

## **Treball final de grau**

**Estudi: Grau en Enginyeria Electrònica Industrial i Automàtica**

**Títol: Domotització d'una vivenda mitjançant una xarxa mallada mesh**

**Document: 1. Memòria**

**Alumne: Abdelaziz Essami**

**Tutor: Albert Figueras Coma**

**Departament: Enginyeria Elèctrica, Electrònica i Automàtica**

**Àrea: Enginyeria de Sistemes i Automàtica**

**Convocatòria (mes/any): Setembre / 2021**

**ÍNDEX**

1. INTRODUCCIÓ .....	4
1.1. Antecedents .....	4
1.2. Objecte.....	4
1.3. Especificacions i abast .....	4
2. DISPOSITIUS DEL SISTEMA DOMÒTIC .....	6
2.1. Fonts d'alimentació .....	6
2.1.1. Adaptador de corrent alterna.....	6
2.1.2. Regulador de voltatge .....	8
2.1.3. Bateries .....	9
2.2. Xarxa sensorial.....	11
2.2.1. Característiques dels mòduls .....	12
2.2.2. Funcions dels mòduls .....	13
2.2.3. Topologia de la xarxa.....	14
2.3. Microcontroladors.....	15
2.3.1. Característiques principals.....	16
2.3.2. Funcions dels microcontroladors.....	16
2.4. Controlador .....	17
2.4.1. Característiques principals.....	18
2.4.2. Funcions del controlador local.....	19
2.5. Sensors .....	20
2.5.1. Sensor de monòxid de carboni.....	21
2.5.2. Sensor de presència .....	22
2.5.3. Sensor de humitat.....	23
2.5.4. Sensor de temperatura .....	24
2.5.5. Sensor de lluminositat.....	24
2.6. Actuadors .....	25
2.6.1. LED.....	26
2.6.2. Relé .....	26

2.6.3. Alarma sonora .....	27
2.7. Altres components del sistema.....	28
2.7.1. Router WiFi.....	28
2.7.2. Programes .....	29
3. PROGRAMACIÓ .....	30
3.1. Mòduls XBee.....	31
3.2. Nodes.....	34
3.2.1. Node coordinador .....	35
3.2.2. Node actuator .....	36
3.3. Controlador .....	36
3.3.1. Broker MQTT .....	37
3.3.2. OpenHAB.....	39
3.3.3. Automatitzacions.....	43
3.3.4. Base de dades.....	47
4. COMUNICACIÓ .....	49
4.1. Zigbee .....	49
4.1.1. Característiques.....	49
4.1.2. Interferències .....	51
4.2. OpenHAB .....	52
4.2.1. Base de dades .....	52
4.2.2. Interfície gràfica d'usuari .....	53
4.2.3. Aplicació Telegram .....	57
4.2.4. Aplicació mòbil per l'accés remot .....	58
5. RESUM DEL PRESSUPOST.....	60
6. CONCLUSIONS .....	61
7. RELACIÓ DE DOCUMENTS .....	62
8. BIBLIOGRAFIA.....	63
9. GLOSSARI .....	64
A. CODI INFORMÀTIC .....	66

A.1. Node coordinador .....	66
A.2. Node actuador .....	79
A.3. OpenHAB.....	90
A.3.1. Rules .....	90
B. CÀLCULS.....	101
B.1. Càlcul de les bateries.....	101
B.2. Càlcul dels sensors.....	101
B.2.1. Càlcul del sensor de monòxid de carboni .....	102
B.2.2. Càlcul del sensor de humitat.....	103
B.2.3. Càlcul del sensor de temperatura .....	104
B.3. Càlcul dels divisors de tensió.....	104
B.4. Càlcul dels relés.....	105
C. INSTAL·LACIÓ ELÈCTRICA.....	107
C.1. Generalitats .....	107
C.2. Previsió de càrregues .....	107
C.3. Condicions de subministrament .....	109
C.4. Empresa subministradora .....	109
C.5. Escomesa i caixa general de protecció .....	109
C.6. Línia repartidora.....	110
C.7. Conjunt de protecció i mesura .....	111
C.8. Derivació individual.....	112
C.9. Interruptor control de potència i interruptor general automàtic .....	113
C.10. Quadre general de protecció i comandament.....	113
C.11. Distribució general .....	113
C.12. Posada a terra .....	114
C.13. Característiques de la instal·lació elèctrica .....	115

## **1. INTRODUCCIÓ**

### **1.1. Antecedents**

El desenvolupament dels serveis en el núvol, el baix cost de maquinari i la millora dels protocols sens fils han creat moltes oportunitats de millora per les vivendes domòtiques i la expansió del IoT.

### **1.2. Objecte**

L'objecte és la realització del disseny de la domotització d'una vivenda de baix cost utilitzant el fog computing, o altrament dit, la computació en la boira. Aquesta tecnologia habilita la comunicació entre els dispositius IoT i el núvol mitjançant un controlador localitzat entremig d'aquests. Aquest farà de capa intermèdia en la xarxa on es decidirà quins anàlisis es faran amb les dades, quines es processaran localment, quines s'enviaran cap al núvol i quines ens interessarà emmagatzemar.

D'aquesta forma aconseguirem escurçar les vies de comunicació entre el núvol i els dispositius IoT, o sigui millorar la latència, i reduir el cabal de dades per no saturar la xarxa. També aconseguirem augmentar considerablement la seguretat i privacitat degut a que ara els dispositius IoT no estaran directament exposats al món exterior. Hi haurà un node coordinador que farà de mestre per tractar les dades dels sensors i un node pels actuadors. Els mòduls dels sensors formaran una xarxa mallada entre ells amb l'ajuda de routers per tal d'augmentar la robustesa i fiabilitat de la instal·lació domòtica.

### **1.3. Especificacions i abast**

Respecte l'abast, la capa que s'encarregarà de la computació en la boira estarà formada per un ordinador de placa simple que actuarà com a controlador i passarel·la, tractarà les dades mitjançant un broker MQTT i les enviarà cap a la interfície gràfica d'usuari, la qual estarà feta amb una aplicació de codi obert desenvolupada en Java i que basa el seu entorn de programació sobre el llenguatge Xbase.

El protocol de comunicació entre el controlador i els nodes serà el MQTT, aquest és un protocol de missatgeria i està dissenyat amb un transport de missatgeria de publicació-subscripció extremadament lleuger que és ideal per connectar dispositius IoT. Aquests nodes

utilitzaran microcontroladors equipats amb WiFi de baix cost programats en llenguatge basat en C++. La comunicació entre els mòduls de sensors i el node coordinador es farà amb el protocol Zigbee.

El present projecte s'ha estructurat en tres grans blocs per facilitar la seva comprensió, en primer lloc es realitzarà una exposició de les característiques principals del maquinari utilitzat en el sistema domòtic. Hi hauran dos tipus de maquinari, el principal i el secundari, el principal es dividirà en tres seccions que són, xarxa sensorial, microcontroladors, i controlador local. Mentre que el maquinari secundari consistirà en els sensors, actuadors i altres elements necessaris pel funcionament de la instal·lació.

En segon lloc, es farà una explicació de com s'ha decidit programar el maquinari principal esmentat anteriorment i finalment només quedarà detallar de quina manera es comunicarà l'usuari amb el control domòtic i totes les funcionalitats que li ofereix aquest.

## 2. DISPOSITIUS DEL SISTEMA DOMÒTIC

La domòtica és la tecnologia que automatitza les funcions i instal·lacions d'habitacles o edificis per augmentar el confort, la seguretat i l'estalvi d'energia, basant-se en els nous avenços produïts principalment en tres camps, la electrònica, la informàtica i les comunicacions, que s'integren en un únic sistema aplicat sobre l'edifici que es vol gestionar.

Una de les tecnologies que ha tingut més èxit en l'àmbit de la domòtica és la computació en el núvol, aquesta es basa en treure la càrrega de computació que necessiten els dispositius del sistema domòtic i traspasar-la al núvol. En vivendes on s'exigeix una resposta ràpida d'actuació i baixa sobrecàrrega de comunicacions, sobretot en casos on la xarxa WiFi local crea un tràfic elevat, hem de buscar alternatives a la computació en el núvol, en aquests casos és on entra en escena la computació en la boira.

Aquesta tecnologia trasllada la capacitat de computació i comunicació a prop dels nodes físics del sistema domòtic, d'aquesta forma minimitzem la latència, distribuïm els recursos computacionals i d'emmagatzematge, millorem la mobilitat i localització dels dispositius, facilitem la escalabilitat de la xarxa i augmentem la seguretat gràcies al controlador que farà de passarel·la de boira. Hi haurà un node per la coordinació i tractament de dades sensorials, un pels actuadors i els mòduls necessaris per fer l'adquisició de dades sensorials. A més del controlador local que actuarà com a passarel·la, interfície gràfica d'usuari i broker MQTT.

### 2.1. Fonts d'alimentació

Un dels elements més importants en tota instal·lació és la alimentació, sobretot en el nostre cas, on hem de domotitzar una vivenda amb diferents tipus de circuits. Alguns elements necessitaran alimentació constant i continua mentre que altres es podran alimentar amb bateries, dins d'aquests hi hauran alguns que necessitaran més o menys tensió d'alimentació depenent amb quins tipus de sensors aniran equipats. Seguint la documentació del mòdul XBee series 2, el microcontrolador ESP8266 i la Raspberry Pi, necessitarem una tensió continua d'entre 2,1-3,6 V pel mòdul XBee mentre que pels altres 2 components una de 5 V.

#### 2.1.1. Adaptador de corrent alterna

Un adaptador de corrent alterna, carregador o convertidor AC/DC és un dispositiu elèctric compost d'un transformador que redueix el voltatge de 230 V d'una xarxa domèstica a un valor

menor, d'un rectificador d'ona completa que permet convertir el voltatge altern de 50 Hz de la xarxa en voltatge continu amb un cert rissat residual, i d'un filtre que permet eliminar aquest rissat. Més recentment, nous models d'adaptadors emprenen transistors com a interruptors per a obtenir una ona quadrada a l'entrada, aconseguint d'aquesta manera un adaptador de menor grandària i una major eficiència energètica.

S'ha optat per aquest tipus de fonts d'alimentació perquè només requereixen d'un endoll domèstic per poder funcionar i les corrents que entreguen són més que suficient pels nostres dispositius, a més no ens produiran problemes electromagnètics com els de les fonts commutades. Les seves dimensions són reduïdes respecte altres fonts d'alimentació que requereixen de cablejat excessiu pel seu funcionament. En la següent figura podem veure l'adaptador en qüestió que s'ha triat.



Figura 1. Adaptador de corrent alterna amb entrada jack 2,5 mm

Aquest adaptador s'utilitzarà en tots aquells elements que necessiten tensió continua i constant fixe, com són el node d'actuadors de relés, el node coordinador, els mòduls routers i mòdul de la cuina el qual ha d'estar monitoritzant en tot moment el nivell de monòxid de carboni. Tots aquests són elements que necessitem estar alimentats en tot moment perquè el sistema domòtic funcioni correctament.

Aquest adaptador aportarà una tensió continua de 5 V pel node d'actuadors i el node coordinador, i en cas dels mòduls routers i el de la cuina, s'utilitzarà un regulador de voltatge per reduir la tensió fins a 3,3 V.



### 2.1.2. Regulador de voltatge

Com s'ha explicat anteriorment, necessitem convertir la tensió de sortida del adaptador de corrent fins a 3,3 V pels mòduls XBee, hi han dues alternatives principals per tal de fer aquesta conversió.

La primera opció seria amb un regulador de tensió lineal, aquests circuits permeten mantenir un voltatge constant a la sortida per a entrades de voltatge variables. El seu funcionament es basa en actuar com una resistència variable, ajustant un divisor de tensió per a mantenir sempre la sortida a un voltatge concret. L'energia es dissipa en forma de calor, per la qual cosa és un mètode més ineficient que un convertidor magnètic, i a més només poden obtenir tensions de sortida menors que a l'entrada, i aquest voltatge d'entrada ha de ser superior en una certa quantitat per a poder operar correctament, amb el qual necessitarem un adaptador de corrent d'una tensió superior als 5 V. A canvi, són molt barats, senzills i petits, i no generen cap soroll.

La segona opció seria amb un convertidor DC/DC magnètic, mitjançant un circuit commutador i bobines es poden implementar circuits que permeten reduir o augmentar el voltatge de la entrada amb una alta eficiència energètica, aquests s'anomenen reguladors commutats reductors o elevadors. Tenen major rendiment de conversió, ja que els transistors funcionen en commutació, reduint així la potència dissipada en aquests i la mida dels dissipadors, un desavantatge és la producció de soroll electromagnètic produït per la commutació a freqüències elevades, havent d'apantallar i dissenyar correctament la PCB.

En el nostre cas com ens interessa utilitzar la tensió de 5 V del adaptador per alimentar alguns sensors i a més volem simplificar el disseny dels PCB per estalviar costos i augmentar la eficiència energètica, ens hem decantat per un regulador commutat LM7803 degut a que és el més barat i simple, sempre s'assegurarà que la font continua en tot moment subministri a l'entrada un voltatge constant i superior que el de la sortida en una quantitat suficient com per assegurar el correcte funcionament, només necessitarem afegir uns condensadors de desacoblament per protegir el circuit de possible soroll i millorar la estabilitat. El voltatge d'entrada serà de 5 V i el regulador ens aportarà un voltatge de sortida de 3,3 V, en la següent figura podem veure el regulador triat en qüestió.



Figura 2. Regulador de voltatge commutat LM7803

Els condensadors hauran de ser ceràmics i amb un valor de ESR, o altrament dit resistència sèrie equivalent, petit. Han d'estar connectats el més a prop possible de la entrada i sortida del regulador per evitar la formació de inductàncies parasites, d'aquesta forma mantenim al mínim el soroll generat pel convertidor. També s'han utilitzat PCB de quatre capes amb una distribució adequada de plans de tensió, terra i senyal per tal de disminuir els efectes electromagnètics negatius i protegir els circuits més davant de curtcircuits.

### 2.1.3. Bateries

Les bateries són una forma barata i ràpida d'alimentar dispositius sense la necessitat d'utilitzar endolls, aconseguint així dispositius mòbils que es poden ubicar gairebé en qualsevol lloc. Hi ha dos tipus principals de bateries, primàries o d'un sol ús, que s'esgoten després del seu ús i no poden recuperar la seva càrrega, i secundàries o recarregables, que mitjançant elements químics diferents sí que permeten tornar a emmagatzemar energia un nombre limitat de vegades, encara que a costa d'un menor voltatge a la sortida.

El principal problema de les bateries és la seva durada, ja que acaben esgotant-se després del seu ús. Cada bateria té una capacitat expressada en mAh, que indica la quantitat de càrrega elèctrica que pot lliurar al voltatge nominal. Aquesta capacitat depèn de la grandària de la pila, dels seus components, del voltatge i també de la càrrega a la qual està sotmesa, i és un factor determinant en la viabilitat de l'ús d'aquesta alimentació.

En el nostre cas com necessitarem alimentar els mòduls XBee i els sensors amb els quals aniran equipats, utilitzarem dues piles AA de 1,5 V per aconseguir un total de 3 V. Aquesta tensió està dins el rang d'operativitat del mòdul XBee i de la majoria de sensors que utilitzarem, però en el cas del mòduls equipats amb sensors de presència PIR, haurem d'afegir una altra

pila fent un total de tres piles amb 4,5 V. En la taula següent podem veure les duracions aproximades de quatre tipus de piles diferents.

Bateria	Voltatge(V)	Capacitat(mAh)	Duració segons mode de funcionament (h)		
			Transmissió a 40 mA	Inactiu a 15 mA	Hibernació a 1 $\mu$ A
AA Alcalina	1,50	2.850	49,88	133,00	7.556,82
AAA Alcalina	1,50	1.200	21,00	56,00	3.181,82
AA NiMH	1,20	2.900	50,75	135,33	7.689,39
AAA NiMH	1,20	1.000	17,50	46,67	2.651,52

Taula 1. Duració en hores dels diferents estats d'un mòdul XBee alimentat amb piles

Només s'han estudiat aquestes quatre variants de piles degut a les seves dimensions i baix preu, tenen dimensions bastant reduïdes en comparació amb altres i ens simplifiquen el disseny del projecte. S'utilitzaran piles AA de tipus primàries com les de la figura següent ja que són les que ens donen una major duració de hores de funcionament a un menor preu, les recarregables són més cares i brinden gairebé la mateixa duració.



Figura 3. Piles AA Alcalines Duracell de 1,5 V

Els càlculs de la taula anterior només tenen en compte el consums del mòdul XBee, un cop s'afegeixin els sensors aquesta duració baixarà òbviament, els càlculs en el mode hibernació s'han fet tenint en compte que els mòduls amb piles hibernen cinc minuts i es desperten durant uns dos segons per entrar en mode transmissió, després d'enviar les dades dels sensors tornen a hibernar. Un cop el LED indicador de les piles es vagi apagant, haurem de canviar aquestes per unes noves, però això no serà un problema ja que ho farem cada uns quants mesos degut a la gran duració que ens brinda el mode de hibernació dels mòduls XBee.

Els mòduls XBee tenen dues formes de sortir de l'estat de hibernació, un cop s'acabi el període de hibernació de cinc minuts i també si es detecta moviment amb els sensors PIR. Degut a

que no se sap quantes vegades s'activaran els sensors PIR, només s'ha fet el càlcul del primer cas.

## 2.2. Xarxa sensorial

Els mòduls XBee són l'opció triada per a realitzar la implementació de la xarxa de sensors. Aquests mòduls són perfectes per a cobrir les necessitats del projecte, ja que com es descriurà amb més detall a continuació, compten amb la capacitat de recollir directament informació de sensors digitals i analògics, i transmetre-la a altres mòduls o nodes sense necessitat d'emprar microcontroladors addicionals. Així mateix, també posseeixen sortides digitals i de PWM que els permeten realitzar actuacions bàsiques per si sols, encara que en el nostre cas només s'utilitzaran per obtenir adquirir dades sensorials.

Els principals avantatges d'utilitzar aquests mòduls són les seves dimensions reduïdes i la seva simplicitat, la qual cosa els hi permet ser emprats en una gran varietat de situacions i en espais molt reduïts. Un altre punt clau és el seu molt baix consum energètic i estalvi econòmic al emprar menor nombre de components pel seu funcionament. Per les necessitats de la nostra xarxa domèstica, els mòduls XBee són una opció molt interessant i versàtil.

Només necessitarem components addicionals alhora d'enviar les dades recollides per ser tractades, ja que els mòduls entre ells estan limitats a simplement enviar i rebre dades o canviar remotament l'estat dels seus pins. Això es farà al node coordinador que estarà format per mòdul XBee i un microcontrolador ESP8266. Aquests es trobaran connectats amb un protocol serial asíncron mitjançant els seus pins de transmissió i recepció corresponents, i a mesura que el microcontrolador vagi rebent les dades, aquest les anirà tractant i enviant mitjançant el protocol MQTT cap al controlador local. En la següent figura podem veure com quedaria la xarxa de sensors amb el node coordinador.

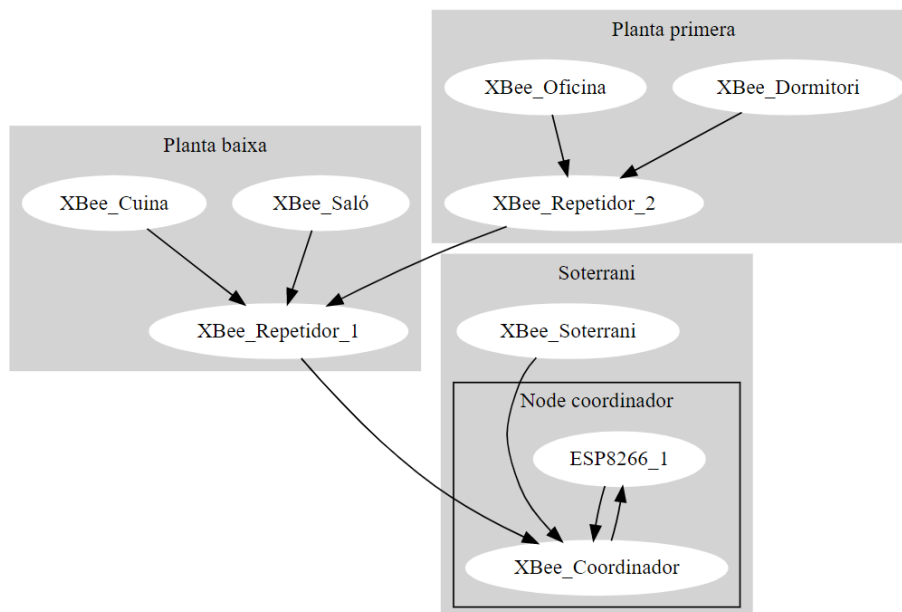


Figura 4. Esquema per plantes de la xarxa sensorial de la vivenda

Un punt molt important a destacar és, en l'esquema anterior només s'han indicat les rutes més clares i estrictes per on passaran les dades, ja que si s'afegien les rutes que es crearien un cop la xarxa mallada s'activés, es saturaria l'esquema de fletxes.

### 2.2.1. Característiques dels mòduls

Actualment hi han tres series del mòduls XBee, la sèrie més coneguda i utilitzada és la series 2. Dins d'aquesta hi han dues versions, la versió estàndard i la PRO. En la següent figura podem veure el model concret de XBee que s'ha triat, aquest correspon a un XBee series 2 estàndard, concretament el XB24-ZB RF. El protocol de comunicació d'aquests mòduls és el ZigBee, més endavant veurem de que tracta i justificarem la seva tria.



Figura 5. Mòdul XBee Series 2 amb antena whip o fuet

Seguidament veurem alguns de les seves característiques tècniques. Els mòduls d'aquesta versió i model operen en la banda de freqüència de 2,4 GHz i es poden alimentar amb un font continua d'una tensió compresa entre 2,1 i 3,6 V, mentre que el model PRO demana un mínim de 3 V per poder alimentar-se. En segon lloc, aquest model té un alt rendiment amb baix cost, el rang d'abast en entorns interiors o urbans és de fins a 40 metres, aconseguint els 120 metres en exteriors i amb línia de visió directa. El XBee PRO té un rang major, uns 100 metres en interior i fins a 1,6 kilòmetres en exterior i amb línia directa de visió.

Tercerament es trobaria el baix consum energètic, el mòdul XBee té un consum màxim de corrent de 40 mA a 3,3 V. Si no està executant cap tasca però està encès, el seu consum és de 15 mA, però si es configura per a entrar en mode d'hibernació, el consum en aquest estat és menor d'1  $\mu$ A. El model PRO té un consum major d'energia a causa de les seves millorades prestacions.

Finalment podríem parlar de la creació avançada de xarxes i alta seguretat, el sistema d'interconnexió dels mòduls compta amb processos de reintent i confirmació de recepció, múltiples direccions disponibles, admet topologies arbre, estrella i malla, a més té la capacitat autònoma i automàtica d'encaminar, reparar i tolerar fallades en les vies de comunicació.

### 2.2.2. Funcions dels mòduls

Les xarxes ZigBee ens permeten estalviar-nos l'esforç de cablejar gràcies a la seva comunicació sense fil, això també ens permet reubicar actuadors o sensors fàcilment. Aquestes xarxes compten amb fins a tres tipus d'elements, el coordinador, un o més routers i un o més end devices.

Un coordinador és el que s'encarrega de crear la xarxa, administrar les direccions i dur a terme funcions de seguretat i manteniment de la xarxa. En el nostre cas, aquest és el que recollirà totes les dades dels sensors i les enviarà cap a un microcontrolador per ser tractades. És imprescindible que el coordinador estigui directament alimentat a la xarxa elèctrica, ja que aquest és el que manté habilitada la xarxa sensorial en tot moment.

Els routers són els dispositius que s'ocupen de la comunicació entre tots els dispositius de la xarxa. Aquests han de triar quina és la millor ruta per a comunicar dos punts prou allunyats com per a impossibilitar la seva connexió directa i gestionar aquesta comunicació. Aquests dispositius solen estar alimentats per una presa de corrent ja que el seu funcionament és

imprescindible i és necessari que estiguin sempre funcionant. Ja que la vivenda està composta per tres plantes, hi haurà un mòdul router en la planta baixa i en la primera.

Els end devices actuen com a terminacions d'una xarxa, i les seves capacitats inclouen unir-se a xarxes, enviar dades mostrejades o rebre comandes per activar sortides. Com no necessiten estar contínuament en funcionament, requereixen d'un maquinari més reduït i amb menor consum energètic. Per a això, els end devices poden entrar periòdicament en mode d'estalvi d'energia, o altrament dit, mode hibernació. Tot End device necessita un router o coordinador com a dispositiu mestre, el qual ajuda als dispositius a unir-se a les xarxes i emmagatzema els missatges dirigits a ells en cas que estiguin en estat d'hibernació. Disposem de cinc estàncies per domotitzar, en total hi hauran cinc mòduls fent de end devices.

Els mòduls routers també poden tenir incorporats sensors i enviar les seves dades mostrejades cap altres mòduls, d'aquesta forma farien dues funcions simultàniament i podríem aprofitar les seves entrades i sortides per afegir algun actuator o sensor en específic.

### 2.2.3. Topologia de la xarxa

El protocol ZigBee treballa amb diverses topologies de xarxa per a interconnectar els mòduls existents, aquestes són la estrella, arbre i malla que es troben representades en la figura següent.

