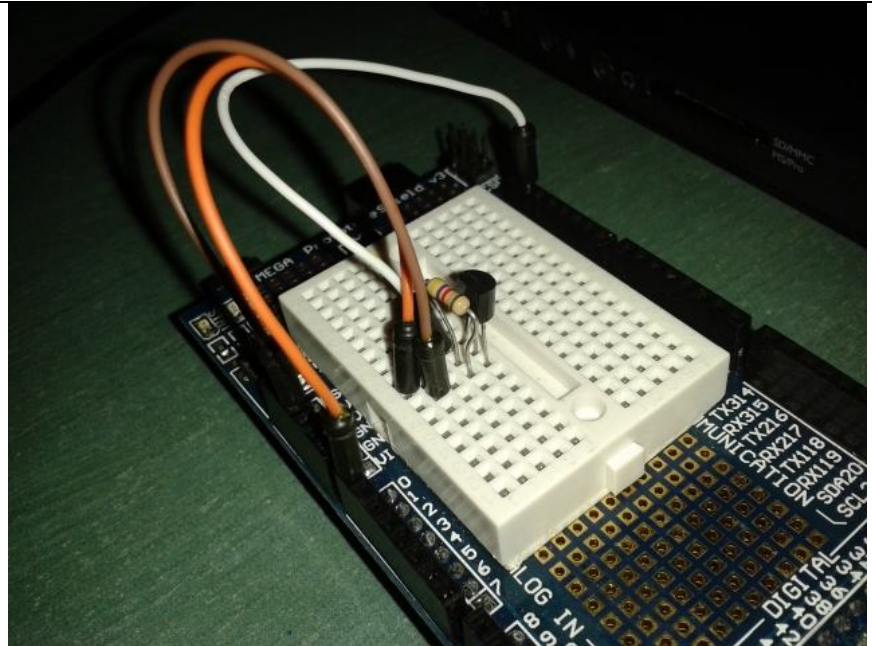


Termistor 1: Hem connectat el termistor tal com mostro en el diagrama de la figura 9 en la pàgina 20: La pota esquerra al GND, la del mig als 5v de l'Arduino, i la pota de la dreta a l'entrada 0 de l'Arduino. Però al no funcionar correctament vam haver de substituir-lo pel següent termistor.



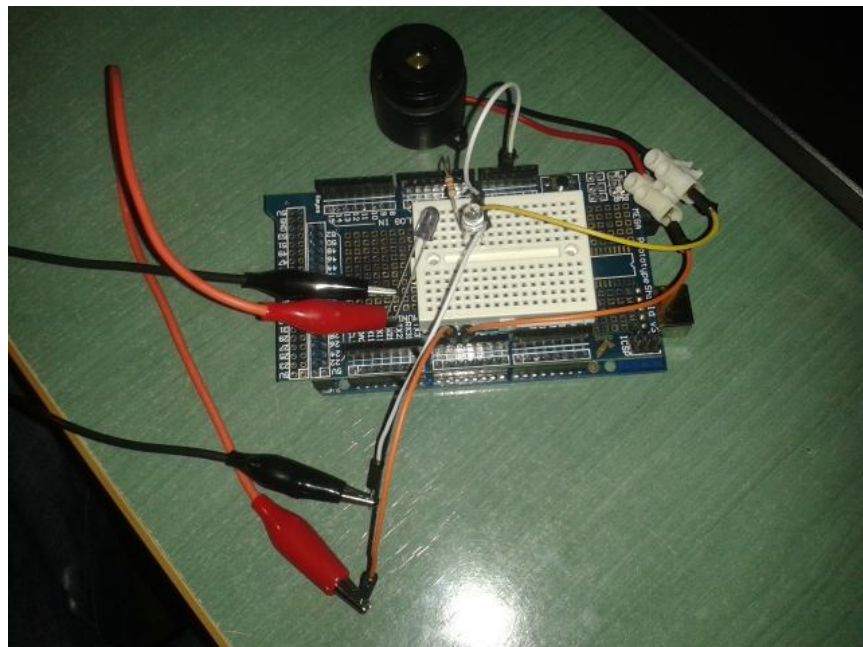
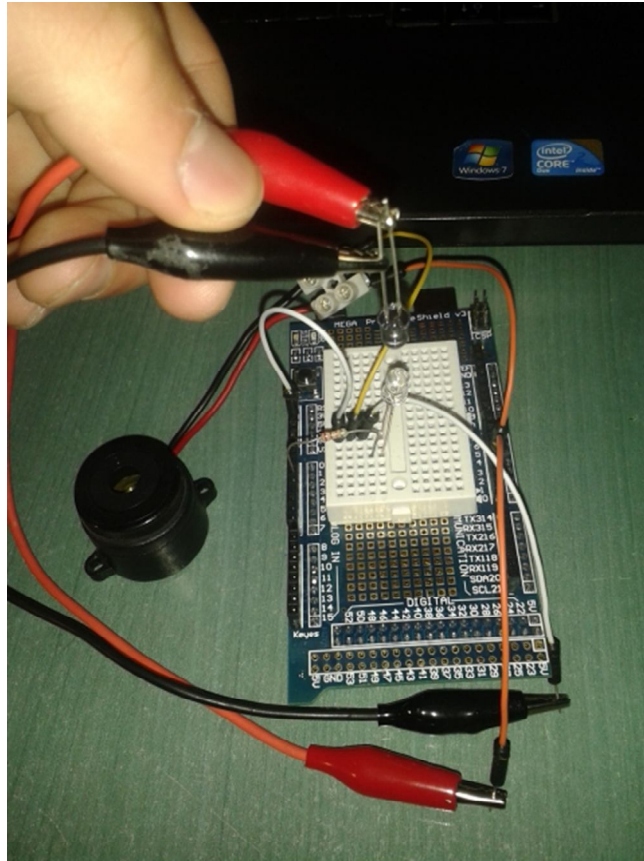


