



SION: Sistema de Informació i Obtenció de Notificacions

Nom Estudiant

Alfredo Garcia Lerma

Nom Consultor

Jordi Bécares Ferrés

dimarts, 12 gener de 2016



Aquesta obra està subjecta a una llicència de [Reconeixement-NoComercial-SenseObraDerivada 3.0 Espanya de Creative Commons](https://creativecommons.org/licenses/by-nc-nd/3.0/es/)

Copyright © 2016 Alfredo Garcia Lerma.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation; with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts.

A copy of the license is included in the section entitled "GNU Free Documentation License".

© Alfredo Garcia Lerma

Reservats tots els drets. Està prohibit la reproducció total o parcial d'aquesta obra per qualsevol mitjà o procediment, compresos la impressió, la reprografia, el microfilm, el tractament informàtic o qualsevol altre sistema, així com la distribució d'exemplars mitjançant lloguer i préstec, sense l'autorització escrita de l'autor o dels límits que autoritzi la Llei de Propietat Intel·lectual.

FITXA DEL TREBALL FINAL

Títol del treball:	<i>SION: Sistema de Informació i Obtenció de Notificacions</i>
Nom de l'autor:	<i>Alfredo Garcia Lerma (frodo969)</i>
Nom del consultor:	<i>Jordi Bécarres</i>
Data de lliurament (mm/aaaa):	<i>16-01-2016</i>
Àrea del Treball Final:	<i>Sistemes Encastats</i>
Titulació:	<i>Grau en Enginyeria Informàtica</i>
Abstract:	
<p>A fact is that we cannot discuss here the last few years has become more expensive electricity bill considerably. The causes behind this situation can be many and varied: possible lack of raw materials, an increase in cups or alternative energy production is a plot of the oil corporations. The causes of this situation, despite being quite interesting and motivating for study, are beyond the scope of this document and we do not focus on the cause that has generated, but how we can reduce its impact.</p> <p>Obviously we are all aware of that means that we turn up the received light our homes, to be applied corrective measures: lowering the heating, use energy saving light bulbs, etc. primarily with measures that generate savings, in order to absorb that increase, watching the investment, for example, light bulbs, which cannot exceed the savings generated.</p> <p>But what happens to a business when you turn up the electricity bill. Well firstly the production costs increase, and either increases the price of the product manufactured and loses competitiveness assumes either that side and try to reduce Moreover That increase production. In short, you get straight is a loss of competitiveness.</p>	
Paraules clau (entre 4 i 8):	
Analitzador, energia, Arduino, Raspberry, programari lliure, Modbus.	

Índex

SION: Sistema de Informació i Obtenció de Notificacions.....	i
1. Introducció.....	1
1.1. Justificació.....	1
1.2. Descripció del projecte	3
1.3. Objectius del TFG.....	4
1.4. Enfocament i mètode seguit.	4
1.5. Planificació del projecte.....	5
1.5.5. Canvis en la planificació	7
1.6. Productes emprats	7
1.7. Productes obtinguts.....	9
1.8. Descripció d'altres capítols.....	10
2. Antecedents	10
2.1. Estat de l'art	10
2.2. Estudi de mercat	27
3. Descripció funcional	28
3.1. Nucli SION.....	29
3.2. Visualització de dades.....	31
4. Descripció detallada	31
4.1. Base de dades	31
4.2. Nucli SION.....	33
4.3. Arduino.....	38
4.4. Interfície Web	43
5. Viabilitat tècnica	47
6. Valoració econòmica	47
7. Conclusions.....	50
7.1. Conclusions.....	50
7.2. Proposta de millores.....	51
7.3. Autoavaluació.....	51
8. Glossari	52
9. Bibliografia.....	52
10. Annexos	54
1.- Configuració Raspberry Pi	54
2.- Configuració Arduino.....	57
3.- Estructura dades MySQL	63

Llista de figures

Figura 1: metodologia tradicional.	4
Figura 3: mètode àgil. Font: www.northware.mx	5
Figura 4: planificació del projecte	6
Figura 5: eines emprades SION	8
Figura 6: elements Arduino	8
Figura 7: funcionament toroide. Font:wikipedia.org	9
Figura 8: diferents plaques del mercat. Font www.raspberrypi.org	12
Figura 9: característiques Arduino Mega. Font: arduino.cc	13
Figura 10: Arduino Ethernet Shield. Font: arduino.cc	14
Figura 11: RS485 mòdul per Arduino. Font: arduino-info	14
Figura 12: construcció de la trama Modbus TCP. Font prosoft-Technology.....	19
Figura 13: cable utp. Font wikipedia	20
Figura 14: trama Ethernet. Font Tonenbaum. Redes de computadores	21
Figura 15: bus RS485. Font: comunicaciones Modbus Serie Akros.....	21
Figura 16: RS485. Font wikipedia	22
Figura 17: diagrama de blocs SION	29
Figura 18: arquitectura SION.....	30
Figura 19: interacció dels equips.....	30
Figura 20: diagrama entorn web.....	31
Figura 21: taules base dades SION	32
Figura 22: diagrama de flux SION	33
Figura 23: classe Analitzador	34
Figura 24: diagrama estat analitzador	35
Figura 25: diagrama de flux lectura dades	36
Figura 26: tractament de registres.....	37
Figura 27: diagrama de flux actualització estat	38
Figura 28: diagrama de flux Arduino	39
Figura 29: captura del monitor sèrie en mode DEBUG	40
Figura 30: trama Modbus	43
Figura 31: diagrama d'accés	44
Figura 32: diagrama de flux login SION	44
Figura 33: canvi estat i valors web SION	46
Figura 34: pressupost PC mitjà. Font appinformàtica.....	48
Figura 35: pressupost Raspberry Pi. Font Amazon	49
Figura 36: pressupost S.O. Font appinformatica	49
Figura 37: pressupost Scada Indusoft 150 tags. Font smartInstruments	49

1. Introducció

Està demostrat que una de les millors eines que tenen al seu abast les empreses per a obtenir una millora en la competitivitat són les normes de qualitat, que afavoreixen la presa de decisions d'una forma lògica i suportada en dades concretes. Com feríem en la nostra llar, una empresa té que aplicar ajustos en matèria de consum energètic, però no ho té que fer d'una forma instintiva. Per exemple una empresa no deu d'apagar les llums de la planta per estalviar, sense analitzar el seu impacte per exemple, en matèria de prevenció de riscos, ja que una manca d'il·luminació pot generar accidents. Però si es detecta que hi ha zones sobre il·luminades si que es podria aplicar aquesta acció.

Afortunadament avui dia existeixen normes de qualitat que ens serveixen de guia a l'hora d'implantar sistemes de tot tipus, des de la qualitat alimentària (ISO 22000) o la ISO 5001 que tracta sobre l'estalvi energètic.

Aquesta última norma ens servirà de suport a l'hora de la presa de decisions, així com la seva avaluació i control.

Tampoc anem a parlar excessivament d'esta norma, ja que existeix força documentació i organitzacions on es pot cercar informació. On si es centrarem és en un apartat fonamental de la norma, l'auditoria energètica.

L'auditoria energètica consisteix en un anàlisi detallat, basat en un procediment sistemàtic, de les instal·lacions de la empresa així com de les pautes d'utilització, a través de mesures per a determinar el nivell d'eficiència energètica de la instal·lació i detectar així possibles fonts d'ineficiència.

Per a poder realitzar una auditoria d'aquests tipus cal poder enregistrar les dades referents al seu funcionament energètic dels equips durant el seu temps de funcionament.

Evidentment no es pot col·locar a una persona amb un multímetre anotant les dades en un full, cal automatitzar eixa tasca, de forma que eixes dades es puguin transformar en informació en poc temps.

El present document tracta de dissenyar y programar un sistema que permeti capturar les dades d'un nombre indeterminat d'analitzadors de xarxes (no cal dir que el nombre sempre serà major que 1) i permeti desar eixa informació en una base de dades per a la seva posterior consulta mitjançant informes predefinits o gràfiques. Tot el sistema estarà basat en programari lliure. Aquesta informació serà accessible mitjançant qualsevol navegador web compatible i actualitzat del mercat, ja sigui des de la mateixa xarxa on està ubicat el nostre servidor web, o be si es considera necessari i segur, accedir a ella a través d'una xarxa externa.

El projecte no només tractarà el disseny del programari necessari, si no que també es fabricaran eines basades en maquinari lliure que permetin reduir els seus costos d'implantació.

1.1. Justificació

Avui no existeix una solució lliure per a resoldre el problema d'adquisició de dades, es a dir, no hi ha un programari que permeti accedir a les lectures d'un grup d'analitzadors sense tindre que realitzar una gran despesa en matèria de llicències.

Es evident que aquest tipus de llicències ofereixen un suport i un treball de desenvolupament darrere d'elles, però que en molts casos, sobre tot en els casos de petites empreses allunyen la possibilitat de la inversió de la PIME en matèria d'auditoria energètica. Per a obtenir una instantània del cost real d'una instal·lació d'un sistema d'aquest estil posem un exemple real:

Aquest sistema està compost per els següents elements:

- Analitzadors, en concret CIRCUTOR CVM-C10¹ aquest equip ens permet el registre d'energies amb mesura de 4 quadrants (consum i generació), per a instal·lacions de mitja i baixa tensió, tant en circuits trifàsics a 3 o 4 fils o dues fases amb o sense neutre. El preu per equip és de: 268,5²€.
- Per a que cada analitzador funcioni cal dotar-lo d'uns elements anomenats transformadors toroïdals. Aquests elements són capaços mitjançant les seves propietats electromagnètiques de capturar el corrent que circula dins d'ells. Més informació a: <http://www.sc.ehu.es/sbweb/fisica/electromagnet/magnetico/cmagnetico.html> . El cost d'aquests productes varia segons la precisió i la mida dels mateixos, així com del tipus de muntatge. El cost pot anar des dels 20€ fins als 100€. Cal recordar que estem analitzant corrent trifàsica, per lo que necessitarem 3 transformadors toroïdals per punt d'anàlisi de xarxa.
- Desplegament de les comunicacions e instal·lacions del equips. Una vegada decidit el nombre d'equips que es van a instal·lar cal muntar-los al seu lloc i oferir-los comunicació per a poder capturar les seves dades de forma remota. En la majoria de plantes no es poden emprar sistemes sense fil, ja que el soroll elèctric que genera la maquinaria fa inviable un sistema WIFI o similar. El cost del desplegament variarà segons la distribució de la planta, separació entre els equips i xarxes ja existents a la planta o el nombre d'equips a implantar.
- Una vegada hem realitzat el desplegament, ens queda dur les comunicacions a un equip que controli les peticions d'informació així com que sigui capaç de desar les dades i poder transformar-les més tard en informació. Enlloc de emprar ordinadors industrials, que tenen un cost força elevat i unes característiques força limitades, utilitzarem ordinadors clònics, amb una configuració bàsica recomanada de:
 - Processador: Intel I3.
 - Ram: 4 GB.
 - Sistema Operatiu: Windows 7.
 - Disc Dur: 1 SSD per a un accés ràpid al històric de les dades i un altre per a backups locals.
 - Monitor: mínim 21"

¹ <http://circuitor.es/es/productos/medida-y-control/analizadores-de-redes-fijos/analizadores-de-redes/serie-cvm-c10-detail>

² http://www.muntanerelectro.com/wp-content/uploads/2014/03/40distribucion-SP_LR.pdf

Com es pot observar el sistema operatiu té que ser Windows, ja que no existeixen aplicacions de visualització y control de dades que no treballen amb altres sistemes operatius, almenys de forma massiva al territori Espanyol. Tot i existir d'altres, estan poc implantades al mercat espanyol, i no disposen de la confiança ni del suport que tenen altres productes.

- Sistema de visualització i control SCADA. Tal i com ja hem comentat abans existeixen múltiples solucions d'aquest tipus al mercat. Ens centrarem en dues, una anomenada WinCC³ desenvolupada per Siemens i l'altra Indusoft⁴ desenvolupada per Wonderware recentment adquirida per Schneider Electric. Ambdós disposen d'un sistema de llicències similar. Per a cada projecte tens que establir el nombre de variables que creus que vas a utilitzar. Tant en un com en l'altre tens que adquirir llicències de desenvolupament i llicències d'execució, anomenades de Runtime. Per exemple, per a un projecte mitjà d'uns 30 analitzadors de xarxes, es solen emprar unes 3000 variables, el que suposa un cost realment alt, prop dels 3000€. Aquestes llicències de Runtime han de ser adquirides per a cada projecte, ja que s'instal·len en l'equip on s'executarà el SCADA, a banda cal adquirir la llicència de desenvolupament que be pot ser emprada en diferents projectes també té un cost similar.

A tots els costos que es mostren dalt cal incloure les hores de programació del SCADA per a realitzar els informes, establir comunicacions, dibuix i disseny de les pantalles, etc.

Per lo que ara cal analitzar: és necessari un software tan car?, es evident que si el que pretenem es controlar una planta, moure motors, cintes, fabricar productes i controlar elements crítics es pot justificar una inversió en aquest tipus de software, però per a la captura seqüencial d'informació és necessari un cost tan gran?

El que pretén aquest document és que a la finalització del mateix sigui possible la captura d'informació d'un grup de X analitzadors de xarxes amb un cost força reduït, i utilitzant sempre que sigui possible maquinari i programari lliure.

1.2. Descripció del projecte

SION (Sistema de Informació i Obtenció de Notificacions) es un sistema que permetrà, mitjançant petites modificacions del codi, la captura de dades de diferents analitzadors de xarxes.

La solució que proposem és una Raspberry Pi que realitza consultes de forma seqüencial als analitzadors instal·lats.

³ <http://w3.siemens.com/mcims/human-machine-interface/en/visualization/software/scada/pages/default.aspx>

⁴ <http://www.indusoft.com/>

Aquesta informació es desa en una estructura de dades que representa la informació (voltatge, intensitat, THD, consum...) de cada equip.

Per a poder obtenir un històric de les dades per a un posterior anàlisi, es desarà cada X segons en una base de dades les lectures realitzades de cada equip.

El sistema te que ser capaç de generar gràfiques dels diferents equips (corrents, consums).

La segona part del projecte serà una placa Arduino que serà un pont entre els dos tipus de protocols emprats, per una banda Modbus TCP que correspon a la Raspberry Pi i per altra banda Modbus RTU que es la versió del protocol que utilitzen els analitzadors.

1.3. Objectius del TFG.

1. Programació d'un sistema d'adquisició de dades en Python capaç de comunicar mitjançant un protocol estàndard, com es Modbus TCP amb diferents equips de mesures de dades energètiques. Aquest software ha de córrer en una Raspberry Pi
2. Que aquestes dades siguin transformades en informació mitjançant informes predefinits i gràfiques explicatives.
3. Que aquesta informació pugui ser accedida de forma descentralitzada, es a dir des de qualsevol punt, mitjançant navegadors Web. Per lo que la Raspberry ha de ser alhora servidor web i servidor de bases de dades.
4. El sistema ha d'estar programat en programari lliure. En concret emprarem Python, MySQL i Lighttpd, a banda de dos plataformes de maquinari lliure, Arduino i Raspberry.
5. Creació d'un sistema mixt de maquinari i programari que sigui capaç de traduir les trames que emetrà la Raspberry en Modbus TCP a telegrams Modbus RTU.

1.4. Enfocament i mètode seguit.

Com en qualsevol tipus de projecte, la primera tasca que cal fer es decidir quina metodologia de treball es va a seguir durant el seu transcurs. Per això calia avaluar quin dels dos mètodes principals permetrien una millor obtenció de resultats.

En primer lloc es va avaluar la metodologia tradicional, que es basa en l'execució de forma seqüencial d'un seguit d'etapes que du com a finalització un producte de software acabat.



Figura 1: metodologia tradicional.

La metodologia tradicional ofereix uns avantatges que calia valorar, com la simplicitat del seu seguiment i la obligació de generar documentació, que

facilitaria força el treball d'aquesta memòria, però té un problema prou gran, i es la no reversibilitat.

Si disposes d'un equip gran de treball es "fàcil" dur a terme aquesta metodologia, ja que es factible que les errades puguin ser resoltes a temps, mentre que en equips de poca gent, o com és el cas, formats per una sola personal, detectar o millor dit no detectar un problema por acabar per tancar un projecte.



Figura 2: mètode àgil. Font: www.northware.mx

Fiquem com a exemple una part important del projecte, les comunicacions. El mètode tradicional hagués obligat a que les proves no es feren fins a que producte no estigués completament finalitzat, eixa part del codi no s'hagués testejat correctament i si haguera errors de programació no s'hagueren pogut resoldre fins al final, on inclús hagués sigut difícil si discernir si les errades eren produïdes per la part de les comunicacions del programa o be per una part de disseny del maquinari.

Per això s'opta per una metodologia àgil, ja que cada iteració dona com a resultat codi utilitzable en parts posteriors del projecte, i com a factor important es que no cal tornar enrere, ja que les modificacions formen part intrínseca de la pròpia metodologia.

1.5. Planificació del projecte

Com en qualsevol tipus de projecte cal marcar unes fites temporals que tenen que ser acomplides, si no els projectes poden eternitzar-se, quedant-se enclavats en les seves primeres fases i deixant molts pocs recursos a les parts més importants com són les de test i documentació. Per això i per a poder

racionalitzar els recursos el projecte que te una duració aproximada de 4 mesos (1 semestre) va ser dividit en les següents fases.

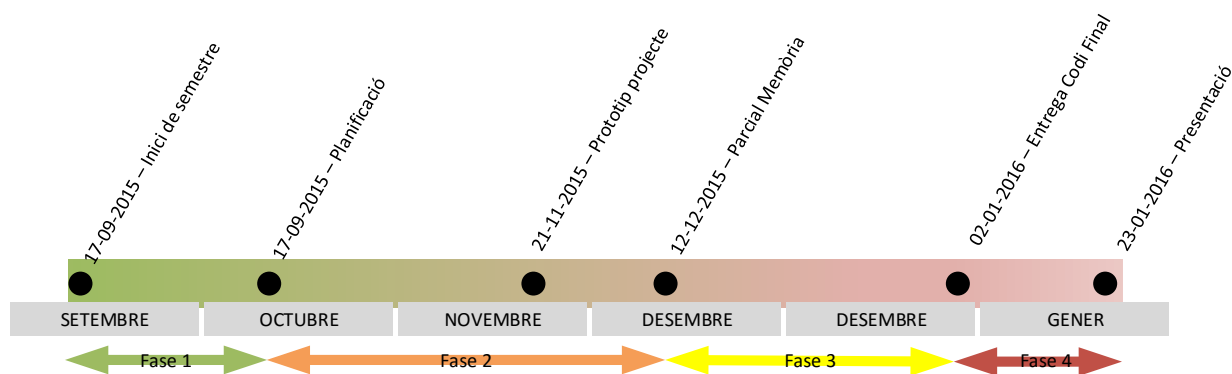


Figura 3: planificació del projecte

1.5.1. FASE 1: planificació

La primera fase marca l'estudi del projecte, eines disponibles i viabilitat teòrica del mateix. Una vegada avaluades totes les opcions es planteja una planificació temporal amb fites a assolir.

Data Inici: 17-09-2015

Data Fi: 17-10-2015

Duració: 1 mes

Objectius: planificació del projecte i fites a assolir.

1.5.2. FASE 2: prototipatge

Disseny del producte.

Data Inici: 17-10-2015

Data Fi: 12-12-2015

Duració: 2 mesos.

Objectius: muntatge inicial del producte, consta de dues entregues, la primera constarà de la situació del producte i finalitzarà amb l'entrega parcial de la memòria.

1.5.3. FASE 3: finalització

Finalització del producte i proves de funcionament.

Data Inici: 12-12-2015

Data Fi: 02-01-2016

Duració: 3 setmanes.

Objectius: implementació del codi generat, proves en entorns reals.

1.5.4. FASE 4: documentació

Generació de la documentació del projecte.

Data Inici: 02-01-2016

Data Fi: 23-01-2015

Duració: 3 setmanes.

Objectius: documentar el projecte (memòria, manuals) i presentació del mateix.

1.5.5. Canvis en la planificació

Diferents motius han generat que el projecte no hagi arribat sobre tot a les primeres entregues, per tant les fases 1 i 2 s'han vist afectades.