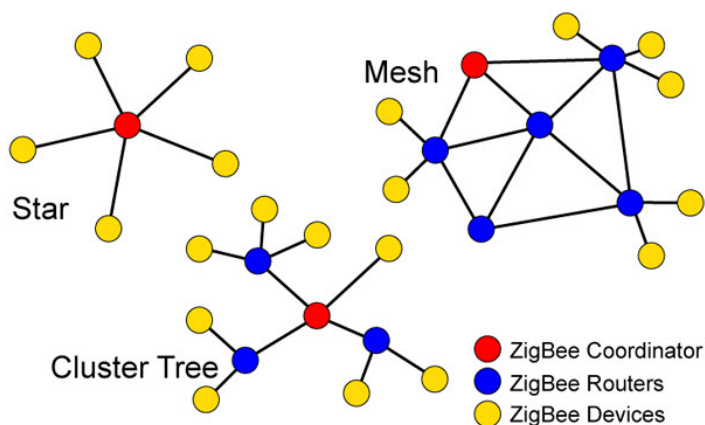


Figura 6. Topologia de les xarxes ZigBee

En el nostre cas s'ha triat la topologia en malla o altrament dit mesh, aquesta variant és la més fiable ja que cada node està connectat a diversos nodes i no a un només de manera que, si un dels routers fallés, el coordinador establirà una nova ruta perquè la comunicació no

s'interrompi. A més, gràcies a que és una xarxa sense fil, supera el principal desavantatge d'aquesta topologia que és l'excessiu cablejat, sobretot si estem parlant d'una instal·lació domòtica en una vivenda amb moltes habitacions o plantes.

Tindrem els mòduls localitzats en les estàncies que es volen domotitzar per tal d'obtenir informació i poder actuar segons aquesta en les tasques d'automatització corresponents. Per augmentar la robustesa i fiabilitat de la xarxa s'afegiran mòduls routers en la planta baixa i en la primera per tal de que les dades puguin arribar correctament al node coordinador, aquesta formació desencadenarà en una xarxa de tipus malla.

### 2.3. Microcontroladors

Pel que fa al transceptor WiFi, fins fa poc, les comunicacions basades en WiFi eren relativament cares, però el llançament de transceptors com el ESP8266 han suposat una revolució en l'ecosistema IoT.

El NodeMCU, que es troba a la figura següent, és una plataforma oberta de IoT que inclou tant programari, funcionant sobre el SoC ESP8266, desenvolupat per Espressif System, com a maquinari, basat en el mòdul ESP-12. Aquest presenta una solució completa de WiFi en un SoC integrat desenvolupat per a satisfer les necessitats dels usuaris en implementacions d'eficiència energètica i dissenys compactes relacionats directament amb el IoT.



Figura 7. NodeMCU V3 amb el SoC ESP8266

A més, ofereix més avantatges com la incorporació d'un regulador de tensió integrat de 3,3 V, així com un port USB de programació. Es pot programar mitjançant el IDE de Arduino que és



la forma més popular de fer-ho i és la qual ha estat empleada per a l'elaboració d'aquest projecte.

### 2.3.1. Característiques principals

Aquesta placa pot alimentar-se amb una tensió continua d'entre 3 i 9 V, les entrades i sortides funcionen a una tensió nominal de 3,3 V però estan protegides i poden admetre fins a 6 V de senyal segons el seu manual, en total tenim 13 entrades i sortides amb les quals podem afegir multitud de sensors o actuadors. També compta amb un convertidor analògic digital de 10 bits però en aquest projecte no en fem cap ús d'aquest.

El regulador integrat ens aporta una tensió de 3,3 V en tres pins diferents, això ens facilita la alimentació d'altres dispositius com els mòduls XBee. El seu consum varia depenent de l'ample de banda del senyal WiFi en el qual està transmetent, en els pitjors dels casos pot arribar fins als 170 mA mentre que en els millors casos baixa fins als 50 mA.

### 2.3.2. Funcions dels microcontroladors

Hi hauran dos microcontroladors com aquest en total la instal·lació, un pel node actuator i l'altre pel node coordinador. En el node coordinador, aquest microcontrolador només estarà connectat al mòdul XBee coordinador bàsicament, i el seu programari consistirà en tractar les dades dels sensors i enviar-les cap al controlador local mitjançant el broker MQTT. Es farà un primer filtratge de dades en el mateix microcontrolador i un més endavant que ho farà el broker MQTT, així les dades que hi arribaran al controlador local, o sigui a la interfície d'usuari, seran els que més ens interessaran. Aquest és el principi de funcionament de la computació en la boira, on el controlador local és el principal protagonista.

En el node actuator, hi haurà un maquinari molt més extens que en el cas anterior, en aquest node tindrem el circuit de control i potència que s'encarregarà de commutar les càrregues elèctriques de la vivenda. Rebrà les ordres des de la interfície d'usuari manualment o bé després que s'hagi activat alguna tasca automatitzada, mitjançant el protocol MQTT. Aquest node es trobarà el més a prop possible del quadre elèctric per tal d'empalmar les línies elèctriques amb els relés, per a cada element que es vulgui controlar hi haurà una línia elèctrica pertinent. En la següent figura podem observar com els dos nodes descrits anteriorment i el controlador local estan interconnectats entre ells.

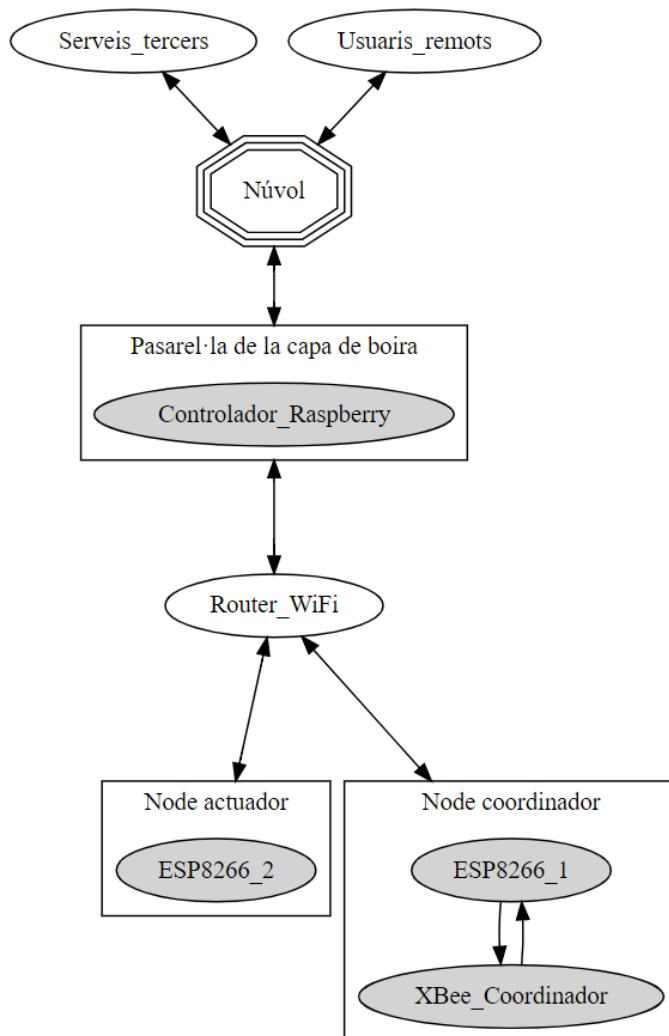


Figura 8. Esquema de la interconnexió del controlador amb els nodes i el núvol

## 2.4. Controlador

El controlador local domòtic és el responsable d'establir les comunicacions entre els dos nodes i l'usuari. Els seus requisits de maquinari són realment reduïts, ja que només ha de ser capaç d'executar el programari necessari per fer anar el servidor MQTT i oferir una interfície gràfica per l'usuari. Tenint en compte aquestes restriccions, gairebé qualsevol ordinador modern connectat a la xarxa podria ser utilitzat com a controlador, no obstant això, hi ha una àmplia gamma de dispositius integrats anomenats SBC, ordinador de placa única, que normalment inclouen un processador amb la necessària potència de processament i capacitat d'emmagatzematge, a més de tenir dimensions petites, baix cost i consum d'energia reduït.

En l'actualitat es poden trobar diversos models d'ordinadors d'aquest tipus, però per a la realització d'aquest projecte ens centrarem en la Raspberry Pi ja que és la opció més utilitzada de totes i de les més econòmiques.

#### 2.4.1. Característiques principals

La Raspberry Pi és un ordinador de placa única de baix cost que, en la seva tercera versió i model B, està compost per un processador Broadcom a 1,20GHz de velocitat de rellotge i una RAM de 1GB. La seva major novetat respecte als models anteriors va ser la inclusió de WiFi i Bluetooth 4.1 Low Energy sense necessitat d'adaptadors, per la connexió a la xarxa també es disposa de Ethernet per a endollar un cable RJ-45. En la següent figura la podem veure representada des de una vista superior.



Figura 9. Raspberry Pi 3 model B

Per al seu funcionament, n'hi ha prou amb afegir un mitjà d'emmagatzematge com potser per exemple una targeta de memòria SD i endollar-ho al corrent mitjançant qualsevol carregador de tipus micro USB. S'ha escollit el model B de la Raspberry Pi 3 degut a que és la més estesa i comunament utilitzada, i sobretot per la seva disposició al mercat.

En la següent figura podem observar els consums energètics dels diferents models de Raspberry Pi, els quals han sigut mesurats en quatre comprovacions diferents. S'ha mesurat el consum en estat de repòs, quan càrrega la interfície gràfica, amb visualització d'un vídeo amb una resolució de 1080p i amb la realització d'una captura de vídeo de 1080p.

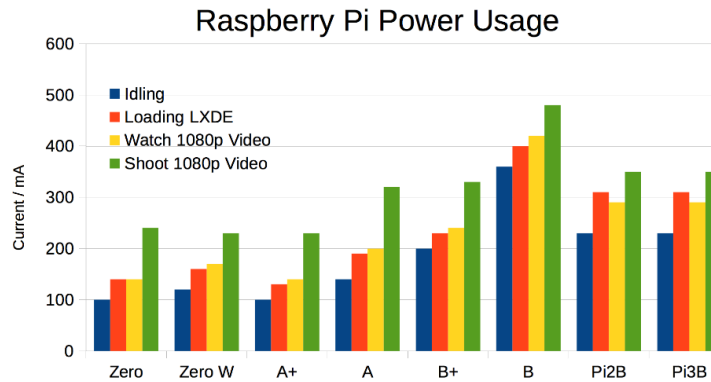


Figura 10. Consums energètics de les Raspberry Pi

Veiem que els consums dels models Pi2B i Pi3B són més elevats que els dels models més antics i els models Zero i Zero W, encara que aquests últims estan destinats a aplicacions de baixa demanda i consumeixen menys degut a que el seu maquinari és menys potent. S'ha de dir també que a mesura que van augmentant les prestacions també augmenta el consum, però com es veu en la figura anterior els dos últims models aconseguen millors prestacions sense haver de pujar gaire el consum, així que podem dir que són bastant eficients.

Un altre punt a destacar és que aquest model triat té suficients recursos per respondre ràpidament als esdeveniments i peticions que es produeixen en un sistema domòtic habitual. Altres models més antics presenten temps de resposta i arrencada més lents i es podrien saturar fàcilment.

#### 2.4.2. Funcions del controlador local

Aquest controlador serà l'encarregat de fer anar la interfície gràfica d'usuari i el broker MQTT, estarà connectat a la xarxa WiFi domèstica mitjançant el router i tindrà una IP estàtica com en el cas dels dos microcontroladors que fan de node actuador i node coordinador. Per crear la interfície d'usuari s'ha d'instal·lar el programari de codi obert dedicat exclusivament a les automatitzacions domèstiques OpenHAB i el broker mosquitto com a servidor MQTT, més endavant veurem què tracten i perquè s'han escollit aquests en qüestió.

Per les necessitats que ens hem proposat dins del projecte, hem optat per instal·lar directament una imatge adaptada del sistema operatiu de la Raspberry Pi Raspbian pel programari OpenHAB anomenada OpenHABian. Aquesta imatge està basada en el Raspbian

Lite, així que la càrrega de processos innecessaris en la placa es redueix enormement. També és un requisit la instal·lació de Java per tal que la aplicació OpenHAB pugui funcionar.

OpenHAB farà de client i estarà subscript a les variables de les dades sensorials dins del broker MQTT, mentre que el node coordinador farà també de client però aquest serà el que publicarà les dades. El broker filtrarà les dades dels sensors de manera que no es dupliquin. En el cas del node actuador, aquest estarà subscript a les variables sobre les que actuarem com són les llums i electrodomèstics, les quals publicarà OpenHAB al broker.

Tota la part explicada anteriorment pot funcionar sense cap connexió a internet, ja que el protocol MQTT només necessita una xarxa IP per tal d'utilitzar-la en les comunicacions TCP/IP entre el broker i els dos nodes. Seria un sistema domòtic funcionant en la intranet. Però en el nostre cas ens interessa la connexió a internet perquè el controlador farà de passarel·la, aquesta a part de fer la conversió de protocols per permetre l'accés d'usuaris remots, guardarà algunes dades en una base de dades i crearà unes tasques automatitzades per tal de domotitzar la vivenda. Aquestes tasques i altres processos faran ús de la internet per accedir a serveis de tercers com podrien ser serveis meteorològics, bots per aplicacions com Telegram, geolocalització de coordenades i el protocol NTP, que és un protocol de sincronització de rellotges, per determinar la data i hora en temps real.

## 2.5. Sensors

Els sensors són aquells elements que transformen les magnituds físiques o químiques de l'entorn en elèctriques per tal de poder llegir-les i d'aquesta forma posteriorment actuar d'acord al seu estat. Es disposarà de diversos tipus de sensors, hi hauran sensors de presència, lluminositat, monòxid de carboni, temperatura i humitat. La comunicació de tots els sensors serà a través del protocol ZigBee que utilitzen els mòduls XBee. Aquests mòduls compten amb deu pins d'entrades i sortides, quatre d'aquests es poden utilitzar com a entrades analògiques. Aquests entrades i sortides només suporten tensions entre 2,8 i 3,4 V, així que hauré de vigilar amb els valors de tensions que donen els sensors a les seves sortides.

Les entrades analògiques en canvi només poden llegir fins a 1,2 V, això vol dir que hauré d'escalar les sortides d'aquells sensors que la seva lectura es veurà afectada amb un divisor de tensió, però sabent que la impedància d'entrada del mòdul XBee és de l'ordre de 10 M $\Omega$  ens permet prescindir de l'ús d'un seguidor de voltatge o altrament dit buffer, degut a que la precisió dels sensors no es veurà molt afectada.

La majoria dels sensors s'alimentaran entre 3 i 3,3 V, els sensors de presència PIR a 4,5 V amb unes tres piles AA i el sensor de monòxid de carboni a 5 V directament des de l'adaptador de corrent alterna.

Els sensors que necessitin algun tipus de calibratge s'hauran de consultar les seves fitxes tècniques pertinents.

### 2.5.1. Sensor de monòxid de carboni

Durant un incendi o una avaria per exemple, és gairebé igual de perillós l'enverinament per inhalació de monòxid de carboni com el propi foc. Això és degut a que el CO és inodor, incolor i molt tòxic.

El sensor de monòxid de carboni MQ-7 s'utilitza tant a nivell domèstic com industrial, utilitza el gas SnO<sub>2</sub> per detectar la presència de monòxid de carboni en l'aire, aquest gas té una conductivitat més reduïda en absència del monòxid de carboni i augmenta aquesta a mesura que augmenta la concentració de CO. Concretament aquest sensor proporciona dues sortides, una digital i una analògica amb sortida de voltatge lineal, s'utilitzarà només la analògica ja que s'instal·larà una alarma a la vivenda que monitoritzarà el nivell de CO i actuarà d'acord a les instruccions del programari que hi haurà al controlador domòtic. S'haurà d'escalar la sortida perquè s'adapti a la màxima lectura possible del mòdul XBee de 1,2 V. En la següent figura podem observar el sensor triat.



Figura 11. Sensor de monòxid de carboni MQ-7

Aquest sensor necessita fer un procés de preparació abans de començar a obtenir dades correctes de CO, el procés en qüestió consisteix en encendre l'escalfador que porta internament durant un mínim de 48 hores per tal d'escalfar la resistència interna que

reaccionarà posteriorment al gas CO. El rang de detecció de concentracions del sensor és d'entre 20 i 2000 ppm de CO, encara que en l'àmbit domèstic el rang de funcionament serà molt més reduït òbviament. Aquest procés d'escalfament farà que aquest sensor sigui el que més consumeixi de tota la instal·lació domòtica amb un valor màxim de corrent de 151 mA. Com que el mòdul XBee de la cuina estarà alimentat amb l'adaptador de corrent de 5 V i aquest procés d'escalfament es farà només en la posada en marxa de la instal·lació, podem concloure que no afectarà negativament el rendiment energètic del sistema domòtic.

### 2.5.2. Sensor de presència

Els sensors de presència més comuns són els PIR, sensor infraroig passiu, aquests detecten canvis en la radiació de la llum infraroig dins del seu camp de visió. El sensor en particular que s'ha utilitzat ens dona un pols alt quan detecta moviment i un pols baix en el cas contrari, manté el puls alt quan segueix detectant presència mitjançant uns temporitzadors interns. La seva sortida lògica arriba fins els 3,3 V en cas de detectar moviment i consumeix un corrent màxim en estat inactiu de 50  $\mu$ A, per el que és compatible amb la màxima tensió permesa de les entrades digitals del mòdul XBee i una alimentació basada en bateries.

Els sensors de presència haurien d'estar alimentats a 5 V per assegurar que el corrent sigui suficientment gran com per activar el regulador de voltatge de 3,3 V intern i així activar la seva sortida digital, però amb 4,5 V es pot aconseguir igualment. En la següent figura es mostra el sensor triat.



Figura 12. Sensor de presència PIR 287-18001

Per poder despertar el mòdul XBee mitjançant el sensor PIR hem d'utilitzar un inversor lògic per convertir el pols alt en un pols baix, ja que així ho especifica a la seva fitxa tècnica.

Aquest sensor també ens permet ajustar els valors de la distància màxima a la que pot detectar moviment i el angle de detecció amb un potenciòmetre, i el temps d'activació de la sortida lògica després d'haver detectat moviment amb un altre potenciòmetre. Aquest en particular presenta un rang de detecció de 5 metres i un camp de visió d'un màxim de 120°.

### 2.5.3. Sensor de humitat

Els sensors de humitat relativa detecten la concentració de vapor d'aigua en l'ambient i l'expressen en un valor dins el rang entre 0 i 100 %, encara que en casos reals domèstics només ens interessarà el rang entre 20 i 80 %.

El sensor que s'ha triat per aquesta tasca és el HIH-5030 el qual compta amb una precisió de lectura del  $\pm 3\%$  RH i permet alimentar-se a partir de 2,7 V, el qual és ideal en sistemes a bateria en els quals el subministrament és nominal de 3 V.

La sortida de voltatge gairebé lineal d'aquest sensor permet la seva incorporació directa a un controlador o un altre dispositiu i gràcies al seu consum típic de només 200  $\mu$ A, és molt adequat per a sistemes a bateria de molt baix consum. En la següent figura es mostra el sensor en particular.



Figura 13. Sensor de humitat HIH-5030

Haurem d'escalar la seva sortida fins a un valor màxim de 1,2 V perquè el mòdul XBee ho pugui llegir correctament i així obtenir valors raonables de humitat relativa.



#### 2.5.4. Sensor de temperatura

El TMP36 és un sensor de temperatura de baix voltatge i baix consum, proporciona una sortida analògica i suficient precisió per a la majoria de sistemes d'automatització domèstics. Té un rang de lectura entre - 40 i 125 °C, un consum màxim de 50  $\mu$ A i un voltatge d'alimentació mínim de 2,7 V que el converteixen en un aliat dels sistemes alimentats a bateria. Proporciona una sortida de voltatge proporcionalment lineal a la temperatura centígrada que mesura amb un precisió de  $\pm 1$  °C, en la següent figura podem observar el sensor en qüestió.

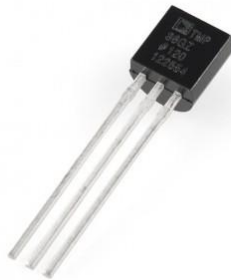


Figura 14. Sensor de temperatura TMP36

En aquest cas no necessitarem escalar la sortida del sensor fins als 1,2 V perquè el rang de temperatures que haurem de llegir en la vivenda no arribarà a aquest valor. Segons la fitxa tècnica, una temperatura de 50 °C equival a una sortida de 1 V per exemple, la qual no és una temperatura habitual en una vivenda.

#### 2.5.5. Sensor de lluminositat

Un fotoresistor o LDR és un component electrònic que varia la seva resistència amb l'augment de la intensitat de la llum incident. Una LDR va ser seleccionada a causa de la impossibilitat d'utilitzar sensors digitals com el TSL2561, el qual pot mesurar en un rang de 0,1 - 40000 lux i funciona mitjançant un protocol I2C. No és possible la seva implementació perquè la versió dels mòduls XBee utilitzats en aquest projecte no és la programable, sinó la estàndard, que és la més barata i àmpliament utilitzada. En la figura següent podem veure el sensor triat.



Figura 15. LDR GL5549

Una LDR no és tan precisa com aquests sensors digitals, no pot mesurar els lux i la seva sortida depèn de la temperatura a la qual funciona. No obstant això, l'exactitud de la LDR és suficient per a distingir entre escenaris foscos i clars, ja que l'objectiu d'aquest sensor és obtenir una estimació aproximada del nivell de claredat o foscor per poder actuar sobre les llums de la vivenda i automatitzar alguns processos.

## 2.6. Actuadors

Els actuadors són dispositius que generen una acció després de rebre una ordre d'un controlador, d'entre els principals tipus d'actuadors existents en el mercat i empleats en domòtica, la acció que es genera sol ser de tipus elèctrica. En el nostre cas, necessitarem un node actuator capaç de controlar les llums de la vivenda, una alarma de 230 V i l'estat d'electrodomèstics com la rentadora i la caldera d'aigua, per aquesta tasca de control tot o res s'ha triat com a actuator el relé. Aquest node actuator es trobarà a prop del quadre elèctric per tal de connectar les 8 línies a controlar amb el propi node. També es podria millorar i ampliar la automatització de la vivenda afegint com a actuadors a motors pel control de persianes per exemple, però degut a que la majoria de dispositius i càrregues de la vivenda funcionen amb un control tot o res, la implementació de relés ja cobreix una molt ampla part de la domotització de la vivenda.

El protocol de comunicació entre el node actuator i el controlador domòtic serà el MQTT, estarà connectat a la xarxa elèctrica mitjançant un adaptador de corrent altera de 5 V i estarà protegit amb aïllament galvànic de forma directa per mitjà d'optoacobladors, aquesta protecció requereix de l'ús d'una segona font per alimentar els relés. L'optoacoblador aconseguirà aquest aïllament elèctric entre el circuit de control i el circuit de potencia ja que internament el senyal passa del circuit del díode emissor de llum al circuit del fotodetector a través de fotons només.

A part dels relés, també disposarem de LED indicadors en els diversos nodes i mòduls.

### 2.6.1. LED

Un díode LED permet proporcionar informació visual a l'usuari sobre l'estat d'un sistema, il·luminar elements en substitució de bombetes incandescents tradicionals, i fins i tot poden actuar en bandes fora de l'espectre visible, com a díodes infrarojos. Es basa en un díode que, en rebre voltatge en els seus borns, permet als electrons recombinar-se amb els buits o càrregues positives, alliberant energia en forma de fotons. Aquesta és la forma més senzilla d'actuació.

En la següent figura s'observa el LED utilitzat, tots els LED serà de color vermell ja que són els que consumeixen menys energia.



Figura 16. LED vermell

Aquests LED serviran per tal d'entendre si estan correctament alimentats els nodes i els mòduls XBee, per saber en quin estat es troben els mòduls o si els relés s'estan activant correctament.

### 2.6.2. Relé

El relé és un interruptor accionat per un electroimant, en aplicar una tensió en els borns d'una bobina, aquesta genera un camp magnètic que atrau una armadura fèrria cap al nucli, això fa que el terminal d'entrada deixi d'estar connectat al terminal normalment tancat i passi a establir contacte amb el terminal normalment obert. En cessar el voltatge, el relé torna a l'estat de repòs. El relé permet controlar càrregues de potència més elevades que altres tipus d'interruptors, per la qual cosa se sol emprar per al control de llums o electrodomèstics. En la següent figura es pot observar el model de relé triat, el qual funciona a 5 V i pot suportar corrents de fins a 10 A amb una tensió alterna 250 V.



Figura 17. Relé Songle de 5V

Els relés s'utilitzaran per a controlar l'estat de llums de les cinc estàncies a domotitzar, pel control de la rentadora i la caldera, i per controlar l'alarma de 230 V. En total tindrem un total de 8 relés, això vol dir que haurem d'utilitzar 8 sortides digitals del ESP8266 i que aquestes es puguin utilitzades com a sortides en tot moment. S'actuarà sobre aquests relés des de la interfície gràfica d'usuari del sistema domòtic remotament de dues formes, o bé manualment, o després de que s'hagi activat alguna automatització de la vivenda com pot ser la d'encendre els electrodomèstics en les hores vall per reduir el consum energètic. S'assegurarà que els electrodomèstics hagin acabat la seva posada en marxa abans d'activar-los remotament, específicament en el cas de la rentadora i de la caldera d'aigua, i que en funcionament no superin les intensitats nominals màximes suportades pel relé.