Termistor 2: Per a connectar el segon termistor hem hagut de seguir el datasheet de la pàgina 22 que ens mostra com hem de connectar la resistència de 4700Ω que hem d'incorporar al nostre sensor. A part d'aixo no té cap altre complicació, una pota va a terra i les altres a 5v i a la pota 0 de les entrades analògiques, amb la resistència entre voltatge i pota.





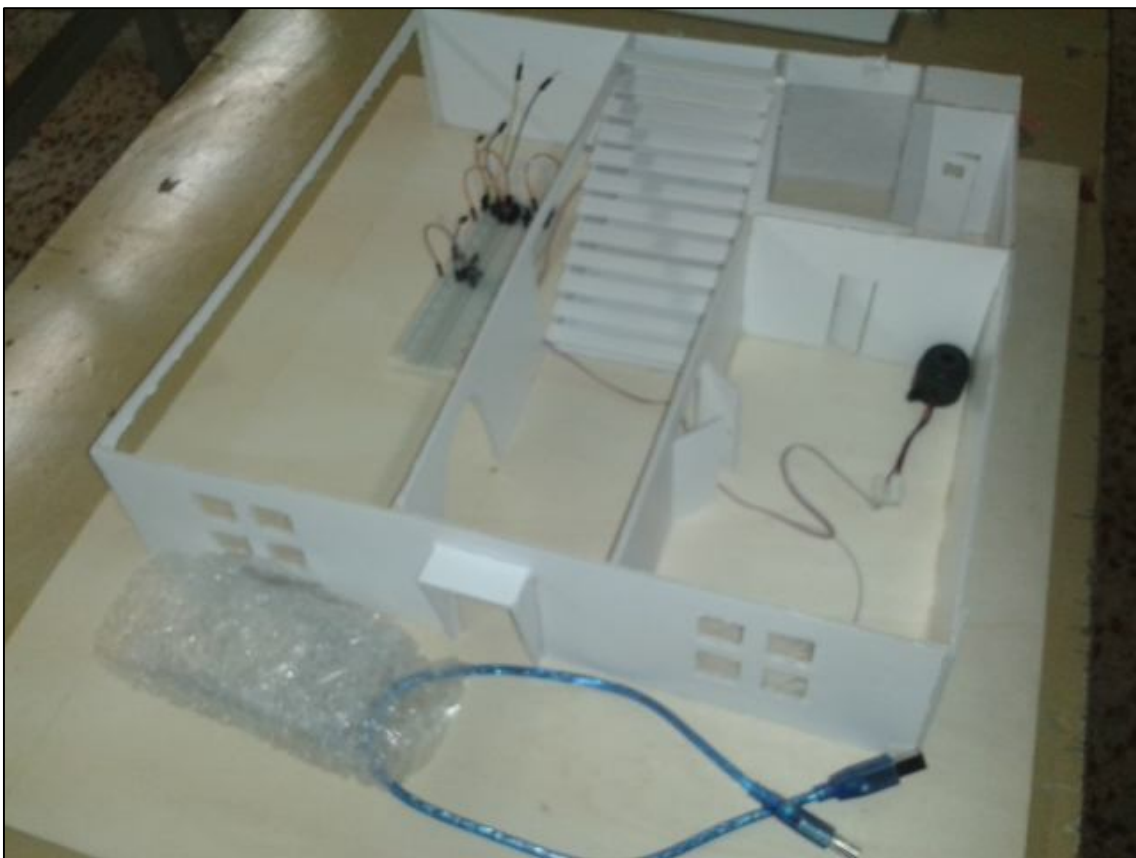
Alarma: Per connectar els elements que van formar la nostra alarma (receptor IR, emissor IR i timbre), vam haver d'incorporar una resistència de 330Ω a l'emissor i una resistència de 22000Ω al receptor entre els ànodes i la sortida/entrada. El timbre el vam connectar directament a la sortida corresponent. Els dos càtodes i el negatiu del timbre els hem connectat a terra (GND de l'Arduino).