En la direcció del projecte es deuria haver observat que la possibilitat de que algun dels membres de l'equip pogués per exemple caure malalt, o be que decidís deixar el projecte. Afortunadament s'ha pogut reconduir el projecte, no pas modificant-lo, sinó modificant les fases, on pràcticament les fases 1 i 2 s'han convertit en la mateixa.

1.6. Productes emprats

A nivell pedagògic ha sigut un projecte molt interessant, ja que he emprat un fum de coneixements de la carrera, així con diferents tecnologies i eines per a dur a terme el producte final.

Per a la part del cor de SION s'ha emprat Python com a llenguatge de programació, principalment amb tres llibreries: *Pymodbus*, per a les comunicacions en Modbus, *PyMySQL* per al diàleg amb la base de dades i finalment *loggin*, que ens serveix per al *debug* i controls d'excepcions de l'aplicació. El sistema operatiu sobre el que corre SION és una distribució gnu/Linux anomenada *Raspbian* que es una versió de *Debian* sobre dispositius en arquitectura ARM.

El maquinari sobre el que se suporta aquesta part del projecte es un Raspberry Pi B+.

Seguidament podem observar un diagrama esquematitzant les eines emprades.

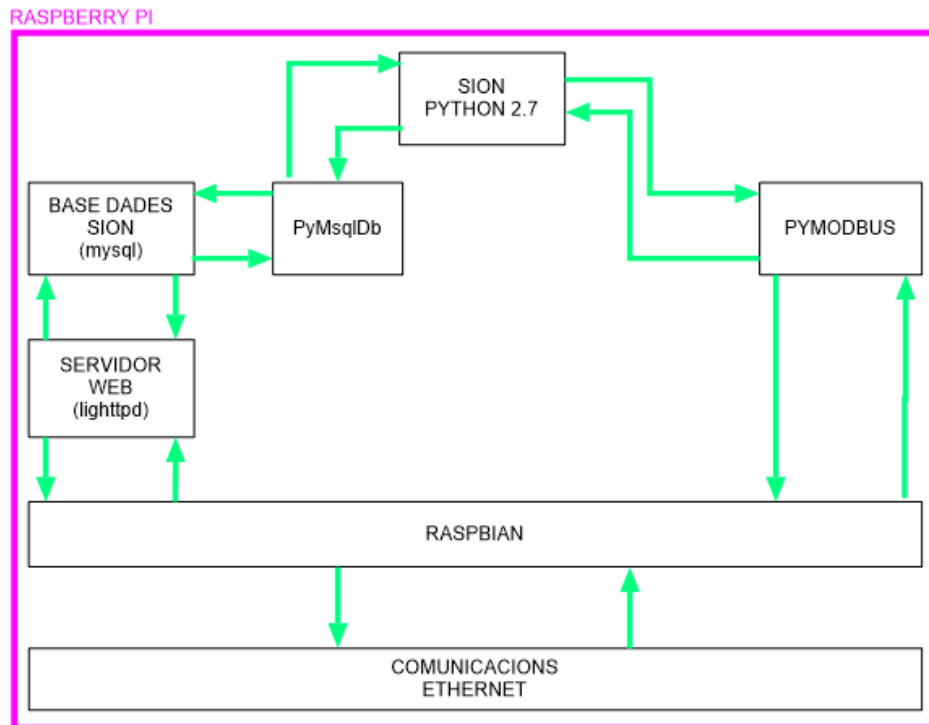


Figura 4: eines emprades SION

La part del servidor web, es a dir la transformació de les dades en informació, se ha realitzat mitjançant PHP, que permet la comunicació entre un client web i la base de dades mitjançant peticions asíncrones en *Ajax*. Per a visualitzar la informació s'ha emprat HTML5 i CSS, així com per a dinamitzar les dades s'ha emprat *Javascript* en concret *JQuery*.

Com a eina de programació s'ha emprat *Geany*, que es un editor de text gratuït que funciona tant el Linux com en *Microsoft Windows*®.

Per altra banda tenim la part dedicada a Arduino, que com a eina de programació s'ha emprat el seu propi IDE. La següent imatge mostra el material emprat per a desenvolupar aquesta part.

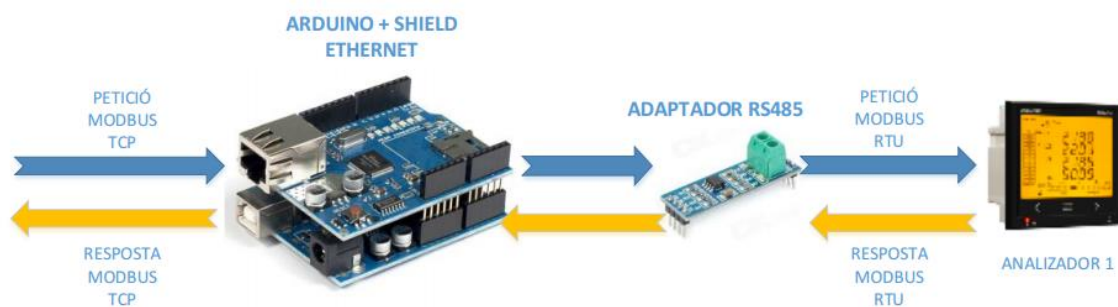


Figura 5: elements Arduino

Finalment la part del analitzador de xarxes, que és l'encarregat d'obtenir les dades directament de la xarxa elèctrica. Per a que el sistema funcioni cal dotar-lo tant d'un analitzador tal com indica la figura 5, com d'un toroide que sigui capaç de capturar el flux de corrent elèctrica que transita per un conductor.

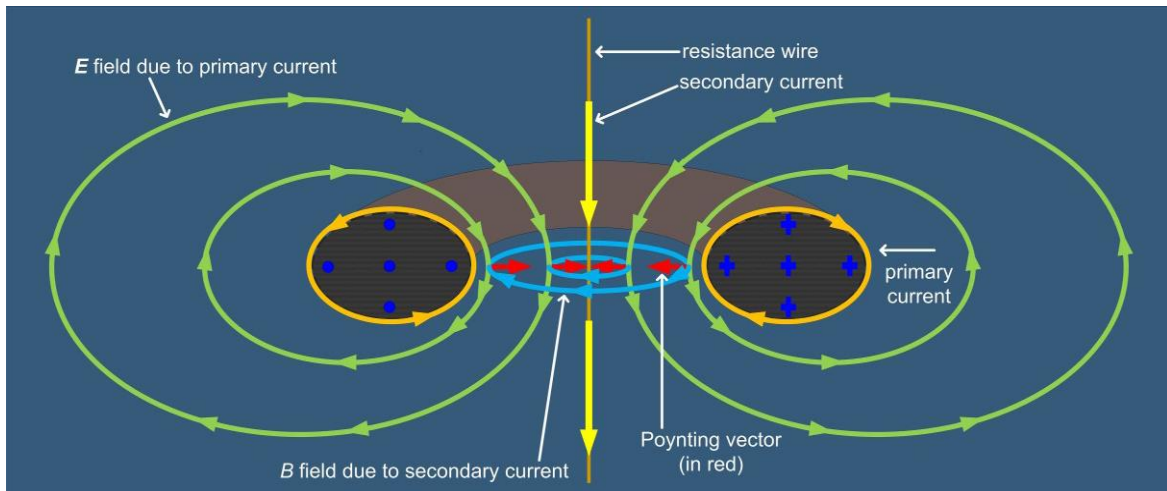


Figura 6: funcionament toroide. Font:wikipedia.org

Per a no estendre'ns massa direm que un transformador toroidal és una bobina que rodeja un nucli, aquesta bobina genera corrent elèctrica gràcies al camp electromagnètic que genera un conductor conforme travessa el propi toroide tal i com es pot observar esquemàticament a la figura 6. Com ja es conegut els camps electromagnètics generats de forma radial pel conductor segueixen un sentit de gir que es pot predir gràcies a la regla de la mà dreta⁵. Esta regla dicta en quin sentit es generarà el corrent elèctric en el toroide, per lo qual indica que el toroide té posició. Si es gira el toroide, els valors que s'obtenen seran negatius, per lo que descomptarà consum, ja que no existeixen per a l'analitzador corrents negatives, si no elements generadors o consumidors d'energia.

1.7. Productes obtinguts

El conjunt de diferents tecnologies emprades donen com a resultat un conjunt anomenat SION. SION són les sigles de Sistema de Informació i Obtenció de Notificacions, si be en aquest cas s'empra per a obtenir les dades d'un analitzador de xarxes.

El que s'obté és un programa que corre dins d'una Raspberry Pi i que mitjançant consultes realitzades de forma síncrona llegeix les dades d'un o més analitzadors de xarxes que es comuniquin mitjançant el protocol Modbus TCP. Els equips que van dotats amb targetes de xarxa Ethernet i que suportin Modbus TCP són força cars, per lo que habitualment s'empren passarel·les Modbus TCP cap a Modbus RTU. En nostre sistema també disposa d'aquest tipus de passarel·les però fabricades i programades en Arduino.

El sistema fa les consultes als equips connectats, desa les dades en una base de dades, que pot ser remota o local, i serveix eixes dades en forma de informació mitjançant un servidor web que ell mateix incorpora. A banda de mostrar la informació capturada, indica el estat de l'equip (si està en fallada o no).

Resumint el que hem obtingut es un sistema que pot substituir un SCADA comercial, amb més funcionalitats que un SCADA comercial, ja que la seva visualització es multi plataforma i multi dispositiu, sense limitacions d'equips a

⁵ https://es.wikipedia.org/wiki/Regla_de_la_mano_derecha

connectar, i que permet treballar amb qualsevol equip que accepti aquest protocol.

Junt a la memòria no s'entregarà el codi font del nucli de SION, però sí l'estructura de la base dades així com el seu manual d'assemblatge i el codi font de la part web de visualització de les dades. També es podrà trobar un manual d'usuari de la plataforma i un manual de configuració de l'entorn necessari. Seguint tots els manuals es deu de poder replicar per complet el sistema SION.

1.8. Descripció d'altres capítols

Els següents capítols de la memòria tractaran de la **situació actual** en matèria sistemes d'anàlisi i obtenció de dades i de la viabilitat del projecte mitjançant un breu **estudi de mercat**.

Seguidament ja es tractarà en profunditat el **disseny realitzat** així com les decisions preses per a la seva elaboració.

Després d'una **valoració del producte**, el quantificarem i compararem amb productes comercials, conclouríem el present document amb una **valoració final i millores** que poden incorporar les següents versions.

2. Antecedents

2.1. Estat de l'art

2.1.1. Python

De forma breu podem dir que Python es un llenguatge de propòsit general interpretat, es a dir que no requereix compilar-lo per a poder-lo executar, el que el torna força útil per a per a la realització de *scripts*. Alguns autors l'han definit com *object-oriented scripting Language*⁶ (Lutz, 2009)

La idea primera de Python es la de poder crear codi comprensible i mantenible d'una forma àgil i potent, per això i per la seva potència e integració en diferents plataformes i la seva orientació a objectes el fan una bona opció per a desenvolupar aquest tipus de projectes.

Python va ser creat per Guido Van Rossum a finals dels anys 80 en el CWI⁷ en Països Baixos, com a curiositat podem dir que el nom de Python es agafat per l'afició de Guido pels conjunt còmic Monty Python. El seu objectiu era crear un llenguatge de programació orientat a objectes senzill que servis per a realitzar tasques dins d'Unix que es feien mitjançant C. Guido ha passat per diverses companyies importants del món d'internet, com Google o Zope Corporation, actualment treballa com a desenvolupador a Dropbox.

Aquest llenguatge de programació s'empra en multitud de projectes i empreses punteres en el sector l'utilitzen, com ILM (Industrial Light & Magic), Philips, D-link⁸.

Actualment es pot desenvolupar en dues versions la 2.X i la 3.X, en concret la 2.7.11 i la 3.5.1. Concretament existeixen dues versions ja que la 2.x és l'antiga i permet actualitzar i mantenir el codi ja escrit, mentre que la 3.x és la nova

⁶ Llenguatge de scripts orientat a objectes

⁷ CWI: Centre para les Matemàtiques y la Informàtica (Centrum Wiskunde & Informatica)

⁸ El llistat complet es pot trobar en : <https://www.python.org/about/success/>

orientació que vol prendre el llenguatge com per exemple el suport a UNICODE.

Si es pensa desenvolupar un codi completament nou, i no dependre de llibreries de tercers, es més interessant treballar en la nova versió, sobretot per qüestions de manteniment. Però si el producte que volem desenvolupar, com es el nostre cas, pensa utilitzar llibreries de tercers és millor desenvolupar en la versió 2.x, ja que evitarem problemes de compatibilitat en les llibreries per possible falta de manteniment de les mateixes.

Un altra part important es la llicència⁹ de Python. Python es gratuït (GLP) inclòs si s'empra per a sistemes empresarials, i aquí es la clau del seu gran desenvolupament en industria.

Resumint, Python es un llenguatge multi plataforma, interpretat, interactiu amb un fum de llibreries i de sintaxis clara ja que no empra ni claus ni punts ni comes, senzillament utilitza tabulacions. Per això es una opció clara a l'hora de desenvolupar projectes on no es té molt clar en quina plataforma pot acabar funcionant.

2.1.2. Raspberry Pi

Per a comprendre millor l'origen d'aquest producte cal remuntar-se als principi dels anys 1980, on la BBC, si el canal de televisió públic anglès, va llançar una sèrie de petits ordinadors domèstics anomenats BBC Micro. La idea principal d'aquests equips era apropar la informàtica al xiquets de l'època. D'aquest dispositiu existiren dos models, el model A i el model B.

Ara viatgem una mica al futur, fins al 2009 que es va crear la fundació Raspberry Pi, amb la idea de desenvolupar un producte similar al dels anys 1980, però evidentment adaptat als temps actuals. La historia de la placa fins arribar a la seva producció massiva es força curiosa i es pot llegir a la wikipedia en el següent enllaç: https://es.wikipedia.org/wiki/Raspberry_Pi però creguem que estendrà massa el text i no aportaria informació rellevant al document.

Actualment es poden trobar dues plaques convivint al mercat, la Raspberry Pi 1 Model B+ (que es la que s'ha emprat per al projecte) i la Raspberry Pi Model B 2 (objecte de desig) la figura 7 compara no sols els dos models esmentats anteriorment, si no que les compara amb altres plaques amb característiques similars del mercat.

⁹ <https://docs.python.org/3/license.html>

	Pi 1 B+	Pi 2 B	BBB	Edison	CI20
CPU	Arm11	Cortex A7	Cortex A8	Atom + Quark	MIPS
Cores	1	4	1	2 + 1	2
Clock	700MHz	900MHz	1000MHz	500MHz	1200MHz
GPU	Videocore IV	Videocore IV	PowerVR SGX530	None	PowerVR SGX540
Memory	512MB	1GB	512MB	1GB	1GB
USB Ports	4	4	2	1*	2
Flash	None	None	2GB	4GB	8GB
Storage	microSD	microSD	microSD	microSD*	SD
Network	10/100	10/100	10/100	None	10/100
GPIO	40-pin	40-pin	2x46-pin	70-pin Hirose	40-pin
Wifi	No	No	No	Yes	Yes
Bluetooth	No	No	No	Yes	Yes
RRP	\$35	\$35	\$49	\$85*	\$65

Figura 7: diferents plaques del mercat. Font www.raspberrypi.org

Un altre gran avantatge és el sistema operatiu que podem fer córrer aquestes plaques, en concret per al nostre projecte hem emprat Raspbian, un sistema operatiu basat en Debian però ajustat per a poder treballar en arquitectures ARM, en concret fem la versió Raspbian Jessie, amb el nucli 4.1.

Com podem observar tenim un hardware que es capaç de suportar un sistema operatiu gràfic i tota la llibertat de treball que ofereixen els sistemes operatius en base Linux. Sobre aquesta plataforma pot córrer sense cap problema un servidor de dades Mysql i un servidor Web així com executar aplicacions en Python o C de forma nativa.

Una part molt interessant d'aquests dispositius són els ports GPIO, que permeten actuar sobre el món físic directament, o bé actuant o llegint entrades digitals. Els ports GPIO també es poden configurar com a UART, SPI i I2C que ofereix opcions de comunicacions amb un fum de dispositius.

Raspberry s'ha convertit tant en un joguet per als aficionats a la electrònica i la informàtica, com en un media center fantàstic gràcies a les distribucions dedicades, com en una eina per al internet de les coses. Pel seu reduït cost es també una opció per a substituir equips que tenen que fer tasques repetitives i que no requereixen una gran quantitat de capacitat de processament.

En un projecte d'eficiència energètica també caldria parlar del consum de l'equip, aquest necessita únicament un adaptador de corrent de 5v i 2 ampers, o sigui que dona un consum de 10w. Però també pot treballar amb menys corrent. Actualment el projecte està funcionant amb un carregador de mòbil de 1 amper, o sigui que té un consum de 5w. Com a comparació una font ATX d'un PC convencional actual solen ser de 500w.

2.1.3. Arduino

En l'any 2006 els estudiants del institut IVREA (Itàlia) empraven plaques que utilitzaven el microcontrolador Basic Stamp, que tenia un cost proper als 100\$. Tant en el 2006 com avui 100\$ es un cost excessiu per a un producte educatiu. Això i l'imminent tancament del fabricant de les plaques, feia necessari trobar noves opcions. Massimo Banzì donava classes en IVREA, i junt amb altres estudiants i professionals donaren forma al projecte.

Per a evitar que tingués el mateix problema que les plaques llicenciades, que si tancava la empresa, desapareixeria el producte, desenvoluparen tant el maquinari com el programari baix codi obert, de forma que la comunitat pogués continuar-lo.

Com hem comentat en aquesta breu introducció, la plataforma no és simplement una placa de programari sinó que disposa del seu propi entorn o IDE de programació.

El seu llenguatge de programació està basat en C++, i de fet suporta instruccions i llibreries en C i C++ si es segueixen les instruccions corresponents.

Al ser de codi obert, moltes empreses s'han dedicat a fabricar o be clons seus abaratint el seu preu o be la fabricació de Shields (escuts) que no són altra cosa que plaques que encaixen als pins de la placa Arduino i que ofereixen diferents funcionalitats que van des de: connexió Ethernet i convertir-lo en un servidor web, enviament de SMS, control de motors, ect.

A la web d'Arduino o be en una simple cerca per internet podrem trobar projectes per a realitzar amb un cost de desenes d'euros.

Un últim apunt sobre el codi Arduino, els escuts dels que hem parlat solen vindre amb les seves pròpies llibreries, el que fa que el seu ús sigui molt simple.

En referent al maquinari de la placa existeixen diferents configuracions, la majoria d'elles basades en la família de microcontroladors ATmega.

Degut a la gran quantitat de membres de la que disposa la família Arduino inclourem únicament les dades de la placa que hem emprat al nostre projecte, la Arduino Mega.

Microcontroller	ATmega2560
Operating Voltage	5V
Input Voltage (recommended)	7-12V
Input Voltage (limit)	6-20V
Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16
DC Current per I/O Pin	20 mA
DC Current for 3.3V Pin	50 mA
Flash Memory	256 KB of which 8 KB used by bootloader
SRAM	8 KB
EEPROM	4 KB
Clock Speed	16 MHz
Length	101.52 mm
Width	53.3 mm
Weight	37 g

Figura 8: característiques Arduino Mega. Font: arduino.cc

La placa Arduino Mega és la placa que més connexions disposa, en concret disposa de 54 pins GPIO, dels quals alguns tenen funcions especialitzades com 4 ports series, interrupcions externes, lectura escriptura analògica amb una resolució de 8 bits, comunicació SPI i TWI.

El consum energètic de la placa també és irrisori, ja que estem parlant de 5v i 80mA com a màxim, el que dona un consum de 4w.

ARDUINO ETHERNET SHIELD

La Arduino Ethernet Shield dota al Arduino de comunicació Ethernet 10/100 Mb. Una vegada muntada sobre una placa Arduino compatible, en poc temps disposarem de la connectivitat necessària, inclòs podem dotar a la nostra placa de connexió a internet. Està basat en Wiznet W5100¹⁰ Ethernet chip, proveeix una pila TCP-IP i UDP-IP. Suporta fins a 4 sockets (connexions) recurrents. També disposa de ranura per a targeta microSD, que ben be es podria utilitzar per escriure un fitxer de diari de les transaccions realitzades. Però este punt quedaria per a pròximes versions.