Els relés s'activaran amb un pols baix i estaran configurats amb una posició normalment oberta, d'aquesta forma per encendre qualsevol càrrega s'haurà de tancar el circuit prèviament activant el relé. Es podrien afegir també interruptors físics pel control de les càrregues, així l'usuari ho podria controlar des de un lloc físic també. Els interruptors enviarien una comanda a OpenHAB per tal d'activar les càrregues i es podran comunicar entre ells mitjançant múltiples formes com poden ser KNX, MQTT, ZigBee, Z-Wave o Bluetooth.

### 2.6.3. Alarma sonora

Un sistema d'alarma és un element de seguretat passiva, això significa que no eviten una situació de perill però que sí són capaços d'advertir per poder actuar contra aquesta, complint així, una funció dissuasiva enfront de possibles problemes greus com la intrusió de persones, la presència d'agents tòxics o qualsevol situació que sigui anormal per l'usuari.

Es disposarà d'una alarma sonora situada a la planta baixa de la vivenda, aquesta ens advertirà sobre els nivells de monòxid de carboni existents en la cuina i funcionarà d'una manera específica que es detallarà més endavant en l'apartat de programació. L'alarma seleccionada per aquesta tasca és la que es mostra en la següent figura.



Figura 18. Alarma sonora RS 144-5946

Té un consum màxim de 100 mA funcionant a 230 V d'alimentació alterna, una protecció IP54 i proporciona un nivell de pressió sonora mínima de 100 dB(A).

## 2.7. Altres components del sistema

### 2.7.1. Router WiFi

Un dels aspectes claus del sistema domòtic proposat en aquest document és la comunicació mitjançant el protocol MQTT. Aquesta comunicació només és possible si els dispositius estan connectats a un router WiFi que els permeti l'accés a la xarxa assignant-los adreces IP. Per defecte, un router WiFi assigna direccions IP dinàmiques, això suposa un problema pel correcte funcionament del sistema domòtic ja que si un dispositiu perd momentàniament la connexió amb el router, aquest li assignarà una IP diferent un cop es torni a connectar.

En principi s'utilitzarà el router de la companyia de telefonia existent, si aquest per alguna raó no dona abast en rendiment o rang de senyal, s'utilitzarà un de més potent. Aquesta configuració es realitzarà en la configuració avançada de la LAN, on s'afegirà l'adreça IP i la màscara de subxarxa per a les noves adreces d'IP. En resum, s'etiquetaran els dispositius en el servidor DHCP del router i s'assignarà una adreça IP del rang del tipus de dispositiu i se l'associarà amb la MAC. D'aquesta manera, el dispositiu sempre es connectarà a l'adreça IP estàtica. La primera IP en declarar serà la de la Raspberry com es veu en la següent figura, es triarà òbviament una IP lliure, consecutivament s'assignaran les IP del node actuador i el node coordinador.

### Add Static DHCP - Home Network

Nombre del dispositiu	DESKTOP-C1JB83N
Direcció MAC	70 : 85 : c2 : 06 : a8 : 1c
IP	192 . 168 . 0 . 10

Figura 19. Configuració del servidor DHCP per assignar IP estàtiques

## 2.7.2. Programes

A part dels components esmentats en els apartats anteriors, és important destacar també les eines i programes necessaris per la programació i comunicació del sistema domòtic.

Per tal de programar els mòduls XBee, la última versió del programa XCTU, aquest permet realitzar la programació dels mòduls per tal de definir quin serà el coordinador, qui farà de routers i quins de end devices. A més de configurar el període de hibernació dels end devices, actualitzar el programari intern dels mòduls i determinar de quina manera es comunicaran entre ells. Es necessita un adaptador USB per connectar el mòdul a l'ordinador.

Per programar els microcontroladors ESP8266, la última versió del programa Arduino IDE, aquest ens permetrà programar els xips d'una forma senzilla com si es tractes de plaques Arduino, amb un llenguatge de programació C++.

Per tal de configurar correctament la Raspberry, s'ha d'instal·lar dins d'aquesta la última versió del sistema operatiu openHABian, també la última versió de Java i del servidor Mosquitto. Aquest últim és el broker MQTT pel qual passaran totes les dades sensorials i comandes.

Per la creació de la interfície gràfica d'usuari del sistema domòtic, la última versió del programa OpenHAB, aquest ens permetrà amb l'ajuda de la última versió del Visual Code Studio, crear i programar eficientment la interfície gràfica del que seria la aplicació de control i les automatitzacions de tota la vivenda amb un llenguatge basat en Java i Xbase.

### 3. PROGRAMACIÓ

En aquest capítol es detallarà el plantejament, els passos seguits i la configuració realitzada per aconseguir la automatització de la vivenda. Conforme es va anar plantejant el sistema domòtic que s'exposa en aquest capítol, van anar sorgint una sèrie de restriccions les quals van fer que el desenvolupament del sistema hagués d'estructurar-se entorn d'elles, a més, el codi dels dispositius s'ha elaborat de manera que pugui ser modificat fàcilment en un futur per a poder afegir més sensors, actuadors o qualsevol altra tasca automatitzada.

El projecte es basa en una xarxa ZigBee mallada que seguirà un sistema distribuït. En un sistema distribuït, els diferents components es comuniquen i es coordinen per aconseguir un objectiu comú. Això és el que es proposa en la computació en la boira, les passarel·les locals, altrament dit els controladors, es poden comunicar entre ells i amb el núvol per aconseguir un objectiu comú. El terme distribuït implica que els recursos de càlcul i d'emmagatzematge es distribueixen per tota la xarxa en llocs estratègics per permetre que el sistema respongui més ràpid i també per disminuir la càrrega computacional del núvol. Això es realitza movent part de la potència computacional cap a la vora mitjançant passarel·les de boira, que realitzen certes tasques que requereixen una resposta ràpida.

La càrrega computacional es reparteix entre el controlador i els nodes. El controlador serà l'encarregat de fer de passarel·la amb un servidor MQTT, aquesta enllaçarà el protocol dels dispositius IoT al núvol, i a més, guardarà algunes dades en una base de dades local per tal de poder utilitzar-les en la creació de les tasques automatitzades de la vivenda. A part de fer anar el broker MQTT, el controlador també farà de interfície gràfica d'usuari.

El node coordinador serà l'encarregat de fer el primer tractament a les dades dels sensors i el node actuator de controlar les càrregues de la vivenda. L'adquisició de les dades dels sensors la faran els mòduls XBee.

En la instal·lació domòtica s'hauran de programar i configurar principalment els tres elements descrits anteriorment que són el controlador, el dos nodes i els mòduls XBee. En la següent figura es pot apreciar l'esquema general de tota la instal·lació domòtica.

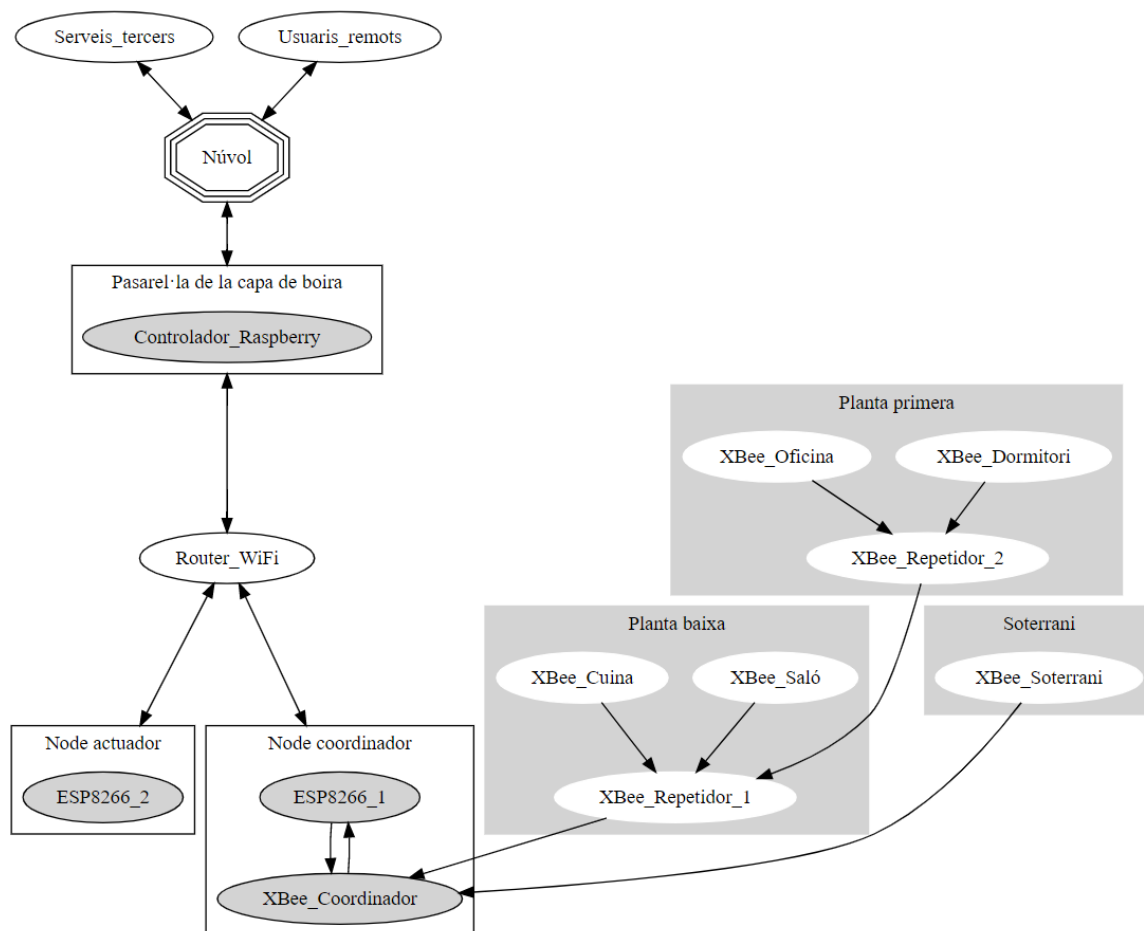


Figura 20. Esquema general de les comunicacions de la instal·lació domòtica

### 3.1. Mòduls XBee

La ràdio compta amb un microprogramari que permet especificar al dispositiu quin rol complirà en la xarxa, així com una sèrie de paràmetres de configuració que recullen totes les funcions i opcions. Aquestes opcions inclouen la definició de paràmetres com la direcció de 16 bits, l'adreça del dispositiu, l'estat dels pins, mostrejos de senyals i moltes altres característiques del circuit integrat. El canvi en aquests paràmetres pot realitzar-se mitjançant el programari XCTU o mitjançant missatges rebuts pel mòdul tant per connexió serial directa com a través de la pròpia xarxa.

Per a poder comunicar-se i connectar-se amb la resta de la xarxa, els XBee necessiten algun tipus d'interfície de comunicació. Així, per exemple, encara que el mòdul rebi alimentació i funcioni, necessitarà configurar el seu programari per a poder adoptar el rol que es desitgi en la xarxa de nodes. Per això, és necessari que es connecti a un PC equipat amb el programari XCTU. El mòdul XBee compta amb un UART, punt d'entrada i sortida de dades del mòdul des



d'un dispositiu amfitrió. El mòdul UART també fa tasques com la comprovació de la paritat i el sincronisme, necessàries per a la comunicació. En la taula 2 podem veure la configuració dels mòduls XBee.

Configuracions dels mòduls XBee								
Funció	Coordinator	End device 1	End device 2	End device 3	End device 4	End device 5	Router 1	Router 2
Mode	API 1	AT	AT	AT	AT	AT	AT	AT
PAN ID	5555	5555	5555	5555	5555	5555	5555	5555
JV	NO	SI	SI	SI	SI	SI	SI	SI
DH	-	0x0	0x0	0x0	0x0	0x0	0x0	0x0
DL	-	0x0	0x0	0x0	0x0	0x0	0x0	0x0
NI	Coordinator	Soterrani	Cuina	Salo	Dormitori	Oficina	Plant. baixa	Plant. primera
SM	-	5	-	5	5	4	0	0
ST (x1ms)	-	0x7D0	-	0x7D0	0x7D0	0x7D0	-	-
SP (x10ms)	0x7D0	0x7D0	-	0x7D0	0x7D0	0x7D0	0x7D0	0x7D0
SN	0x0F	0x0F	-	0x0F	0x0F	0x0F	0x0F	0x0F
SO	-	0x06	-	0x06	0x06	0x06	-	-
D0	-	ADC0	ADC0	ADC0	ADC0	ADC0	-	-
D1	-	ADC1	ADC1	ADC1	ADC1	ADC1	-	-
D2	-	ADC2	ADC2	ADC2	ADC2	ADC2	-	-
D4	-	DI4	-	DI4	DI4	-	-	-
IR (x1ms)	-	0x800	0x800	0x800	0x800	0x800	-	-

Taula 2. Configuració amb XCTU dels mòduls XBee

Aquesta seria la configuració dels mòduls XBee, veiem com tenim un coordinador, dos routers i cinc end devices. Totes les dades numèriques es troben en hexadecimal ja que així és com està determinat pel programari XCTU. S'ha actualitzat el seu firmware a l'última versió estable i s'ha procedit a la seva configuració emplenant només els paràmetres necessaris.

Respecte al mode, existeixen dos tipus diferents, el mode transparent AT i el mode API. En el mode transparent el mòdul XBee funciona de manera que tota la informació és transmesa i rebuda per la antena sense cap addició d'altres dades secundàries com les adreces de destí. El mode transparent té moltes limitacions, però és idoni en casos on ens interessa crear una xarxa mallada amb un únic punt de control que seria el coordinador com és el nostre cas, els mòduls entre ells es comunicaran enviant les dades dels sensors i el coordinador analitzarà de quin mòdul s'ha enviat cada dada per poder organitzar-les més endavant. Per aquest motiu tots els mòduls excepte el coordinador es trobaran en mode AT.

Perquè el coordinador pugui analitzar les dades sensorials s'haurà de trobar en mode API, aquest mode ofereix una interfície estructurada on les dades són comunicades a través de paquets organitzats i en un determinat ordre. Això permet establir una comunicació complexa entre mòduls sense haver de definir un protocol propi. En aquest mode, les dades enviades estan estructurats en una trama. Aquesta trama conté el missatge i informació addicional com la qualitat del senyal, les adreces d'origen i destí entre altres informacions.

Tots comparteixen la mateixa PAN ID com hauria de ser, ja que aquesta és la identificació de la xarxa personal on es trobaran tots els mòduls. Només necessitem que el coordinador estigui en mode API perquè pugui llegir el format de les trames correctament per poder tractar-les posteriorment al node coordinador i enviar-les al broker MQTT. Tots els altres mòduls es troben en mode AT, ja que només hauran d'enviar les dades sensorials cap al coordinador directament o mitjançant els routers localitzats a la planta baixa o la planta primera.

Els paràmetres DH i DL fan referència a la adreça de destí a la que volem enviar les dades, aquesta sempre serà una adreça de 64 bits on 32 bits es determinen en DH i els altres 32 bits en el DL. Com que en el nostre cas ho enviem tot cap al coordinador, indiquem 0x0 en els mòduls on desitgem que l'adreça de destí sigui la del coordinador. La opció JV, es refereix a que la verificació de canals està activada, això vol que tots els mòduls comprovaran a quin canal es troba el coordinador i s'hi afegiran sempre que canviï, el coordinador en la primera connexió analitzarà quins dels canals Zigbee de 2,4 GHz és el millor per connectar-se, basant-se en dades com el tràfic i el soroll creat per les xarxes WiFi o altres aparells emissors en la mateixa freqüència. En l'apartat de comunicació s'explicarà en profunditat els possibles problemes que podria ocasionar una mala selecció del canal Zigbee.

El paràmetre NI és la etiqueta que utilitzem per referenciar el mòdul XBee amb una identificació entenedora. Tota la descripció anterior serviria per configurar el XBee perquè sàpiga on s'ha de connectar i amb qui, ara parlarem del funcionament del propi mòdul quan estigui mostrejant les dades. El paràmetre SO indica que està habilitat el mode sleep de llarga durada.

Depèn d'on es trobi el mòdul en la vivenda tindrà uns certs sensors instal·lats, en el cas del soterrani, saló i dormitori tindrem els mateixos sensors. En aquests tres casos utilitzarem el mode cinc del sleep mode, en aquest mode el mòdul es desperta després de cinc minuts (SP\*SN) apagat, envia una mostra de cada sensor a un període de mostreig de 2,048 segons (IR) i es queda despert durant 2 segons (ST) abans de tornar a dormir. El mòdul despertarà

periòdicament com s'ha comentat abans però també ho farà si detecta un canvi en la entrada digital 9 anomenada Sleep RQ, aquest canvi ho gestionarà el sensor de presència PIR a través d'un inversor lògic, ja que en casos com aquests necessitem que s'activin les automatitzacions pertinents perquè s'ha detectat moviment. Els mòduls amb el mode quatre del sleep mode es comportaran igual que el descrit anteriorment però sense la opció de gestionar l'estat del mòdul XBee amb la entrada digital Sleep RQ, això és degut a que els mòduls aquests no tindran un sensor de presència PIR. Els mòduls amb un sleep mode de 0 sempre es trobaran encesos com és el cas del mòdul coordinador també.

Finalment només queda la declaració dels pins que volem activar com entrades analògiques o digitals.

### 3.2. Nodes

Els nodes contindran principalment un microcontrolador NodeMCU cadascun equipat amb el xip ESP8226 i es programaran amb el IDE de Arduino en un llenguatge basat en C++. En tots dos nodes s'haurà de definir l'adreça IP del broker MQTT i els paràmetres d'usuari i contrasenya per accedir a la xarxa WiFi local. També s'haurà de definir els microcontroladors com a clients i els tòpics MQTT als quals ens volem subscriure o publicar. Hi haurà tants tòpics com variables, aquestes poden fer referència a valors de sensors o estat d'actuadors.

La comunicació es basa en uns tòpics que el client que publica el missatge crea i els clients que desitgin rebre'l s'han de subscriure's a ell. La comunicació pot ser d'un a un, o d'un a molts. Un tòpic es representa mitjançant una cadena i té una estructura jeràrquica, cada jerarquia se separa amb " / ", també podem triar si ens volem subscriure a totes les branques d'una jerarquia, múltiples nivells, o només a unes branques determinades, un únic nivell. La jerarquia aplicada en el projecte consta de tres nivells on en primer lloc ens referim a la casa, després a la planta, seguidament a la estància i finalment al tipus de sensor o actuator.

En la comunicació MQTT entre les plaques NodeMCU i la Raspberry Pi, s'ha considerat que la forma més còmoda quant a reutilització de codi es refereix és establint els tòpics dels missatges d'una forma jeràrquica. Si la placa NodeMCU actua com a subscriptora d'un tòpic determinat, aquesta rebrà missatges de tipus publicació amb els tòpics oportuns. Però si actua com a publicadora, enviarà missatges tipus publicació. S'ha estimat oportú que els missatges enviats per part de la placa NodeMCU siguin de manera periòdica, sent el període d'enviament

de missatges MQTT configurable, aquest període dependrà del temps de mostreig i hibernació dels mòduls XBee.

### 3.2.1. Node coordinador

Aquest node estarà format per una placa NodeMCU i un mòdul XBee que farà de coordinador, aquest rebrà les dades dels sensors i les tractarà per poder ser enviades cap al controlador mitjançant MQTT. S'haurà de comprovar quins dels cinc end devices és el que ha enviat les dades mitjançant la seva adreça de 16 bits, aquest procés de selecció és necessari degut a que necessitem saber quin ho ha enviat per poder assignar-li el tòpic MQTT corresponent i també perquè no tots els end devices tindran instal·lats els mateixos sensors.

En la taula següent podem observar amb quins tòpics treballarà aquest node, els quals són bàsicament dades dels sensors.

Estància	Sensor	Tòpic MQTT
Soterrani	Temperatura	casa / st / soterrani / temperatura
	Humitat	casa / st / soterrani / humitat
	Lluminositat	casa / st / soterrani / lux
	Presencia	casa / st / soterrani / pir
Cuina	Temperatura	casa / pb / cuina / temperatura
	Humitat	casa / pb / cuina / humitat
	CO	casa / pb / cuina / co
Saló	Temperatura	casa / pb / salo / temperatura
	Humitat	casa / pb / salo / humitat
	Lluminositat	casa / pb / salo / lux
	Presencia	casa / pb / salo / pir
Dormitori	Temperatura	casa / pp / dormitori / temperatura
	Humitat	casa / pp / dormitori / humitat
	Lluminositat	casa / pp / dormitori / lux
	Presencia	casa / pp / dormitori / pir
Oficina	Temperatura	casa / pp / oficina / temperatura
	Humitat	casa / pp / oficina / humitat
	Lluminositat	casa / pp / oficina / lux

Taula 3. Tòpics MQTT del node coordinador

### 3.2.2. Node actuator

Aquest node estarà format per una placa NodeMCU i el circuit de potència, aquest rebrà les comandes MQTT des del controlador i commutarà les càrregues de la vivenda. En la taula següent podem observar amb quins tèmics treballarà aquest node.

Estància	Actuador	Tòpic MQTT
Soterrani	Llums	casa / st / soterrani / llums
	Rentadora	casa / st / soterrani / rentadora
	Caldera	casa / st / soterrani / caldera
Cuina	Llums	casa / pb / cuina / llums
Saló	Llums	casa / pb / salo / llums
	Alarma	casa / pb / salo / alarma
Dormitori	Llums	casa / pp / dormitori / llums
Oficina	Llums	casa / pp / oficina / llums

Taula 4. Tèmics MQTT del node actuator

### 3.3. Controlador

El controlador, que serà un ordinador de placa simple, farà de passarel·la entre el núvol i els nodes físics, i farà de interfície gràfica d'usuari per poder mostrar les dades dels sensors i enviar les comandes als actuadors des de OpenHAB. Aquesta comunicació entre els nodes i el controlador s'implementarà en MQTT, més endavant veurem de que tracta aquest protocol. Totes les dades que es mouen cap al núvol, o viceversa, passen per la passarel·la, una passarel·la IoT també pot denominar-se passarel·la intel·ligent o nivell de control ja que pot fer tasques més complexes que una passarel·la comú.

En les vivendes domòtiques alguns sensors generen desenes de milers de punts de dades per segon. Una passarel·la proporciona un lloc per a pre-processar aquestes dades localment en la fog layer, altrament capa de boira, abans d'enviar-los al núvol. Quan les dades s'agreguen, es resumeixen i s'analitzen tàcticament en la capa de boira, es minimitza el volum de dades que han de ser enviats al núvol, la qual cosa pot tenir un gran impacte en els temps de resposta i en els costos de transmissió de la xarxa.

Encara que el volum de dades no fos el problema, és aconsellable distribuir correctament la càrrega computacional del sistema domòtic, per aquesta raó el controlador a més de fer de passarel·la, farà de base de dades emmagatzemant algunes dades dels sensors, i també farà

anar les automatitzacions de la vivenda fent una monitorització i anàlisi constants tant de les dades locals com de les externes. Per tant, OpenHAB, juntament amb Java 11 i el broker MQTT Mosquitto estan desplegats en el controlador que serà una Raspberry Pi 3 model B amb sistema operatiu OpenHABian.

### 3.3.1. Broker MQTT

MQTT és un protocol de missatgeria i està dissenyat amb un transport de missatgeria de publicació/subscripció extremadament lleuger que és ideal per connectar dispositius remots de pocs recursos amb un codi relativament simple i una amplada de banda de xarxa mínima. Actualment, MQTT s'utilitza en una gran varietat d'indústries, com l'automoció, la fabricació, les telecomunicacions, el petroli i el gas. Aquestes característiques han fet que ràpidament es converteixi en un protocol molt utilitzat en la comunicació de sensors i, conseqüentment, dins de les IoT. En relació amb el mecanisme de comunicació amb MQTT dins del projecte, consistirà en publicar periòdicament els valors dels sensors amb un temporitzador i a llegir les comandes pels actuadors.

La configuració del servidor MQTT, altrament dit el broker, s'ha realitzat mitjançant la instal·lació de Mosquitto en el sistema operatiu OpenHABian, de tal forma que convertim a la Raspberry Pi en l'eix central de l'estructura MQTT. En la següent figura podem observar amb quins clients treballa el broker.