5.2.3 Maqueta automatitzada

En aquestes imatges observem el procés d'automatització que va produir-se a la nostra llar, començant per connectar a la protoboard els elements diversos, després cablejant-la amb múltiples cables junts, i finalment la maqueta tota muntada i preparada per a la demostració.







5.3 Programa

Aquí està adjunt el programa:

```
import processing.serial.*;
// Importa la llibreria per llegir el port serie

import cc.arduino.*;
//Importa la llibreria per connectarse amb arduino

PImage img;      //declarem l'objecte de la imatge
PFont f;         //declarem l'objecte del tipus de lletra

boolean button1 = false; /*declarem la variable dels botons i
                           fem que comencin desactivats*/
boolean button2 = false;
boolean button3 = false;
boolean button4 = false;
boolean button5 = false;
boolean button6 = false;
boolean button7 = false;
boolean button8 = false;
boolean button9 = false;
boolean button10 = false;
boolean button11 = false;

int x1 = 80;     //Botó Persiana Saló Pujar
int y1 = 130;
int w1 = 35;
int h1 = 220;

int x2=80; //Botó Persiana Saló Baixar
int y2=350;
int w2=35;
int h2=220;
```




```
int x3 = 172; //Botó Temperatura casa
int y3 = 298;
int w3 = 40;
int h3 = 40;

int x4= 515; //Botó Llum Menjador
int y4= 420;
int w4= 40;
int h4= 40;

int x5= 879; //Botó Persiana nens l pujar
int y5= 607;
int w5= 96;
int h5= 20;

int x6= 879; //Botó Persiana nens l baixar
int y6= 586;
int w6= 96;
int h6= 20;

int x7= 677; //Botó Llum Pares
int y7= 486;
int w7= 40;
int h7= 40;

int x8= 170; //Botó Llum exterior
int y8= 630;
int w8= 50;
int h8= 50;
```



```
int x9= 1040; //Botó Persiana nens 2 pujar
int y9= 607;
int w9= 96;
int h9= 20;

int x10= 1039; //Botó Persiana nens 1 baixar
int y10= 586;
int w10= 96;
int h10= 20;

int x11= 1106; //Botó Activar Alarma
int y11= 29;
int w11= 51;
int h11= 20;

//Pins digitals

    //Pis superior

int PinLed2= 3; //Llum Interior Pares
int PinMotor11= 4; //Motor Persiana nens 1 puja
int PinMotor12= 5; //Motor Persiana nens 1 baixa
int PinMotor21= 6; //Motor Persiana nens 2 puja
int PinMotor22= 7; //Motor Persiana nens 2 baixa

    //Pis inferior
int PinMotor31= 10; //Motor Persiana Saló puja
int PinMotor32= 11; //Motor Persiana Saló baixa
int PinLed1= 9; //Llum Interior Menjador
int PinTimbre= 12; //Alarma
int PinLed3= 13; //Lum Exterior
```



```
//Pins analogics
int PinTermistor= 0;

//Variables de valor
float val_termistor;

Arduino arduino;

void setup() { //configuració del programa

    size(1276,696); //obrim un recuadre del tamany del planol
    img= loadImage("Planol.jpg"); //insertem la imatge del planol i la font
    f= loadFont("HighTowerText-Italic-40.vlw");

    arduino = new Arduino(this, Arduino.list()[0], 57600); //Busquem la nostra arduino
    arduino.pinMode(PinTermistor, Arduino.INPUT); //Assenyalem si son entrades o sortides
    arduino.pinMode(PinMotor11, Arduino.OUTPUT);
    arduino.pinMode(PinMotor12, Arduino.OUTPUT);
    arduino.pinMode(PinMotor21, Arduino.OUTPUT);
    arduino.pinMode(PinMotor22, Arduino.OUTPUT);
    arduino.pinMode(PinMotor31, Arduino.OUTPUT);
    arduino.pinMode(PinMotor32, Arduino.OUTPUT);
    arduino.pinMode(PinLed1, Arduino.OUTPUT);
    arduino.pinMode(PinLed2, Arduino.OUTPUT);
    arduino.pinMode(PinLed3, Arduino.OUTPUT);
    arduino.pinMode(PinTimbre, Arduino.OUTPUT);
}
```



```
void draw() { //Programa (el que es comprovarà constantment)
  background (0); //Fons negre
  image(img,0,0); //imatge per sobre el fons
  val_termistor= arduino.analogRead(PinTermistor); //llegim el valor del termistor
                                                    //i el posem dins la variable

  if (button1) { //Si cliquem el botó 1 :
    stroke(52,216,111); //pinta de verd el recuadre del botó 1
    arduino.digitalWrite(PinMotor31, Arduino.HIGH); //encen el motor "31" (motor de pujada)
  }
  else { //sinó
    stroke(255,0,0); //pinta de vermell el recuadre del botó 1
    arduino.digitalWrite(PinMotor31, Arduino.LOW); //apaga el motor "31"
  }
  noFill(); //no omplenis el recuadre dels colors
  rect(x1,y1,w1,h1); //marca el recuadre de coordenades x1,y1,w1,h1

  if (button2) { //el mateix que al botó 1 però pel botó 2 què es per baixar el motor
    stroke(52,216,111);