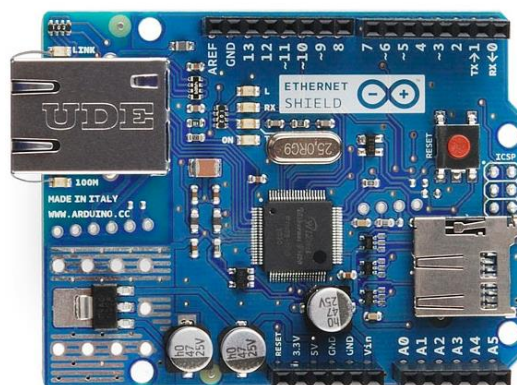


Figura 9: Arduino Ethernet Shield. Font: arduino.cc

RS485 SERIAL COMMUNICATION

Per a dotar de comunicació serial al nostre equip hem contat amb aquest petit dispositiu basat en el micro MAX485¹¹ que permet establir una xarxa RS485 entre el nostre conjunt u qualsevol altre node.

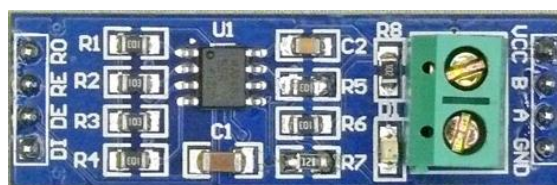


Figura 10: RS485 mòdul per Arduino. Font: arduino-info

¹⁰

http://www.wiznet.co.kr/Sub_Modules/en/product/Product_Detail.asp?cate1=5&cate2=7&cate3=26&pid=1011

¹¹ http://yourduino.com/sunshop2/index.php?l=product_detail&p=323

Per a oferir aquest tipus de comunicació seria suficient amb el xip MAX485 i unes poques resistències, però hem optat per adquirir el sistema ja muntat que ofereix un millor acabat i facilita les proves posteriors.

2.1.4. MySQL

MySQL es un sistema de bases de dades relacional, multifil i multiusuari, amb més de sis milions d'instal·lacions (wikipedia, 2015).

Una de les claus de MySQL es el seu tipus de llicència, disposa d'un sistema dual. Es a dir, si vols desenvolupar un producte amb MySQL de forma privativa, tens que adquirir la llicència corresponent, mentre que si el que vas a desenvolupar es obert podràs disposar d'una llicència GNU GPL. Açò es possible per que MySQL es propietat de Sun Microsistems, que disposa drets d'autor sobre part del codi.

Per al nostre projecte l'emprem no per la seva llicència, ja que podríem optar per altres sistemes de bases de dades amb un altre tipus de llicenciamnt, si no per la quantitat de documentació existent a la web, així com les llibreries existents tant per a Python com per a PHP.

Al existir eixe suport la instal·lació d'aquests sistemes es força senzilla i el seu rendiment està molt optimitzat.

La versió que se instal·la en Raspbian es la 5.5.44-0+deb8u1. A l'hora de redactar el present document l'última versió disponible era la 5.7, però es clar per a altres plataformes que no són ARM.

2.1.5. Lighttpd

A l'hora de triar un servidor web cal veure primer de tot quines necessitats tenim, en el nostre cas la nostra limitació principal no serà pas el nombre d'usuaris recurrents ni un us intensiu d'ell, el nostre principal requisit serà que consumeixi pocs recursos, recordem que disposem de 512 mB de RAM. Un altre requisit fonamental serà la capacitat d'integració amb altres tecnologies, com PHP, i la nova HTML 5, que parlarem més endavant.

La millor descripció del producte la podem llegir a la seva pàgina web:

Seguretat, velocitat, compliment i flexibilitat - tots ells descriuen lighttpd (pron light i .), Que és ràpidament redefinir l'eficiència d'un servidor web; ja que està dissenyat i optimitzat per a entorns d'alt rendiment. Amb una petita empremta de memòria en comparació amb altres servidors web, la gestió eficaç de la cpu - càrrega, i conjunt de funcions avançades (FastCGI, SCGI, autenticació, Sortida - compressió, URL - Reescriptura i molts més) lighttpd és la solució perfecta per a tots els servidors que està patint problemes de càrrega. I el millor de tot, és de codi obert amb llicència sota la llicència BSD revisada.

En la mateixa web del producte podem observar una comparació amb el altre gran rival del servidor web triat Apache 2 (<https://www.lighttpd.net/benchmark/>). Podem creure o no les mostres que ens dona el propi fabricant, però si que podem dir que a l'hora de servir les pàgines web es força més ràpid que un servidor Apache, i si mirem el rendiment de la Raspberry a l'hora de servir una pàgina web pràcticament no es veu afectada per eixa activitat.

2.1.6. Modbus

Modbus presenta una estructura de missatges que pot ser reconeguda per diferents dispositius independentment del tipus de medi físic emprat, així com de la versió emprada, ja sigui RTU o TCP.

Modbus defineix un únic mestre (es qui realitza les interrogacions) i tot un seguit d'esclaus, com a mínim un, que executen o responen les peticions realitzades per el mestre.

MODBUS RTU

Els nostres analitzadors (en davant els anomenarem esclaus) únicament disposen d'aquest protocol de comunicacions, per lo que és força important el seu estudi.

Modbus RTU s'empra sobre una connexió física RS485. La seva estructura de connexió es pot veure a la imatge 4, on podem observar la simplicitat del seu connexionat.

Una vegada establert el mitjà físic, ens queda observar que transmetrem per ell. La primera par de la trama és una seqüència d'activació, dona igual els bits que siguin el que importa es que ocupin 3,5 vedades el temps de transmissió d'un caràcter. Una vegada monitoritzat eixe temps el receptor sap que es van a enviar la resta de la trama que està composta per:

Taula 1: trama Modbus RTU

	NOMBRE ESCLAU	FUNCIÓ	DADES	CRC
Mida	1 byte	1 byte	N bytes	2 bytes

El **nombre d'esclau** es un nombre que identificarà al esclau corresponent de la xarxa, una vegada el mestre ha llançant una trama Modbus, únicament respondrà l'esclau que tingui eixe número assignat. El nombre mínim es 1, ja que el valor 0 representa un multicast, per lo que podrien respondre tots els integrants de la xarxa, però aquesta part del protocol no està tan universalitzada com la resta.

La **funció** es la acció que ha de realitzar el esclau que llegeixi el missatge, generalment són aquestes les accions a realitzar:

Taula 2: funcions de lectura Modbus

Funció	Descripció	Memòria relacionada	Operant
0x01	1 - Read Coil Status	@00000 a @09999	Bit
0x02	2 – Read Input Status	@10000 a @19999	Bit
0x03	3 – Read Holding Registers	@40000 a @49999	Registres
0x04	4 – Read Input Registers	@30000 a @39999	Registres

Existeixen unes altres que serveixen per a escriure dades en el esclau, anomenades Force Single Coil o Preset Single Register, però que no les anem a utilitzar en aquest projecte, ja que únicament realitzarem lectures del esclau.

En concret ens centrarem en les funcions 0x03 i 0x04, lectura de registres, ja que la tasca que farem sempre serà llegir registres de dades dels esclaus.

Les **dades** tindran una mida depenent de la funció a emprar, per exemple en el nostre cas, llegir un valor del esclau, el bloc de dades tindrà la següent lectura:

Taula 3: exemple trama Modbus

NOMBRE ESCLAU	FUNCIÓ	DADES	CRC
0x01	0x03	0x00 0x00 0x00 0x0A	0xC4 0x34

El que vol dir la trama de la taula 4 es lo següent: vull que el esclau número 1 hem llegeixi el registres (0x03 = Read Holding Registers) començant per la direcció @40001 fins a la @40010.

El **CRC** ja serà explicat en el punt dedicat a ell, si que tornarem a dir que el polinomi generador es el $CRC_{16} = X^{16} + X^{15} + X^2 + 1$.

Una vegada rebuda la resposta per l'esclau, aquest es disposa a tractar la petició, si la comprovació del CRC es correcta, i tant la funció es suportada per l'esclau, respondrà així:

Taula 4: resposta Modbus

NOMBRE ESCLAU	FUNCIÓ	Nombre de bytes	Valor del Primer	Valor del Segon	Valor del últim	CRC
---------------	--------	-----------------	------------------	-------	-----------------	-----------------	-----

		llegits	Registre		Registre	Registre	
0x01	0x03	0x14	0x00 0x00	0x00 0x03	0x00 0x03	0xCB 0x2E

Una vegada rebuda la trama de la resposta, comprovaríem el seu CRC i si es correcte desariem el seu valor.

MODBUS TCP

Un avantatge gran del protocol Modbus es que no difereix massa d'una versió a una altra, es a dir no existeixen grans diferències entre els tipus, mes enllà del mitjà físic i el encapsulament de les peticions Modbus RTU dins de datagrames TCP/IP.

Per això en aquesta versió no utilitzarem el mitjà RS485, si no que emprarem el mitjà Ethernet. En poques paraules TCP/IP permet el intercanvi de blocs de dades entre equips d'una mateixa xarxa. Com es sabut és un estàndard mundial i serveix de base a Internet. La funció principal del TCP es garantir que els paquets de dades es reben correctament, mentre que IP s'assegura que els missatges s'aborden i es col·loquen correctament. Recordeu que la combinació de TCP/IP no es més que un protocol de transport, i no defineix com les dades es tenen que interpretar (aquest es el treball del protocol d'aplicació, Modbus en el nostre cas).

Resumint Modbus TCP utilitza TCP/IP i Ethernet per portar les dades de l'estructura del missatge Modbus entre dispositius compatibles, combinant una xarxa física Ethernet, amb un enrutament estàndard TCP/IP i un mètode de representació de dades Modbus. Bàsicament el missatge Modbus TCP/IP és una comunicació Modbus RTU dins d'una xarxa Ethernet + TCP/IP. La següent imatge il·lustra aquest concepte.

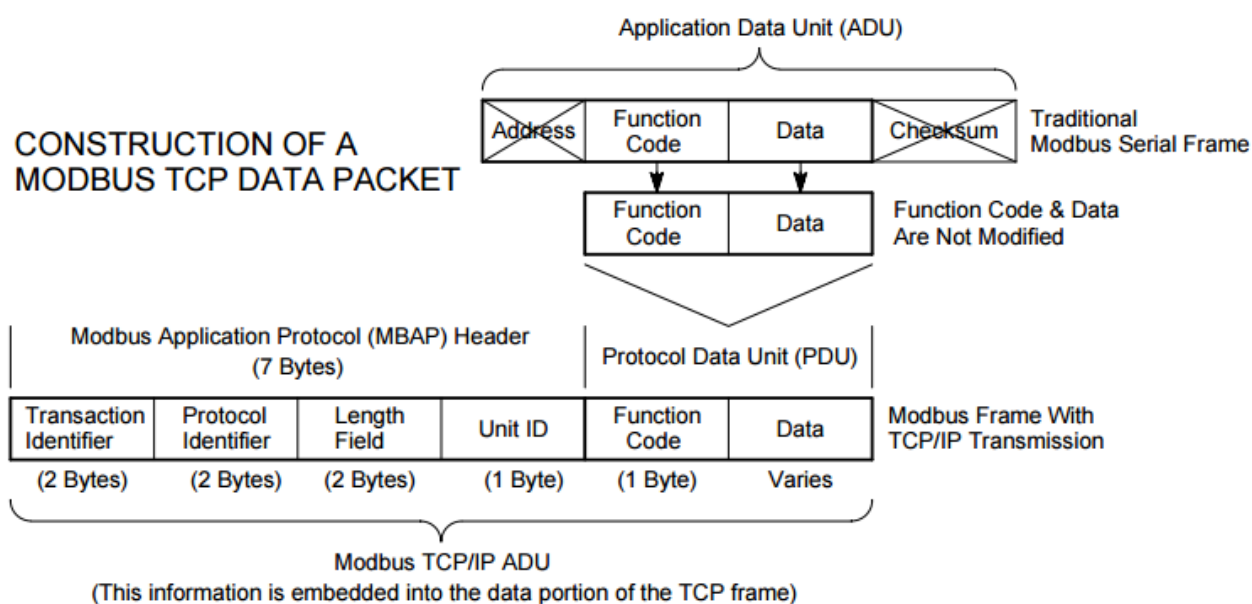


Figura 11: construcció de la trama Modbus TCP. Font prosoft-Technology

Com es pot observar en la il·lustració 6, per a transformar el protocol RTU a TCP tenim que fer les següents tasques:

1. Agafar una petició RTU.
2. Eliminar d'eixa petició tant el CRC com l'adreça del esclau.
3. Incorporar les dades a una nova petició Modbus RTU amb la següent estructura:
 - *Transaction identifier* (id de transacció): número seqüencial per a portar un control de la finestra de transmissió. No es obligatori. Pot ser zero.
 - *Protocol Identifier* (id de protocol). Actualment es zero, està reservat per a futures extensions.
 - *Length*, longitud de la trama incloent: identificador d'esclau, codi de funció i les dades.
 - *Unit ID*, identificador de l'esclau.
 - *Function Code*, codi de funció. Igual que el Modbus RTU.
 - *Data*, dades igual que el Modbus RTU.

Per a completar una comunicació Modbus TCP cal indicar que les transmissions es tenen que realitzar cap al port 502, reservat explícitament per a Modbus aplicacions.

2.1.7. Ethernet

El estàndard IEEE 802.3 comunament anomenat Ethernet és fins ara el més comú de les LAN (Local Àrea Network) cablejades. En aquest tipus de xarxes cada integrant es comunica mitjançant el protocol Ethernet i es connecta a la xarxa mitjançant equips anomenats Switch amb un enllaç punt a punt. Aquests

Switch disposen de diferents ports a cada un dels quals es pot connectar un integrant de la xarxa. El seu treball es el de retransmetre els paquets que li arriben per el port corresponent on es troba connectat el destinatari del missatge.

Les dades que s'envien entre els equips es transmeten mitjançant mitjans de transmissió guiats, que en el nostre cas seran cables de coure de par trenat. El par trenat consta de dos fils de coure aïllats, i trenats de forma helicoidal entre si, a la següent imatge podem veure aquest tipus de fil.

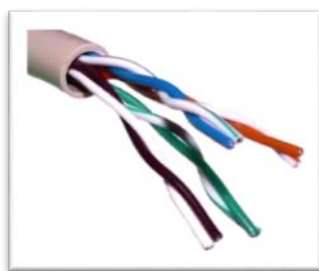


Figura 12: cable utp. Font wikipedia

El trenat es deuen a que un fil de coure recte, ens transforma en una antena simple, per lo que si es trenen les ones generades es cancel·len i el cable irradia amb menys força. El fil mostrat a la imatge 12 es un fil UTP (Par Trenat sense Blindatge) de 4 parells. D'aquests quatre parells el protocol Ethernet de 100Mbps (100BaseTX, mirar taula 1) emprà 2 parells, un per a cada direcció, es a dir un per a l'emissor i l'altre per al receptor, que conforma una comunicació full-duplex (sempre que s'utilitzin switch i no hub que puguin generar dominis de col·lisió) que permet la comunicació bidireccional al mateix temps.

Taula 5: tecnologies Ethernet. Font wikipedia

Tecnología	Velocidad de transmisión	de	Tipo de cable	Distancia màxima
10Base2	10 Mbit/s		Coaxial	185 m
10BaseT	10 Mbit/s		Par Trenat	100 m
10BaseF	10 Mbit/s		Fibra òptica	2000 m
100BaseT4	100 Mbit/s		Par Trenat (categoria 3UTP)	100 m
100BaseTX	100 Mbit/s		Par Trenat (categoria 5UTP)	100 m
100BaseFX	100 Mbit/s		Fibra òptica	2000 m
1000BaseT	1000 Mbit/s		4 pares trenats (categoria 5e ó 6UTP)	100 m
1000BaseSX	1000 Mbit/s		Fibra òptica (multimodo)	550 m
1000BaseLX	1000 Mbit/s		Fibra òptica (monomodo)	5000 m

Una vegada ja hem assentat els precedents del mitjà físic per a la transmissió de dades, ens queda fer una breu explicació del funcionament del enviament de les trames de dades entre els integrants de la xarxa.

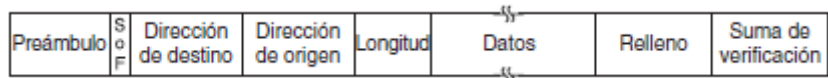


Figura 13: trama Ethernet. Font Tonenbaum. Redes de computadores

El preàmbul té una mida de 8 bytes on cada un conté el següent patró de bits: 10101010, amb l'excepció del últim byte que te aquesta forma: 10101011, que indica al receptor que està a punt de començar la transmissió. Les dues direccions que venen a continuació tenen un mida de 6 bytes cada una, cal remarcar sobre les direccions de origen i de destí es que s'envia la MAC (Media Acces Control), aquesta direcció es única globalment, es a dir no existeixen dues adreces iguals, d'aquesta forma es factible retornar el missatge al seu emissor corresponent. Si l'adreça de destí està conformada de la següent forma FF:FF:FF:FF:FF:FF vol indicar que es una adreça de multidifusió, es a dir que es transmetrà a tots els integrants de la xarxa. Els camps com longitud, dades i farcit no requereixen massa explicació. El que si resulta més interessant es la suma de verificació, ja que aquest tipus verificació l'emprarem en altres protocols.

2.1.8. RS485

Menys conegut és el seu nom oficial TIA-485-A es un estàndard que defineix la capa física de comunicacions mitjançant aquest protocol. Aquest tipus de xarxa està recomanada per a entorns amb molt de soroll i a distàncies llargues, ja que suporta distàncies fins a 1200 metres.

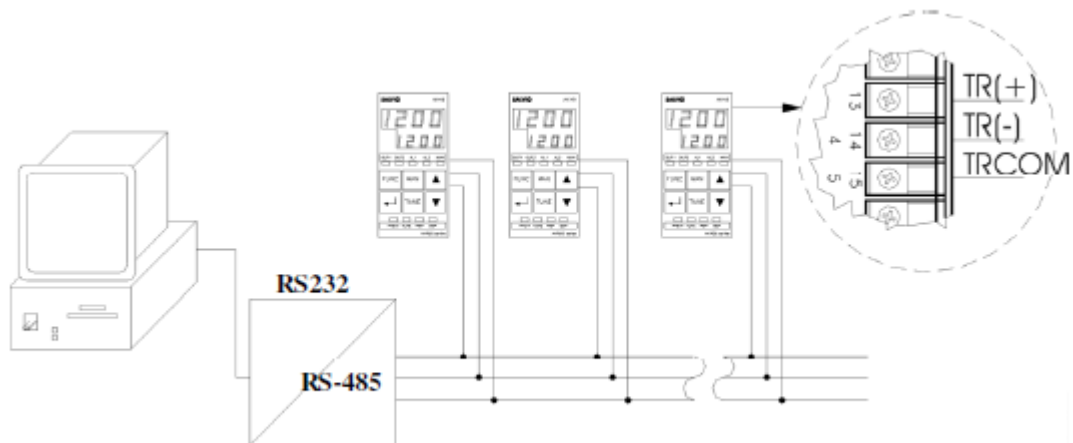


Figura 14: bus RS485. Font: comunicaciones Modbus Serie Akros

Lo més interesant d'aquest medi físic, és el baix cost d'implementació de la xarxa, ja que ofereix velocitats de 35Mbit/s fins a 10 metres, i 100 kbit/s en 1200 metres.

RS-485 permet implementar topologies de bus lineal, emprant únicament un parell de cables, cada equip (anomenat node, estació o dispositiu) penja de

eixe parell de cables, de forma que la seva desconnexió no implica el tall de la xarxa.

El protocol RS-485 únicament expressa les característiques elèctriques del emissor i del receptor, no implica cap protocol de comunicacions. A continuació es mostra un detall d'aquestes característiques.

TIA-485-A (Revisión del EIA-485)	
Patrón	ANSI / TIA / EIA-485-A-1998 Aprobado: 03 de marzo 1998 Reafirmó: 28 de marzo 2003
Medios físicos	Cable comunicadas Balanced
Topología de la red	Punto a punto, Multi-caer, de múltiples puntos
Dispositivos máximos	Al menos 32 unidades de carga
Distancia máxima	No especificado
Modo de operación	Los diferentes niveles del receptor: binario 1 (OFF) (Voa-VOB <-200 mV) binario 0 (ON) (Voa-VOB > 200 mV)
Señales disponibles	A, B, C
Tipos de conectores	No especificado

Figura 15: RS485. Font wikipedia

Per a finalitzar el bus de comunicació i sempre que la distància sigui prou important, cal incloure una resistència de 120 Ohms, per a tancar el circuit.