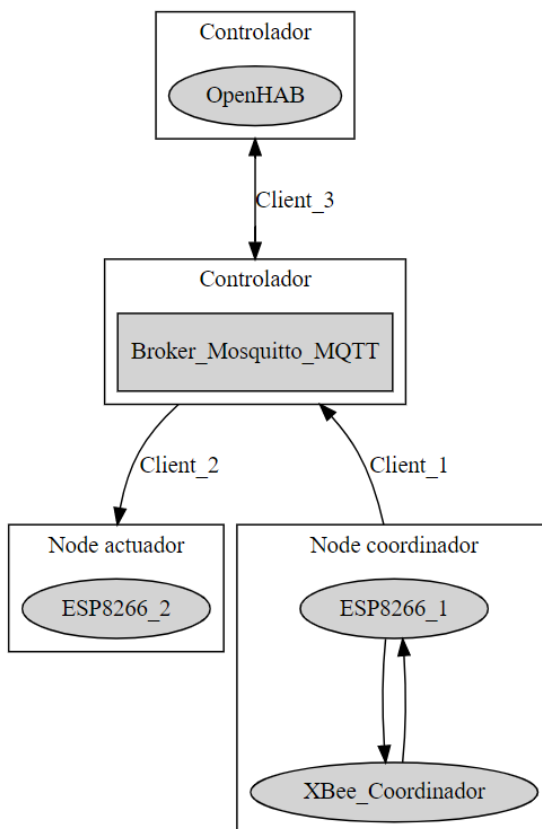


Figura 21. Esquema de les comunicacions MQTT

Després de la instal·lació del broker Mosquitto, s'ha modificat el seu arxiu de configuració on s'han definit alguns paràmetres pel correcte funcionament del sistema domòtic. Com a mínim s'han de declarar el port i la adreça IP del broker. Cadascun dels tres clients descrits anteriorment pot tenir la seva pròpia configuració de paràmetres mentre que es mantinguin intactes els valors del port i la adreça IP del broker MQTT. Per tal d'augmentar la seguretat de les comunicacions s'ha cregut que és necessària la incorporació d'un usuari i contrasenya per tal d'accedir al broker, encara es podria blindar més la seguretat xifrant les connexions amb TLS/SSL. En la següent taula es pot observar els paràmetres que seguiran el broker i els clients connectats a ell, això s'aplica tant al node coordinador i actuador com a OpenHAB.

Paràmetre	Valor
Adreça IP del broker	192.168.0.10
Port	1883
Usuari	admin
Contrasenya	domotica
QoS	0

Taula 5. Configuració del broker MQTT

Amb aquesta configuració tindriem un servidor MQTT atenent connexions pel port 1883. Els clients per tal de poder comunicar-se amb el broker han d'utilitzar els paràmetres de la taula anterior, l'únic paràmetre que es pot canviar de valor és el QoS, aquest fa referència a la qualitat de servei de la comunicació, i cada client pot triar amb quina de les tres opcions es vol comunicar, les quals són 0, 1 o 2. El valor de QoS és inversament proporcional al rendiment de les comunicacions, un QoS de 2 indica que s'hauran de fer més operacions per assegurar que s'ha rebut només una vegada el missatge, mentre que amb un QoS de 0 s'envia el missatge només un cop i no es verifica si s'ha rebut o no. En aquests dos casos no hi hauran duplicats mentre que en el cas del QoS 1 sí que hi poden haver perquè segueix enviant fins que es confirma que s'ha rebut.

### 3.3.2. OpenHAB

Com s'ha descrit en els apartats anteriors, per alleugerar la càrrega computacional en els nodes, es trasllada una part d'aquesta cap al controlador. En aquest és on s'estarà executant el programari que tractarà i mostrarà les dades dels sensors i actuadors d'una forma gràfica perquè el usuari pugui tenir una bona accessibilitat amb el sistema domòtic.

En el nostre cas s'ha optat per OpenHAB, aquesta és una plataforma o entorn de programari de codi obert, amb l'objectiu d'implementar un sistema de control domòtic d'internet de les coses. Aquest està totalment enfocat a les aplicacions domòtiques i és de codi obert lo que ens proporciona una gran flexibilitat alhora d'afegir dispositius de diferents tecnologies i combinar-los en la nostra vivenda. Té un gran suport continu de part de la seva comunitat, i utilitza un motor potent per dissenyar regles, amb variables com el temps i esdeveniments amb codi, disparadors, accions, notificacions i control de veu. S'utilitzarà la última versió estable de OpenHAB 3.

Hi ha dues formes de configurar o programar la aplicació OpenHAB, la primera és una forma automàtica que consisteix en la vinculació dels dispositius un per un des de la interfície amigable de la pròpia aplicació. La segona forma està adreçada a usuaris més experimentats i és la manera més òptima, consisteix en la vinculació dels dispositius mitjançant uns arxius de configuració que es troben en la ruta on està instal·lat OpenHAB. Els arxius que configurarem són els arxius localitzats a les carpetes things, items, sitemaps, rules, services i persistence, més endavant s'explicarà en què consisteixen. Per tal d'agilitzar el procés de configuració i programació s'ha utilitzat el programari Visual Code Studio ja que aquest compta amb un plugin dedicat a OpenHAB que ens permet connectar-nos remotament al controlador



inserir la seva adreça IP, usuari i contrasenya. Des de l'accés remot podrem modificar els arxius descrits anteriorment d'una forma òptima i ràpida, en la figura següent podem observar com queda l'accés remot a al controlador.

Aquest plugin també té la habilitat de comprovar si la sintaxi del codi és correcte o no, així ens assegurem de validar línia per línia el codi escrit. La aplicació en sí funciona sobre Java però per la programació dins d'aquesta s'utilitza un llenguatge basat en Xbase.

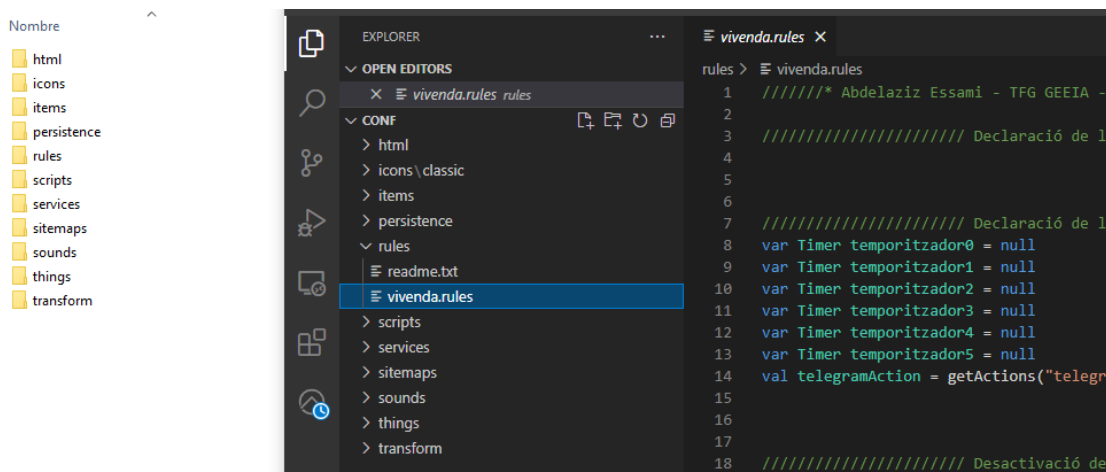


Figura 22. Accés remot a la carpeta conf de OpenHAB

OpenHAB utilitza diferents elements per organitzar i administrar el seu programari, a continuació veurem una breu explicació dels elements que s'han utilitzat.

En primer lloc, les things són qualsevol entitat física que pot ser afegida al sistema domòtic, poden proporcionar més d'una funció com podria ser la mesura de temperatura i al mateix temps operar com un detector de presència. El concepte thing no només es centra en dispositius físics, sinó que també pot representar un servei web o qualsevol font de informació. En el nostre cas només tenim things associades a fonts d'informació com són el broker MQTT, Telegram per la creació d'un bot amb alertes interactuables pel telèfon mòbil, la estació meteorològica OpenWeathermap, el geolocalitzador Astro i el servidor NTP per tal d'obtenir la data i hora locals.

Aquests cinc elements descrits anteriorment només funcionaran un cop s'hagin afegit correctament com a bindings, aquests serien com uns adaptadors de programari, la seva funció seria habilitar que les things estiguin disponibles en el sistema domòtic.

Segonament, els items representen les propietats i capacitats de la instal·lació domòtica de l'usuari, tant en la interfície d'usuari com en el programari. Poden emmagatzemar dades com nombres, strings o estat d'un interruptor per exemple. En el nostre cas tindrem tants items com elements que ens interessi mostrar a la interfície gràfica d'usuari. Cada item estarà enllaçat a una thing mitjançant uns elements anomenats channels, aquest enllaç s'anomena link. Una thing pot tenir tants channels com items enllaçats, mentre que un item només pot tenir un channel.

Finalment, els sitemaps bàsicament són els que preparen els items perquè es puguin visualitzar en la interfície gràfica mitjançant diferents elements visuals com gràfiques o mapes per exemple. Hi haurà dos sitemaps, un per visualitzar l'estat dels sensors i actuadors de la vivenda i un altre per la estació meteorològica. A part dels sitemaps, OpenHAB ofereix unes interfícies d'usuari integrades que són més amigables per mostrar informació a l'usuari.

A continuació en la taula 6 veurem com s'han de configurar els serveis de OpenHAB dins de la carpeta services, aquests contindran quins elements s'utilitzaran dels bindings, serveis de persistence, interfícies d'usuaris integrades de OpenHAB entre altres. Principalment per la interfície d'usuari s'utilitzaran els bindings de MQTT, NTP i OpenWeatherMap. Els dos bindings restants estaran presents en la programació de les automatitzacions.

Addons	Elements
Bindings	MQTT, NTP, Astro, OpenWeatherMap, Telegram
Persistence	InfluxDB
UI	basic, paperui
Altres	openhabcloud

Taula 6. Addons instal·lats en OpenHAB

Veiem que tenim també un servei de persistence, el qual permet emmagatzemar dades, anomenat InfluxDB i un altre servei anomenat openhabcloud, el primer correspon a la base de dades que explicarem més endavant i el segon correspon a la versió de la aplicació mòbil de OpenHAB ja que el controlador només s'hi pot accedir de forma local. Per tal d'accedir a OpenHAB des de qualsevol lloc connectat a internet, és necessària la creació d'un compte en la pàgina oficial de OpenHAB i la autenticació amb un usuari i contrasenya que s'hauran d'haver creat anteriorment. OpenHAB també ofereix unes interfícies d'usuari integrades que són les UI, en el nostre cas només s'han utilitzat dues, una on es mostraran els sitemaps i un altre on hi haurà la interfície gràfica d'usuari.

Com s'ha comentat anteriorment, principalment per la interfície d'usuari s'utilitzaran els bindings de MQTT, NTP i OpenWeatherMap. En les següents dues taules podem veure com estan configurades les things de NTP i OpenWeatherMap dins de la carpeta things, la configuració de MQTT per a OpenHAB és igual a la configuració descrita a la taula 5.

Paràmetre	Valor
Servidor	nl.pool.ntp.org
refreshInterval	60
refreshNtp	30

Taula 7. Configuració del binding NTP

El binding NTP s'utilitza per mostrar les actualitzacions de la data i hora locals des d'un servidor NTP, amb els valors anteriors les noves actualitzacions de temps es publiquen al bus d'esdeveniments cada 60 segons i es consultaria al servidor cada 30 minuts.

Element	Paràmetre	Valor
Compte OpenWeatherMap	apikey	cd1b10f5f4b610f6d80429a624dc4d24
	refreshInterval	10
	language	es
Estació meteorològica	location	41.9741672, 2.7949886
	forecastHours	0
	forecastDays	7

Taula 8. Configuració del binding OpenWeatherMap

El binding OpenWeatherMap s'utilitza per accedir a un servei en línia que proporciona dades meteorològiques globals a través d'una API, incloses dades meteorològiques actuals, pronòstics, prediccions immediates i dades meteorològiques històriques per a qualsevol ubicació geogràfica. S'ha d'haver creat un compte prèviament a la pagina oficial de OpenWeatherMap i per tal d'accedir a aquest hem de proporcionar la clau apikey, s'actualitzaran les dades cada 10 minuts i es mostraran a la interfície gràfica en castellà. L'estació meteorològica necessitarà saber sobre quina localització aportarà les dades i de quina manera, la localització serà la de la vivenda expressada com a latitud i longitud i proporcionarà dades meteorològiques actuals i pronòstics diaris dels 7 pròxims dies.

Per accedir a la interfície gràfica d'usuari de forma local s'ha d'accedir a la adreça IP del controlador des d'un navegador d'internet indicant també el port en el qual treballa OpenHAB, el port per defecte sempre serà el 8080, tal i com podem veure a la taula 9.

Paràmetre	Valor
Adreça URL	http://192.168.1.10:8080
Usuari	admin
Contrasenya	domotica

Taula 9. Dades necessàries per accedir a la interfície gràfica de OpenHAB

### 3.3.3. Automatitzacions

Les automatitzacions són la part més important de la instal·lació domòtica, són les que milloren la eficiència energètica de la vivenda, ens protegeixen contra desastres artificials augmentant la seguretat i proporcionen un gran nivell de confort. Amb la incorporació dels bindings Telegram i Astro, i l'alt nivell de programació de OpenHAB amb Xbase, aconseguim crear unes automatitzacions molt eficients. Totes les automatitzacions s'hauran de programar dins de l'arxiu rules, on s'hauran de programar diferents regles que conjuntament crearan les automatitzacions que necessitem. Totes les regles han de seguir una estructura determinada com la que es presenta en la següent figura, aquesta estructura va marcada amb els elements de color rosa i consisteix en una expressió condicional.

```
rule "1-Reset Alarma CO" // aquesta regla s'ha creat per casos on l'usuari
when
    // no pugui apagar l'alarma desde el mòbil
    System started // Quan arrenca el sistema
then
    PB_Salo_Alarma.sendCommand(OFF) // Apaguem Alarma
end
```

Figura 23. Regla pel reinicialitzar la alarma de CO

Abans que una regla comenci a funcionar, cal activar-la. OpenHAB ofereix diferents formes d'activació com pot ser a partir del valor o estat del items de sensors o actuadors, variables de temps, estats del propi sistema o a partir dels channels o canvis d'estat de les things com del geolocalitzador Astro o el bot de Telegram per exemple. Totes les automatitzacions enviaran avisos, missatges o canvis d'estat mitjançant el bot de Telegram, aquest bot es trobarà instal·lat dins l'aplicació de Telegram de l'usuari en el seu mòbil. En les següents dues taules podem veure com estan configurades les things de Astro i Telegram dins de la carpeta things.

Channel	Paràmetre	Valor
astro:sun:local	geolocation	41.9741672, 2.7949886
	interval	60
astro:moon:local	geolocation	41.9741672, 2.7949886
	interval	60

Taula 10. Configuració del binding Astro

El binding Astro s'utilitza per calcular diverses dades posicionals i temporals del Sol i la Luna. També pot calcular nivells de radiació solar durant el dia, concretament de la radiació directa, difusa i total. En el nostre cas només utilitzarem dades relacionades amb el Sol com la posta o la sortida del Sol, així que haurem d'indicar com a mínim la latitud i longitud de la localització de la vivenda.

El paràmetre interval es refereix al període expressat en segons amb el qual es calculen dades posicionals com l'elevació i l'azimut, en el nostre cas no s'han utilitzat aquestes dades dins del projecte però pot resultar interessant utilitzar-les en un futur pel control i posicionament d'unes plaques solars.

Paràmetre	Valor
Nom del bot	domo789bot
chatlds	208664423
botToken	1804132199:AAHZtpLgiQXQQFdZ1F9w7BehRnZyLjo9m0w

Taula 11. Configuració del binding Telegram

El binding Telegram permet enviar i rebre missatges entre la plataforma OpenHAB i la pròpia aplicació de Telegram, és una comunicació que permet a l'usuari assabentar-se de què està succeint dins de la vivenda a través d'un telèfon mòbil, i que aquest pugui interactuar davant d'alguns successos, com l'activació de l'alarma de CO per exemple.

Prèviament s'ha d'haver creat el bot dins de Telegram. Un cop creat haurem de buscar els paràmetres de chatlds i botToken que ens ha atorgat la aplicació i afegir-los dins de l'arxiu things en els apartats de la taula 11. Amb el paràmetre chatlds identificarem el bot en qüestió i amb el botToken xifrarem les comunicacions. Aquest bot tindrà la tasca de informant-nos de canvis d'estat d'actuadors, avisos o alertes que puguin aparèixer dins de la vivenda.

Per la creació de les automatitzacions, la extensió de codi OpenHAB del Visual Code Studio ofereix suport per a la construcció de regles. Inclou comprovacions de sintaxi i coloració,

validació amb marcadors d'error i assistència de contingut com plantilles. Això facilita molt la creació de regles.

La primera automatització consistirà en la creació d'una alarma de monòxid de carboni, s'haurà de monitoritzar el nivell de monòxid de carboni mitjançant el mòdul XBee localitzat a la cuina i alertar a l'usuari depenent en quin rang es trobi el nivell de CO. S'han programat quatre regles per poder crear aquesta automatització on cadascuna d'elles fa una tasca diferent. Una regla consistirà en trobar en quin rang es troba el nivell de CO, hi hauran tres rangs i en cadascun d'aquests activarem la alarma després d'un cert temps. El temps d'activació de l'alarma dependrà del rang en que es trobi el nivell de CO, quant més contaminat estigui l'aire més ràpid saltarà l'alarma. En el cas de superar els 300 ppm de CO en l'aire s'activarà l'alarma immediatament, en els altres casos s'anirà avisant a l'usuari constantment mitjançant el bot de Telegram fins que s'hagi esgotat el temps del temporitzador i seguidament s'activarà l'alarma.

Hi haurà dues regles que un cop s'hagi activat l'alarma enviaran un pregunta amb el bot de Telegram on l'usuari podrà interactuar amb OpenHAB. L'usuari podrà triar si desitja desactivar o no l'alarma des de el seu mòbil, si per alguna raó amb aquesta forma no es pot desactivar l'alarma, hi haurà un altra regla que la desactivarà un vegada l'usuari hagi reiniciat el controlador.

La segona automatització consistirà en encendre els electrodomèstics en les hores vall per tal de reduir el consum energètic i consegüentment abaratir el preu de la factura elèctrica. L'1 de juny de 2021 va entrar en vigor la nova tarifa de la llum, la qual implica canvis en els peatges d'accés i, per tant, en els períodes de discriminació horària. Aquest nou canvi suposa una discriminació horària dividida en tres períodes que són el període punta, pla i vall. El període vall és el més econòmic, es troba en els dies laborables entre les 0h a les 8h i totes les hores del cap de setmana i festius. Per aquesta raó s'han creat dues regles, una per cada electrodomèstic.

En la vivenda s'encendran la rentadora i la caldera d'aigua a les 6h de la matinada i funcionaran durant les següents dues hores fins a apagar-se, això ho faran en tots els dies menys en els caps de setmana. Es comprovarà que cap d'aquests dos electrodomèstics s'ha utilitzat en les últimes 24h amb l'ajuda de la base de dades InfluxDB, aquesta registrarà l'estat d'aquests electrodomèstics cada minut per veure si s'han encès o no. En cas de que s'ha utilitzat algun dels dos durant el dia, aquest no es tornarà a encendre a la hora programada

de les 6h. Tots els canvis d'estat o avisos s'informaran a l'usuari mitjançant la aplicació de Telegram.

La tercera automatització consistirà en convertir les llums del soterrani, saló i dormitori en llums intel·ligents, això s'ha aconseguit amb la instal·lació de sensors de presència i lluminositat entre altres mecanismes interns de les regles com temporitzadors i la detecció de la posta de sol amb el geolocalitzador Astro. Les llums intel·ligents només començaran a funcionar a partir de la posta de Sol, per això s'han creat dues regles que determinen quan surt el Sol i quan es pon. Només en el cas del soterrani es podran encendre les llums sense haver de complir la condició de la posta del Sol, això és degut a que el soterrani no rebrà tanta llum natural com les altres estàncies i es trobarà més fosc la majoria del dia. Això s'ha aconseguit llegint dos channels del binding Astro i consegüentment modificant l'estat del mode nit depenent de quin channel s'ha activat. En la següent figura podem observar quins són aquests dos channels.

Estat	Channel
Mode nit OFF - Sortida del Sol	astro:sun:home:rise#event
Mode nit ON - Posta del Sol	astro:sun:home:set#event

Taula 12. Channels Astro relacionats amb l'estat del Sol

Un cop estiguem en mode nit, s'encendran les llums si es compleixen dues condicions més, que són la detecció de moviment en algun sensor de presència PIR i un nivell determinat de obscuritat en les LDR. Per aquesta tasca s'han creat tres regles, un per cada estància. Un cop s'hagin encès les llums hauran de mantenir-se enceses si es segueix detectant moviment, per aquesta tasca també s'han utilitzat tres regles. Les llums es mantindran enceses durant 10 segons i si no es torna a detectar moviment s'apagarien, per cada cop que el sensor de presència detecta moviment es reinicialitza el temporitzador dels 10 segons. Tots els canvis d'estat de les llums i del mode nit s'informaran a l'usuari mitjançant la aplicació de Telegram.

Després de tota aquesta explicació de les automatitzacions, una correcta distribució de la càrrega computacional cobra més sentit, ja que no ens interessa fer totes aquestes tasques als nodes o al núvol, això només baixaria el rendiment del sistema domòtic i augmentaria el tràfic de dades i la latència. És molt més interessant utilitzar un maquinari potent com la Raspberry Pi en una capa intermèdia entre el núvol i els nodes, s'utilitza aquest maquinari per aquesta raó i per altres que s'han explicat en els blocs anteriors.

### 3.3.4. Base de dades

OpenHAB pot emmagatzemar dades mitjançant els serveis de persistence, les dades es poden recuperar en un moment posterior, per exemple per restaurar el sistema després d'un reinici, per preparar gràfics per a la visualització en una interfície d'usuari o per utilitzar-les dins de les automatitzacions.

OpenHAB guarda els estats dels items en una base de dades, la qual va lligada a un servei de persistence determinat, podem tenir més d'un servei de persistence carregat, i cadascun d'aquests pot ser configurat independentment. De les diferents opcions que brinda OpenHAB, s'ha utilitzat una base de dades anomenada InfluxDB degut a la seva popularitat.

Aquesta servirà per millorar la eficiència de les automatitzacions, bàsicament la seva funció és la d'emmagatzemar les dades que ens interessa analitzar per poder consultar-les posteriorment i actuar segons el seu valor. Algunes de les funcions que aporta són la possibilitat d'analitzar un cert volum de dades i extreure d'aquest valors com el màxim, mínim o la mitjana, també podem comprovar si l'estat d'algun item ha canviat en les últims minuts, hores o dies, o per exemple fer la suma d'un cert nombre de dades a partir d'un determinat moment. Totes les operacions que pot fer la base de dades han d'anar acompanyades del període de temps del qual volem analitzar. En el nostre cas, només s'acudirà a la base de dades InfluxDB per comprovar si els electrodomèstics s'han engegat durant el dia, però és bastant clar que es poden implementar automatitzacions molt més potents amb l'ús d'aquestes bases de dades.

En la següent taula podem veure com està configurada la base de dades en qüestió, l'arxiu que s'haurà de modificar serà el influxdb que es troba a la carpeta services.

Paràmetre	Valor
url	http://192.168.1.10:8086
version	V2
user	domotica
password	adminadmin
token	XvucKS3heCpAHQSS-SFuve_h6LOjWoJ0U
retentionPolicy	domotica

Taula 13. Configuració de la base de dades Influxdb



Veiem com aquesta estarà en execució dins del controlador en un port determinat, la versió utilitzada serà la segona versió i la més recent. S'haurà d'accedir amb un usuari i contrasenya, i es necessitarà d'un token per xifrar les dades transmises. L'últim paràmetre fa referència a quin lloc s'hauran d'enviar els dades, aquest s'haurà d'haver creat prèviament accedint a la direcció URL de la base de dades.

També s'haurà d'editar l'arxiu influxdb localitzat a la carpeta persistence, en aquest haurem d'especificar de quina manera volem guardar les dades a la base de dades. Es pot fer un accés a la base de dades cada minut, hora, dia o per cada canvi d'estat del item. En el nostre cas s'ha optat per guardar l'estat del items dels electrodomèstics cada minut i per cada canvi d'estat, mentre que l'estat dels altres items es guardarà per cada canvi d'estat només ja que així és com està determinat per defecte.

## 4. COMUNICACIÓ

El sistema domòtic proposat en aquest projecte es basa en una xarxa mallada, aquesta xarxa està formada per uns nodes. Els protocols de comunicació dels dispositius domòtics sempre s'han enfocat en eficiència energètica, seguretat, confort i en la accessibilitat entre l'usuari amb el sistema de control de la seva vivenda. La majoria d'aquests protocols utilitzen una infraestructura de cablejat independent com és KNX i LonWorks o utilitzen la infraestructura existent com és el cas de X10. En el nostre projecte s'ha optat per un protocol de comunicació amb una infraestructura WSN, o sigui, amb una xarxa sense fil de sensorització, aquestes han estat utilitzades en diferents camps com són la indústria 4.0 entre altres. El creixement de les IoT ha fet que les noves aplicacions basin les seves comunicacions en xarxes sense fil degut a la seva flexibilitat i baix cost de implementació.

Un dels objectius d'aquest projecte és utilitzar el protocol ZigBee que es troba implementat en els mòduls XBee. Aquests dispositius tenen la capacitat de llegir els sensors, digitals o analògics, directament i transmetre aquesta informació al seu node coordinador o router per si mateix sense l'ajuda d'un tercer. De la mateixa manera, posseeixen sortides PWM, digitals i analògiques.

Aquesta arquitectura ens permet aconseguir un cost d'implementació bastant baix en comparació amb els sistemes domòtics comercials ja que aquests utilitzen controladors complexos i d'elevat cost a més que limiten la versatilitat. Amb el descrit anteriorment i amb la utilització d'un protocol de comunicació obert com és el MQTT per la comunicació entre nodes i controlador, i amb la incorporació de OpenHAB com a interfície gràfica d'usuari aconseguim un sistema domòtic flexible, robust, obert, eficient i fàcil de mantenir.