    arduino.digitalWrite(PinMotor32, Arduino.HIGH);
  }
  else {
    stroke(255,0,0);
    arduino.digitalWrite(PinMotor32, Arduino.LOW);
  }
  noFill();
  rect(x2,y2,w2,h2);
}
```



```
if (button3) {
  stroke(52,216,111);
}
else{
  stroke(255,0,0);
}

noFill();
rect(x3,y3,w3,h3);

/*Taula sensor de temperatura en la que hem calibrat el termistor per passar
de la resistència què ens dona a la temperatura què correspon*/

if ((button3)&&(720<val_termistor)&&(val_termistor<725)) {
  /*si es compleixen les tres condicions (botó 3 encès, i el valor del termistor
entre aquests paràmetres, llavors:*/
  textAlign(CENTER); //centra el text
  fill(0,0,0);
  textSize(38); //Tamany de la lletra 38
  text("Temperatura: 3°C", width/2-25, 60); /*escriu "Temperatura: 3°C"
a la meitat de la pantalla i 60 pixels per sota del màxim d'altura*/
}
else { //sinó no facis res
}
if ((button3)&&(715<val_termistor)&&(val_termistor<720)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 3.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(710<val_termistor)&&(val_termistor<715)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 4°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(705<val_termistor)&&(val_termistor<710)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 4.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(700<val_termistor)&&(val_termistor<705)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(695<val_termistor)&&(val_termistor<700)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 5.5°C", width/2-25, 60);  
}  
else {  
}
```




```
if ((button3)&&(690<val_termistor)&&(val_termistor<695)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 6°C", width/2-25, 60);
}
else {
}
if ((button3)&&(685<val_termistor)&&(val_termistor<690)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 6.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(680<val_termistor)&&(val_termistor<685)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 7°C", width/2-25, 60);
}
else {
}
if ((button3)&&(675<val_termistor)&&(val_termistor<680)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 7.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(670<val_termistor)&&(val_termistor<675)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 8°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(665<val_termistor)&&(val_termistor<670)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 8.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(660<val_termistor)&&(val_termistor<665)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 9°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(655<val_termistor)&&(val_termistor<660)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 9.5°C", width/2-25, 60);  
}  
else {  
}
```



```
if ((button3)&&(650<val_termistor)&&(val_termistor<655)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 10°C", width/2-25, 60);
}
else {
}
if ((button3)&&(645<val_termistor)&&(val_termistor<650)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 10.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(640<val_termistor)&&(val_termistor<645)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 11°C", width/2-25, 60);
}
else {
}
if ((button3)&&(635<val_termistor)&&(val_termistor<640)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 11.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(630<val_termistor)&&(val_termistor<635)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 12°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(625<val_termistor)&&(val_termistor<630)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 12.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(620<val_termistor)&&(val_termistor<625)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 13°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(615<val_termistor)&&(val_termistor<620)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 13.5°C", width/2-25, 60);  
}  
else {  
}
```



```
if ((button3)&&(610<val_termistor)&&(val_termistor<615)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 14°C", width/2-25, 60);
}
else {
}
if ((button3)&&(610<val_termistor)&&(val_termistor<615)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 14.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(605<val_termistor)&&(val_termistor<610)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 15°C", width/2-25, 60);