2.1.4. HTML, JavaScript, PHP i Ajax

HTML es un acrònim de Hyper Textual Markup Language¹², emprat per a la creació de pàgines web. El seu funcionament és simple: les parts que corresponen a diferents seccions de la pàgina o elements que tenen que rebre un tractament diferent van precedits per unes marques que indiquen de quin tipus són i com te que representar-les el navegador.

El publicador de la pàgina web puja un codi, en aquest tipus de marques al servidor web, quan un client fa una petició d'eixa pàgina el servidor envia tal qual el codi ha sigut escrit pel publicador, i es cada navegador web el que interpreta les marques i les representa. Tot i ser un estàndard gestionat per la W3C tots els navegadors no representen de la mateixa forma el codi, tot i

¹² Llenguatge de marques de hipertext

entendre de quin tipus de marca es tracta. Este exemple demostra eixa situació:

El següent codi representa una llista no ordenada dins d'un altra.

```
<html>
<head><title>Diferències de codi</title></head>
<body>

<h1>Llistat de cursos</h1>

<ul>
<li>1.0</li>
<li>2.0</li>
<li>
  <ul>
    <li>2.1</li>
    <li>2.2</li>
    <li>2.3</li>
  </ul>
</li>
</ul>
</body>
</html>
```

Ara veurem el seu comportament en diferents navegadors:

Taula 6: diferències entre navegadors

Firefox versió: 42.0	Chrome versió: 47.0.2526.106 m
Llistat de cursos <ul style="list-style-type: none">• 1.0• 2.0• <ul style="list-style-type: none">◦ 2.1◦ 2.2◦ 2.3	Llistat de cursos <ul style="list-style-type: none">• 1.0• 2.0• <ul style="list-style-type: none">◦ 2.1◦ 2.2◦ 2.3

Com es pot observar tot i que cada vegada les diferències són menors, si que existeixen i poden donar lloc a errades l'hora de representar les dades en diferents navegadors. Conforme el codi es torna més complex les diferències poden ser majors.

Com podem observar el codi escrit abans té poc que veure amb les impressionants pàgines web que es poder veure actualment, tot i estar en el

mateix llenguatge, calen instruccions addicionals per a donar-li una estructura més agradable.

```
<html>
<head>
<style>
ul#menu li {
    display:inline;
}
</style>
<title>Diferències de codi</title></head>
<body>

<h1>Llistat de cursos</h1>

<ul id="menu">
<li>1.0</li>
<li>2.0</li>
<li>
    <ul>
        <li>2.1</li>
        <li>2.2</li>
        <li>2.3</li>
    </ul>
</li>
</ul>
</body>
</html>
```

Per a poder generar eixes noves característiques cal implementar un afegit anomenat CSS, acrònim de Cascade Style Sheet¹³.

El CSS ens permet indicar-li al navegador mes dades de com mostrar el codi HTML descarregat, apliquem ara al codi anterior una mica de CSS:

La nova secció de codi que hem inclòs¹⁴, emmarcada dins les etiquetes `<style></style>` li indica al navegador que el element és una llista no ordenada (marca `Llista`) que s'identifica amb el nom únic de *menu* es mostrarà en la mateixa línia, o sigui en horitzontal, els resultats són els següents:

¹³ Full de estils en cascada

¹⁴ No cal incloure el tot el CSS a la pàgina, se li pot indicar una adreça d'on descarregar el CSS

Taula 7: llista CSS

Firefox versió: 42.0	Chrome versió: 47.0.2526.106 m
Llistat de cursos	Llistat de cursos
1.0 2.0	1.0 2.0
2.1 2.2 2.3	2.1 2.2 2.3

Com podem observar, tornen a existir diferències. CSS tot i estar definit també per la W3C també es representat per els navegadors de forma diferent, inclús hi ha opcions que no són representables per diferents navegadors.

Actualment la versió actual de HTML es la versió 5 coneguda com HTML5 que ofereix funcionalitats tals com la reproducció de vídeo sense la necessitat de codi de tercers o una definició més extensa de tipus de marques per a afavorir la web semàntica. De CSS anem per la versió 3, que ofereix funcionalitats com transparència d'objectes una millor reorganització dels elements a la pàgina, ect.

Ara se ens presenta uns altres problemes, com són els diferents tipus de dispositius com tabletas o telèfons mòbils que actualment són uns grans consumidors de pàgines web i per altra banda el navegadors no actualitzats.

La part menys complicada, però mes tediosa es la retro compatibilitat. Per a ajudar hi ha eines que ens permeten ajustar el codi a versions mes antigues i només caldria incloure en les capçaleres que depenent de quina versió de navegador emprès utilitzat una pàgina de CSS o un altra. Per això es una bona forma de treballar separar el codi CSS de la pàgina en HTML, ja que el codi és més fàcil de mantenir.

Per altra banda tenim els nous formats de visualització, o lo que es diu actualment, que la web sigui *Responsive Desing*, es a dir sensible als dispositius en la que se exposa, adequant la seva mida. Existeixen diferents formes de fer-ho, però la més habituals són les *media quèries*.

Les *media quèries*, són una regla de CSS que permet seleccionar unes regles CSS depenent de la mida del dispositiu on es representa, el navegador quan rep la pàgina interpreta eixa regla i aplica una o altra fulla o regla CSS depenent de la seva mida de pantalla, o més concretament de la mida de la finestra, ja que si tenim el navegador a una mida que no és la de pantalla completa en un equip d'escriptori, la pàgina es sensible i s'adaptarà a les noves mesures, per a facilitar la seva lectura.

Com hem dit al principi el que ens envia el servidor web no deixa de ser un text remarcat amb etiquetes que el nostre navegador interpreta de forma estàtica i a cada nova acció cal fer una nova petició al servidor web, descarregant la pàgina web de forma completa altra vegada, opció que consumeix recursos

innecessàriament ja que si per exemple únicament volem que ens aparegui una finestra d'accés o verificar si un camp de un formulari està buit. Esta limitació també ha sigut superada mitjançant 2 tecnologies, JavaScript i Ajax.

Començarem per JavaScript. Tot i semblar-se en el nom al altre llenguatge de programació, a JAVA¹⁵ no es pareix pràcticament en res. JavaScript és un llenguatge interpretat de banda del client, es a dir el navegador descarrega del servidor, de dins de la pàgina web o be en pàgines accessòries com en CSS, el codi i el navegador l'executa. JavaScript pot realitzar un fum de funcions que van des de canviar les propietats CSS d'un objecte, per exemple que estigues ocult i degut a una acció de l'usuari mostrar-lo o realitzar operacions matemàtiques per a comprovar que una data es correcta fins i tot incloure nou codi a la pàgina web. El seu llançament es va fer en 1997 mitjançant el estàndard ECMAScript 1, actualment en juny del 2015 es va publicar ECMAScript 6 que es la versió actual. Avui en dia fins i tot els navegadors de dispositius mòbils suporten JS¹⁶. Gràcies a aquest suport i la seva potència i flexibilitat s'han desenvolupat llibreries com JQuery¹⁷ que permet manipular codi HTML de forma senzilla i creat animacions molt interessants com les ja clàssiques finestres modals. Altres llibreries emprades al projecte són ChartJS¹⁸ que permet la creació de gràfiques amb un format visual molt agradable per a l'usuari.

Però continuem amb el mateix problema, per a enviar per exemple una consulta que ens torni un simple número cal demanar tota la pàgina web a l'hora i descarregar-la completa. Açò es soluciona mitjançant Ajax.

Ajax acrònim de *Asynchronous JavaScript and XML*¹⁹ d'una forma breu direm que el que fa el sistema es enviar una consulta al servidor web, o be per la petició del usuari d'una consulta concreta, el servidor web la respon, però no respon en forma de pàgina web completa, si no que respon en format XML, d'aquesta forma la resposta pot ser tractada mitjançant JS i així actualitzar contingut de la pàgina depenent de la resposta del servidor sense tenir que descarregar-la de forma sencera. Lo que s'aconsegueix mitjançant aquesta tècnica són pàgines amb contingut dinàmic i un menor consum d'ample de banda i de recursos tant del client com del servidor.

Concretament al nostre projecte hem emprat AJAX, que enlloc d'obtenir la resposta del servidor en format XML, la rebem en JSON que també podrà ser tractada pel nostre navegador.

Ja tenim controlada la part del client, podem visualitzar la pàgina web, fer peticions de forma asíncrona i síncrona, però hi ha un darrer problema: el HTML es estàtic.

HTML no deixa de ser un fitxer de text que no es pot modificar, esta opció pot ser interessant per a visualitzar documents que no es tenen que modificar, però nosaltres volem llegir dades que canvien cada pocs segons. Solució: PHP.

PHP (acrònim de *Hypertext Preprocessor*) és un llenguatge de programació, interpretat de banda del servidor, es a dir l'executa el servidor i no cal compilar. Quan nosaltres enviem una petició al servidor del tipus <http://www.mipagina.com/index.php> li estem preguntant al servidor de

¹⁵ [https://es.wikipedia.org/wiki/Java_\(lenguaje_de_programaci%C3%B3n\)](https://es.wikipedia.org/wiki/Java_(lenguaje_de_programaci%C3%B3n))

¹⁶ Acrònim de JavaScript

¹⁷ <https://jquery.com/>

¹⁸ <http://www.chartjs.org/>

¹⁹ JavaScript i XML asíncron

mipagina.com per una pàgina anomenada index.php. El servidor al rebre esta petició executa el codi PHP que hi ha en la pàgina, per exemple una consulta a una base de dades, espera la resposta i la retorna al codi PHP. La funció darrera de PHP es retornar, la majoria de vegades, codi HTML perfectament format, però amb les noves dades injectades on correspongui en la pàgina.

Enlloc de que retorni un codi HTML podem fer que faci accions, com per exemple esborrar un fitxer del servidor o pujar-lo, enviar un correu electrònic o retornar unes dades en format XML o JSON.

PHP va actualment per la versió 5.5 i en la seva pàgina web²⁰ ofereix tota la documentació de l'immens conjunt de llibreries que el conformen, així com exemples pràctics.

2.2. Estudi de mercat

Durant la introducció d'aquest document ja es parla de preus i de lo car que és desenvolupar una eina d'anàlisi de xarxes mitjançant software comercial i les limitacions d'ús que aquest tipus de programes ofereixen, per lo que tampoc crec que sigui necessari tornar a explicar-lo ací. Tampoc entrarem en la diferència de preu d'un PC industrial enfront de una Raspberry Pi, on als dos casos el nostre producte te avantatges evidents.

El que ara toca plantejar es si seria viable comercialment la producció d'aquest tipus de producte, es a dir introduir un producte que el seu nucli principal s'executa en un equip que funciona mitjançant el carregador del mòbil i que el seu maquinari s'empra tant com a centre multimèdia²¹ o per a emular jocs antics de recreatius²².

La resposta pot estar a la història (o millor dit llegenda²³) recent de l'enginyeria. Es comenta que la NASA en plena cursa espacial amb els russos, volia desenvolupar un bolígraf per que els seus cosmonautes de les missions Apollo pogueren emprar-lo en gravetat zero, el cost del projecte es va disparar i al final abandonaren el projecte, mentre que la solució dels russos va ser simple i econòmica, emprar un llapis.

Si be la història no és del tot real serveix per a fixar una màxima en l'enginyeria, la simplicitat es la sofisticació definitiva (Leonardo DaVinci). No tot el mon pot o vol acceptar aquesta màxima, i mes avui en un mon tan comercial com l'actual on el sinònim de marca es sinònim de fiabilitat, on el major repte es que el client final accepti un producte nou i desconegut.

Per això caldria centrar-se no en els beneficis econòmics directes d'un producte front a un altre, que és un punt molt fort, sinó als beneficis accessoris del producte.

El projecte no està pensat per a que empreses que ja tenen un sistema plantejat migrin al nostre, perquè és Linux dins d'una Raspberry que es comunica amb un Arduino, que és lo màxim en software lliure. No l'idea d'aquest projecte és un altra. Poder que les petites empreses siguin capaces de dur a terme Sistemes de Gestió energètica, a un cost molt reduït, el que directament els afectarà augmentant els beneficis econòmics, i també i lo més

²⁰ <https://secure.php.net/manual/es/index.php>

²¹ <http://openelec.tv/get-openelec>

²² <http://blog.petrockblock.com/retropie/>

²³ <http://www.physics.org/facts/apollo-nasa-pen.asp>

important del projecte, reduir la empremta ecològica que deixa eixa empresa i que si que ens afecta a tots.

Jo crec que es un producte orientat a empreses que volen fer alguna cosa, o be per a millorar la seva competitivitat o be preocupats per el mediambient i que no poden arribar a destinar recursos suficients a aquest propòsit.

Una altra bondat de SION és que es multiplataforma, està escrit en Python, que es pot executar tant en una Raspberry com en un PC actual, amb uns requisits mínims. El mateix passa amb MySQL, Lighttpd i PHP que es poden executar en qualsevol sistema operatiu comercial, per lo que el sistema pot ser implementat en possibles clients que desconfien de productes que funcionin amb carregadors de mòbils.

3. Descripció funcional

El SION (Sistema de Informació i Obtenció de Notificacions) es un sistema que permetrà la captura de dades de diferents analitzadors de xarxes, en concret i per a aquest cas emprarem un model concret que hem utilitzat per a les proves, així com de tots els components de hardware i programari que formaran part del projecte. La solució que proposem és una Raspberry Pi que realitza consultes de forma seqüencial als analitzadors instal·lats. Aquesta informació es desa de forma permanent en una estructura de dades que representa la informació (voltatge, intensitat, THD, consum..) de cada equip. El temps entre peticions dependrà del nombre d'equips connectats a la xarxa. Per a poder obtenir un històric de les dades per a un posterior anàlisi, es desarà la resposta de cada equip si es vàlida en una base de dades (en MYSQL). Aquest temps ens permetrà estudiar les fluctuacions energètiques en un moment de temps concret. Es recomana un mínim de 4 mesures per hora, cada 15 minuts, per a obtenir unes 100 mesures per equip al dia. Cal que ens aturem en aquest punt ja que tenim que recordar que estem mesurant consum energètic, així com alguns valors que indiquen la qualitat de la senyal, no pas estem analitzant la qualitat del sistema de subministrament energètic, ja que pel nombre d'equips emprats en un sistema de captura industrial (més de 30 analitzadors) no es poden obtenir mesures en temps real, per lo que el nombre de captures de dades així com la seva freqüència no és gaire important, que el que importa es obtenir un acumulat per interval de temps. Cada vegada que desem a la base de dades, s'obtindrà el consum de cada equip analitzat. El sistema té que ser capaç de generar gràfiques dels diferents equips, en concret la versió avaluada realitza gràfiques de corrent, recuperant les dades de la base de dades.

En un primer grup tenim els paquets de comunicació que extrauen les trames TCP i Ethernet i ens retornen un codi hexadecimal que el nostre sistema pot tractar. Aquest software pertany al propi driver de comunicació de la Raspberry. Una vegada extret el codi hexadecimal de la consulta, aquest passa al programa en Python que és el nucli central de tot el sistema. Aquest programa s'encarrega de crear les consultes als diferents analitzadors, tractar les respostes, transformant-les en informació que s'encarregarà de desar en la base de dades MySQL. Una vegada les dades han sigut desades al SGBD, podrà ser consultat mitjançant el propi servidor web de la Raspberry.

Aquest punt és el que cal explicar de forma més concreta. L'avantatge principal de fer-ho mitjançant un servidor web i consultes en Ajaj (de forma asíncrona) es que el sistema sigui multiplataforma, multidispositiu i permeti concurrència i a

banda podrem millorar el nostre apartat de disseny. SION no és simplement un Microcomputador que realitza consultes de forma seqüencial a una base de dades, es un sistema de comunicacions.

Degut a les distàncies entre els equips emprarem dos estàndards de comunicació: Ethernet i RS485. Emprarem el primer per a comunicar la nostra Raspberry Pi amb els adaptadors Modbus TCP a Modbus RTU. La pregunta lògica seria: perquè cal dos protocols diferents si tenen que fer el mateix? La solució és ben senzilla, la reducció de costos. La instal·lació de per exemple 32 analitzadors amb cable UTP requeria 32 cables UTP, un per cada analitzador, més a banda els cost d'instal·lació d'aquest cables. Mentre que mitjançant RS485 amb un únic parell de cables podem connectar els 32 al mateix adaptador. Aquests adaptadors podem ser comercials o be podem fabricar-ne un nosaltres mateix mitjançant una placa Arduino i la seva Shield Ethernet que ofereix comunicació a la nostra placa.

Descompondrem la petició que es fa mitjançant Modbus TCP al adaptador i reenviar-la mitjançant algun dels adaptadors disponibles al mercat que ofereix comunicació RS485 a una placa Arduino. La resposta dels Esclaus Modbus seria tractada de la mateixa manera, arribaria la resposta (o esgotaria el temps d'espera) i la placa reenviaria mitjançant Ethernet la resposta al seu emissor. L'esquema de funcionament es pot veure a la figura 5.

3.1. Nucli SION

El nucli SION controla les comunicacions entre els diferents analitzadors que hi poden haver a la xarxa, captura les dades i les desa a la base de dades per a posteriorment ser revisades per l'usuari.

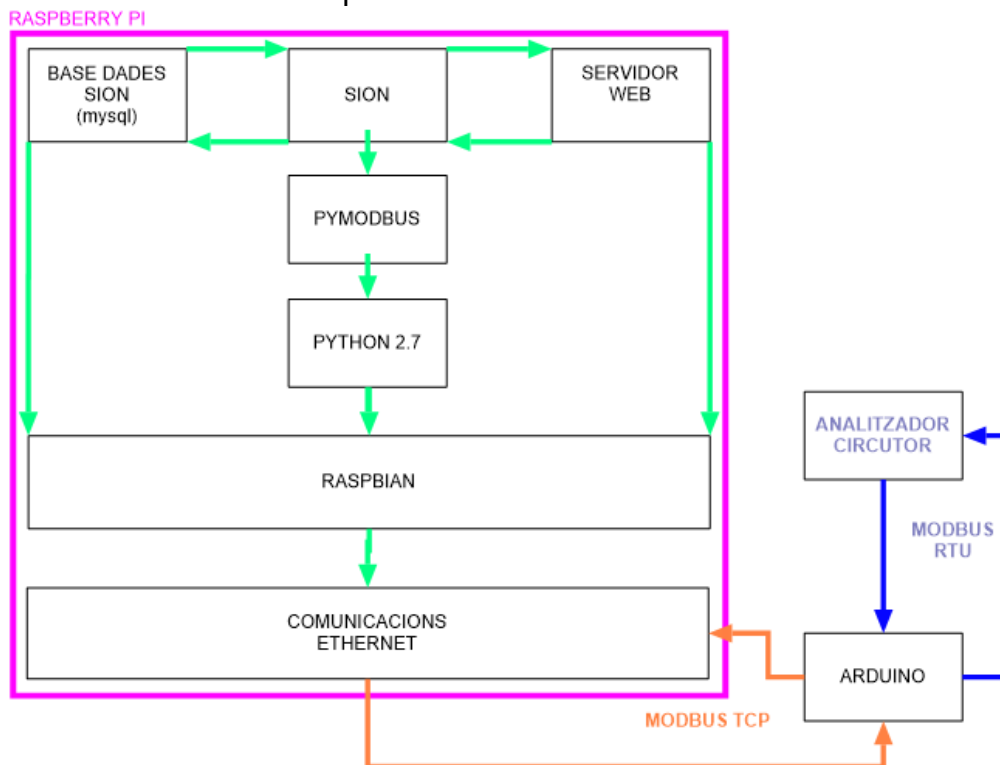


Figura 16: diagrama de blocs SION

La arquitectura de xarxa seria la següent:

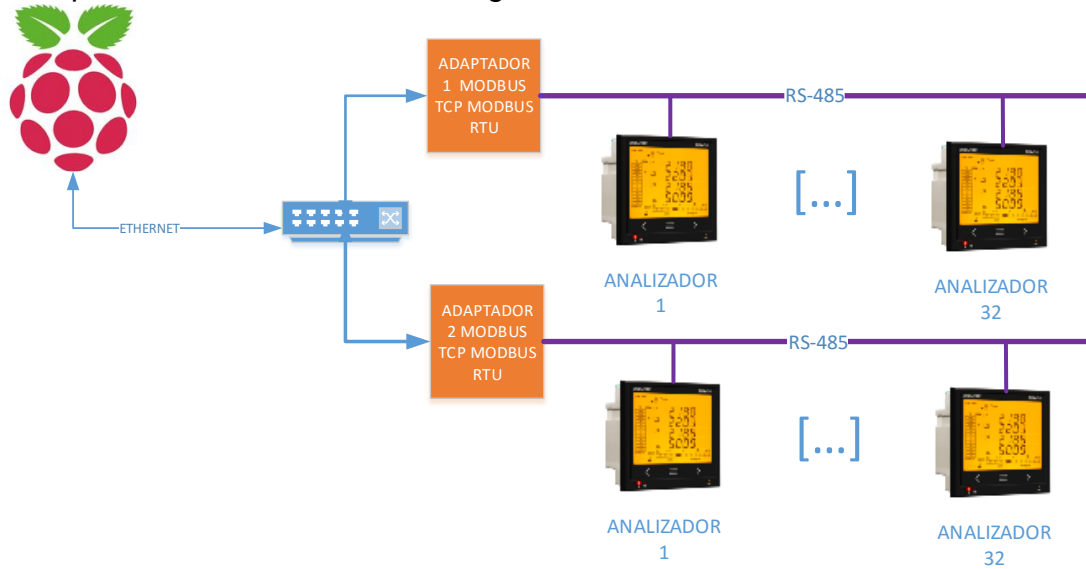


Figura 17: arquitectura SION

La Raspberry Pi, representada l'esquema mitjançant el seu logo, rep per la connexió Ethernet de la que disposa tant les respostes dels adaptadors Modbus com les peticions de clients demanant la visualització dels resultats.