### 4.1. Zigbee

#### 4.1.1. Característiques

La funcionalitat d'aquest protocol es basa en la facilitat per a comunicar sense fils dispositius electrònics de baix consum, d'una manera segura i amb una velocitat d'enviament de dades baixa. El baix consum elèctric d'aquests dispositius, la topologia en malla i la seva senzilla integració electrònica faciliten el seu ús en la domòtica. Aquesta tecnologia ens permet estalviar-nos l'esforç de cablejar gràcies a la seva comunicació sense fil, el qual també ens permet reubicar actuadors o sensors fàcilment.

En la següent taula podem observar les característiques de ZigBee i d'algunes de les alternatives tecnologies que poden ser utilitzades en la domòtica.

Protocols de comunicació sense fils en entorns domèstics			
Característica	ZigBee	Z-Wave	ISM genèric
Freqüència	2,4-2,483 GHz	868 MHz (Europa)	900 MHz-2,4 GHz-5 GHz
Velocitat de transferència	250 kb/s	9,6 kb/s	25-384 kb/s
Màxim rang TX/RX	10-50 m	> 100 m	> 100 m
Sensibilitat RX típica	-87 a -98 dBm	-104 dBm	-112 a -117 dBm
Potència TX típica	0 a +5 dBm	-20 a +5 dBm	+13 dBm
Topologia en malla	Si	Si	No
Corrent en mode transmissió	35 mA	36 mA	290 mA

Taula 14. Característiques dels protocols domèstics sense fils

Una possible alternativa és Z-Wave, de Zensys, principalment enfocat a domòtica, específicament al control i monitoratge de sistemes domèstics, és menys flexible però més econòmic que ZigBee. Opera en la banda de freqüències ultra altes, que li atorga un bon rang sobretot perquè es troba fora dels 2,4 GHz que es troben bastant saturats, però la seva velocitat de transferència és baixa. Més endavant tractarem quins problemes podrien aparèixer si treballem en aquest rang de freqüències.

Si bé és una alternativa bastant atractiva, s'ha optat pel protocol ZigBee per el suport que li brinda la seva comunitat i els estudis sobre ell, a l'ésser un protocol més veterà. A més, ZigBee permet fer tants salts com sigui necessari per transmetre les dades mentre que Z-Wave només en permet fer-ne quatre, aquests salts serien la funció que farien els routers. Pel que respecte a la sensibilitat RX típica, aquesta fa referència a la mínima senyal discernible que es detectaria en mode recepció, mentre que la potència TX típica fa referència a la capacitat de guany amb la qual es transmeten les dades. Totes dues característiques són bastant semblants en els tres protocols esmentats en la taula anterior, tot i que variarien segons el dispositiu utilitzat.

A efectes comparatius, s'afegeixen les característiques genèriques bàsiques d'un dispositiu transceptor de banda ISM (industrial, científica i mèdica), pel consum en aquest cas s'ha seleccionat el del ESP8266. Addicionalment, si bé WiFi i Bluetooth també són protocols sense fils, no s'enfoquen en control i monitoratge de sistemes, sent excessivament potents i sobre qualificats per a les necessitats del present projecte. Existeixen altres protocols i sistemes per a xarxes domòtiques que no són sense fils però que també s'empren amb una certa

freqüència, com l'estàndard KNX. Encara que les seves característiques poden ser comparables o fins i tot superiors en rendiment a aquests protocols sense fils, en emprar cablejat són menys pràctiques i no són l'objecte d'aquest projecte.

Per aquests motius en el nostre sistema domòtic, concretament els mòduls XBee, es comuniquen entre ells amb el protocol ZigBee, que és un de les tecnologies més populars per crear xarxes sense fils de sensorització. També s'utilitza ZigBee perquè és ideal per enviar dades en intervals periòdics reduint consum energètic ja que els sensors solen estar alimentats amb bateries generalment. Amb la utilització d'aquest protocol també s'ha tingut en compte la possibilitat de poder ampliar la xarxa de sensors degut al creixement continu de les IoT i la facilitat que es crearia per poder introduir noves tasques automatitzades de la vivenda.

#### 4.1.2. Interferències

La comunicació entre els mòduls XBee en la xarxa mallada es farà mitjançant la freqüència de 2,4 GHz en uns canals determinats. Això vol dir que l'amplada de banda de la xarxa WiFi domèstica no haurà de suportar el trànsit intern pel qual es comuniquen aquests mòduls, per la qual cosa tota la amplada de banda queda lliure per el sistema domòtic, això s'aconseguirà triant els canals ZigBee de manera que no s'interposin amb el canals del WiFi que utilitza la xarxa domèstica. En la següent figura podem veure els canals d'arreu del món de ZigBee interposats amb els canals WiFi.

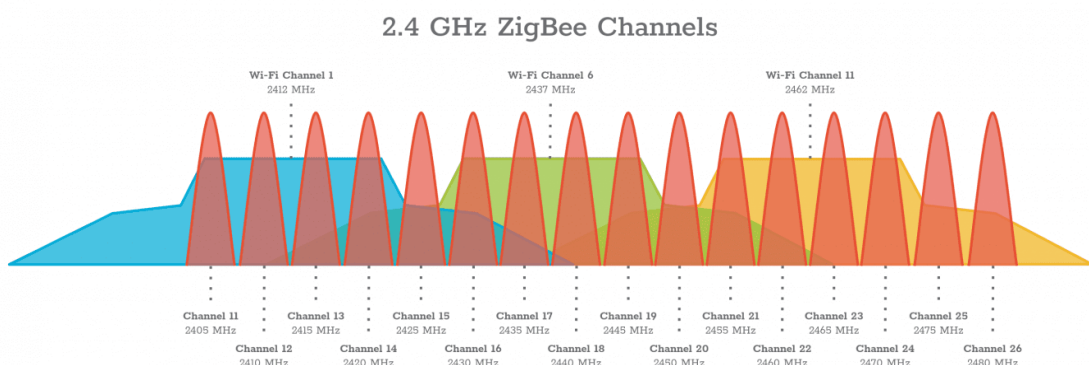


Figura 24. Canals de 2,4 GHz interposats de ZigBee i WiFi

Aquesta comunicació sense fil encara que no es realitza en les mateixes bandes reservades per a la xarxa WiFi domèstica, la qual també utilitza la freqüència de 2,4 GHz, podria crear problemes de interferència creuada entre els senyals dels mòduls XBee i la xarxa WiFi de la vivenda. Aquest problema ha estat estudiat en el passat i es va concloure que, per a evitar la

interferència creuada en la banda de 2,4 GHz, un amplada de banda de 20MHz hauria de deixar-se desocupat entre els canals operatius de la xarxa WiFi i els mòduls XBee. També s'ha de dir que existeixen solucions comercials per aquest problema. Per exemple, hi han fabricants que proporcionen passarel·les programables per connectar els mòduls XBee amb una IP a la xarxa WiFi, d'aquesta forma els mòduls XBee passarien a treballar en el mateix entorn que els dispositius WiFi amb una IP, però són relativament cares. Una altra alternativa, que seria la més ideal, seria canviar els mòduls actuals per uns amb capacitat de treballar en la freqüència de 900 MHz, encara que s'hauria de comprovar si està permès treballar en aquesta banda segons el país en que ens situem ja que aquesta banda de freqüència no és universal.

S'ha de dir també, que encara que hi hagin interferències creuades no afectaria severament el funcionament del sistema domòtic, ja que hi han factors com la distància entre els nodes que podrien reduir, si és possible, aquestes interferències aproximant físicament els nodes. Un altre factor és el tràfic habitual de la xarxa WiFi en una vivenda, el qual sol ser bastant lleuger i ajuda a mantenir la comunicació estable entre els mòduls. Per això, s'ha de triar amb especial atenció el canal de la banda de freqüència operativa en el nostre cas, així aconseguim maximitzar el percentatge de paquets entregats amb èxit.

El coordinador és l'encarregat de crear la xarxa amb la qual es comunicaran els mòduls XBee, aquest durant la posada en marxa fa un anàlisi de tots els canals als quals s'hi pot connectar i tria automàticament el canal més qualificat.

## **4.2. OpenHAB**

En aquest apartat veurem d'una manera gràfica com funcionen i es comuniquen les diferents parts del sistema domòtic.

### **4.2.1. Base de dades**

En els apartats anteriors s'ha explicat com funciona la base de dades InfluxDB, i ara es mostrarà gràficament aquest funcionament en les següents figures.



Figura 25. Canvi d'estat dels electrodomèstics dins de la base de dades

En la figura anterior s'observa l'estat dels electrodomèstics rentadora i caldera, s'han accionat manualment des de la interfície de OpenHAB. En aquest cas s'ha apagat la caldera i s'ha encès la rentadora, la base de dades enregistra tots aquests canvis de d'estat a més de llegir l'estat cada minut.

En la següent figura podem veure l'estat dels sensors PIR que tenim distribuït per la vivenda, veiem com canvien d'estat d'acord al moviment de les persones dins de la vivenda.



Figura 26. Canvi d'estat dels PIR dins de la base de dades

#### 4.2.2. Interfície gràfica d'usuari

Aquesta serà la forma amb la qual l'usuari podrà comunicar-se amb el control domòtic. Tenim dues formes d'utilitzar la interfície gràfica d'usuari, hi ha una versió més moderna i completa i un altra més bàsica i antiga, la primera s'ha creat a partir d'una edició manual dins el bloc de

pagines de OpenHAB i la segona s'ha creat a partir de codi dins l'aplicació de Visual Code Studio modificant la carpeta sitemaps.

En les següents figures podem veure la interfície gràfica més completa, a part del control domòtic conté la estació meteorològica incorporada també.

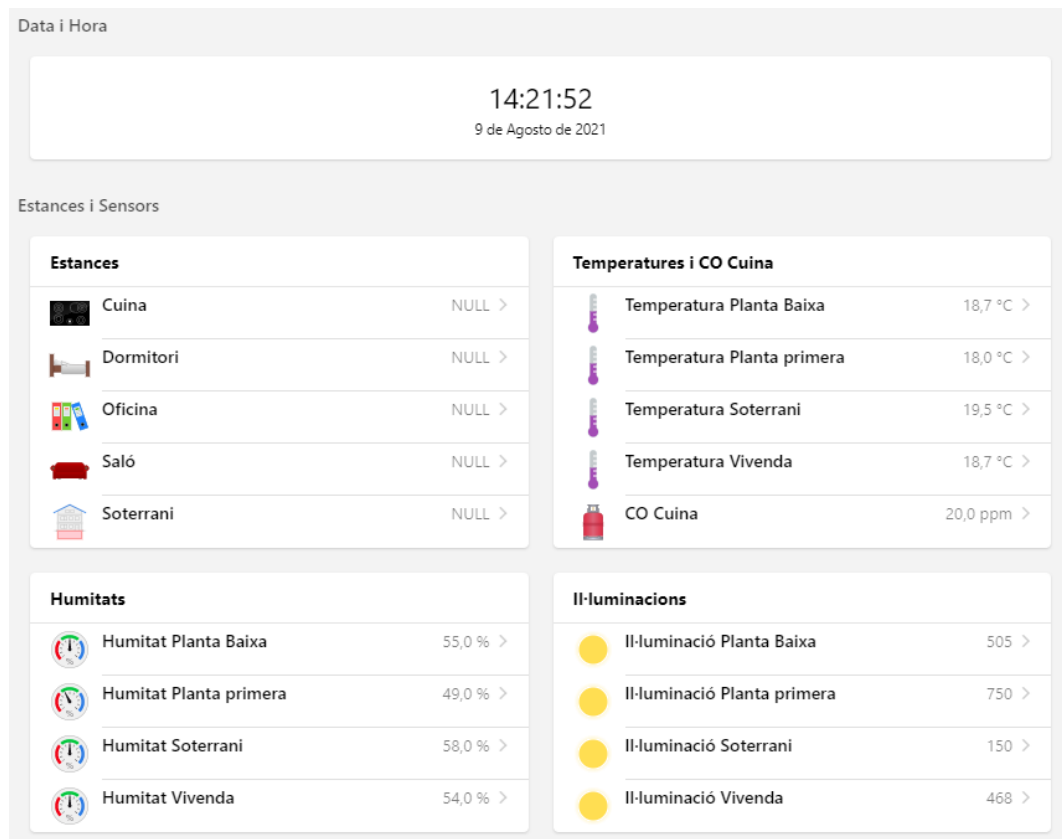


Figura 27. Pàgina d'inici de la interfície gràfica d'usuari 1

Veiem com tenim diferents blocs on mostrem la hora i la data, més a baix les temperatures i humitats de la vivenda per plantes, i també el nivell de lluminositat i un control per estances que veurem desplegat més endavant.

En totes les pestanyes que compten amb una fletxa en la seva part dreta s'obrirà un desplegable mostrant el contingut d'aquestes, aquesta forma de presentar les dades ajuda a mantenir una organització i un ordre que facilita la comprensió i lectura de les dades.

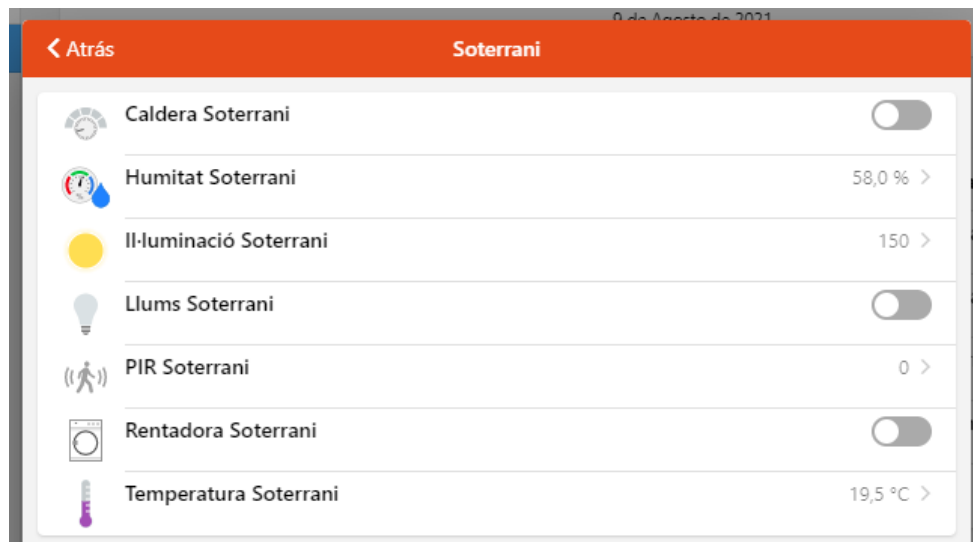


Figura 28. Menú desplegable de l'estança soterrani

Aquest és el desplegable que s'obre si seleccionem soterrani dins del bloc de les estances, veiem com mostra tots els elements que formen la domotització del soterrani. Dins de la pàgina inicial també tenim un bloc pels actuadors i per la estació meteorològica.

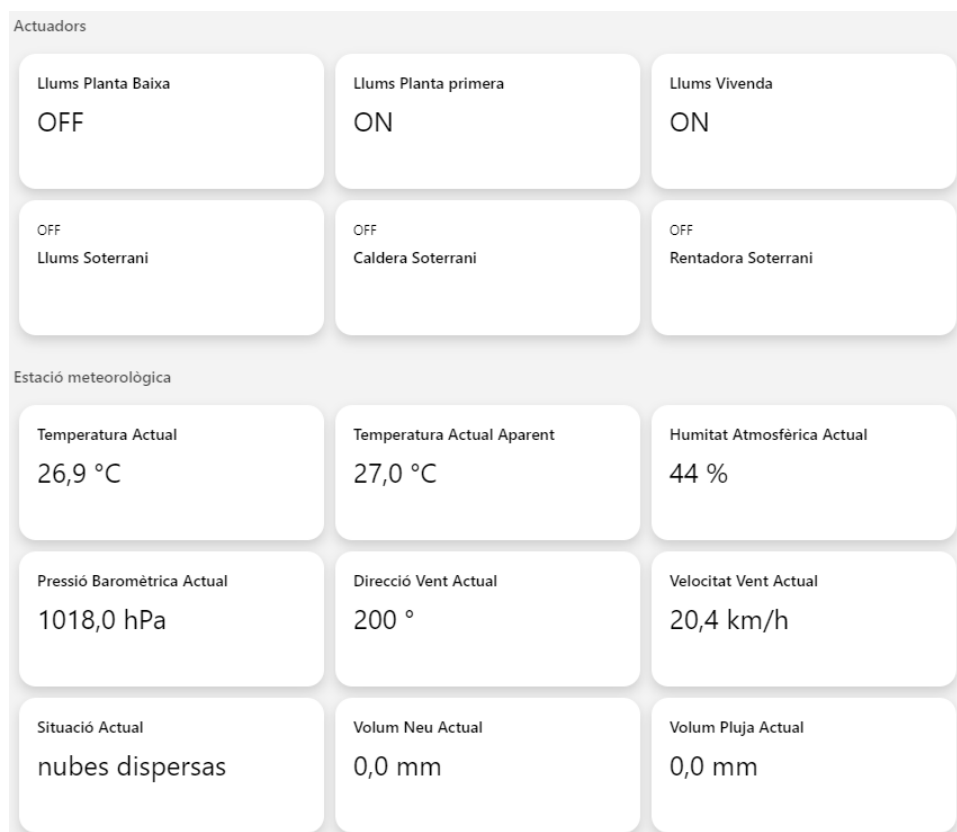


Figura 29. Pàgina d'inici de la interfície gràfica d'usuari 2



En aquest bloc dedicat als actuadors tenim les llums i electrodomèstics, les llums estan dividides per planes i per cada planta s'obre un desplegable, en el cas del soterrani com només hi ha una llum no s'obre cap desplegable. L'estació meteorològica proporciona moltes dades però només s'han seleccionat les més interessants com és la temperatura, humitat, pressió, velocitat del vent i volum de la pluja.

En canvi la segona forma d'interfície gràfica feta amb el Visual Code Studio, no ofereix una accessibilitat i comoditat tan bona com la primera. En les següents figures podem veure com queda el control domòtic.

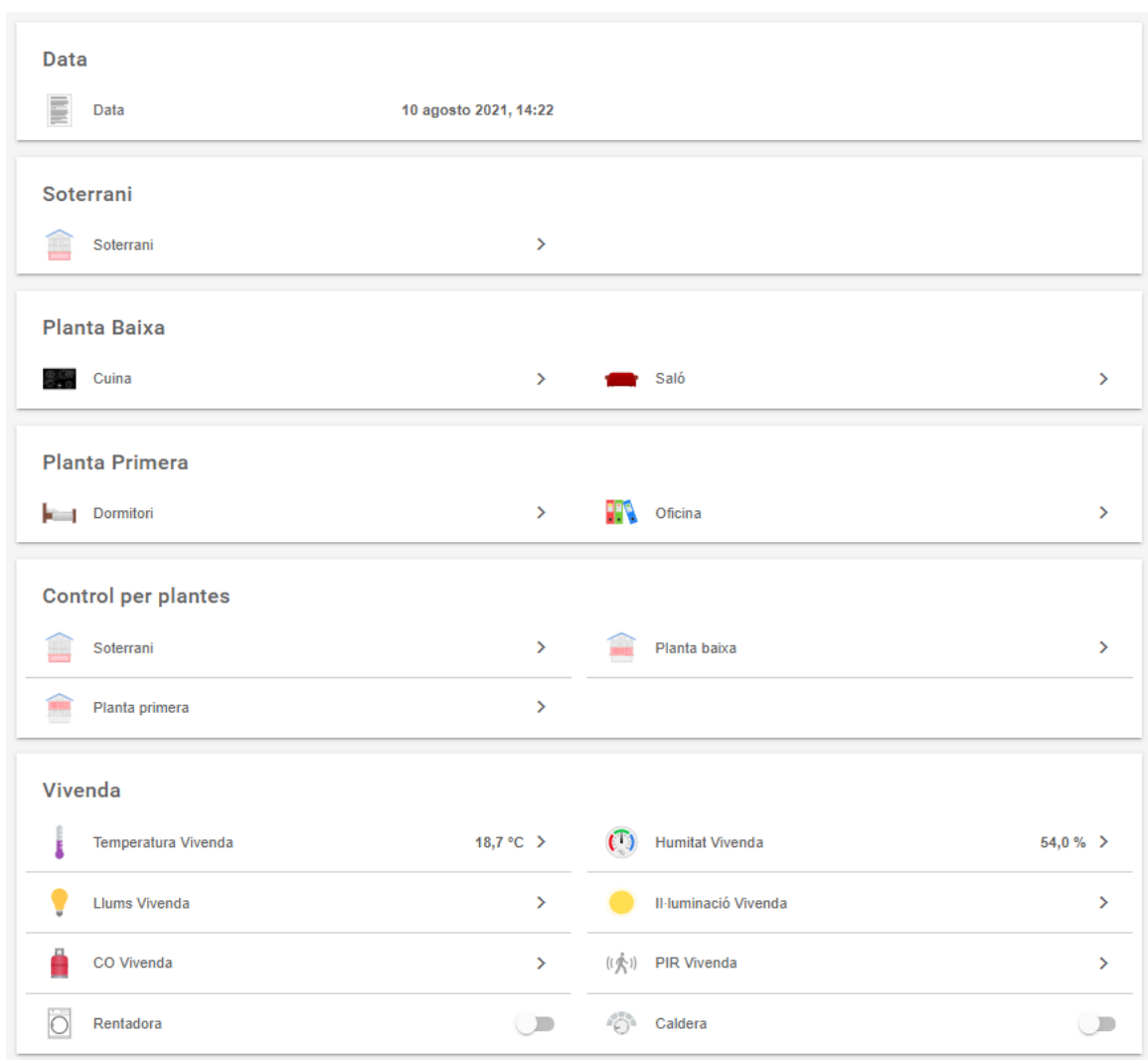


Figura 30. Pàgina d'inici de la interfície gràfica d'usuari utilitzant els Sitemaps

### 4.2.3. Aplicació Telegram

S'ha creat un bot dins de la aplicació Telegram pel telèfon mòbil que ens avisarà de què està succeint dins de la vivenda respecte les automatitzacions que s'han activat i sobre altres esdeveniments. S'ha configurat d'una manera que l'aplicació esborraria tots els avisos antics de més de 24 hores. En la següent figura podem veure alguns exemples del seu funcionament.

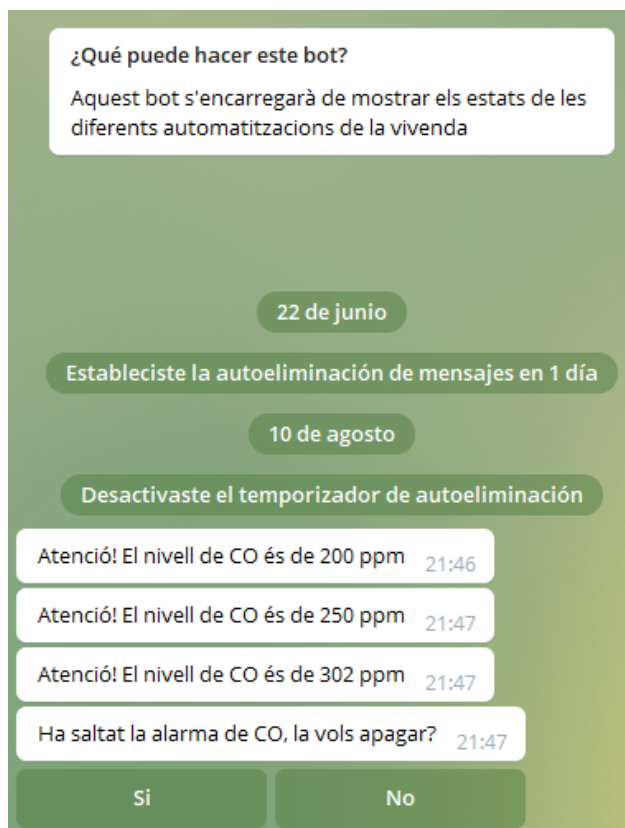


Figura 31. Exemple d'un disparament de l'alarma de CO dins el bot de Telegram

En l'anterior figura veiem com ha saltat l'alarma després de que el nivell de CO hagi superat els 300 ppm, des del mòbil podrem apagar l'alarma i mentre el nivell es mantingui dins del rang de toxicitat seguirem rebent avisos constants.

Hi ha més casos on rebríem avisos al mòbil com per exemple quan s'ha encès alguna llum de la vivenda perquè ha detectat moviment, quan s'activa al mode nit a la posta del Sol o quan s'engega algun electrodomèstic a les hores vall del mercat elèctric.

#### 4.2.4. Aplicació mòbil per l'accés remot

Fins ara tots els accessos que s'han mencionat per accedir al control domòtic de OpenHAB són des de la xarxa local només, ja que el servidor funciona dins de la Raspberry Pi i només es connecta a Internet per fer anar alguns serveis como l'estació meteorològica.

Per aquesta raó el connector al núvol de OpenHAB li permet connectar el seu servidor d'automatització de la llar local a una instància de OpenHAB Cloud. Amb aquesta instància podem connectar-nos al nostre control domòtic des de pràcticament qualsevol lloc amb el telèfon mòbil. A part d'aquest accés remot proporciona la possibilitat de crear diferents usuaris i assignar privilegis, a més de poder activar unes notificacions pel telèfon mòbil gratuïtes.

S'ha d'haver configurat amb anterioritat instal·lant el corresponent Binding i creant un compte a la pàgina oficial de OpenHAB, ens demanarà a part del e-mail i contrasenya, unes dues dades que trobarem dins del directori on s'ha instal·lat OpenHAB i que podem veure en la següent taula.