}
else {
}
if ((button3)&&(600<val_termistor)&&(val_termistor<605)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 15.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(595<val_termistor)&&(val_termistor<600)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 16°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(590<val_termistor)&&(val_termistor<595)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 16.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(585<val_termistor)&&(val_termistor<590)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 17°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(580<val_termistor)&&(val_termistor<585)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 17.5°C", width/2-25, 60);  
}  
else {  
}
```




```
if ((button3)&&(575<val_termistor)&&(val_termistor<580)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 18°C", width/2-25, 60);
}
else {
}

if ((button3)&&(570<val_termistor)&&(val_termistor<575)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 18.5°C", width/2-25, 60);
}
else {
}

if ((button3)&&(565<val_termistor)&&(val_termistor<570)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 19°C", width/2-25, 60);
}
else {
}

if ((button3)&&(560<val_termistor)&&(val_termistor<565)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 19.5°C", width/2-25, 60);
}
}
```



```
if ((button3)&&(555<val_termistor)&&(val_termistor<560)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 20°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(550<val_termistor)&&(val_termistor<555)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 20.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(545<val_termistor)&&(val_termistor<550)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 21°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(535<val_termistor)&&(val_termistor<540)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 21.5°C", width/2-25, 60);  
}  
else {  
}
```



```
if ((button3)&&(530<val_termistor)&&(val_termistor<535)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 22°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(525<val_termistor)&&(val_termistor<530)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 22.5°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(520<val_termistor)&&(val_termistor<525)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 23°C", width/2-25, 60);  
}  
else {  
}  
if ((button3)&&(515<val_termistor)&&(val_termistor<520)) {  
    textAlign(CENTER);  
    fill(0,0,0);  
    textSize(38);  
    text("Temperatura: 23.5°C", width/2-25, 60);  
}  
else {  
}  
}
```



```
if ((button3)&&(510<val_termistor)&&(val_termistor<515)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 24°C", width/2-25, 60);
}
else {
}
if ((button3)&&(505<val_termistor)&&(val_termistor<510)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 24.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(500<val_termistor)&&(val_termistor<505)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 25°C", width/2-25, 60);
}
else {
}
if ((button3)&&(495<val_termistor)&&(val_termistor<500)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 25.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(490<val_termistor)&&(val_termistor<495)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 26°C", width/2-25, 60);
}
else {
}
if ((button3)&&(485<val_termistor)&&(val_termistor<480)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 26.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(480<val_termistor)&&(val_termistor<485)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 27°C", width/2-25, 60);
}
else {
}
if ((button3)&&(475<val_termistor)&&(val_termistor<480)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 27.5°C", width/2-25, 60);
}
else {
}
```



```
if ((button3)&&(470<val_termistor)&&(val_termistor<475)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 28°C", width/2-25, 60);
}
else {
}
if ((button3)&&(465<val_termistor)&&(val_termistor<470)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 28.5°C", width/2-25, 60);
}
else {
}
if ((button3)&&(460<val_termistor)&&(val_termistor<465)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 29°C", width/2-25, 60);