La interacció entre els diferents equips de la xarxa seria la següent:

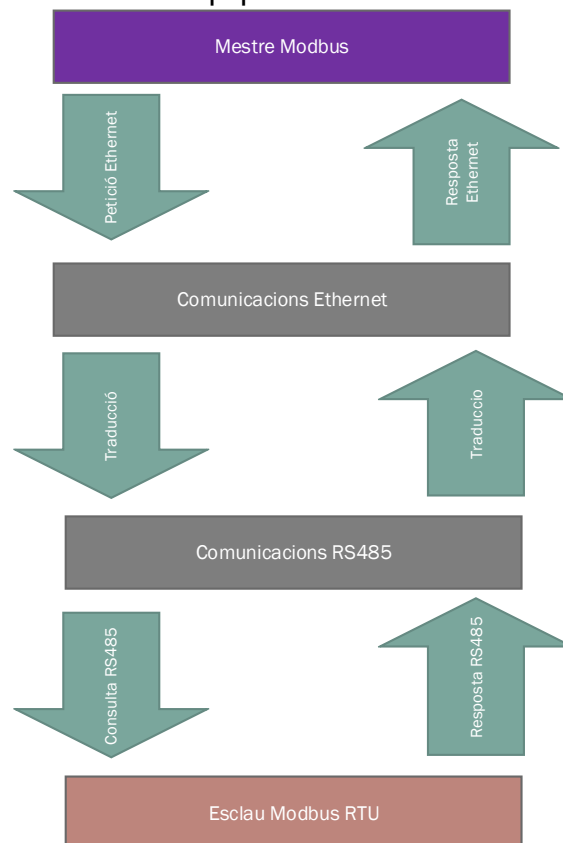


Figura 18: interacció dels equips

En el nostre cas el mestre Modbus serà la nostra Raspberry Pi, que llança peticions de dades mitjançant Ethernet i Modbus TCP. Posteriorment les trames es retransmeten ja en el protocol correcte gràcies a la passarel·la

Arduino, qui envia la petició de dades l'esclau corresponent. L'esclau respon i tot el procés es produeix a l'inrevés.

3.2. Visualització de dades

Enlloc de crear una interfície d'usuari gràfica per a que funcionés directament sobre l'equip en el que corre el nucli de SION s'ha optat per un entorn web que garanteix que la visualització de les dades pugui ser realitzada per qualsevol tipus de navegador amb accés a la xarxa local on s'executa el nostre producte. Per a garantir un accés segur de fora de la LAN a l'aplicació, enlloc de redirigir el port 80 i donar-li accés a internet, per a millorar la seguretat, ha optat per establir una VPN, d'aquesta forma la informació estaria doblement assegurada, per una banda amb el registre d'usuari a la pròpia pàgina web i per altra banda un accés segur i xifrat mitjançant una connexió VPN.

El diagrama de blocs del producte resultant seria el següent:

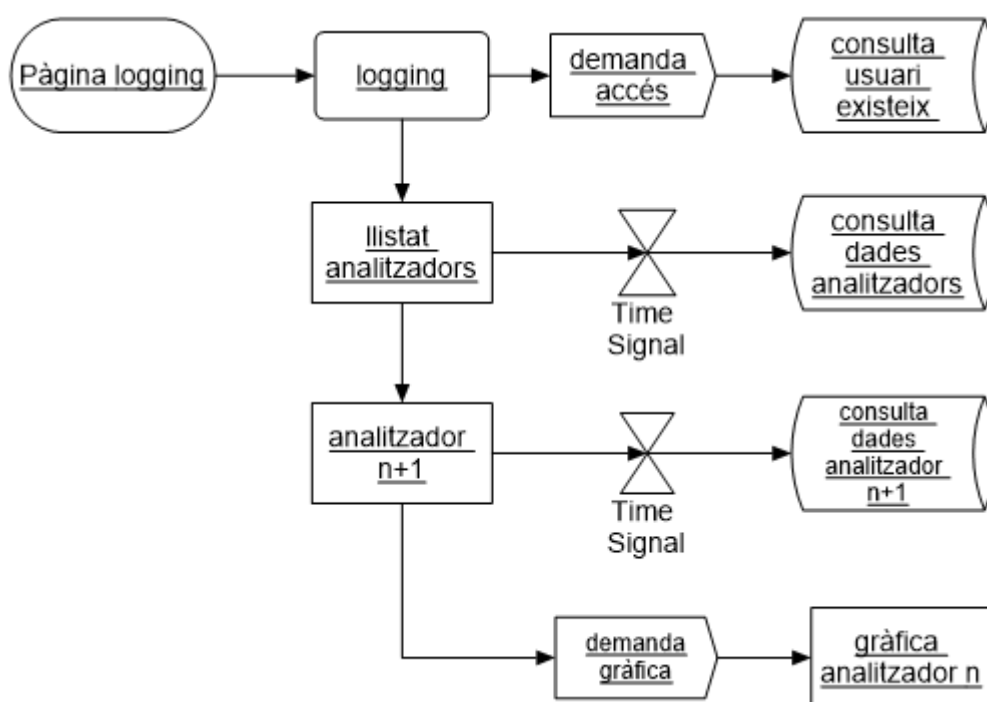


Figura 19: diagrama entorn web

4. Descripció detallada

A continuació es procedirà a detallar el funcionament del producte SION. La descripció es farà en diferents parts. Les parts són les següents:

- Descripció de la base de dades.
- Descripció del nucli SION.
- Descripció de la passarel·la Modbus TCP – Modbus RTU
- Descripció del entorn web.

4.1. Base de dades

Una base de dades permet desar dades d'una forma que posteriorment es pot recuperar i mitjançant un tractament darrer transformar-la en informació.

El més important de esta base de dades no es la seua capacitat relacional, si no la velocitat d'accés a les dades i sobre tots que consumirà pocs recursos. Per aquests motius i tal i com s'exposa a l'apartat 2.1.4 s'ha optat per MySQL com a gestor de base de dades.

L'estructura de la taula es molt senzilla, amb 1 taula per als usuaris, 1 taula per al llistat dels analitzadors existents, i enlloc de desar totes les dades en una única taula, s'ha optat per desar les dades de cada analitzador amb una taula pròpia per a d'ell. Les taules no estan relacionades, ja que les dades estan separades per taules independents tal com mostra el seu disseny.

listado analizadores	users	analizador_1
id	user_id	id lectura
id_esclavo	email	dateTime
IP_esclavo	password	tension_fase_1
nombre	nick	corriente_fase_1
activado		tension_fase_2
enFallo		corriente_fase_2
enAlarma		tension_fase_3
estado		corriente_fase_3
		energiaActiva
		energiaReactivaInductiva
		energiaReactivaCapacitativa
		energiaAparente
		frecuencia
		tension_L1_L2
		tension_L2_L3
		tension_L1_L3
		THD_V_L1
		THD_V_L2
		THD_V_L3
		THD_A_L1
		THD_A_L2
		THD_A_L3
		potenciaActiva
		potencialInductiva
		potenciaCapacitativa
		cosenoPhi
		factorPotencia

Figura 20: taules base dades SION

Per a reduir al màxim la mida de cada inserció a la base de dades, cada atribut de cada taula està conformat d'acord a les dades que te que contenir i després mostrar per a tenir que realitzar les menors modificacions necessàries. L'annex 3 mostra en SQL el disseny de la base de dades així com tipus de dades de cada taula.

Cada vegada que es realitza una inserció a la taula de cada analitzador, es procedeix a crear una clau primària i única que identifica la inserció, aquest identificador no es més que un nombre sencer amb auto increment. A l'hora es crea una marca temporal a cada camp, que ens servirà després per a poder recuperar la informació en posteriors consultes.

4.2. Nucli SION

Hem optat per descriure primer la base de dades ja que SION treballa directament sobre ella, i es part fonamental del procés.

En primer lloc mostrarem un diagrama de flux del funcionament del sistema.

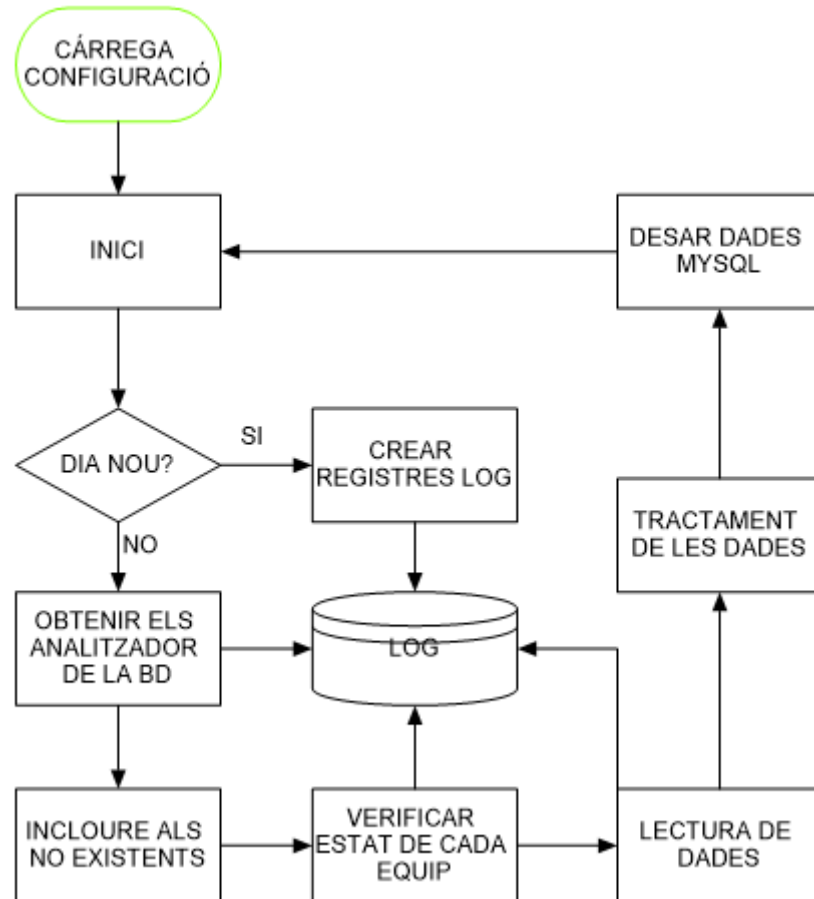


Figura 21: diagrama de flux SION

El primer que fa el sistema quan inicia és carregar la configuració com en tots els programes, carregar llibreries i inicialitzar variables.

Seguidament el primer que fa és avaluar el dia actual o si es la primera arrancada, si és així els que fa es crear uns nous diaris d'activitat: un per al nucli, un per la base de dades i un tercer per a les llibreries de comunicacions Modbus. Estos tipus de diàries provenen d'una llibreria bàsica de Python anomenada 'logging' que no sols permet crear logs per pantalla, si no que també permet crear logs en el disc dur i personalitzats. Com a apunt direm que esta llibreria es capaç d'enviar un correu electrònic amb les dades de l'alarma al correu configurat. Aquesta llibreria disposa de distints nivells de mostra de missatges:

- **DEBUG:** el més baix, desaria totes les accions i els seus resultats, consultes a base de dades...
- **INFO:** informació general del funcionament del programa.
- **WARNING:** informa de situacions anòmales que no poden generar aturada del programa.
- **ERROR:** fallada que pot provocar un tancament del programa.

- CRITICAL: situació crítica insalvable.

Des de l'interior del codi del programa es possible modificar aquests paràmetres, per defecte el nucli SION està configurat en ERROR²⁴.

Una vegada ha finalitzat la creació dels logs, desa la data actual en una variable. En cada cicle verifica que el dia es el mateix que te gravat, si no és així torna a crear nous fitxers de log per al dia actual.

Seguidament va a la base de dades i li fa una consulta preguntant-li les dades de tots els analitzadors existents i per a cada analitzador existent crea un objecte de la classe Analitzador. Aquesta classe conté els següents atributs i herències:

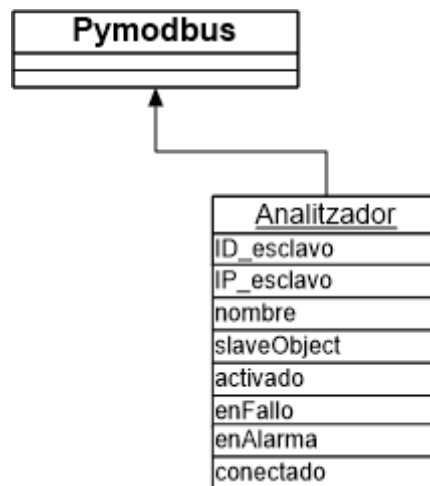


Figura 22: classe Analitzador

La classe Analitzador té pràcticament els mateixos atributs que la taula de MySQL de cada analitzador, a excepció de SlaveObject que genera l'herència de PyModbus, i l'atribut connectat.

Abans de crear cada analitzador recorre un vector d'objectes de la classe Analitzador, si eixe analitzador no existeix en el vector, crea un objecte d'aquest tipus i l'afegeix al vector.

Aquesta funció es fa cada 15 segons o cada vegada que es connecta per primera vegada el programa SION.

Una vegada ha finalitzat de comprovar les dades de la consulta a la base de dades, torna a recórrer el vector per a comprovar si tots els analitzadors estan connectats. Si no ho estan intenta connectar-los, en cas de que no pugui fica el analitzador en fallada, ficant a 1 l'atribut 'enAlarma', en cas de que es pugui connectar fixa el seu valor a 0.

També pot succeir que l'usuari hagi decidit desconnectar l'analitzador de la xarxa, en eixe cas tancaríem la comunicació de l'objecte i esborraríem totes les alarmes del mateix. Aquesta es la única forma d'interacció que te l'usuari amb el sistema com es pot observar al següent diagrama:

²⁴ El sistema de logs es ascendent,es a dir si el marques com a DEBUG, mostrarà les accions de DEBUG més la INFO, més els WARNINGS i els ERRORS.

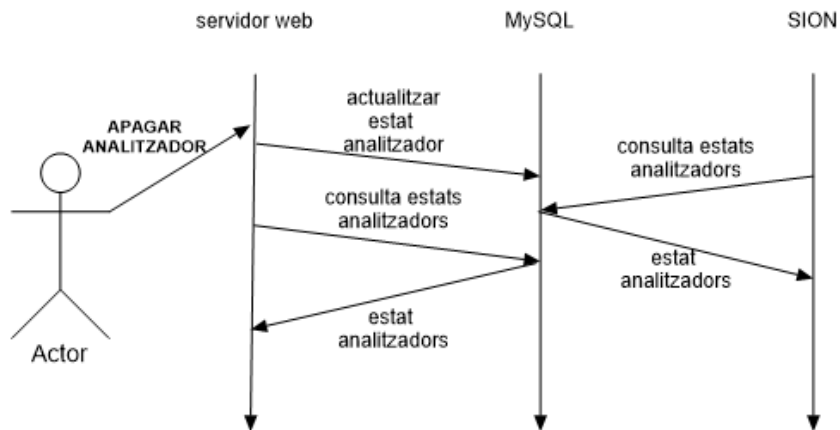


Figura 23: diagrama estat analitzador

Les comunicacions són asíncrones, amb temps de reacció que podem oscil·lar depenent del nombre d'analitzadors connectats, però aquest tema el tractarem més endavant.

Si tenim com a mínim 1 analitzador actiu, es pot iniciar un cicle de lectura. Per aquest sistema hem comptat que tots els analitzadors són iguals i disposen dels mateixos mapes de memòria on dirigir les peticions Modbus.

En cas de que no fos així el sistema es podria adaptar a diferents mapes de memòria preestablerts i depenent del tipus d'analitzador sol·licitar unes adreces o unes altres.

Les peticions de lectura es fan de forma síncrona, es fa una petició i s'espera la resposta, es fa així per una única causa. El sistema es un sistema híbrid, es a dir treballa amb dos protocols de comunicació que empen dues capes físiques de comunicació distintes, Modbus Ethernet que és full dúplex, i Modbus RTU que treballa sobre un par de coure simple, per lo que llançar moltes peticiones no te sentit si només l'esclau pot atendre una a la vegada.

Es un punt delicat aquest, ja que els timeout del protocols sèrie són molt lents, prop d'un segon cada petició, i el sistema reintenta 3 vegades cada una en cas de timeout. Per lo que un sistema amb analitzadors que no responguin es torna molt lent.

No sols les peticions de lectura de cada analitzador es fan de forma cíclica, ni no també les lectures de tots els equips, es a dir, agafem l'analitzador 1 i fem les seves peticions de lectura, quan acaba de respondre o esgota els timeout, passem a l'analitzador 2 que fa les peticions de lectura i així de forma seqüencial fins que esgota tots els analitzadors connectats del vector.

Es podria haver programat un sistema per fils que canviés l'estat de cada analitzador i un altre fil que executés les ordres de lectura, però el temps de modificació dels equips haguera sigut el mateix, ja que fins que els fils no acabaren no es podrien sincronitzar les dades.

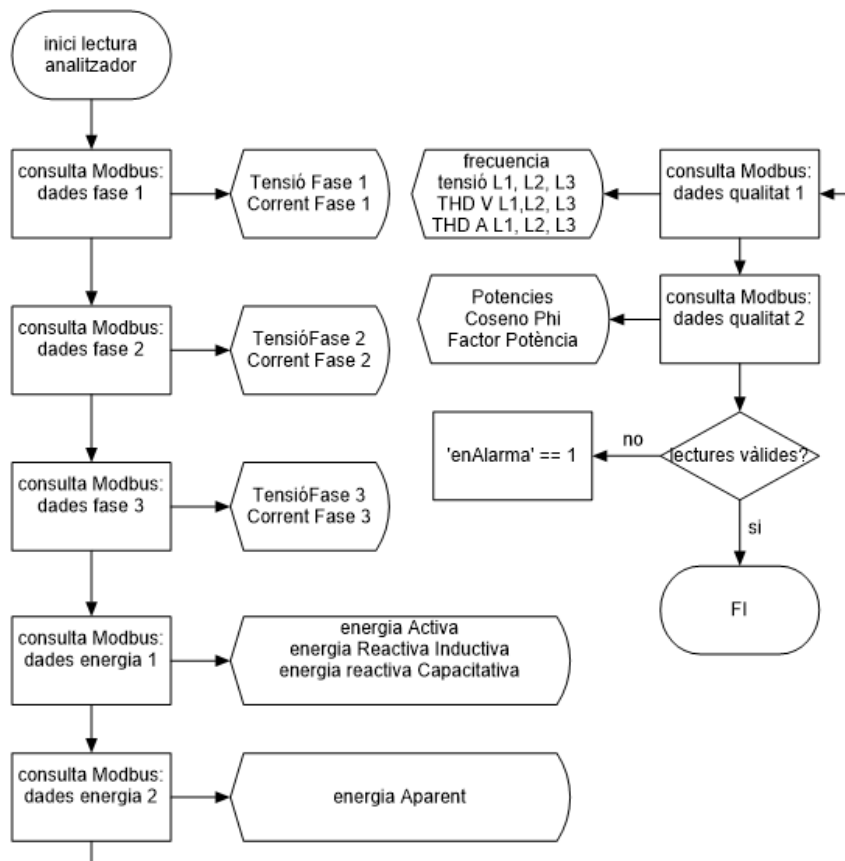


Figura 24: diagrama de flux lectura dades

Previ a desar les dades en la taula de MySQL corresponent, cal tractar-les ja que no es poden desar directament, ja que no tenen el format correcte. Quan es fa una consulta a un esclau Modbus torna els registres amb la mida d'una paraula, es a dir 16 bits, però cada registre de l'analitzador està compost per una doble paraula o sigui 32 bits, si volem representar nombres menors a 32767 (16 bits amb signe) no tenim cap problema i no caldria fer cap tractament, però en la majoria dels casos si que cal, per exemple el consum ve donat en un nombre sense signe de 32 bits o sigui un valor màxim de 4.294.967.295 w si que cal compondre la paraula de 32 bits.