Paràmetre	Valor
UUID	f070c505-e010-4e1a-8a98-0b40c7906b3a
Secret	qbwiFsO358WRTPNNASB3

Taula 15. Característiques dels protocols domèstics sense fils

Una vegada tenim configurat tot, procedirem a instal·lar l'aplicació de OpenHAB dins del telèfon mòbil, depenent del sistema operatiu d'aquest s'haurà d'accedir a un lloc o un altre. En el nostre cas s'utilitza Android així que hem accedit a la plataforma de distribució digital d'aplicacions mòbils Google Play. En les següents figures podem observar com és exactament l'accés remot des d'un telèfon mòbil a OpenHAB.

Veiem com el format visual de l'aplicació mòbil és el mateix que el de la interfície gràfica feta amb Visual Code Studio modificant l'arxiu sitemaps, també comptem amb la possibilitat d'utilitzar l'altra interfície gràfica més moderna però les mides de la pantalla d'un telèfon mòbil són bastant reduïdes i no ens proporcionen una bona presentació.

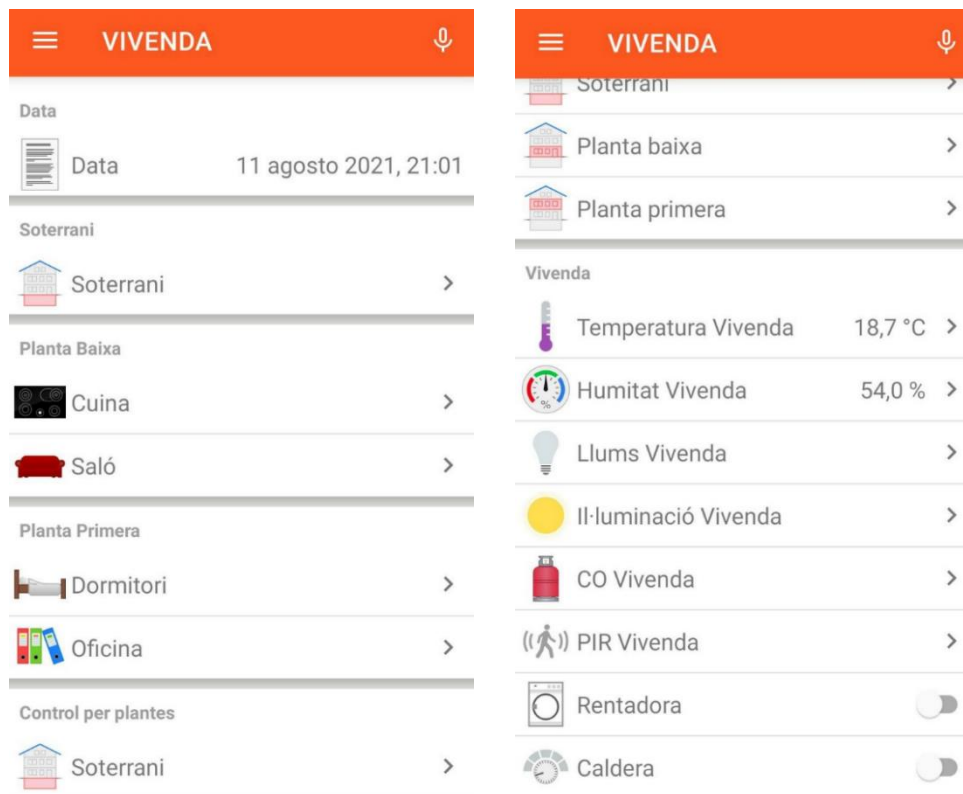


Figura 32. Pàgina d'inici de la interfície gràfica d'usuari al telèfon mòbil



Figura 33. Menú desplegable de l'estança soterrani al telèfon mòbil

En la figura anterior s'observa com és el desplegable de la planta soterrani, compta amb les mateixes dades que el desplegable vist anteriorment de la interfície gràfica moderna.

## **5. RESUM DEL PRESSUPOST**

El projecte de la domotització d'una vivenda mitjançant una xarxa mallada mesh ascendeix a un cost de sis mil dos-cents vint-i-dos amb setanta-tres cèntims, sense IVA.

## 6. CONCLUSIONS

El següent projecte compleix amb el Reglament Electrotècnic de Baixa Tensió i amb les instruccions tècniques complementàries necessàries per dur a terme la instal·lació domòtica, també compleix amb les normatives dels diferents àmbits que formen part del projecte.

Finalment podríem dir que s'han complert els objectius d'aquest projecte veient com la utilització d'un protocol com és el cas de ZigBee i la seva xarxa mallada ha resultat ser molt eficient energèticament i que la comunicació entre els nodes amb el protocol MQTT ha facilitat la futura incorporació de dispositius amb recursos limitats i que puguin funcionar més endavant com sensors o actuadors. La utilització de la xarxa mallada ha sigut fonamental ja que és la única forma segura i robusta per aconseguir connectivitat en una vivenda intel·ligent, especialment quan ens interessa afegir més i més dispositius en el futur.

La tecnologia de la computació en la boira ens ha permès dividir la càrrega computacional entre els nodes, el controlador domòtic i el núvol, a més de poder concentrar en el controlador una gran part d'aquesta ja que aquest és l'element més potent de la instal·lació, per aquests motius es pot afirmar que aquesta infraestructura de boira proporciona respostes gairebé en temps real i redueix significativament la latència si la comparem amb altres infraestructures com és la computació en el núvol. L'únic inconvenient que es podria destacar són les interferències creuades que apareixen en ambients conflictius entre els dispositius WiFi i ZigBee, i que les comunicacions MQTT no estan xifrades. Les solucions d'aquests dos inconvenients s'han detallat en els blocs anteriors.

Abdelaziz Essami

Graduat en Enginyeria Electrònica Industrial i Automàtica

Salt, 1 de setembre de 2021

## **7. RELACIÓ DE DOCUMENTS**

Aquest projecte està format per cinc documents que són la memòria, els plànols, el plec de condicions, l'estat d'amidaments i el pressupost.

## 8. BIBLIOGRAFIA

DIYLOT. ZigBee Communication Protocol Tutorial for Smart Home. (<https://diyi0t.com/zigbee-communication-protocol-tutorial-for-smart-home/>, 7 de maig de 2021).

GITHUB. Community forum. (<https://github.community/>, 15 de maig de 2021).

HUI CHENG. Designing Efficient Smart Home Management with IoT Smart Lighting: A Case Stud. (<https://www.hindawi.com/journals/wcmc/2020/8896637/>, 25 de març de 2021).

JECRESPOM. MQTT. (<https://aprendiendoarduino.wordpress.com/2018/11/19/mqtt/>, 5 de maig de 2021).

JECRESPOM. ZigBee. (<https://aprendiendoarduino.wordpress.com/2016/11/16/zigbeexbee/>, 5 de maig de 2021).

JUANMA. Hacia el Fog Computing, ¿en qué punto estamos?. (<https://iofutura.com/2019/06/hacia-el-fog-computing-en-que-punto-estamos/>, 20 de maig de 2021).

OPENHAB. Documentation. (<https://www.openhab.org/docs/>, 5 de maig de 2021)

REGLAMENTO ELECTROTÈCNICO DE BAJA TENSIÓN (RD842/2002). McGrawHill. Madrid. 2012.

WIKIPEDIA, la enciclopedia libre. Enciclopèdia web. (<http://en.wikipedia.org/wiki/MainPage>, 5 maig de 2021).



## 9. GLOSSARI

AC: Corrent altern.

API: Interfície de programació d'aplicacions.

AT: És un mode de transmissió de dades definit per XBee, també s'anomena mode transparent.

C++: És un llenguatge de programació.

DC: Corrent continu.

DHCP: És un protocol de xarxa que permet als nodes d'una xarxa IP obtenir els seus paràmetres de configuració automàticament.

IoT: Internet de les coses. Es refereix, en termes d'informàtica, a una xarxa d'objectes físics de la vida quotidiana interconnectats mitjançant sensors, programari i altres tecnologies per tal d'intercanviar dades amb altres dispositius i sistemes a través d'Internet.

IP: Protocol d'Internet.

KNX: És una associació que ha fusionat el bus EIB amb uns altres dos, ja existents a Europa i s'estableix com a una alternativa de l'automatització.

LAN: Xarxa d'àrea local.

LDR: És un dispositiu que varia el seu valor de resistència elèctrica depenent de la quantitat de llum que incideix sobre ell.

LED: Díode emissor de llum.

MAC: Significa control d'accés a mitjans. És un identificador únic per a les interfícies de xarxa.

MQTT: És un protocol de missatgeria publicació/subscripció basat en el protocol TCP/IP.

NTP: És un protocol de sincronització de rellotges sobre una xarxa de dades de latència variable i mitjançant transmissió de paquets.

PCB: Placa de circuit imprès.

PIR: Sensor infraroig passiu.

PWM: Modulació per amplada de polsos.

QoS: Qualitat del servei.

RAM: Memòria d'accés aleatori.

REBT: Reglament electrotècnic de baixa tensió.

RF: Radiofreqüència.

SOC: Sistema en un xip.

SSL/TLS: Són protocols criptogràfics que proporcionen comunicacions segures per una xarxa, generalment Internet.

TCP: És un protocol d'internet encarregat d'informar del destí de les dades permetent la creació de connexions segures.

USB: Bus Universal en Serie.

WiFi: És una tecnologia que permet la interconnexió sense fils de dispositius electrònics.

Xbase: És el terme genèric per a tots els llenguatges de programació que deriven del llenguatge de programació dBase, originalment publicat per Ashton-Tate.

## A. CODI INFORMÀTIC

En aquest annex trobarem el codi informàtic que s'ha utilitzat en el projecte. Tenim dos nodes cadascun equipat amb un ESP8266 i l'aplicació de OpenHAB.

### A.1. Node coordinador

Aquest seria el codi que es troba dins del ESP8266 del node coordinador.

```
//// TFG-GEEIA Domotica ABDELAZIZ ESSAMI ///
```

```
//// Node coordinador ////
```

```
// Declaració de les llibreries
```

```
#include <SoftwareSerial.h>
```

```
#include <PubSubClient.h>
```

```
#include <ESP8266WiFi.h>
```

```
// Declaració de l'adreça IP del servidor MQTT Mosquitto, usuari i contrasenya
```

```
#define broker_mqtt "192.168.0.10"
```

```
#define user_mqtt "admin"
```

```
#define pass_mqtt "domotica"
```

```
// Declaració del ESP8266 com a client
```

```
WiFiClient espClient;
```

```
PubSubClient client(espClient);
```

```
// Paràmetres per la connexió WiFi, SSID i contrasenya
```

```
#define ssid "client1" // SSID de la xarxa wifi
```

```
#define password "hola1234" // contrasenya de la xarxa wifi
```

```
// Declaració de les cadenes de caràcters pel payload que contindrà les dades dels sensors

char msgbuf[50];

// Declaració de variable per controlar temps de mostreig

long ultim_msg = 0;

// Soterrani - Declaració de variables

float temp1, hum1, lux1 = 0.0;

int pir_int1 = 0;

// Cuina - Declaració de variables

float temp2, hum2, co2 = 0.0;

// Saló - Declaració de variables

float temp3, hum3, lux3 = 0.0;

int pir_int3 = 0;

// Dormitori - Declaració de variables

float temp4, hum4, lux4 = 0.0;

int pir_int4 = 0;

// Oficina - Declaració de variables

float temp5, hum5, lux5 = 0.0;

// Declaració dels topics mqtt

// Soterrani

#define st_temp "casa/st/soterrani/temperatura"

#define st_hum "casa/st/soterrani/humitat"

#define st_pir "casa/st/soterrani/pir"
```

```
#define st_lux "casa/st/soterrani/lux"

// Cuina

#define cu_temp "casa/pb/cuina/temperatura"

#define cu_hum "casa/pb/cuina/humitat"

#define cu_co "casa/pb/cuina/co"

// Saló

#define sa_temp "casa/pb/salo/temperatura"

#define sa_hum "casa/pb/salo/humitat"

#define sa_pir "casa/pb/salo/pir"

#define sa_lux "casa/pb/salo/lux"

// Dormitori

#define do_temp "casa/pp/dormitori/temperatura"

#define do_hum "casa/pp/dormitori/humitat"

#define do_pir "casa/pp/dormitori/pir"

#define do_lux "casa/pp/dormitori/lux"

// Oficina

#define of_temp "casa/pp/oficina/temperatura"

#define of_hum "casa/pp/oficina/humitat"

#define of_lux "casa/pp/oficina/lux"

// Declaració dels nous pins per la comunicació serial de la XBee

SoftwareSerial mySerial(13, 15); // RXD1 | TXD1

// Definició del procés per la connexió WiFi
```

```
void setup_wifi()

{

    delay(10);

    WiFi.begin(ssid, password); // començem connexio wifi

    Serial.println();

    Serial.print("Esperant connexió WiFi...");

    while (WiFi.status() != WL_CONNECTED)

    {

        delay(500); // mentre no es connecta esperem....

        Serial.print(".");

    }

    Serial.println("");

    Serial.println("Connexió WiFi OK");

    Serial.println("Adreça IP: ");

    Serial.println(WiFi.localIP());

}

// Funció per fer la primera connexió amb el client i les reconexions necessàries

void reconnect() {

    // Loop fins hi hagi connexió

    while (!client.connected()) {

        Serial.print("Intentant fer la comunicació MQTT...");

        // Intent de connexió
```

```
if (client.connect("ESP8266Client",user_mqtt,pass_mqtt)) {

    Serial.println("Connectat");

}

else {

    Serial.print("No s'ha pogut connectar amb el broker, rc=");

    Serial.print(client.state());

    Serial.println("Esprerem 5 segons");

    // esprem 5s i tornem a intentar

    delay(5000);

}

}

}

// Funcions pel tractament i conversió de les dades a les seves magnituds llegibles

float convr_temp(int x)

{

    float resultat;

    resultat = (float)x * 1200 / 1024.0; // Vref Xbee=1.2V Convertim a mV

    resultat = (resultat / 10) - 50; // mV a °C (Offset de 50°C i 10mV de sensibilitat)

    return resultat;

}

float convr_hum(int x)

{
```

```
float resultat;

resultat = (float)x * 1.2 / 1024.0;

resultat = resultat * ((-0.1515/0.00636)+(1.0/1.2*0.00636)); // Vref=1.2V

return resultat; // Càlculs amb Tamb de 25°C

}

float convr_lux(int x)

{

float resultat;

resultat = (float)x;

return resultat;

}

int convr_pi(byte x)

{

int resultat;

if (x == 0x10) { resultat = 1; }

else { resultat = 0; }

}

// funció conversió CO cuina

float convr_co(int x)

{

float volts,resultat;

float RS,RS_gas,R0,ratio;
```



```
float R2 = 2000.0;

volts = (float)x * 1.2 / 1024.0;

RS = ((1.2 * R2) / volts) - R2;

R0 = RS / 1;

RS_gas = (1.2 - volts) / volts;

ratio = RS_gas / R0;

float ppm = 1538.46 * ratio;

resultat = pow(ppm,-1.709);

return resultat;

}

// Adquisició de les dades dels sensors

void parseData()

{

    // Definició del paquet de bytes rebuts del XBee End Device

    byte packBytes[27];

    // Definició de variables pel tractament i emmagatzematge de dades

    byte pir;

    int dataByteTmp, dataByteHum, dataByteLux, dataByteCO;

    // Un cop tinguem el total de 27 bytes que formarien una trama

    if (mySerial.available() == 27) //XBee/UART1/pins 0 and 1

    {

        Serial.println("Reading package."); //Serial port
```

```
// Verifiquem que la trama rebuda comença amb 0x7E per assegurar-nos de que és correcta
```

```
if (mySerial.read() == 0x7E)

{

    // Guardem els bytes un per un fins l'últim que seria el checksum [27]

    packBytes[0] = 0x7E;

    for (int i = 0; i < 27; i++)

    {

        packBytes[i + 1] = mySerial.read();

    }

    // Soterrani

    if (packBytes[12] == 0x7D) // Adreça XBEE Soterrani

    {

        pir = packBytes[20]; // Com és digital EN D4 dona en 1 = 0x10 i 0 = 0x00,
s'agafa primer per ordre de lectura

        // El 19 seria la primera dada digital pero la tenim a 0 porque nomes
pertany al grup de d10 d11 d12

        // 1 byte = 8 bits ---- 2^8 ----- 256, nosaltres tenim cada dada analogica
ve formada per 2 bytes

        dataByteHum = ((int)packBytes[21]) * 256 + int(packBytes[22]); //AD0 -> HIH

        dataByteLux = ((int)packBytes[23]) * 256 + int(packBytes[24]); //AD1 -> Ldr

        dataByteTmp = ((int)packBytes[25]) * 256 + int(packBytes[26]); //AD2 -> Tmp

        pir_int1 = convr_pi(pir);

        temp1 = convr_temp(dataByteTmp);

        hum1 = convr_hum(dataByteHum);
```

```
lux1 = convr_lux(dataByteLux);

}

// Cuina

if (packBytes[12] == 0xDD) // Adreça XBEE Cuina

{

    // 1 byte = 8 bits ---- 2^8 ----- 256, nosaltres tenim cada dada analogica
ve formada per 2 bytes

    dataByteHum = ((int)packBytes[21]) * 256 + int(packBytes[22]); //AD0 -> HIH

    dataByteCO = ((int)packBytes[23]) * 256 + int(packBytes[24]); //AD1 -> CO

    dataByteTmp = ((int)packBytes[25]) * 256 + int(packBytes[26]); //AD2 -> Tmp

    temp2 = convr_temp(dataByteTmp);

    hum2 = convr_hum(dataByteHum);

    co2 = convr_co(dataByteCO);

}

// Saló

if (packBytes[12] == 0x33) // Adreça XBEE Soterrani

{

    pir = packBytes[20]; // Com és digital EN D4 dona en 1 = 0x10 i 0 = 0x00,
s'agafa primer per ordre de lectura

    // El 19 seria la primera dada digital pero la tenim a 0 perque nomes
pertany al grup de d10 d11 d12

    // 1 byte = 8 bits ---- 2^8 ----- 256, nosaltres tenim cada dada analogica
ve formada per 2 bytes

    dataByteHum = ((int)packBytes[21]) * 256 + int(packBytes[22]); //AD0 -> HIH
```

```
dataByteLux = ((int)packBytes[23]) * 256 + int(packBytes[24]); //AD1 -> Ldr

dataByteTmp = ((int)packBytes[25]) * 256 + int(packBytes[26]); //AD2 -> Tmp

pir_int3 = convr_pi(pir);

temp3 = convr_temp(dataByteTmp);

hum3 = convr_hum(dataByteHum);

lux3 = convr_lux(dataByteLux);

}

// Dormitori

if (packBytes[12] == 0x56) // Adreça XBEE Soterrani

{

    pir = packBytes[20]; // Com és digital EN D4 dona en 1 = 0x10 i 0 = 0x00,
s'agafa primer per ordre de lectura

    // El 19 seria la primera dada digital pero la tenim a 0 porque nomes
pertany al grup de d10 d11 d12

    // 1 byte = 8 bits ---- 2^8 ----- 256, nosaltres tenim cada dada analogica
ve formada per 2 bytes

    dataByteHum = ((int)packBytes[21]) * 256 + int(packBytes[22]); //AD0 -> HIH

    dataByteLux = ((int)packBytes[23]) * 256 + int(packBytes[24]); //AD1 -> Ldr

    dataByteTmp = ((int)packBytes[25]) * 256 + int(packBytes[26]); //AD2 -> Tmp

    pir_int4 = convr_pi(pir);

    temp4 = convr_temp(dataByteTmp);

    hum4 = convr_hum(dataByteHum);

    lux4 = convr_lux(dataByteLux);

}
```

```
// Oficina

if (packBytes[12] == 0x88) // Adreça XBEE Soterrani

{

    // 1 byte = 8 bits ---- 2^8 ----- 256, nosaltres tenim cada dada analogica
    ve formada per 2 bytes

    dataByteHum = ((int)packBytes[21]) * 256 + int(packBytes[22]); //AD0 -> HIH

    dataByteLux = ((int)packBytes[23]) * 256 + int(packBytes[24]); //AD1 -> Ldr

    dataByteTmp = ((int)packBytes[25]) * 256 + int(packBytes[26]); //AD2 -> Tmp

    temp5 = convr_temp(dataByteTmp);

    hum5 = convr_hum(dataByteHum);

    lux5 = convr_lux(dataByteLux);

}

}

}

// Inicialització de la funció WiFi, del broker mqtt i de la funció callback per
llegir les comandes

void setup()

{

    // Velocitat del port serie físic i del SoftwareSerial

    mySerial.begin(9600);

    Serial.begin(9600);

    // Inicialitzem la funció del WiFi
```

```
setup_wifi();

// definim el broker mosquitto i el port

client.setServer(broker_mqtt, 1883);

}

void loop()

{

// Fem la primera connexió del client i les següents que siguin necessàries

if (!client.connected())

{

reconnect();

}

// Inicialitzem la funció de tractament de dades

parseData();

// Les dades dels sensors PIR i CO es publiquen lo més ràpid possible degut a que
les necessitem

// per fer anar les automatitzacions de la vivenda pel control de les llums
intel·ligents i l'alarma

// Pir-Soterrani

String pir_s1 = String(pir_int1);

client.publish(st_pir, (char*) pir_s1.c_str());

// Pir-Saló

String pir_s3 = String(pir_int3);

client.publish(sa_pir, (char*) pir_s3.c_str());
```

```
// Pir-Dormitori

String pir_s4 = String(pir_int4);

client.publish(do_pir, (char*) pir_s4.c_str());

// CO-Cuina

client.publish(cu_co, dtostrf(co2, 4, 2, msgbuf));

// Publiquem les dades dels sensors restants cada un cert temps

long temps_ara = millis();

if (temps_ara - ultim_msg > 5000)

{

    ultim_msg = temps_ara;

    // tractem les variables amb 2 enters i 2 decimals de precisió i les convertim a
    string

    // Soterrani

    client.publish(st_temp, dtostrf(temp1, 4, 2, msgbuf));

    client.publish(st_hum, dtostrf(hum1, 4, 2, msgbuf));

    client.publish(st_lux, dtostrf(lux1, 4, 2, msgbuf));

    // Cuina

    client.publish(cu_temp, dtostrf(temp2, 4, 2, msgbuf));

    client.publish(cu_hum, dtostrf(hum2, 4, 2, msgbuf));

    // Saló

    client.publish(sa_temp, dtostrf(temp3, 4, 2, msgbuf));

    client.publish(sa_hum, dtostrf(hum3, 4, 2, msgbuf));

    client.publish(sa_lux, dtostrf(lux3, 4, 2, msgbuf));
```

```
// Dormitori

client.publish(do_temp, dtostrf(temp4, 4, 2, msgbuf));

client.publish(do_hum, dtostrf(hum4, 4, 2, msgbuf));

client.publish(do_lux, dtostrf(lux4, 4, 2, msgbuf));

// Oficina

client.publish(of_temp, dtostrf(temp5, 4, 2, msgbuf));

client.publish(of_hum, dtostrf(hum5, 4, 2, msgbuf));

client.publish(of_lux, dtostrf(lux5, 4, 2, msgbuf));

}

// Fem una connexió en cada loop amb client per refrescar dades per publicar i
mantenir la connexió

client.loop();

}
```