}
else {
}
if ((button3)&&(455<val_termistor)&&(val_termistor<460)) {
    textAlign(CENTER);
    fill(0,0,0);
    textSize(38);
    text("Temperatura: 29.5°C", width/2-25, 60);
}
else {
}
```




```
if ((button3)&&(450<val_termistor)&&(val_termistor<455)) {
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura: 30°C", width/2-25, 60);
}
else {
}

//fi de taula de valors

if (button4) {
  stroke(52,216,111);
  arduino.digitalWrite(PinLed1, Arduino.HIGH);
}
else {
  stroke(255,0,0);
  arduino.digitalWrite(PinLed1, Arduino.LOW);
}
noFill();
rect(x4,y4,w4,h4);

if (button5) {
  stroke(52,216,111);
  arduino.digitalWrite(PinMotor11, Arduino.HIGH);
}
else {
  stroke(255,0,0);
  arduino.digitalWrite(PinMotor11, Arduino.LOW);
}
noFill();
rect(x5,y5,w5,h5);
```



```
if (button6) {  
  stroke(52,216,111);  
  arduino.digitalWrite(PinMotor12, Arduino.HIGH);  
}  
else {  
  stroke(255,0,0);  
  arduino.digitalWrite(PinMotor12, Arduino.LOW);  
}  
noFill();  
rect(x6,y6,w6,h6);  
  
if (button7) {  
  stroke(52,216,111);  
  arduino.digitalWrite(PinLed2, Arduino.HIGH);  
}  
else {  
  stroke(255,0,0);  
  arduino.digitalWrite(PinLed2, Arduino.LOW);  
}  
noFill();  
rect(x7,y7,w7,h7);
```



```
if (button8) {
  stroke(52,216,111);
  arduino.digitalWrite(PinLed3, Arduino.HIGH);
}
else {
  stroke(255,0,0);
  arduino.digitalWrite(PinLed3, Arduino.LOW);
}
noFill();
rect(x8,y8,w8,h8);

if (button9) {
  stroke(52,216,111);
  arduino.digitalWrite(PinMotor21, Arduino.HIGH);
}
else {
  stroke(255,0,0);
  arduino.digitalWrite(PinMotor21, Arduino.LOW);
}
noFill();
rect(x9,y9,w9,h9);

if (button10) {
  stroke(52,216,111);
  arduino.digitalWrite(PinMotor22, Arduino.HIGH);
}
else {
  stroke(255,0,0);
  arduino.digitalWrite(PinMotor22, Arduino.LOW);
}
noFill();
rect(x10,y10,w10,h10);
```



```
if (button11) {
  stroke(52,216,111);          //pinta de verd el botó
}
else{
  stroke(255,0,0); //pinta de vermell el botó
}
noFill();
rect(x11,y11,w11,h11);

if ((button11) && (val_termistor<450)) {          //botó alarma antiincendis
  stroke(52,216,111);          //pinta de verd el botó
  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura alta: Activant sistemes antiincendis", width/2-25, 60);
  arduino.digitalWrite(PinTimbre, Arduino.HIGH); //encen el timbre
}
else {          //si el botó no esta clicat
  stroke(255,0,0); //pinta de vermell el botó
  arduino.digitalWrite(PinTimbre, Arduino.LOW); //apaga el timbre
}

if ((button11) && (val_termistor>600)) {          //botó alarma antiincendis

  textAlign(CENTER);
  fill(0,0,0);
  textSize(38);
  text("Temperatura baixa: Activant calefacció", width/2-25, 60);
}
else {          //si el botó no esta clicat
}
```



```
}  
void mousePressed() {  
    if (mouseX > x1 && mouseX < x1 + w1 && mouseY > y1 && mouseY < y1 + h1 &&  
mousePressed){  
        button1 = !button1;  
    }  
    if (mouseX > x2 && mouseX < x2 + w2 && mouseY > y2 && mouseY < y2 + h2 &&  
mousePressed){  
        button2 = !button2;  
    }  
    if (mouseX > x3 && mouseX < x3 + w3 && mouseY > y3 && mouseY < y3 + h3 &&  
mousePressed){  
        button3 = !button3;  
    }  
    if (mouseX > x4 && mouseX < x4 + w4 && mouseY > y4 && mouseY < y4 + h4 &&  
mousePressed){  
        button4 = !button4;  
    }  
    if (mouseX > x5 && mouseX < x5 + w5 && mouseY > y5 && mouseY < y5 + h5 &&  
mousePressed){  
        button5 = !button5;  
    }  
    if (mouseX > x6 && mouseX < x6 + w6 && mouseY > y6 && mouseY < y6 + h6 &&  
mousePressed){  
        button6 = !button6;  
    }  
    if (mouseX > x7 && mouseX < x7 + w7 && mouseY > y7 && mouseY < y7 + h7 &&  
mousePressed){  
        button7 = !button7;  
    }  
    if (mouseX > x8 && mouseX < x8 + w8 && mouseY > y8 && mouseY < y8 + h8 &&  
mousePressed){  
        button8 = !button8;  
    }  
    if (mouseX > x9 && mouseX < x9 + w9 && mouseY > y9 && mouseY < y9 + h9 &&  
mousePressed){  
        button9 = !button9;  
    }  
    if (mouseX > x10 && mouseX < x10 + w10 && mouseY > y10 && mouseY < y10 + h10 &&  
mousePressed){  
        button10 = !button10;  
    }  
    if (mouseX > x11 && mouseX < x11 + w11 && mouseY > y11 && mouseY < y11 + h11 &&  
mousePressed){  
        button11 = !button11;  
    }  
}
```