No és que només el nombre ve partit en dos paraules, si no que cal fer un intercanvi per a que el nombre sigui vàlid. El que fem es desplaçar 16 bits el que seria la part alta de la paraula i després sumar-li la part baixa d'acord a la imatge següent:

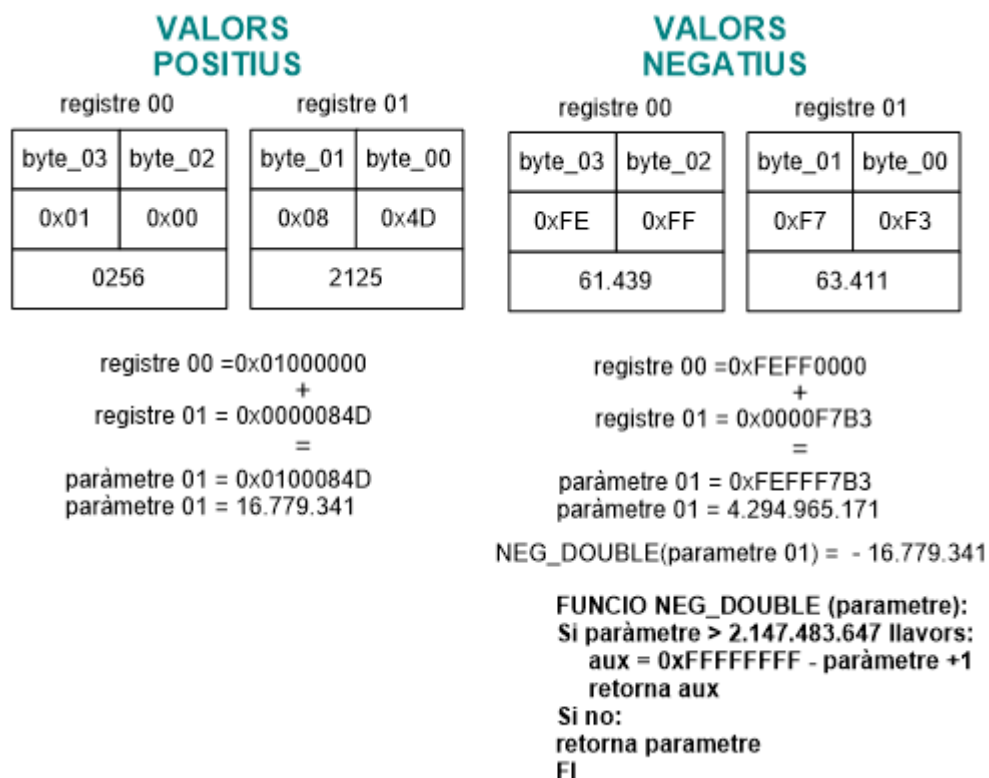


Figura 25: tractament de registres

Abans de poder desar les dades a la base cal fer un últim tractament d'elles, i es donar-li les unitats corresponents. La taula de memòria del analitzador tracta les dades de forma que retorni nombres sencers, es a dir si el valor que mostra per pantalla per exemple es de 221,5 volts l'analitzador el desa com a 2215 volts. A continuació es mostra una taula amb les modificacions fetes a cada camp.

Taula 8: tractament de les unitats

Magnitud	Unitats	Modificació
Tensió Fase 1	V x 10	Dividir per 10
Corrent Fase 1	Mili ampers	Dividir per 1000
Tensió Fase 2	V x 10	Dividir per 10
Corrent Fase 2	Mili ampers	Dividir per 1000
Tensió Fase 3	V x 10	Dividir per 10
Corrent Fase 3	Mili ampers	Dividir per 1000
Potència Activa	w	
Potència Inductiva	w	
Potència Capacitativa	w	
Cossen Phi	X 100	Dividir per 100
Factor de Potència	X100	Dividir per 100
Freqüència	Hz x 10	Dividir per 10
Tensió Línia L1 – L2	V x 10	Dividir per 10
Tensió Línia L2 – L3	V x 10	Dividir per 10
Tensió Línia L1 – L3	V x 10	Dividir per 10
THD corrent i tensió	% x 10	Dividir per 10

Cal sincronitzar les dades ja que SION té que tornar el feedback de l'estat cap a la base de dades per a que l'usuari pugui tenir constància del estat del equip. Este estat s'actualitza cada 10 segons com a mínim. L'estat d'un analitzador pot variar durant l'execució d'una lectura, si aquesta lectura es realitza de forma anòmala o no hi ha resposta de l'esclau, SION canviarà l'estat de l'objecte analitzador ('enAlarma' = 1), en el moment de la fallada de la lectura. Després durant l'actualització de l'estat de l'analitzador modificarà l'estat del equip a la base de dades ('estado' = 5). Seguidament veurem un esquema de funcionament de la actualització de l'estat:

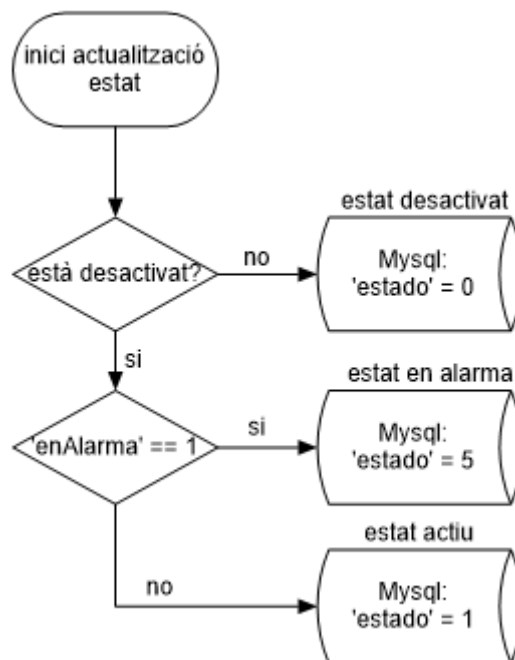


Figura 26: diagrama de flux actualització estat

4.3. Arduino

En aquest moment ja som capaços d'enviar trames Modbus TCP per la xarxa cap a algun esclau Modbus concret, però el que succeeix es que no tenim cap esclau Modbus TCP. Els nostres esclaus són Modbus RTU.

Per això cal transformar la trama que envia la Raspberry a un protocol físic de comunicacions que els nostres esclaus siguin capaços d'entendre i respondre, i una vegada han respost, recuperar eixa resposta i traduir-la de forma que la Raspberry sigui capaç de tractar-la.

Aquesta es la feina del Arduino, la de traductor de protocols, deu de traslladar de Modbus TCP cap a Modbus RTU.

No ens aventurarem en aquest punt a parlar de la connexió de maquinari necessari per a dur a terme aquesta tasca, ja que a l'Annex 2 es troba tota la documentació necessària per a replicar el dispositiu. En centrarem en el programari necessari per a que funcioni.

La figura 27 mostra el diagrama de flux del funcionament de la passarel·la, com es pot observar el seu funcionament és autònom es a dir no requereix de cap tipus d'interacció amb cap usuari, amb la excepció es clar, de les peticions de trames de comunicacions.

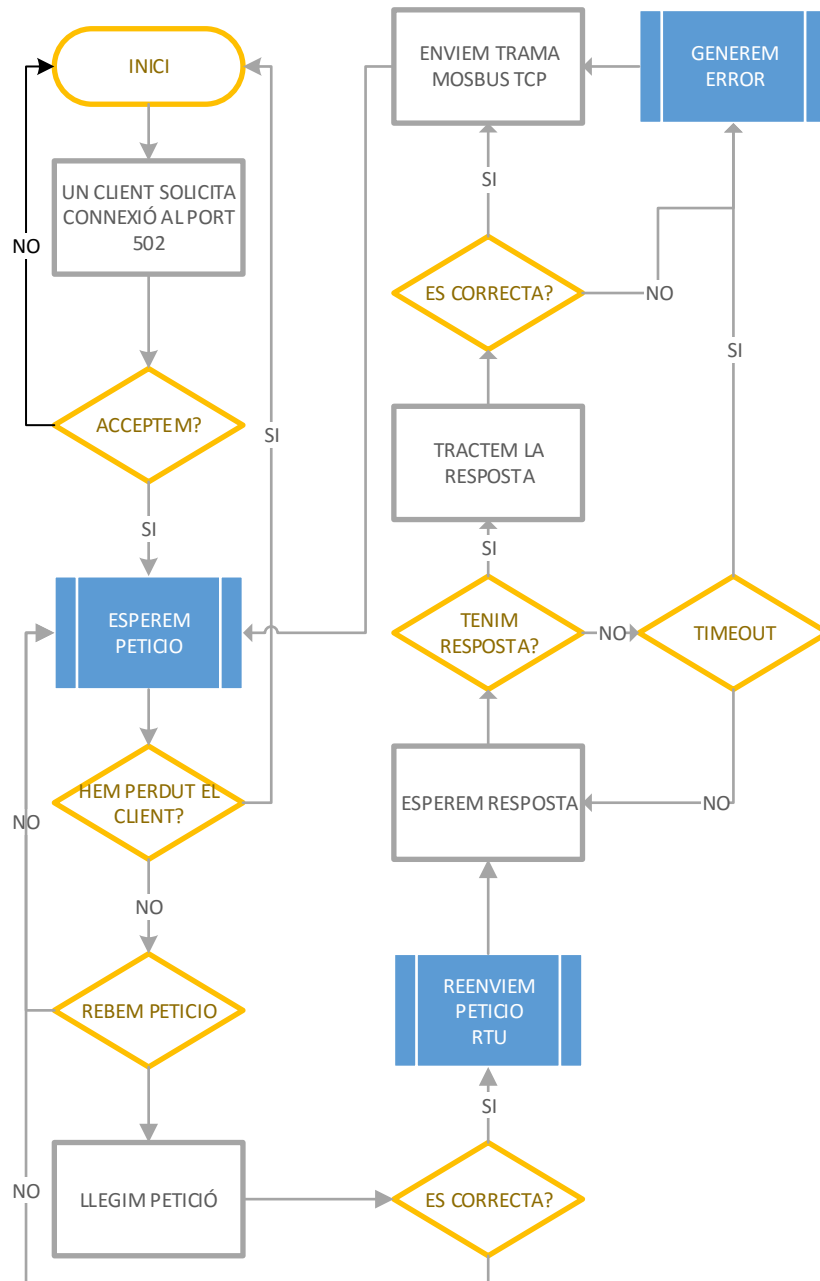


Figura 27: diagrama de flux Arduino

Per a poder fer funcionar el sistema cal primer de tot aconseguir que la nostra Arduino es converteixi en un servidor TCP, o sigui un equip esperi peticions a un port concret, en particular al port 502. Aquesta funció es possible gràcies a la llibreria Ethernet²⁵ d'Arduino. Aquesta llibreria controla de forma transparent

²⁵ <https://www.arduino.cc/en/Reference/Ethernet>

l'escut d'Ethernet del Arduino. Aquesta llibreria no sols permet crear servidors, sinó que també permet la creació de clients tants TCP com UDP.

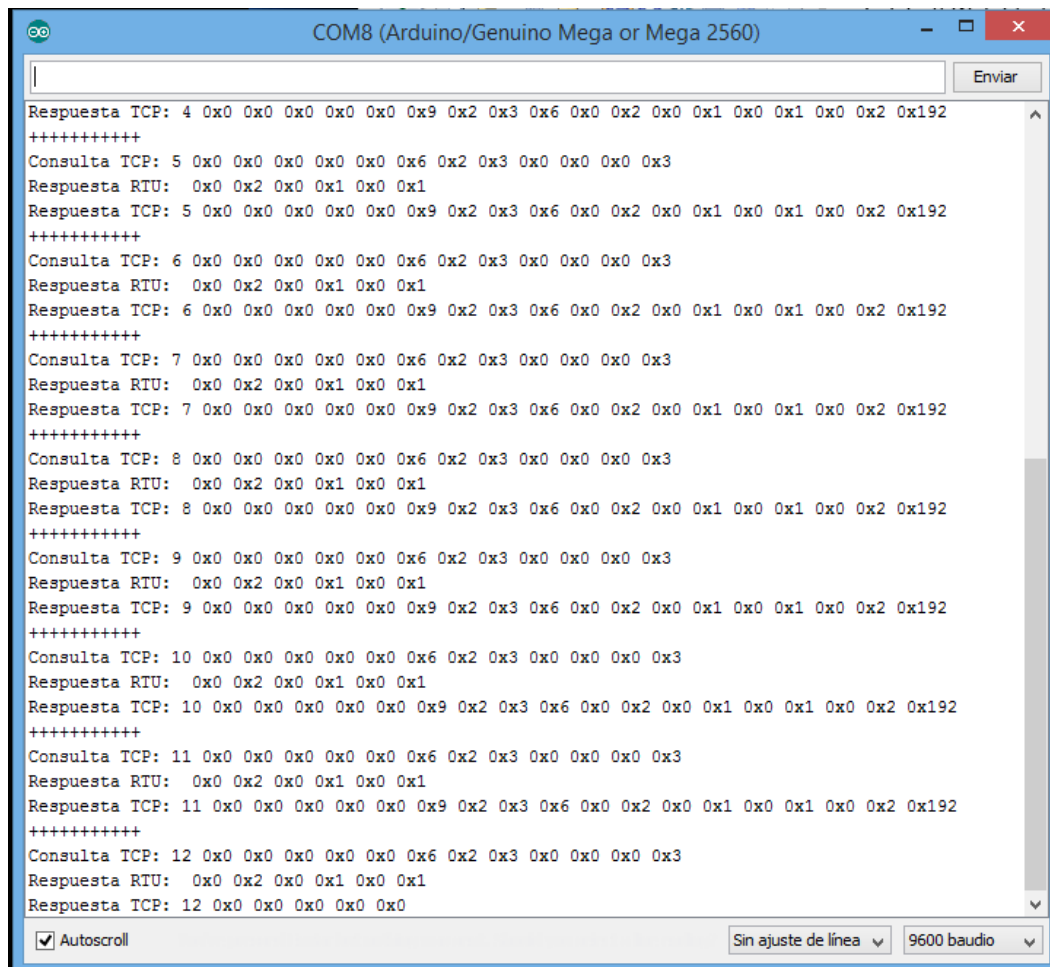
En cas de necessitat es serveix el codi font de la llibreria per a poder realitzar les modificacions necessàries si fos el cas. Però en el nostre cas no ha sigut així i la llibreria ens ha servit amb el seu estat original.

Per a poder inicialitzar la llibreria i així crear una instància d'un servidor TCP cal introduir els següents valors:

- Adreça MAC
- Adreça IP del servidor (estàtica clar).
- Port del servidor (en el nostre cas 502).

Si instanciem correctament la classe disposarem d'un servidor TCP en poques línies de codi.

El següent pas serà inicialitzar els ports series necessaris per al projecte. En concret emprarem 2, un per a les comunicacions amb els esclaus Modbus, i l'altra per a tasques de DEBUG. Per defecte Arduino empra el port serial 1 per a les comunicacions amb el monitor sèrie i poder mostrar per pantalla les trames enviades i rebudes.



```
COM8 (Arduino/Genuino Mega or Mega 2560)
Respuesta TCP: 4 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 5 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 5 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 6 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 6 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 7 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 7 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 8 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 8 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 9 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 9 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 10 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 10 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 11 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 11 0x0 0x0 0x0 0x0 0x0 0x9 0x2 0x3 0x6 0x0 0x2 0x0 0x1 0x0 0x1 0x0 0x2 0x192
+++++++
Consulta TCP: 12 0x0 0x0 0x0 0x0 0x0 0x6 0x2 0x3 0x0 0x0 0x0 0x3
Respuesta RTU: 0x0 0x2 0x0 0x1 0x0 0x1
Respuesta TCP: 12 0x0 0x0 0x0 0x0 0x0 0x0
```

Figura 28: captura del monitor sèrie en mode DEBUG

El port restant l'emprarem per a les comunicacions amb els esclaus Modbus. Tot i emprar els port sèrie del Arduino per la comunicació amb els esclaus no es una comunicació directa, cal activar un pin del mòdul RS485 a nivell alt per a

poder començar a transmetre, i tornar-lo a ficar a nivell baix per a establir el mòdul en mode escolta.

Una vegada tot el maquinari configurat i correctament connectat, ja estem en disposició de rebre peticions per part del mestre Modbus, o sigui la nostra Raspberry que executa SION.

Quan reben la petició de connexió per part d'un client i l'acceptem esperem que aquest ens transmeti un petició Modbus, una vegada aquesta petició es rebuda, el sistema comença la seva descomposició. Tal i com es mostra en la figura 11, una trama Modbus TCP està composta per una trama Modbus RTU menys els bytes de la suma de comprovació. Una vegada descomposta la trama cal incloure la suma de comprovació CRC.

El CRC (comprovació de redundància cíclica) també coneguda com codi polinomial. Aquest sistema s'ha imposat a l'hora del tractament de verificació dades ja que la seva implementació física en un circuit binari és molt senzilla. El seu funcionament és el següent.

Disposarem d'un equip emissor i de receptor, i ambos són coneixedors del polinomi generador. Aquest polinomi ens servirà per a generar la cadena que enviarem i per a comprovar la validesa de les dades rebudes. Aquest polinomi té que acomplir distints requisits: el primer i últim bit tenen que ser 1 i té que ser més petit que la trama a verificar.

El primer que farem serà establir un polinomi generador: $10011 = (x^4+x+1)$ seguidament la trama a enviar serà esta: 1101011111. Com podem observar s'acompleixen tots els requisits expresats abans en referent al polinomi generador.

Seguidament injectarem a la trama un nombre de zeros igual al grau del polinomi generador en aquest cas serien 4, lo que ens quedaria una trama: 11010111110000. Ara ens queda realitzar una divisió binaria entre la trama i el polinomi generador, que no es mes que realitzar un XOR entre la trama i el polinomi generador de forma recurrent. Vegem-ho:

1	1	0	1	0	1	1	1	1	1	0	0	0	0
1	0	0	1	1	↓	↓	↓	↓	↓	↓	↓	↓	↓
	1	0	0	1	1	↓	↓	↓	↓	↓	↓	↓	↓
	1	0	0	1	1	↓	↓	↓	↓	↓	↓	↓	↓
	0	0	0	0	0	1	↓	↓	↓	↓	↓	↓	↓
		0	0	0	0	0	↓	↓	↓	↓	↓	↓	↓
		0	0	0	0	1	1	↓	↓	↓	↓	↓	↓
			0	0	0	0	0	↓	↓	↓	↓	↓	↓
				0	0	1	1	1	1	↓	↓	↓	↓
					0	0	0	0	0	↓	↓	↓	↓
					0	1	1	1	1	0	↓	↓	↓
						1	0	0	1	1	↓	↓	↓
						0	1	1	0	1	0	↓	↓
							1	0	0	1	1	↓	↓
							0	1	0	0	1	0	↓
								1	0	0	1	1	↓
								0	0	0	0	1	0

Per lo que nosaltres enviarem la següent trama:

1	1	0	1	0	1	1	1	1	1	← trama			
Farcit->										0	0	0	0
Residu->												1	0
1	1	0	1	0	1	1	1	1	1	0	0	1	0

El receptor per a comprovar que la trama rebuda es correcta ferà la mateixa divisió però de la trama rebuda. Si el resultat de la divisió es zero, la trama es correcta, si no es així la trama es incorrecta.