## A.2. Node actuador

Aquest seria el codi que es troba dins del ESP8266 del node actuador.

```
//// TFG-GEEIA Domotica ABDELAZIZ ESSAMI ///
```

```
//// Node actuador ////
```

```
// Declaració de les llibreries
```

```
#include <PubSubClient.h>
```

```
#include <ESP8266WiFi.h>
```

```
// Declaració de l'adreça IP del servidor MQTT Mosquitto, usuari i contrasenya
```

```
#define broker_mqtt "192.168.0.10"
```



```
#define user_mqtt "admin"

#define pass_mqtt "domotica"

// Declaració del ESP8266 com a client

WiFiClient espClient;

PubSubClient client(espClient);

// Paràmetres per la connexió WiFi, SSID i contrasenya

#define ssid "client1" // SSID de la xarxa wifi

#define password "hola1234" // contrasenya de la xarxa wifi

// Declaració dels pins dels reles

const int st_llumsr = 4;

const int st_renr = 5;

const int st_caldr = 12;

const int cu_llumsr = 13;

const int sa_llumsr = 15;

const int sa_alarmar = 14;

const int do_llumsr = 0;

const int of_llumsr = 2;

// Declaració dels topics mqtt

// Soterrani

#define st_llums "casa/st/soterrani/llums"

#define st_rent "casa/st/soterrani/rentadora"

#define st_cald "casa/st/soterrani/caldera"
```

```
// Cuina

#define cu_llums "casa/pb/cuina/llums"

// Saló

#define sa_llums "casa/pb/salo/llums"

#define sa_alarma "casa/pb/salo/alarma"

// Dormitori

#define do_llums "casa/pp/dormitori/llums"

// Oficina

#define of_llums "casa/pp/oficina/llums"

// Declaració de les cadenes de caràcters pel payload i el topic

// que contindran les comandes ON/OFF pels actuadors

char state[40];

char topic_s[40];

// Definició del procés per la connexió WiFi

void setup_wifi()

{

    delay(10);

    WiFi.begin(ssid, password); // començem connexio wifi

    Serial.println();

    Serial.print("Esperant connexió WiFi...");

    while (WiFi.status() != WL_CONNECTED)

    {
```

```
    delay(500); // mentre no es connecta esperem...

    Serial.print(".");

}

Serial.println("");

Serial.println("Connexió WiFi OK");

Serial.println("Adreça IP: ");

Serial.println(WiFi.localIP());

}

// Definició de la funció callback que llegeix les comandes que envia OpenHAB

void callback(char* topic, byte* payload, unsigned int length)

{

    memset(state, 0, strlen(state)); // Posem a 0 la cadena de caràcters state

    memset(topic_s, 0, strlen(topic_s)); // Posem a 0 la cadena de caràcters topic_s

    for (int i = 0; i < length; i++)

    {

        state[i] = (char)payload[i]; // Guardem el valor del payload a la variable state

    } // el payload conté les comandes (ON/OFF) en format

string

    for (int i = 0; i < strlen(topic); i++)

    {

        topic_s[i] = topic[i]; // Guardem el valor del topic a la variable topic_s

    } // aquí sabrem a quin topic pertany la comanda enviada més

a dalt
```

```
}

// Funció per fer la primera connexió amb el client i les reconexions necessàries

void reconnect()

{

    // Loop fins hi hagi connexió

    while (!client.connected()) {

        Serial.print("Intentant fer la comunicació MQTT...");

        // Intent de connexió

        if (client.connect("ESP8266Client_1",user_mqtt,pass_mqtt)) {

            Serial.println("Connectat");

            // fem una subscripció als topics per poder llegir-los més endavant

            client.subscribe(st_llums);

            client.subscribe(st_rent);

            client.subscribe(st_cald);

            client.subscribe(cu_llums);

            client.subscribe(sa_llums);

            client.subscribe(sa_alarma);

            client.subscribe(do_llums);

            client.subscribe(of_llums);

        } else {

            Serial.print("No s'ha pogut connectar amb el broker, rc=");

            Serial.print(client.state());
```

```
    Serial.println("Esprerem 5 segons");

    // espem 5s i tornem a intentar

    delay(5000);

}

}

}

void setup()

{

    // Definició dels pins dels reles

    pinMode(st_llumsr, OUTPUT);

    pinMode(st_rentr, OUTPUT);

    pinMode(st_caldr, OUTPUT);

    pinMode(cu_llumsr, OUTPUT);

    pinMode(sa_llumsr, OUTPUT);

    pinMode(sa_alarmar, OUTPUT);

    pinMode(do_llumsr, OUTPUT);

    pinMode(of_llumsr, OUTPUT);

    // Definició de l'estat inicial dels reles

    // Serà una connexió Normalment Obert, high = off

    // No actuem sobre les altres sortides ja que es necessiten pel boot

    digitalWrite(st_llumsr, HIGH);
```

```
digitalWrite(st_rentr, HIGH);

digitalWrite(st_caldr, HIGH);

digitalWrite(cu_llumr, HIGH);

digitalWrite(sa_alarma, HIGH);

// Velocitat del port serie

Serial.begin(9600);

// Inicialitzem la funció del WiFi

setup_wifi();

// Definim el broker mosquitto i el port

client.setServer(broker_mqtt, 1883);

// Habilitem la funció callback

client.setCallback(callback);

}

void loop()

{

    if (!client.connected())

    {

        reconnect();

    }

    client.loop();

    // Comprovem si a la variable state ha arribat una comanda ON/OFF i a quin topic pertany

    // Soterrani llums
```

```
if (strcmp(state, "ON") == 0 && strcmp(topic_s, st_llums) == 0)

{

    Serial.println("Llums Soterrani ON");

    digitalWrite(st_llumsr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, st_llums) == 0)

{

    Serial.println("Llums Soterrani OFF");

    digitalWrite(st_llumsr, HIGH);

}

// Soterrani Rentadora

if (strcmp(state, "ON") == 0 && strcmp(topic_s, st_rent) == 0)

{

    Serial.println("Rentadora ON");

    digitalWrite(st_renr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, st_rent) == 0)

{

    Serial.println("Rentadora OFF");

    digitalWrite(st_renr, HIGH);

}

// Soterrani Caldera
```

```
if (strcmp(state, "ON") == 0 && strcmp(topic_s, st_cald) == 0)

{

    Serial.println("Caldera ON");

    digitalWrite(st_caldr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, st_cald) == 0)

{

    Serial.println("Caldera OFF");

    digitalWrite(st_caldr, HIGH);

}

// Cuina llums

if (strcmp(state, "ON") == 0 && strcmp(topic_s, cu_llums) == 0)

{

    Serial.println("Llums Cuina ON");

    digitalWrite(cu_llumsr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, cu_llums) == 0)

{

    Serial.println("Llums Cuina OFF");

    digitalWrite(cu_llumsr, HIGH);

}

// Saló llums
```



```
if (strcmp(state, "ON") == 0 && strcmp(topic_s, sa_llums) == 0)

{

    Serial.println("Llums Saló ON");

    digitalWrite(sa_llumsr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, sa_llums) == 0)

{

    Serial.println("Llums Saló OFF");

    digitalWrite(sa_llumsr, HIGH);

}

// Saló Alarma

if (strcmp(state, "ON") == 0 && strcmp(topic_s, sa_alarma) == 0)

{

    Serial.println("Alarma Saló ON");

    digitalWrite(sa_alarmar, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, sa_alarma) == 0)

{

    Serial.println("Alarma Saló OFF");

    digitalWrite(sa_alarmar, HIGH);

}

// Dormitori llums
```

```
if (strcmp(state, "ON") == 0 && strcmp(topic_s, do_llums) == 0)

{

    Serial.println("Llums Dormitori ON");

    digitalWrite(do_llumsr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, do_llums) == 0)

{

    Serial.println("Llums Dormitori OFF");

    digitalWrite(do_llumsr, HIGH);

}

// Oficina llums

if (strcmp(state, "ON") == 0 && strcmp(topic_s, of_llums) == 0)

{

    Serial.println("Llums Oficina ON");

    digitalWrite(of_llumsr, LOW);

}

if (strcmp(state, "OFF") == 0 && strcmp(topic_s, of_llums) == 0)

{

    Serial.println("Alarma Oficina OFF");

    digitalWrite(of_llumsr, HIGH);

}

}
```

### A.3. OpenHAB

Aquest seria el codi que es troba dins de la carpeta Rules de OpenHAB i és el que s'encarrega del funcionament de les automatitzacions de la vivenda, hi han altres arxius necessaris per correcte funcionament de OpenHAB que es troben a la carpeta Programa amb el CD adjunt.

#### A.3.1. Rules

```
////////* Abdelaziz Essami - TFG GEEIA - Domotització d'una vivenda mitjançant una
xarxa mallada mesh *////////

//////////////////////////////////// Declaració de les regles automatitzades
////////////////////////////////////

//////////////////////////////////// Declaració de les variables //////////////////////////////////

var Timer temporitzador0 = null

var Timer temporitzador1 = null

var Timer temporitzador2 = null

var Timer temporitzador3 = null

var Timer temporitzador4 = null

var Timer temporitzador5 = null

val telegramAction = getActions("telegram","telegram:telegramBot:domo789bot")

//////////////////////////////////// Desactivació de la alarma en l'arrencada del controlador local
////////////////////////////////////

rule "1-Reset Alarma CO" // aquesta regla s'ha creat per casos on l'usuari

when // no pugui apagar l'alarma desde el mòvil

    System started // Quan arrenca el sistema

then

    PB_Salo_Alarma.sendCommand(OFF) // Apaguem Alarma
```

```
end
```

```
//////////////////////////////////// Regles de funcionament de l'Alarma de CO  
////////////////////////////////////
```

```
rule "1-Monoxid Carboni Cuina"
```

```
when
```

```
    Item PB_Cuina_CO changed // Llegim el sensor de CO cada cop que canvia de valor
```

```
then
```

```
    if ((PB_Cuina_CO.state >= 50) && (PB_Cuina_CO.state < 100)) // Fem un avís  
    repetitiu si es troba en aquest rang
```

```
    {
```

```
        telegramAction.sendTelegram("Atenció! El nivell de CO és de %s ppm",  
PB_Cuina_CO.state.toString)
```

```
        temporitzador0 = createTimer(now.plusHours(1), [ |
```

```
            if ((PB_Cuina_CO.state >= 50) && (PB_Cuina_CO.state < 100)) // Si després  
            de 1 hora seguim en el mateix rang, activem la alarma
```

```
            {
```

```
                PB_Salo_Alarma.sendCommand(ON)
```

```
                temporitzador0 = null
```

```
            }
```

```
        temporitzador0 = null])
```

```
    }
```

```
    else if ((PB_Cuina_CO.state >= 100) && (PB_Cuina_CO.state < 300)) // Fem un avís  
    repetitiu si es troba en aquest rang
```

```
    {
```

```
        telegramAction.sendTelegram("Atenció! El nivell de CO és de %s ppm",  
PB_Cuina_CO.state.toString)
```

```
temporitzador0 = createTimer(now.plusMinutes(10), [ |

    if ((PB_Cuina_CO.state >= 100) && (PB_Cuina_CO.state < 300)) // Si
després de 10 minuts seguim en el mateix rang, activem la alarma

    {

        PB_Salo_Alarma.sendCommand(ON)

        temporitzador0 = null

    }

    temporitzador0 = null])

}

else if (PB_Cuina_CO.state >= 300) // Fem un avís si es troba per sobre d'aquest
valor

{

    telegramAction.sendTelegram("Atenció! El nivell de CO és de %s ppm",
PB_Cuina_CO.state.toString)

    PB_Salo_Alarma.sendCommand(ON) // Activem la alarma directament

}

end

rule "1-Notificació Alarma CO" // Notificació pel telegram indicant que ha saltat
l'alarma

when

    Item PB_Salo_Alarma changed to ON

then

    telegramAction.sendTelegramQuery("Ha saltat la alarma de CO, la vols apagar?",
"commandco", "Si", "No")

end
```

```
rule "1-Apagar Alarma" // El client li haurà aparegut el missatge anterior i tindrà
la possibilitat de apagar l'alarma
```

```
when
```

```
    Item telegramLastMessageReplyId received update commandco
```

```
then
```

```
    if (telegramLastMessage.state.toString == "Si")
```

```
    {
```

```
        PB_Salo_Alarma.sendCommand(OFF)
```

```
telegramAction.sendTelegramAnswer(telegramLastMessageReplyId.state.toString, "Ok,
Alarma apagada.")
```

```
    }
```

```
    else if (telegramLastMessage.state.toString == "No")
```

```
    {
```

```
telegramAction.sendTelegramAnswer(telegramLastMessageReplyId.state.toString, "Ok, No
faré res.")
```

```
    }
```

```
end
```

```
//////////////////////////////////// Activació dels electrodomèstics en hores vall
////////////////////////////////////
```

```
rule "2-Hores Vall Rentadora"
```

```
when
```

```
    Time cron "0 0 6 ? * MON,TUE,WED,THU,FRI *" // Un cop siguin les 06:00h i dia
d'entre setmana, s'activa aquesta regla
```

```
then
```

```
var comprovacio1 = ST_Soterrani_Rentadora.changedSince(now.minusHours(21),
"influxdb") // Comprovem que no s'hagi encès la rentadora durant el dia

if ((ST_Soterrani_Rentadora.state == OFF) && (comprovacio1==false)) //
accedint a la base de dades influxDB

{

    telegramAction.sendTelegram("S'ha engegat la rentadora a las 06:00h") // Si
no s'ha engegat durant el dia, ho farà a la hora indicada durant

    ST_Soterrani_Rentadora.sendCommand(ON) // un
cert temps i es tornarà a apagar després que acabi de comptar el temporitzador.

    temporitzador1 = createTimer(now.plusHours(2),
[ST_Soterrani_Rentadora.sendCommand(OFF) temporitzador1 = null])

}

else if ((ST_Soterrani_Rentadora.state == ON) || (comprovacio1==true))

{

    telegramAction.sendTelegram("La rentadora es troba actualment engegada o ja
s'ha engegat en les últimes 24h")

}

end

rule "2-Hores Vall Caldera"

when

    Time cron "0 0 6 ? * MON,TUE,WED,THU,FRI *" // Un cop siguin les 06:00h i dia
d'entre setmana, s'activa aquesta regla

then

    var comprovacio2 = ST_Soterrani_Caldera.changedSince(now.minusHours(21),
"influxdb") // Comprovem que no s'hagi encès la caldera durant el dia

    if ((ST_Soterrani_Caldera.state == OFF) && (comprovacio2==false)) //
accedint a la base de dades influxDB

    {
```

```
        telegramAction.sendTelegram("S'ha engegat la caldera a las 06:00h") // Si no
s'ha engegat durant el dia, ho farà a la hora indicada durant

        ST_Soterrani_Caldera.sendCommand(ON) // un
cert temps i es tornarà a apagar després que acabi de comptar el temporitzador.

        temporitzador2 = createTimer(now.plusHours(2),
[ST_Soterrani_Caldera.sendCommand(OFF) temporitzador2 = null])

    }

    else if ((ST_Soterrani_Caldera.state == ON) || (comprovacio2==true))

    {

        telegramAction.sendTelegram("La caldera es troba actualment engegada o ja
s'ha engegat en les últimes 24h")

    }

end

////////// Activació/Desactivació del mode nit //////////

rule "3-Alba"

when

    Channel 'astro:sun:home:rise#event' triggered START // Mitjançant el canal de
sortida del sol del binding astro

then

    ModeNit.postUpdate(OFF)

    telegramAction.sendTelegram("Mode Nit OFF")

end

rule "3-Capvespre"

when

    Channel 'astro:sun:home:set#event' triggered START // Mitjançant el canal de
posta del sol del binding astro
```



```
then                                                    // Activem el mode nit

    ModeNit.postUpdate(ON)

    telegramAction.sendTelegram("Mode Nit ON")

end

//////////////////////////////////// Activació de les llums segons la il·luminació i
presència en mode nit //////////////////////////////////////

rule "3-Llums Soterrani ON intel·ligents"

when

    Item ST_Soterrani_Pir changed to 1    // Per cada cop que es detecta moviment, es
procedeix a encendre les llums

then

    // segons el nivell de il·luminació, el
soterrani no haurà de complir la condició del mode nit

    if ((ST_Soterrani_Lux.state <= 200) && (ST_Soterrani_Llums.state == OFF)) // ja
que no tindrà tanta llum natural com les altres estances

        {                                                    // durant
el matí i per tant estarà més fosc

            ST_Soterrani_Llums.sendCommand(ON)

            telegramAction.sendTelegram("Mode Nit: Llums Soterrani ON") // Enviem un avís
al mòbil perquè quedi constància

        }

end

rule "3-Llums Saló ON intel·ligents"

when

    Item PB_Salo_Pir changed to 1    // Per cada cop que es detecta moviment, es
procedeix a encendre les llums

then

    // segons el nivell de il·luminació i si ens trobem
dins del mode nit
```

```
    if ((PB_Salo_Lux.state <= 200) && (ModeNit.state == ON) && (PB_Salo_Llums.state == OFF))

    {

        PB_Salo_Llums.sendCommand(ON)

        telegramAction.sendTelegram("Mode Nit: Llums Saló ON") // Enviem un avís al mòvil perquè quedi constància

    }

end

rule "3-Llums Dormitori ON intel·ligents"

when

    Item PP_Dormitori_Pir changed to 1 // Per cada cop que es detecta moviment, es procedeix a encendre les llums

then

    // segons el nivell de il·luminació i si ens trobem dins del mode nit

    if ((PP_Dormitori_Lux.state <= 200) && (ModeNit.state == ON) && (PP_Dormitori_Llums.state == OFF))

    {

        PP_Dormitori_Llums.sendCommand(ON)

        telegramAction.sendTelegram("Mode Nit: Llums Dormitori ON") // Enviem un avís al mòvil perquè quedi constància

    }

end

rule "3-Llums Soterrani OFF intel·ligents"

when

    Item ST_Soterrani_Pir received update 1 // Mentre hi hagi moviment, haurem de mantenir la llum encesa

then
```

```
    if ((temporitzador3 === null) && (ST_Soterrani_Llums.state == ON)) // En el
primer cicle s'entraria al if, compta el

    {
        // temporitzador
        i si deixa d'existir moviment s'apaguen les llums

        temporitzador3 = createTimer(now.plusSeconds(10), [ |

            ST_Soterrani_Llums.sendCommand(OFF)

            telegramAction.sendTelegram("Mode Nit: Llums Soterrani OFF")

            temporitzador3 = null

        ])

    }

    else

    {
        // Però, si mentres compta
        el temporitz. es torna a detectar moviment

        temporitzador3.reschedule(now.plusSeconds(10)) // llavors es
reinicialitza el temporitz., això ho farà cada cop que detecta moviment

    }

end

rule "3-Llums Saló OFF intel·ligents"

when

    Item PB_Salo_Pir received update 1 // Mentres hi
hagi moviment, haurem de mantenir la llum encesa

then

    if ((temporitzador4 === null) && (PB_Salo_Llums.state == ON)) // En el primer
cicle s'entraria al if, compta el

    {
        // temporitzador i
        si deixa d'existir moviment s'apaguen les llums

        temporitzador4 = createTimer(now.plusSeconds(10), [ |
```

```
        PB_Salo_Llums.sendCommand(OFF)

        telegramAction.sendTelegram("Mode Nit: Llums Saló OFF")

        temporitzador4 = null

    ])

}

else

{
    // Però, si mentres compta
    el temporitz. es torna a detectar moviment

    temporitzador4.reschedule(now.plusSeconds(10)) // llavors es
    reinicialitza el temporitz., això ho farà cada cop que detecta moviment

}

end

rule "3-Llums Dormitori OFF intel·ligents"

when

    Item PP_Dormitori_Pir received update 1 // Mentres hi hagi moviment,
    haurem de mantenir la llum encesa

then

    if ((temporitzador5 === null) && (PP_Dormitori_Llums.state == ON)) // En el
    primer cicle s'entraria al if, compta el

    {
        // temporitzador
        i si deixa d'existir moviment s'apaguen les llums

        temporitzador5 = createTimer(now.plusSeconds(10), [ |

            PP_Dormitori_Llums.sendCommand(OFF)

            telegramAction.sendTelegram("Mode Nit: Llums Dormitori OFF")

            temporitzador5 = null
```

```
    1)
}

else

{
    compta el temporitz. es torna a detectar moviment // Però, si mentres

    temporitzador5.reschedule(now.plusSeconds(10)) // llavors es
    reinicialitza el temporitz., això ho farà cada cop que detecta moviment

}

end
```

## B. CÀLCULS

### B.1. Càlcul de les bateries

En aquest apartat es farà el càlcul de la duració de les piles depenent en quin mode es troben els mòduls XBee. S'han triat unes piles AA per aquests càlculs.

Duració de la bateria en mode transmissió:

$$D_1 = \frac{C}{I} * 0,7 = \frac{2850 \text{ mAh}}{40 \text{ mA}} * 0,7 = 49,87 \text{ hores} \quad (\text{Eq.1})$$

Duració de la bateria en mode inactivitat:

$$D_2 = \frac{C}{I} * 0,7 = \frac{2850 \text{ mAh}}{15 \text{ mA}} * 0,7 = 133 \text{ hores} \quad (\text{Eq.2})$$

En el cas del mode hibernació, sabem que els mòduls hauran d'estar 5 minuts hibernant per enviar dades durant uns 2 segons:

$$\text{Cicle de treball mode transmissió (\%)} = \frac{2 \text{ s}}{302 \text{ s}} * 100 = 0,66 \% \quad (\text{Eq.3})$$

$$D_3 = \frac{D_1}{\text{Cicle de treball mode transmissió}} = \frac{49,87 \text{ h}}{0,0066} = 7556 \text{ hores} \quad (\text{Eq.4})$$

### B.2. Càlcul dels sensors

En aquest apartat justificarem els càlculs del condicionaments dels sensors utilitzats en el projecte, el sensor de presència PIR i la LDR no necessiten cap condicionament. El PIR només proporciona una sortida digital i la LDR només s'utilitza per distingir entre escenaris clars o foscos.

## B.2.1. Càlcul del sensor de monòxid de carboni

El fabricant del sensor de monòxid de carboni MQ-7 ens proporciona una gràfica logarítmica de les corbes de funcionament del sensor depenent de la concentració de CO existent en l'aire. També ens proporciona una fórmula per poder obtenir la equació de la recta que ens interessa, la podem veure a continuació juntament amb la gràfica.

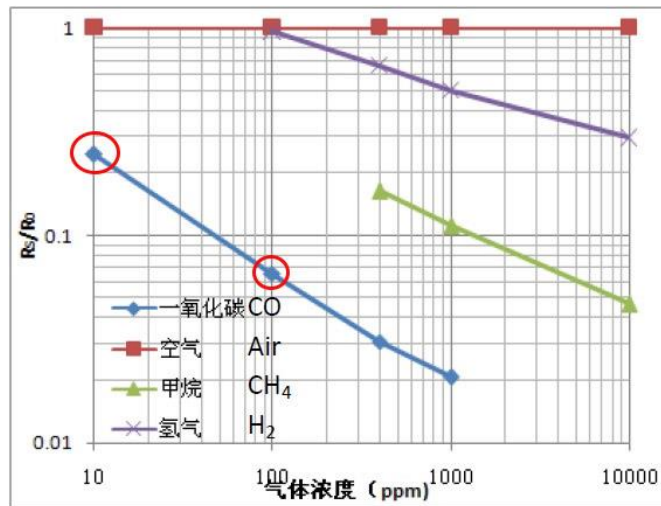


Figura 34. Característiques de la sensibilitat del MQ-7

D'aquesta gràfica s'han extret aquests dos punts per obtenir el valor l'equació de la recta blava, i s'han inserit en la següent equació.

$$F(x) = F_0 * \left(\frac{x}{x_0}\right)^{\frac{\log(F_1/F_0)}{\log(x_1/x_0)}} \quad (\text{Eq.5})$$

$$F(x) = 0,065 * \left(\frac{x}{100}\right)^{\frac{\log(0,25/0,065)}{\log(10/100)}} = 0,0065x^{-0,585} \quad (\text{Eq.6})$$

Sabent que el valor de  $R_s$  dependrà de la lectura de la concentració de CO i d'una resistència fixe  $R_2$ , i el valor de  $R_0$  dependrà de  $R_s$ .

$$\frac{R_s}{R_0} = 0,00065 * \text{ppm}^{-0,585} \quad (\text{Eq.7})$$

$$\text{ppm} = \left( 1538,46 * \frac{R_s}{R_0} \right)^{-1,709} \quad (\text{Eq.8})$$

En l'equació anterior obtenim el valor de la concentració de CO en l'aire que segueix el sensor MQ-7, la relació entre les resistències  $R_s$  i  $R_0$  s'especifica en el datasheet del sensor i s'exposa a continuació.

$$R_s = \left( \frac{1,2 \text{ V} * R_2}{\text{Lectura}} \right) - R_2 \quad (\text{Eq.9})$$

$$R_0 = \frac{1}{R_s} \quad (\text{Eq.10})$$

Pel que respecte al consum màxim del sensor durant l'escalfament de la posada en marxa, aplicarem la següent equació.

$$I = \frac{V}{R} = \frac{5 \text{ V}}{33\Omega} = 151 \text{ mA} \quad (\text{Eq.11})$$

### B.2.2. Càlcul del sensor de humitat

S'ha accedit al datasheet del sensor per poder obtenir l'equació de la recta de la sortida de voltatge que proporciona el fabricant, també ens especifica que aquesta equació s'ha calculat amb una temperatura ambient de 25 °C i només proporciona la humitat relativa, així que no donarà valors del tot exactes però compleix la seva funció informativa dins el projecte.

Primerament haurem de convertir la lectura del XBee a volts, i seguidament aplicar l'equació de sortida del fabricant.

$$\text{Volts} = \text{Lectura} * \frac{1,2 \text{ V}}{1024} \quad (\text{Eq.12})$$

$$\text{Humitat relativa} = \text{Volts} * \left[ \left( -\frac{0,1515}{0,00636} \right) + \left( \frac{1}{1,2 * 0,00636} \right) \right] \quad (\text{Eq.13})$$



### B.2.3. Càlcul del sensor de temperatura

En aquest cas el fabricant ens proporciona la sensibilitat del sensor i tenim un offset de 50 °C degut a que el sensor pot mesurar valors negatius també. Primerament haurem de convertir la lectura del XBee a mili-volts i seguidament aplicar la conversió a graus centígrads.