6. Bibliografia

6.1 Llibres

6.1.1 En format PDF

Aquests els podreu trobar a la meva pàgina web (www.domoticaperlallar.blogspot.com) o bé me'ls podeu demanar contactant amb mi. Les meves dades de contacte apareixen a la pàgina web.

1. Gerhart, J: *"Home Automation and wiring"* (Mc Graw-Hill Professional, 1999)
2. Gavin Smith: *"Arduino Cheat Sheet"*
3. Jonathan Oxeer y Hugh Blemings: *"Practical Arduino"*
4. Daniel Shiffman: *"Learning Processing"*
5. Ignacio Arata, Francisco Arrufat, Pablo Palacios i Santiago Folie: *"Variaciones de una resistencia con la temperatura"*

6.1.2 En format paper

Aquests són llibres de consulta de la biblioteca o de familiars que em van deixar per solucionar algun dels molts problemes que van sorgir al llarg del treball.

1. Ben Fry & Casey Reas: *"Getting started with Processing"*
2. Juan M^a Sarrió García: Manual de ayuda de Processing (adaptació de *"Processing: A programming handbook for visual designers and artists"*)



6.2 Pàgines web

1. http://www.smarthome.com/about_x10.html
2. <http://arduino.cc/forum/index.php?>
3. <http://www.hispavila.com/3ds/atmega/motorescc.html>
4. <http://www.casadomo.com/noticiasDetalle.aspx?idm=10&id=10907&c=6>
5. <http://www.casadomo.com/noticiasDetalle.aspx?c=14>
6. <http://pscmpf.blogspot.com.es/2008/12/arduino-lm35-sensor.html>
7. <http://miqueridopinwino.blogspot.com.es/2012/07/como-utilizar-un-termistor-o-sensor-de.html>
8. <http://tushev.org/articles/electronics/42-how-it-works-ds18b20-and-arduino>
9. <http://itmanageratschool.blogspot.com.es/2012/05/control-de-temperatura-usando-arduino-y.html>
10. <http://www.wikipedia.org/>
11. http://robots-argentina.com.ar/MotorCC_L293D.htm
12. <https://sites.google.com/site/tecnologiavilatzara/home/altresprojectes>
13. <https://www.underground.org.mx/index.php?topic=24860.0>
14. <http://viver.disca.upv.es/~arduino/documentacion/sensores/>
15. <http://www.jeremyblum.com/category/arduino-tutorials/>

6.3 Agraïments als assessors

Agraeixo profundament l'ajuda que m'han brindat les següents persones, que han fet possible continuar el treball en els entrebancs i els moments més difícils:

Elisabet Alsina - Enginyeria en Telecomunicacions

Joan Alós – Enginyeria Agrònoma

Jaume Serra – Física i càtedra en matemàtiques

Roberto Novo Ramudo - Enginyeria en Telecomunicacions

Joan Pellicer - Tecnòleg de l'IES Cendrassos de Figueres.