Cada protocol de comunicacions que executa un CRC ja predefinit, seguidament es mostra una llista en polinomis de diferents protocols, en especial ens interessa el CRC-16 que el que empra el protocol Modbus, que després emprarem.

- **CRC-12:** $X^{12} + X^{11} + X^3 + X^2 + X + 1$
- **CRC-16:** $X^{16} + X^{15} + X^2 + 1$
- **CRC CCITT V41:** $X^{16} + X^{12} + X^5 + 1$
- **CRC-32 (Ethernet):** $= X^{32} + X^{26} + X^{23} + X^{22} + X^{16} + X^{12} + X^{11} + X^{10} + X^8 + X^7 + X^5 + X^4 + X^2 + X + 1$
- **CRC ARPA:** $X^{24} + X^{23} + X^{17} + X^{16} + X^{15} + X^{13} + X^{11} + X^{10} + X^9 + X^8 + X^5 + X^3 + 1$

Una vegada resolt el CRC s'incorpora a la trama a enviar per port sèrie cap al esclau triat. Una vegada l'esclau rep la trama la respon. Si s'esgota el temps de resposta, Arduino omet la trama i retorna al Mestre una trama creada per ell.

Enviem un missatge per el port sèrie i no envia res al mestre per a que esgoti el seu timeout i reenvie la trama o be declari en error eixe analitzador.

Si la resposta és correcta per part de l'esclau, aquesta torna a ser descomposta i tornada a compondre en format Modbus TCP per a ser posteriorment reenviada a SION.

El següent diagrama esposa el funcionament correcte d'una trama:

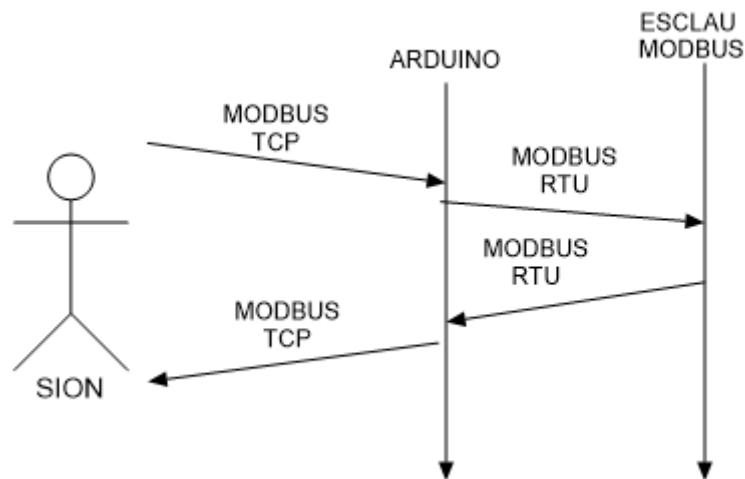


Figura 29: trama Modbus

4.4. Interfície Web

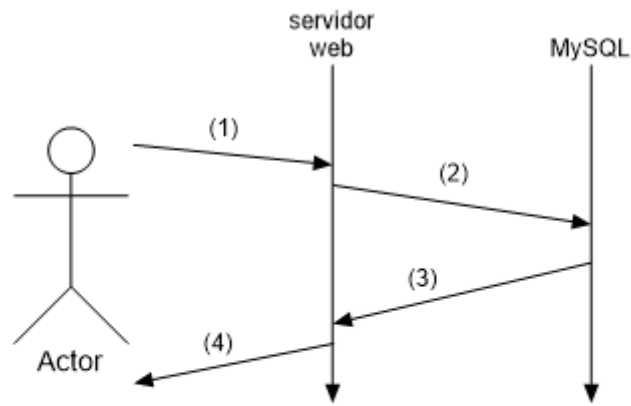
Actualment la majoria de projectes que es realitzen a nivell professional, sobre tot en eines de supervisió industrial, la majoria de clients demanen la possibilitat d'obtenir un accés remot l'equip auditat. No solament es demana que es pugui accedir de forma remota, si no que es vol accedir mitjançant telèfons mòbils o tabletas, fins i tot cap la possibilitat que el usuari disposi enlloc de PC un equip MAC o Linux, on determinats sistemes d'accés remot (com controls ActiveX) no poden funcionar.

Per això la millor opció es accedir mitjançant un sistema que ofereixi un accés remot però en qualsevol sistema operatiu o arquitectura. Un element que és comú a tots els dispositius esmentats abans es que tots disposen de navegador web. Per això s'ha optat que el sistema es pugui controlar mitjançant un navegador web.

Mitjançant aquest tipus de navegador es podrà tant accedir a les dades capturades pel sistema, com generar gràfiques o be desconnectar el analitzador dels cicles de lectura.

El sistema també mostra l'estat del analitzador mitjançant codis de colors que indiquen els estats dels mateixos.

La primera pantalla amb la que ens trobem es una pantalla d'accés mitjançant el mètode clàssic pregunta-resposta. El client envia les dades d'accés, el servidor rep les dades i fa una consulta cap la base de dades MySQL, si el usuari existeix, llavors és directament redirigit cap la pàgina d'inici del projecte, en cas contrari es continuarà mostrant la pàgina actual. El diagrama d'accions seria el següent:



- (1) - Petició d'accés - Conté les dades del usuari
- (2) - Consulta Mysql amb les dades del usuari
- (3) - Resposta Mysql amb el nombre d'usuaris amb eixes dades
- (4) - Si el nombre es Zero, mostra la pàgina de log. Si es 1 mostra la pàgina de inici de SION

Figura 30: diagrama d'accés

Queda pendent millorar el sistema d'accés amb un nombre màxim de peticions d'accés i el retorn d'un missatge d'error. Així com la comprovació mitjançant JS de la validesa dels camps introduïts.

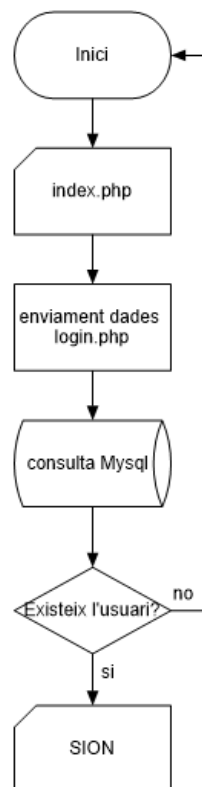


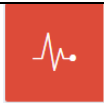


Figura 31: diagrama de flux login SION

Una vegada dins del sistema ens trobem una pàgina amb força informació, ens trobarem una llista dels analitzadors donants d'alta. Cada caixa d'informació ens dona informació sobre l'estat de l'analitzador així com els valors que està llegint.

Taula 9: representació estat analitzadors

Analitzador OK	Analitzador Desconnectat	Analitzador en Alarma																																				
 <table border="1"> <thead> <tr> <th colspan="3">ANALIZADOR 1</th> </tr> <tr> <th>Fase 1</th> <th>Fase 2</th> <th>Fase 3</th> </tr> </thead> <tbody> <tr> <td>4.080 a</td> <td>0.000 a</td> <td>0.000 a</td> </tr> <tr> <td>219.5 v</td> <td>0.0 v</td> <td>0.0 v</td> </tr> </tbody> </table>	ANALIZADOR 1			Fase 1	Fase 2	Fase 3	4.080 a	0.000 a	0.000 a	219.5 v	0.0 v	0.0 v	 <table border="1"> <thead> <tr> <th colspan="3">ANALIZADOR 2</th> </tr> <tr> <th>Fase 1</th> <th>Fase 2</th> <th>Fase 3</th> </tr> </thead> <tbody> <tr> <td>a</td> <td>a</td> <td>a</td> </tr> <tr> <td>v</td> <td>v</td> <td>v</td> </tr> </tbody> </table>	ANALIZADOR 2			Fase 1	Fase 2	Fase 3	a	a	a	v	v	v	 <table border="1"> <thead> <tr> <th colspan="3">ANALIZADOR 1</th> </tr> <tr> <th>Fase 1</th> <th>Fase 2</th> <th>Fase 3</th> </tr> </thead> <tbody> <tr> <td>##.## a</td> <td>##.## a</td> <td>##.## a</td> </tr> <tr> <td>###.# v</td> <td>###.# v</td> <td>###.# v</td> </tr> </tbody> </table>	ANALIZADOR 1			Fase 1	Fase 2	Fase 3	##.## a	##.## a	##.## a	###.# v	###.# v	###.# v
ANALIZADOR 1																																						
Fase 1	Fase 2	Fase 3																																				
4.080 a	0.000 a	0.000 a																																				
219.5 v	0.0 v	0.0 v																																				
ANALIZADOR 2																																						
Fase 1	Fase 2	Fase 3																																				
a	a	a																																				
v	v	v																																				
ANALIZADOR 1																																						
Fase 1	Fase 2	Fase 3																																				
##.## a	##.## a	##.## a																																				
###.# v	###.# v	###.# v																																				

Com podem observar en el moment que l'equip entra en alarma o es desconnecta s'esborren totes les dades de lectura que ofereix, ja que no existeixen noves dades sobre ell. Per aconseguir aquest efecte fem Ajaj i JS.

Cada dos segons fem una consulta mitjançant Ajaj cap a una pàgina anomenada listado_analizadores.php. Aquesta pàgina té una única funció i es la de tornar tant les dades de tots els analitzadors que existeixen en la base de dades de la taula 'listado_analizadores' així com l'última lectura realitzada de cada equip i retorna-ho en format JSON. La resposta a una petició cap a la pàgina 'listado_analizadores.php' es divideix en diverses seccions. La primera correspondria a l'estat de la petició amb el camp 'completado'

```

{"completado":true,
"listado_analizadores":
{"0":{"id":"1","id_esclavo":"1","IP_esclavo":"192.168.2.177","nombre":
"analizador_1","activado":"1","enFallo":"0","enAlarma":"0","estado":"5
"},
"1":{"id":"2","id_esclavo":"2","IP_esclavo":"192.168.2.89","nombre":"a
nalizador_2","activado":"0","enFallo":"0","enAlarma":"0","estado":"0"}
}

```

Seguidament ve una matriu amb els analitzadors de la taula 'listado_analizadores'. La darrera part de la consulta tornaria les últimes insercions de cada analitzador a la seva respectiva taula:

```

"lectura_ana_1":{"0":{"id_lectura":"804","dateTime":"2016-01-08
18:41:43","tension_fase_1":"219.5","corriente_fase_1":"4.080","tensio
n_fase_2":"0.0","corriente_fase_2":"0.000","tension_fase_3":"0.0"
,"corriente_fase_3":"0.000","energiaActiva":"247166","energiaReacti
vaInductiva":"2975","energiaReactivaCapacitativa":"25411","energiaA
parente":"253686","frecuencia":"50.0","tension_L1_L2":"219.5","tensio
n_L2_L3":"0.0","tension_L3_L1":"218.7","THD_V_L1":"1.8","THD_V_L2
":"0.0","THD_V_L3":"0.0","THD_A_L1":"4.3","THD_A_L2":"0.0","THD_A_L
3":"0.0","potenciaActiva":"870","potenciaInductiva":"0","potenciaCa
pacitativa":"90","cosenoPhi":"-0.99","factorPotencia":"0.99"}},
"lectura_ana_2":{"0":{"id_lectura":"1","dateTime":"2016-01-03
08:55:44","tension_fase_1":"0.0","corriente_fase_1":"0.000","tensio
n_fase_2":"0.0","corriente_fase_2":"0.000","tension_fase_3":"0.0","
corriente_fase_3":"0.000","energiaActiva":"0","energiaReactivaInduc
tiva":"0","energiaReactivaCapacitativa":"0","energiaAparente":"0","
frecuencia":"0.0","tension_L1_L2":"0.0","tension_L2_L3":"0.0","tensio
n_L3_L1":"0.0","THD_V_L1":"0.0","THD_V_L2":"0.0","THD_V_L3":"0.0"
,"THD_A_L1":"0.0","THD_A_L2":"0.0","THD_A_L3":"0.0","potenciaActiva
":"0","potenciaInductiva":"0","potenciaCapacitativa":"0","cosenoPhi
":"0.00","factorPotencia":"0.00"}}}

```

Una vegada hem rebut les dades, gràcies a JS i en concret a Jquery actualitzem les dades dels analitzadors. Modificant tant el seu estat (colors) com les seves lectures. A continuació es mostra un diagrama de flux del funcionament del sistema.

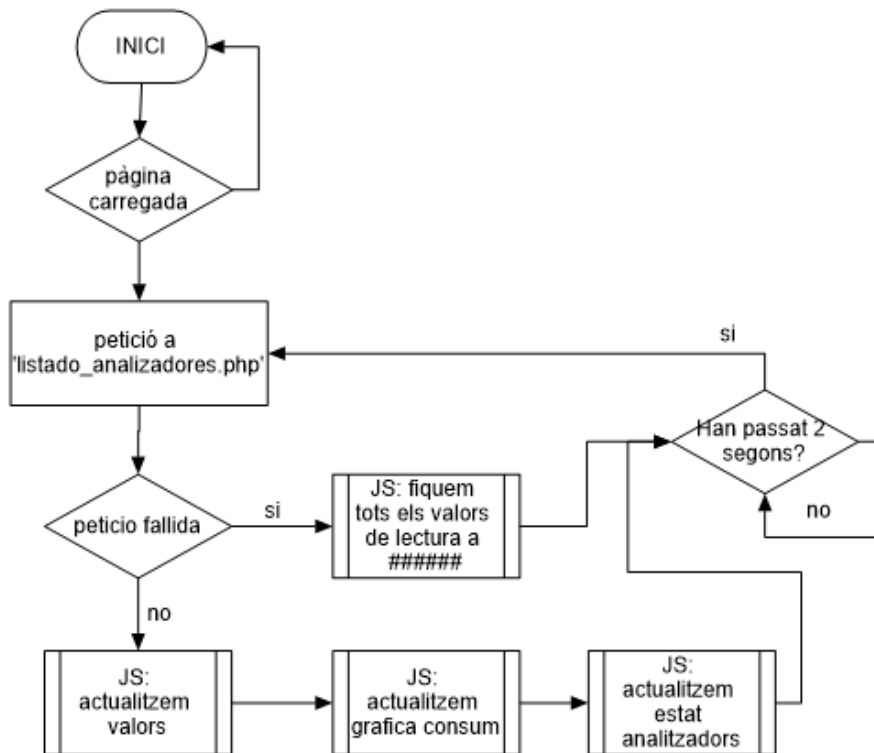


Figura 32: canvi estat i valors web SION

A la figura 32 apareix el subprocés en JS d'actualitzar els valors de la gràfica. La gràfica realment s'actualitza quan es carrega per primera vegada la pàgina. També disposa d'una animació per a la càrrega, si estigués actualitzant eixes dades cada 2 segons, eixa animació seria molesta per a l'usuari.

Per a accedir als detalls de cada analitzador, prenem sobre la icona de cada analitzador. La finestra que se ens obre es idèntica en funcionament amb la diferència que mostra totes les dades del analitzador seleccionat, ja que recupera les dades de la mateixa 'listado_analizadores.php' i de la mateixa forma i mateixos intervals de temps.

L'única diferència en quant a disseny no ve d'aquesta pròpia pàgina, si no en una pàgina que enllaça amb aquesta, que mostra la gràfica de corrents de l'analitzador.

En la pàgina de detalls de l'analitzador es pot seleccionar un rang de dates per a demanar una gràfica. Si aquest rang es superior a 3600 segons, la gràfica mostra resultats per hores, mentre que si es fa en un interval menor la gràfica es mostra per minuts.

Aquesta funcionalitat s'aconsegueix mitjançant una simple operació de resta dels camps enviats a la pàgina 'grafica_corriente.php' si el resultat es menor a 3600 executa esta consulta:

```

"SELECT corriente_fase_1, corriente_fase_2, corriente_fase_3,
DATE(dateTime) as hora FROM analizador_<id_analitzador> WHERE
dateTime BETWEEN <dataInici> AND <dataFi> GROUP BY DATE(dateTime),
HOUR(dateTime)ORDER BY dateTime ASC"
  
```

En cas contrari executaria un altra consulta que retorna les dades agrupades per minuts.

Una vegada retorna les dades la consulta, mitjançant PHP s'omplin els camps en JS per a poder crear la gràfica.

Més informació i exemples d'us al manual de l'aplicació al annex 4

5. Viabilitat tècnica

Una vegada hem realitzat les proves sobre el prototip creat, cal dir que si es factible la seva instal·lació, tant la de la passarel·la Arduino com el nucli SION. Els seus punts forts es sobre tot la modularitat del conjunt, que permet incrementar el nombre d'analitzadors amb un cost mínim temporal de programació, si s'apliquen les millores necessàries al prototip que es comentaran després.

Cal dir que durant els dies de proves ininterrompudes que ha patit l'equip no ha donat sensació d'esgotament si mes no la mida inicial de la targeta per a desat les dades s'ha quedat curta. Caldria implantar unes millores en el sistema de desat de dades per a que no ocupessin tant, estes millores es tractaran més endavant en el apartat corresponent.

Resumint, el prototip de SION es perfectament implementable en un entorn real, fins i tot sense aplicar-li les millores, ja que com hem comentat abans el sistema funciona en entorns reals. Una de les raons per les quals aquest sistema disposa de viabilitat es que el seu origen naix d'una necessitat real, que es la de reduir els costos tant d'equips com de llicències al mon industrial, buscant en tot moment una semblança de funcionament en solucions llicenciades reals implementades.

6. Valoració econòmica

Tal com s'ha comentat abans un dels objectius del projecte era fabricar un sistema similar al que s'està actualment implementant a projectes reals, però amb programari lliure que no suportés els costos de llicències, per això no es valorarà de forma única el producte, si no que es compararan els costos amb un treball amb software llicenciat, eliminant els camps que siguin idèntics per als dos, ja que no aportarien informació comparativa.

En aquest cas s'ha aconseguit l'objectiu i per tant podem eliminar el factor del cost de les llicències, que seria superior a 2000€. També hem comprovat que podem eliminar el PC necessari per a executar el software comercial, i podem executar tot el programari necessari en una Raspberry Pi i un Arduino. Els temps de programació serien similars tant en l'Scada com en SION, ja que cada client demana unes gràfiques i estadístiques concretes i cal personalitzar el seu entorn, logos, imatges de la seva planta, ect. El mateix succeeix amb el manteniment, que sol anar orientat a còpies de seguretat i control d'ocupació d'espai en el disc. Per lo que la taula comparativa quedaria així:

Taula 10: diferència de costos SION vs Indusoft

	Llicenciat	SION
Maquinari	798,70€ (figura 33)	66,91€ (figura 34)
Sistema operatiu	157,00€ (figura 35)	No
Llicència Scada	421,00€ ²⁶ (figura 36)	No
	1376,7 €	66,91 €

Com es pot observar la diferència per a un Scada d'un sol analitzador ja es de 1376,7 €, a este preu caldria sumar-li el temps de desenvolupament de les dues aplicacions, com a mínim 5 dies de treball, més els cost de la instal·lació física del analitzador. Esta última no es pot aportar ja que dependria de la situació física del analitzador, es a dir distància fins al nucli de comunicacions, si caldria fer forat per a poder col·locar el analitzador en el quadre elèctric, els toroides, ect.

Evidentment un dels punts forts de SION es la diferència de preu d'un producte a un altre, a favor es clar de SION.