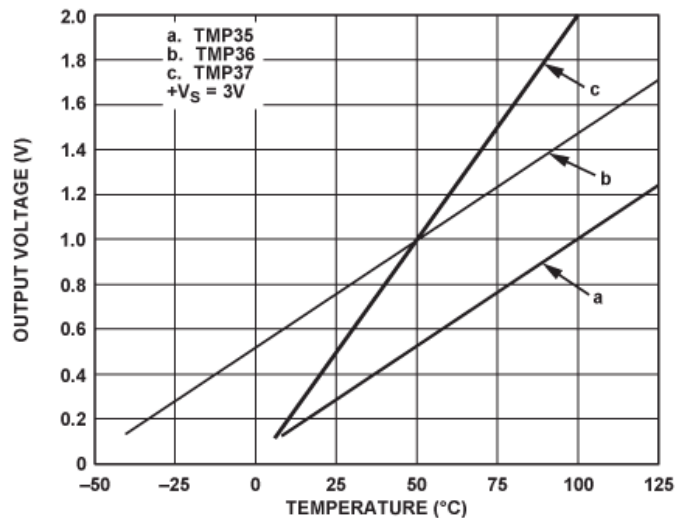


Figura 35. Voltatge de sortida respecte la temperatura mesurada

$$\text{Volts} = \text{Lectura} * \frac{1200 \text{ mV}}{1024} \quad (\text{Eq.14})$$

$$\text{Temperatura} = \frac{\text{Volts}}{10 \text{ mV/}^\circ\text{C}} - 50 \text{ }^\circ\text{C} \quad (\text{Eq.15})$$

### B.3. Càlcul dels divisors de tensió

Els mòduls XBee només poden llegir fins a 1,2V en les entrades analògiques, tots els sensors que la seva lectura pot ser greument afectada aniran equipats amb un divisor de tensió per escalar el voltatge de la seva sortida a un màxim de 1,2V.

El divisor de tensió baixa la impedància d'entrada a les entrades analògiques i podria afectar a la precisió de la lectura dels sensors, però sabent que la impedància d'entrada del mòdul XBee és de l'ordre de 10 MΩ ens permet prescindir de l'ús d'un seguidor de voltatge o altrament dit buffer, degut a que la precisió dels sensors no es veurà molt afectada.

Pels mòduls del soterrani, saló, dormitori i oficina:

$$V_{\text{out}} = V_{\text{in}} * \frac{R_2}{R_1 + R_2} = 3 \text{ V} * \frac{5,6 \text{ K}\Omega}{5,6 \text{ K}\Omega + 8,2 \text{ K}\Omega} = 1,21 \text{ V} \quad (\text{Eq.16})$$

Pel mòdul de la cuina, concretament el sensor de humitat:

$$V_{\text{out}} = V_{\text{in}} * \frac{R_2}{R_1 + R_2} = 3,3 \text{ V} * \frac{4,7 \text{ K}\Omega}{4,7 \text{ K}\Omega + 8,2 \text{ K}\Omega} = 1,20 \text{ V} \quad (\text{Eq.17})$$

Pel mòdul de la cuina, concretament el sensor de CO:

$$V_{\text{out}} = V_{\text{in}} * \frac{R_2}{R_1 + R_2} = 5 \text{ V} * \frac{1,8 \text{ K}\Omega}{1,8 \text{ K}\Omega + 5,6 \text{ K}\Omega} = 1,21 \text{ V} \quad (\text{Eq.18})$$

#### B.4. Càlcul dels relés

Sabent que necessitem superar la intensitat d'excitació del relé per commutar i que ha de treballar en saturació i tall, busquem la intensitat de base necessària per saturar i la intensitat d'entrada del optoacoblador mitjançant la relació de transferència d'intensitat CTR.

$$I_{\text{relé}} = \frac{V}{R_{\text{bobina}}} = \frac{5 \text{ V}}{55 \Omega} = 90,9 \text{ mA} \quad (\text{Eq.19})$$

$$I_C = I_{\text{relé}} = 90,9 \text{ mA} \quad (\text{Eq.20})$$

Aquesta serà la intensitat de base necessària.

$$I_B = \frac{I_C}{\beta} = \frac{90,9 \text{ mA}}{110} = 0,81 \text{ mA} \quad (\text{Eq.21})$$

$$\beta * I_B \gg I_C \quad (\text{Eq.22})$$

Aquest serà el valor de la resistència a la base del transistor.

$$R_B = \frac{V_{BE}}{I_B} = \frac{0,9 \text{ V}}{0,81 \text{ mA}} = 1108,64 \ \Omega \quad (\text{Eq.23})$$

$$\text{CTR (50\%)} = \frac{I_B}{I_F} \quad (\text{Eq.24})$$

Aquesta serà la intensitat d'entrada del optocobrador segons la relació de transferència d'intensitat CTR.

$$I_F = \frac{I_B}{0,5} = \frac{0,81 \text{ mA}}{0,5} = 1,62 \text{ mA} \quad (\text{Eq.25})$$

Aquesta serà la resistència a l'entrada del optocobrador, s'ha tingut en compte la caiguda de voltatge del LED intern del optocobrador i del LED extern indicador.

$$R_O = \frac{V_{VCC} - V_F - V_{LED}}{I_F} = \frac{5 - 1,4 - 2 \text{ V}}{1,62 \text{ mA}} = 985,46 \ \Omega \quad (\text{Eq.26})$$

Ara comprovem que en saturació la intensitat de base pot excitar el relé.

$$I_{F\text{sat}} = \frac{V_{VCC} - V_F - V_{LED} + V_{\text{senyal}}}{R_O} = \frac{5 - 1,4 - 1,2 + 3,3 \text{ V}}{1478 \ \Omega} = 3,85 \text{ mA} \quad (\text{Eq.27})$$

Apliquem la relació de transformació CTR.

$$I_{B\text{sat}} = I_{F\text{sat}} * 0,5 = 1,92 \text{ mA} \quad (\text{Eq.28})$$

D'aquesta forma es comprova la intensitat de base és clarament superior a la del relé.

$$\beta * I_B \gg I_C \quad \text{on} \quad 110 * 1,92 \text{ mA} \gg 90,9 \text{ mA} \quad \text{on} \quad 211,2 \text{ mA} \gg 90,9 \text{ mA} \quad (\text{Eq.29})$$

## **C. INSTAL·LACIÓ ELÈCTRICA**

En aquest capítol es realitzarà la justificació de la instal·lació elèctrica complint amb el Reglament Electrotècnic de Baixa Tensió i les seves Instruccions Tècniques Complementàries, d'acord amb el Real Decret 842/2002.

### **C.1. Generalitats**

Es projecta la instal·lació elèctrica d'il·luminació i electrodomèstics d'una domotització d'una vivenda de tres plantes amb soterrani, planta baixa i primera planta. Les dues plantes superiors tenen una superfície de 140 m<sup>2</sup> cadascuna, mentre que el soterrani només té una superfície de 90 m<sup>2</sup>. La parcel·la és de 16 metres de llarg per 10 metres d'ample. Les estàncies que interessa domotitzar són soterrani, cuina, saló, dormitori i oficina. Es considera que la instal·lació de la vivenda és d'electrificació bàsica degut a que només es vol domotitzar unes estàncies concretes i no la totalitat de la qual, i que la suma de totes les potències no supera els 9,2 kW.

Serà una ampliació a la instal·lació existent, degut a que només ens interessa dissenyar l'ampliació necessària per aquest projecte, no s'entrarà en detall en les característiques de la instal·lació existent ni en aquesta memòria ni en els plànols.

Aquesta instal·lació tindrà límit de potència de 50kW ja que segons la ITC-BT-04 es troba al grup F de vivendes unifamiliars.

El càlcul de la instal·lació consisteix d'una banda en l'elecció i dimensionament de les lluminàries necessàries per a unes correctes condicions de funcionament, per un altre en la previsió de càrregues totals, i finalment el càlcul de les seccions i proteccions dels conductors que ens alimentaran la vivenda, així aconseguirem realitzar una instal·lació segura dins dels paràmetres marcats per els reglaments.

### **C.2. Previsió de càrregues**

Utilitzarem el tipus de lluminària més adient segons la zona que vulguem il·luminar, seguint les recomanacions que determinen quina il·luminació mínima hi ha d'haver en cada zona. D'aquesta manera buscarem un tipus de lluminària que ens permeti estalviar en costos energètics i que ens proporcioni un grau d'il·luminació adequat.

Aplicant els factors de simultaneïtat que hem cregut adients consultant la ITC-BT-25 com a orientació, ens quedaria la següent taula:

Zona	Tipus de lluminària	Quantitat	Potència (W)	Fs	P. Total (W)
L4-Soterrani	Llum LED	4,00	24,00	1,00	96,00
L3-Saló	Llum LED	2,00	24,00	1,00	48,00
L1-Cuina	Llum LED	1,00	24,00	1,00	24,00
L5-Dormitori	Llum LED	1,00	24,00	1,00	24,00
L6-Oficina	Llum LED	1,00	24,00	1,00	24,00
				<b>Total</b>	<b>216,00</b>

Taula 16. Previsió de càrrega d'enllumenat

Els endolls a 230V es consideren a una potència de 3.450W segons la ITC-BT-25, com són també els electrodomèstics com la rentadora o la caldera, en el cas dels electrodomèstics hem optat per la seva potència nominal. Els endolls aniran equipats amb clavilla de posada a terra, la seva col·locació serà sobre les parets a una altura mínima de 30 cm i aquells que es trobin a la cuina a una altura superior a 50 cm.

Aplicant els factors de simultaneïtat i utilització que hem cregut adients ens queda la següent taula:

Línia	Descripció	Quantitat	Potència (W)	Fs	Fu	P. Total (W)
L2	Alarma CO	1,00	23,00	1,00	1,00	23,00
L7	Endolls 230V P.Baixa	3,00	3.450,00	0,20	0,20	414,00
L8	Endolls 230V Soterrani	1,00	3.450,00	1,00	0,20	690,00
L9	Endolls 230V P.Primer	1,00	3.450,00	1,00	0,20	690,00
L10	Rentadora	1,00	2.200,00	1,00	0,50	1.100,00
L11	Caldera	1,00	2.000,00	1,00	0,50	1.000,00
L12	Endolls 230V cuina	1,00	3.450,00	1,00	0,20	690,00
					<b>Total</b>	<b>4.607,00</b>

Taula 17. Previsió de càrrega d'altres línies

Es mostra a continuació el resum de la previsió de càrregues a la següent taula.

Element	Potència (W)
Il·luminació	216,00
Potència	4.607,00
Instal·lació existent	2.000,00
<b>Total</b>	<b>6.823,00</b>

Taula 18. Resum previsió de càrregues

Es pot veure com s'ha tingut en compte la potència que consumeixen les càrregues de la instal·lació ja existent també, la suma d'aquesta i de la instal·lació domòtica a dissenyat ens dona la potència total que haurem de contractar. Aplicant els corresponents factors de simultaneïtat i utilització haurem de contractar un potència normalitzada de 6,9 kW.

### **C.3. Condicions de subministrament**

La tensió de subministrament és a 230 V i monofàsica. Estarà formada per una fase i un neutre.

La freqüència de la línia és a 50Hz.

### **C.4. Empresa subministradora**

La companyia elèctrica Iberdrola serà l'encarregada de proporcionar el subministrament elèctric al nostre local ja que disposa d'unes tarifes que s'adapten a les nostres necessitats.

L'oficina d'atenció al client es troba a la carretera de Barcelona N° 81, de Girona amb telèfon gratuït per a empreses 900 400 408.

### **C.5. Escomesa i caixa general de protecció**

L'alimentació de la xarxa ens arriba subterràniament.

El quadre de control i comandament general s'ha d'instal·lar al costat de la porta principal i el mes a prop possible de l'escomesa i la derivació individual. En el nostre cas s'ha decidit per instal·lar-ho al garatge al costat de la porta principal.

Per al càlcul de l'escomesa o derivació individual seguirem la ITC-BT-15 que ens fa referència a les instal·lacions d'enllaç. La caiguda de tensió màxima no podrà ser superior al 1,5% i els cables enterrats hauran de tenir les qualitats mínimes que ens indica la norma UNE-EN 50086-2-4.

Tenint en compte que la màxima caiguda de tensió segons RBT-ITC-15 és 1,5% en derivacions individuals quan no existeix línia repartidora d'alimentació i que aquesta línia té una llargada de 10 m i serà un únic cable, sense empalmes en tot el seu recorregut i la seva

secció serà uniforme, els cables seran no propagadors d'incendis i amb emissió de fums i opacitat reduïda tal i com s'indica a la ITC-BT-28, els càlculs corresponents són els següents:

$$e = V \cdot \frac{1,5}{100} = 230 \cdot \frac{1,5}{100} = 3,45 \text{ V} \quad (\text{Eq.30})$$

$$S_{ct} = \frac{2 \cdot P \cdot L}{\rho \cdot e \cdot U} = \frac{2 \cdot 6900 \cdot 10}{58 \cdot 3,45 \cdot 230} = 2,99 \text{ mm}^2 \quad (\text{Eq.31})$$

$$I = \frac{P}{V} = \frac{6900}{230} = 30 \text{ A} \quad (\text{Eq.32})$$

Quan fem el càlcul de secció ens surt 2,99 mm<sup>2</sup>, però si mirem la intensitat veiem que segons la taula 5 de la ITC-BT-07 per una intensitat per fase de 30 A ens obliga a instal·lar cables de 6 mm<sup>2</sup>. El cable ha de discórrer per un tub de 32 mm<sup>2</sup>.

$$e(\%) = \frac{2 \cdot 6900 \cdot 10}{58 \cdot 6 \cdot 230} \cdot \frac{100}{230} = 0,75 \% \quad (\text{Eq.33})$$

Tenim una caiguda de tensió inferior al 1,5% màxim establert pel REBT.

El cable de 10 metres de llargada, i enterrat a una profunditat de 0,7 metres, des de la línia de repartiment fins a la caixa general de protecció, serà de coure, unipolar, amb recobriments de XLPE, polietilè reticulat, d'una secció per fase i neutre de 6 mm<sup>2</sup> amb aquestes condicions la caiguda de tensió és del 0,75 %.

Dins la CGP (caixa general de protecció) situada sobre la façana est de la nau a una alçada d'entre 0,7 i 1,8 metres, accessible mitjançant una porta precintada i accessible a peu permanentment. Hi tindrem els fusibles de protecció A, les bases dels fusibles seran bases unipolars tancades (BUC) amb capacitat d'extinció de l'arc.

### C.6. Línia repartidora

La línia repartidora és aquella que enllaça la caixa general de protecció (CGP) amb la centralització de comptadors, en el nostre cas com que només disposem d'un sol usuari la línia repartidora queda eliminada ja que la CGP i el comptador d'energia es troben dins de la mateixa cavitat a l'entrada de la nau.

### C.7. Conjunt de protecció i mesura

La CPM estarà col·locada en un armari destinat exclusivament a aquesta tasca que reuneix els requisits esmentats a la ITC-BT-16 com també estarà constituïda per material aïllant de classe tèrmica A, com a mínim, segons norma UNE 21305. Anirà dins d'un mòdul precintat i encastat dins la paret situat entre 0,7 i 1,8 metres d'altura. Tindrà un grau de protecció mínim contra pols i aigua IP43 i contra impactes IK09. La caixa ha de permetre la lectura directe del comptador i tenir ventilació necessària per evitar condensacions.

El conjunt serà el CPM1-MF2-25A amb base BUC-00 i estarà constituït per un envolupant fabricat en polièster premsat en calent i reforçat amb fibra de vidre, color gris RAL 7035 i doble aïllament. Les bases de fusibles son seccionables amb càrregues de grandària 00 fins a 160A.

Pel càlcul dels fusibles s'ha seguit la ITC-BT-22 i s'ha triat uns fusibles amb una intensitat nominal de 35 A. A continuació veiem la comprovació de les dues condicions que han de complir per reglament sabent que el cable triat suporta una intensitat de 72 A en trifàsic.

$$I_{\text{vivenda}} \leq I_n \leq I_{\text{admissible cable escomessa}} \quad (\text{Eq.34})$$

$$I_{\text{admissible cable escomessa}} = \frac{72 \text{ A}}{\sqrt{3}} = 41,56 \text{ A} \quad (\text{Eq.35})$$

$$30 \text{ A} \leq 35 \text{ A} \leq 41,56 \text{ A} \quad (\text{Eq.36})$$

Veiem que complim la primera condició on la intensitat nominal dels fusibles es troba dins del rang correcte de funcionament.

$$I_f \leq 1,45 * I_{\text{admissible cable escomessa}} \quad (\text{Eq.37})$$

$$I_n * 1,60 \leq 1,45 * I_{\text{admissible cable escomessa}} \quad (\text{Eq.38})$$

$$56 \text{ A} \leq 60,26 \text{ A} \quad (\text{Eq.39})$$

També complim la segona condició on la intensitat a la qual es fon el fusible  $I_f$  ha de ser més petita que la Intensitat que admet el cable multiplicada per un factor de seguretat.



### C.8. Derivació individual

La derivació individual la calcularem seguint la ITC-BT-15, la derivació individual parteix des del ICP fins al quadre de protecció i control de la nau.

Les intensitats màximes admissibles per a conductors de coure en muntatge dels tipus indicats, seran els indicats en la taula 1 de la ITC-BT-19 per conductors del tipus RZ1-K, amb coberta i aïllament de XLPE, i/o EPR, amb una tensió d'aïllament assignada de 0,6/1kV no propagadors d'incendi i amb emissió de fums d'opacitat reduïda, d'acord a les UNE 21.123-4,-5, o UNE 21.002.

Els conductors recorren fins al quadre de protecció i control sota tub superficial protector d'acer inoxidable, aquest ha de permetre ampliar la secció dels conductors en un 100%.

Per a fer el càlcul de la derivació individual la caiguda de tensió no pot ser superior al 1,5%, longitud 10 metres i cables unipolars XLPE de coure.

$$e = V \cdot \frac{1,5}{100} = 230 \cdot \frac{1,5}{100} = 3,45 \text{ V} \quad (\text{Eq.40})$$

$$S_{ct} = \frac{2 \cdot P \cdot L}{\rho \cdot e \cdot U} = \frac{2 \cdot 6900 \cdot 10}{58 \cdot 3,45 \cdot 230} = 2,99 \text{ mm}^2 \quad (\text{Eq.41})$$

$$I = \frac{P}{V} = \frac{6900}{230} = 30 \text{ A} \quad (\text{Eq.42})$$

La secció ens surt de 2,99 mm<sup>2</sup>, segons la taula 1 de la ITC-BT-19 per una intensitat per fase de 30 A ens obliga a instal·lar cables de 6 mm<sup>2</sup> per al tipus d'instal·lació B.

$$e(\%) = \frac{2 \cdot 6900 \cdot 10}{58 \cdot 6 \cdot 230} \cdot \frac{100}{230} = 0,75 \% \quad (\text{Eq.43})$$

Per una longitud de 10 metres, una intensitat per fase de 30 A i el tipus d'instal·lació segons la ITC-BT-19, la secció del cable serà de 6 mm<sup>2</sup> per cada fase i neutre, amb aquesta secció la caiguda de tensió màxima serà del 0,75 %. Així doncs estem dins les especificacions del reglament.

### **C.9. Interruptor control de potència i interruptor general automàtic**

L'interruptor de control de potència (ICP) desconnecta tota la instal·lació quan la potència que consumim és superior a la potència contractada, és a dir dispara per sobrecàrrega. Protegeix la instal·lació de la companyia enfront la nostre.

S'ha col·locat dins de quadre de protecció i control immediatament abans de la instal·lació interior, en un compartiment independent i precintat. L'interruptor és bipolar, té un poder de tall de 6 kA i la intensitat de tall tèrmic és de 30 A.

L'interruptor general automàtic és un dispositiu que funciona com un interruptor de tipus magneto tèrmic i serà l'encarregat de protegir enfront de sobrecàrregues i curtcircuits a la instal·lació interior de l'habitatge al complet. Aquest es troba després de l'ICP i abans de la instal·lació interior. Serà de 32 A i té un poder de tall de 6 kA.

### **C.10. Quadre general de protecció i comandament**

El quadre general de protecció consta de l'interruptor de control de potència (ICP), l'interruptor general automàtic (IGA) i dos interruptors diferencials de dos pols per al circuit d'endolls monofàsics, les línies d'il·luminació i els electrodomèstics amb una sensibilitat de 30mA.

Pels circuits d'il·luminació, endolls monofàsics i electrodomèstics com que generalment són elements amb doble protecció, no fuguen a terra i a més són elements que són susceptibles al contacte amb les persones, per tant instal·larem una sensibilitat de 30mA en aquets elements, així tindrem una tensió de contacte menor en cas de fuga.

Després del diferencial cada línia de la instal·lació esta protegida per un interruptor magneto tèrmic bipolar segons la tensió d'alimentació amb què treballi. Cadascun d'aquest tindrà un poder de tall mínim de 6000V i 6 kA amb la intensitat de tall per sobrecàrrega que li sigui assignada per tal de protegir la línia a què alimenta enfront les sobrecàrregues.

### **C.11. Distribució general**

En la següent taula es mostra com tenim repartides les línies dins del quadre general de protecció i control, el valor dels magnetotèrmics de cada línia, i la secció per la qual discorren els cables dins la vivenda.

Per a l'elecció dels elements de protecció de sobrecàrrega i curtcircuit (PIA), hem de tenir en compte el següent.

$$I. \text{ Càrrega} < I. \text{ N. protecció} < I. \text{ Admissible Cable} \quad (\text{Eq.44})$$

$$I. \text{ Càrrega} = \text{Intensitat de la càrrega del circuit} \quad (\text{Eq.45})$$

$$I. \text{ N. protecció} = \text{Intensitat nominal de l'element de protecció} \quad (\text{Eq.46})$$

$$I. \text{ Admissible Cable} = \text{Intensitat màxima admissible del cable utilitzat} \quad (\text{Eq.47})$$

En la següent taula podem veure les característiques de totes les línies noves.

Línia	U(V)	fdp	P(W)	I(A)	L(m)	Sct (mm <sup>2</sup> )	S (mm <sup>2</sup> )	PIA(A)	Tub (mm <sup>2</sup> )	e(%)
L1	230	1	24,00	0,10	5,30	0,00	1,50	10	16	0,01
L2	230	1	23,00	0,10	4,70	0,00	1,50	10	16	0,00
L3	230	1	48,00	0,21	11,50	0,01	1,50	10	16	0,02
L4	230	1	96,00	0,42	16,50	0,03	1,50	10	16	0,07
L5	230	1	24,00	0,10	14,30	0,01	1,50	10	16	0,01
L6	230	1	24,00	0,10	16,00	0,01	1,50	10	16	0,02
L7	230	1	414,00	1,80	8,20	0,07	2,50	16	20	0,09
L8	230	1	690,00	3,00	5,30	0,08	2,50	16	20	0,10
L9	230	1	690,00	3,00	5,20	0,08	2,50	16	20	0,09
L10	230	1	1.100,00	4,78	6,50	0,15	4,00	20	20	0,02
L11	230	1	1.000,00	4,35	7,90	0,17	4,00	20	20	0,02
L12	230	1	690,00	3,00	10,40	0,15	2,50	16	20	0,03

Taula 19. Distribució general

Les caigudes de tensió màximes a final de la línia compleixen amb el màxim especificat per el reglament d'un 3% per il·luminació i un 5% en la resta de circuits.

### C.12. Posada a terra

La presa de terra es realitza amb varies piques clavades a un mínim de 0,5 metres de fondària sobre el terreny, aquestes piques seran d'acer-coure clavades al llarg del perímetre del terreny i un anell equipotencial constituït amb conductor de coure nuu que uneix entre si totes les piques.

El born de posada a terra es realitzarà mitjançant una caixa desmuntable amb eines, i es podrà mesurar en aquesta el valor de resistència a terra, en aquesta caixa s'hi connectarà el conductor de terra, els de protecció, els de unió equipotencial principal i els de posada a terra funcional si es necessari.

La secció del conductor de terra estarà d'acord al valor de la taula 2 de la ITC-BT-18, i que s'indica en la UNE 20.460-5-54, apartat 543.1.1 i està realitzada amb coure amb una secció mínima de 6 mm<sup>2</sup>.

Per a la instal·lació de posada a terra s'ha dispost de manera que hem rodejat el perímetre de la vivenda amb 8 piquetes de coure de 2 metres de llargada i 14,2 mm de diàmetre, cada una d'elles enterrades a una profunditat mínima de 50 cm. El nombre de piquetes ha estat calculat prèviament tenint en compte la resistència del terreny i la taula 3 de la ITC-BT-18.

Suposem que la resistivitat del terreny és de 100 ohms per metre segons la naturalesa del terreny:

$$R = \rho / L = 100 / (8 \cdot 2) = 6.25 \Omega \quad (\text{Eq.48})$$

Tenim un conductor de coure enterrat de 10 metres:

$$R = 2 \rho / L = (2 \cdot 100) / 10 = 20 \Omega \quad (\text{Eq.49})$$

Vist que tenim dos tipus de elèctrodes haurem de realitzar el valor de la resistència de cadascun i el conjunt que serà la resistència en paral·lel de les dues.

$$1 / R_{\text{total}} = 1/R_p + 1/R_c = 1/6.25 + 1/20 = 4.76 \Omega \quad (\text{Eq.50})$$

A dia 1 de setembre de 2021 la mesura del terra és de 4,76  $\Omega$ .

### **C.13. Característiques de la instal·lació elèctrica**

La instal·lació elèctrica s'ha d'executar d'acord amb el què indica la norma EN-50281-1-2.

El color de l'aïllament exterior dels conductors serà dels colors corresponents al següent codi que marca la ITC-BT-19: Fase R: Negre; Fase S: Marró; Fase T: Gris; Neutre: Blau; Protecció: Groc-Verd. En el nostre cas només hi haurà fase, neutre i conductor de protecció.

Els conductors seran tots de tensió d'aïllament 0,6/1kV, XLPE, amb resistència a la calor i extinció de flama.

Els tubs per on passaran els cables seran d'acer inoxidable i han de seguir les característiques mínimes que marca la taula 3 de la ITC-BT-29.

El quadre elèctric és d'estructura metàl·lica i obertura frontal, ha estat col·locat a una alçada de 1,5 m sobre el nivell del terra.