Fecha: 09/01/2016

Teléfono: _____

Presupuesto de Ordenador



































Familia	Artículos	Uds.	Precio
<input checked="" type="checkbox"/>  Procesadores	 INTEL. CORE I5 4460 3.20GHZ 1150 BOX	<input type="text" value="1"/>	203,00 €
<input checked="" type="checkbox"/>  Placas Base Intel	 GIGABYTE PLACA BASE H81M S1 MATX LGA1150	<input type="text" value="1"/>	57,70 €
<input checked="" type="checkbox"/>  Placas Base Amd		<input type="text" value="0"/>	
<input checked="" type="checkbox"/>  Discos Duros	 W.DIGITAL 1TB 3.5 WD10EZEEX SATA3 7200 64MB WD BLUE	<input type="text" value="1"/>	49,10 €
<input checked="" type="checkbox"/>  Memorias Ram	 KINGSTON DDR3 8GB 1600MHZ HYPERX FURY BLUE CL10	<input type="text" value="1"/>	50,60 €
<input checked="" type="checkbox"/>  Torres	 UNYKA UK 6011 MATX 500W NEGRA	<input type="text" value="1"/>	43,40 €
<input checked="" type="checkbox"/>  Monitores	 MONITOR ACER 23 G236HLB8ID FHD 5MS HDMI DVI VGA	<input type="text" value="1"/>	122,50 €
<input checked="" type="checkbox"/>  Teclados	 LOGITECH MK270 COMBO TECLADO CON RATON OPTICO 2.4G	<input type="text" value="1"/>	28,40 €
<input checked="" type="checkbox"/>  Ventiladores	 SCYTHE REFRIG. KATANA 4 INTEL AMD	<input type="text" value="1"/>	36,50 €
<input checked="" type="checkbox"/>  Multilectores	 COOLBOX LECTOR TARJETAS FLASH Y DNIE USB	<input type="text" value="1"/>	13,60 €
<input checked="" type="checkbox"/>  Grabadoras Dvd	 LG DVD.RW INTERNA 24X BULK SATA SIN SOFTWARE	<input type="text" value="1"/>	13,70 €
<input checked="" type="checkbox"/>  Tarjetas Graficas	 GIGABYTE VGA GTX 750 TI OC 2GB DDR5	<input type="text" value="1"/>	132,00 €
<input checked="" type="checkbox"/>  Ratones	 RATON OPTICO LOGITECH M175 MINI WIRELESS NEGRO	<input type="text" value="1"/>	17,60 €
<input checked="" type="checkbox"/>  Altavoces	 WOXTER ALTAVOCES BIG BASS 95 NEGRO 2.0	<input type="text" value="1"/>	30,60 €
<input type="checkbox"/>  Agregar Artículo...	 ...	<input type="text" value="0"/>	
			798,70 €

Figura 33: pressupost PC mitjà. Font *appinformàtica*

²⁶ El preu correspon a una llicència *Lite interface de Indusoft* amb 150 variables. En un sistema equivalent al nostre podríem connectar 1 o dos analitzadors.

Cesta

	Precio	Cant.
 <p>Samsung Evo MB-MP16D/EU - Tarjeta de memoria Micro SDHC de 16 GB (UHS-I Grade 1, Clase 10) por Samsung En stock <input type="checkbox"/> Es un regalo Más información Eliminar Guardar para más tarde</p>	EUR 8,98	1
 <p>Raspberry Pi 2 Model B - Placa base (ARM Quad-Core 900 MHz, 1 GB RAM, 4 x USB, HDMI, RJ-45) por Raspberry Pi En stock <input type="checkbox"/> Es un regalo Más información Eliminar Guardar para más tarde</p>	EUR 39,95	1
 <p>Raspberry Pi 8193655 - Caja para Raspberry Pi B+ y Pi2, color negro por Raspberry Pi En stock <input type="checkbox"/> Es un regalo Más información Eliminar Guardar para más tarde</p>	EUR 7,99	1
 <p>Nuevo! Cargador Micro USB para Raspberry Pi 2, B+, B, A & Banana Pro & Pi (5V, 2000 mA) por MAREL En stock <input type="checkbox"/> Es un regalo Más información Eliminar Guardar para más tarde</p>	EUR 9,99	1

Pilot V-ball rt 07 - Blistér de 3 bolígrafos (punta mediana de b... se ha eliminado de Cesta.

Subtotal (4 productos): EUR 66,91

Figura 34: presupost Raspberry Pi. Font Amazon

Familia	Artículos	Uds.	Precio
<input checked="" type="checkbox"/> Sistemas Operativos	WINDOWS 7 PROFESIONAL 64 BITS OEM DVD	1	157,00 €
<input type="checkbox"/> Agregar Artículo...	...	0	
			157,00 €

Figura 35: presupost S.O. Font appinformatica



InduSoft Web Studio v7.1 + SP1

Condición Nuevo

Licencia SCADA InduSoft Web Studio v7.1+SP1 Runtime, Lite Interface

Codigo: IND-15002NT-RT

Precios incluyen IGV

[Tweet](#)
[Compartir](#)
[Google+](#)

[Pinterest](#)

[Compartir en Facebook](#)

[Enviar a un amigo](#)

[Imprimir](#)

\$460.20

Cantidad

1

+ -

Añadir al carrito

Aceptamos
Targetas de
crédito y débito



Figura 36: presupost Scada Indusoft 150 tags. Font smartInstruments

7. Conclusions

7.1. Conclusions

Per a observar la correcta finalització del projecte farem una taula amb els objectius declarats a la planificació inicial del projecte.

Taula 11: llista objectius i resultats

Objectiu	Resultat
<i>Programació d'un sistema d'adquisició de dades en Python capaç de comunicar mitjançant un protocol estàndard, com es Modbus TCP amb diferents equips de mesures de dades energètiques. Aquest software ha de córrer en una Raspberry Pi</i>	Complit al 100%
<i>Que aquestes dades siguin transformades en informació mitjançant informes predefinitos i gràfiques explicatives.</i>	El sistema seria capaç de generar els informes, però al prototip de SION no s'han programat informes. Gràfiques si.
<i>Que aquesta informació pugui ser accedida de forma descentralitzada, es a dir des de qualsevol punt, mitjançant navegadors Web. Per lo que la Raspeberry ha de ser alhora servidor web i servidor de bases de dades.</i>	Complit al 100%
<i>El sistema deu estar programat en programari lliure. En concret emprarem Python, MySQL i Lighttpd, a banda de dos plataformes de maquinari lliure, Arduino i Raspberry.</i>	Complit al 100%
<i>Creació d'un sistema mixt de maquinari i programari que sigui capaç de traduir les trames que emetrà la Raspberry en Modbus TCP a telegrams Modbus RTU.</i>	Complit al 100%

El segon punt del objectiu es podria dir que s'ha acomplit ja que el sistema desa la informació, per exemple de consum i si que seria capaç d'emetre informes, però essent sincers, aquesta part no s'ha programat per manca de temps.

A nivell pedagògic podem dir que la realització de la pràctica ha sigut molt interessant, tenia moltes ganes en aprofundir en un llenguatge com Python, ja que havia escoltat que es molt potent. Havia fet altres treballs en Python per a la universitat, sobre tot a l'àrea d'Intel·ligència Artificial, però no havia emprat comunicacions. Ara definitivament puc dir que aquest llenguatge ofereix unes possibilitats increïbles, sobre tot per la quantitat d'informació i documentació que existeix a la xarxa.

7.2. Proposta de millores.

Conforme vas fent la documentació, ens anem adonant de defectes que pot tenir el producte o millor dit de millores que el producte deuria patir per a se més funcional. Les propostes són les següents:

- Creació d'un panell de control per a la gestió tant d'usuaris com de analitzadors. Tindre un millor control d'accessos desant a la base de dades l'últim accés d'un usuari, així com tenir la capacitat de crear nous usuaris o esborrar-los. El mateix amb els analitzadors, tenir la capacitat de crear o esborrar analitzadors. Creació de diferents nivells d'usuari.
- Que la mateixa Raspberry Pi fora servidor VPN, d'aquesta forma no caldria tenir que tocar la infraestructura de la xarxa de la empresa. Senzillament cadria redirigir el port de OpenVPN cap a la Raspberry. D'aquesta forma disposaríem d'un accés remot i segur. També podria succeir que la planta no disposes de connexió a internet, llavors podríem emprar un mòdem 3G a un port USB de la Raspberry Pi per a obtenir comunicació a internet.
- Evitar connexions al mateix esclau. Quan creem un analitzador de xarxes li assignem una IP, en cas de que tinguéssim més d'un analitzador penjant de la mateixa passarel·la crearíem tantes connexions a la mateixa passarel·la com elements fossin necessaris. Això no es necessari quan amb una única connexió seria suficient. Deuríem agrupar les connexions per IP no per objecte creat.
- Crear mapes per a diferents tipus d'analitzadors o equips. Així quan crearem un analitzador només caldria dir de quin tipus es.
- Ampliar el model a altres protocols de comunicació, per exemple Profinet, un protocol de Siemens sobre Ethernet, el que ens permetria controlar no sols analitzadors si no també altre tipus d'equips com variadors de freqüència o PLC.

7.3. Autoavaluació.

Funciona. Eixe podria ser el resum del treball. Si l'objectiu principal d'este projecte era demostrar que en entorns empresarials es poden reduir els costos d'implementació emprant programari lliure. Demostrat queda.

També queda demostrat que podem emprar maquinari lliure per a poder desenvolupar productes que poden competir amb maquinari llicenciat.

També queda demostrat el mal ús a l'hora d'emprar maquinari, ja que per una instal·lació similar a esta, cal un ordinador amb unes característiques que fan que el preu de l'equip es dispari.

Per eixa part estic satisfet, tot el que m'he proposat al començar este projecte han estat assolits, però per altra banda hem queda l'espina que podria haver estat millor programat i haver desenvolupat el codi per a posteriors reutilitzacions, enlloc de dirigir el producte a un resultat concret.

Si mes no se m'obre una porta cap a un altre projecte. Dissenyar un Scada lliure basat en Python.

8. Glossari

- SCADA: *Supervisory Control And Data Acquisition* (Supervisió, Control i Adquisició de Dades).
- UNIX: Sistema operatiu portable, multitasca i multiusuari.
- C: Llenguatge de programació.
- UNICODE: estàndard de codificació de caràcters.
- Debian: sistema operatiu que empra en nucli Linux.
- ARM: arquitectura, família de microprocessadors produïts per la empresa ARM Holdings.
- GPIO: *General Purpose Input Output* (Entrada Eixida de propòsit General).
- UART: *Universal Asynchronous Receiver-Transmitter* (Transmissor-Receptor Asíncrono Universal).
- SPI: *Serial Peripheral Interface* (Interface Serial de Perifèrics).
- I2C: *Inter Integrated Circuit* (bus de comunicació: Circuit Inter-Intergrats).
- ATX: *Advanced Technology Extended*, estàndard per a millorar la funcionalitat del E/S actuals.
- IDE: *Integrated Development Environment* (Ambient de Desenvolupament Integrat).
- RAM: *Random Acces Memory*, (Memòria d'Accés Aleatori).
- W3C: *World Wide Web Consortium*, (Consorti D'Internet).
- XML: extensive Markup Language, (Llenguatge Extensiu de Marques).
- JSON: JavaScript Object Notation, format per al intercanvi lleuger de dades.
- VPN: Virtual Private Network, Xarxa Privada Virtual.
- PLC: Programmable Logic Controller (Programador Lògic Programable).

9. Bibliografia

Alvarez Miguel Angel desarrolloweb.com [En línia] // desarrolloweb.com. - desarrolloweb.com, 19 / 11 / 2003. - 06 / 01 / 2016. - <http://www.desarrolloweb.com/articulos/1325.php>.

api.jquery [En línia] // api.jquery. - JQuery Foundation, 2016. - 01 de 11 de 2015. - <http://api.jquery.com/>.

Arduino Arduino Librerias [En línia] // Arduino Librerias. - Arduino, 2016. - 10 de 9 de 2015. - <https://www.arduino.cc/en/Reference/Libraries>.

D'Andrea Edgar MySQL 5 [Libro]. - Barcelona : InforBooks, 2010. - Vol. 1.

Flanagan David JavaScript the definitive Guide [Libro]. - Estats Units d'Amèrica : O'Reilly, 1998. - Vol. 1.

Gallen Collins Pymodbus Documentacion [En línia] // Pymodbus Documentacion. - Python, 2009. - 10 de 10 de 2015. - <http://pythonhosted.org/pymodbus/>.

Lutz Mark Learning Python [Llibre]. - Estats Units D'Amèrica : O'Really, 2009. - 978-0-596-15806-4.

Manual de PHP [En línia] // Manual de PHP. - PHP Group, 2016. - 10 de 11 de 2015. - <https://secure.php.net/manual/es/index.php>.

Ullman Larry PHP paso a paso [Libro]. - Madrid : Anaya Multimedia, 2009. - Vol. 1.

wikipedia wikipedia [En línia] // wikipedia. - wikipedia, 28 de 11 de 2015. - 12 de 01 de 2016. - <https://es.wikipedia.org/wiki/SCADA>.

wikipedia wikipedia [En línia] // wikipedia. - wikipedia, 26 de 11 de 2015. - 06 de 01 de 2016. - <https://es.wikipedia.org/wiki/MySQL>.

wikipedia wikipedia [En línia] // wikipedia. - wikipedia, 11 de 12 de 2014. - 12 de 01 de 2016. - https://es.wikipedia.org/wiki/Universal_Asynchronous_Receiver-Transmitter.

wikipedia wikipedia [En línia] // wikipedia. - wikipedia, 28 de 11 de 2015. - 12 de 01 de 2016. - <https://es.wikipedia.org/wiki/I%C2%B2C>.

wikipedia wikipedia [En línia] // wikipedia. - wikipedia, 28 de 12 de 2015. - 12 de 01 de 2016. - <https://es.wikipedia.org/wiki/JSON>.

wikipedia wikipedia en espanyol [En línia] // wikipedia en espanyol. - wikipedia, 2015 / 06 / 30. - 06 / 01 / 2016. - https://es.wikipedia.org/wiki/BBC_Micro.

10. Annexos

1.- Configuració Raspberry Pi

Per a que el sistema sigui capaç de tractar amb el nostre sistema cal instal·lar una sèrie de programari específic. Aquesta guia comença quan la Raspberry ja te la configuració de idioma i de fitxers correcta (extensió del sistema).

Canvi de IP dinàmica a IP fixa

Amb la configuració bàsica completa, la primera modificació serà assignar una IP fixa al sistema, ja que sinó no serem capaços d'accedir al nostre sistema d'una forma correcta, ja que pot succeir que el nostre router canviï la IP del nostre equip.

Es pot fer de dues formes, la primera a través de l'entorn gràfic, però com serà habitual ho farem des de el terminal de Raspbian.

Al terminal teclegem:

```
sudo nano /etc/network/interfaces
```

Ara se ens obrirà l'editor de texts nano, cerquem la línia: *iface eth0 inet dhcp* aquesta línia fa referència a la connexió cablejada, en cas de que s'optés per una configuració mitjançant un adaptador WIFI, caldria cercar *iface wlan0 inet dhcp*

```
iface eth0 inet static
address 192.168.2.79
netmask 255.255.255.0
gateway 192.168.2.1
```

Com hem dit abans canviem els valor que apareixen per els nostres, tal com diu la caixa de text superior. Guardem els canvis polsant Crtl + X i acceptem desar els canvis.

Ja disposem d'una IP fixa dins del nostre rang.

Instal·lació Servidor Web

Com ja disposem d'una IP fixa, el nostre sistema ja permet accedir a d'ell d'una forma estable, ara queda instal·lar l'eina que permetrà accedir a les dades del SION.

Com a servidor web hem optat per Lighttpd²⁷, un servidor lleuger que cobreix totes les nostres necessitats, a banda també instal·larem PHP, que ens permetrà executar consultes des de la web, i que la nostra aplicació sigui capaç de mostrar dades de forma dinàmica.

A la nostra terminal teclegem:

```
sudo apt-get -y install lighttpd
```

²⁷ <https://www.lighttpd.net/>

El procés es automàtic, el sistema no ens preguntarà res.

Ara instal·larem PHP:

```
sudo apt-get -y install php5-common php5-cgi php5
```

El procés també es automàtic i ara només ens queda fer que el nostre servidor web reconegui PHP, ho farem mitjançant les següents instruccions de terminal:

```
sudo ligthy-enable-mod fastcgi-php #habilitar el control de PHP
sudo service lighttpd force-reload #reiniciar el servidor web
```

Una modificació que queda per fer es donar permisos al usuari Pi (que ve per defecte) per a poder modificar el directori del servidor web, així podrem incloure el nostre codi en ell.

```
sudo chown -R www-data:www-data /var/www #donem permisos al grup
www-data
sudo chmod 775 /var/www # permisos d'escriptura
sudo usermod a- -G www-data pi # incloquem al usuari Pi dins de www-
data
sudo reboot #reiniciem el sistema
```

Una vegada reiniciat el sistema ja podrem modificar els fitxers al directori del servidor web.

Instal·lació de Mysql

Com a gestor de base de dades hem optat per mysql per a fer esta tasca, ja que es àmpliament suportat per la majoria de tots els llenguatges de programació. A la nostra terminal teclejarem:

```
sudo apt-get -y install mysql-server mysql-client phpmyadmin
```

A banda instal·larem phpmyadmin, una excel·lent utilitat per al control i gestió de mysql.

```
sudo ln -s /usr/share/phpmyadmin/ /var/www/phpmyadmin
```

Durant la instal·lació ens demanarà les contrasenyes de "root" per a mysql server, la teclegem. També ens demanarà quin servidor web anem a emprar, seleccionarem lighttpd. Ens dirà si volem carregar la configuració per defecte o be volem un altra, la configuració per defecte ens serveis perfectament. També ens preguntarà la contrasenya del root de la base de dades administrativa, la teclegem i llestos.

Una darrera configuració serà la de crear un enllaç simbòlic cap al directori de phpmyadmin per a que sigui accessible des de la web. Teclegem:
Ara ja podem accedir des de la web mitjançant les contrasenyes introduïdes abans.

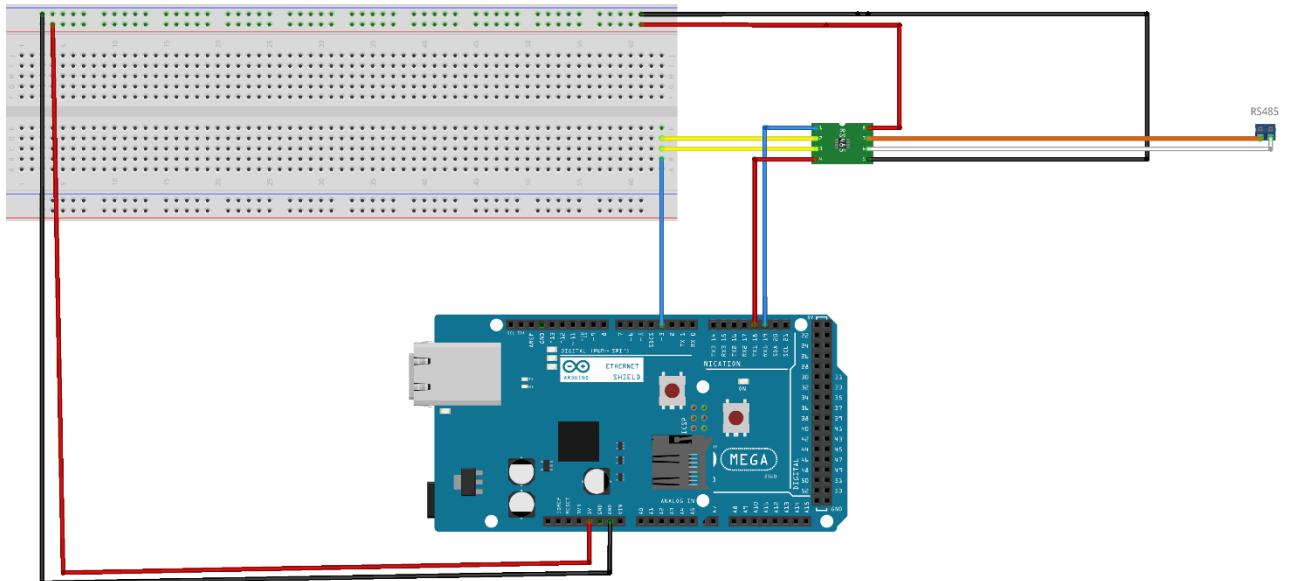
Llibreries Python

Queda un últim punt per a poder fer funcionar el sistema i es instal·lar les llibreries de Python necessàries, en concret Pymodbus i PyMySQL. La primera ofereix comunicació Modbus al sistema i l'altra dona l'opció de treballar amb bases de dades Mysql. La instal·lació es força senzilla:
Tot el procés es automàtic i no cal fer res més.

```
sudo apt-get -y install python-pymodbus python-mysqldb
```

2.- Configuració Arduino

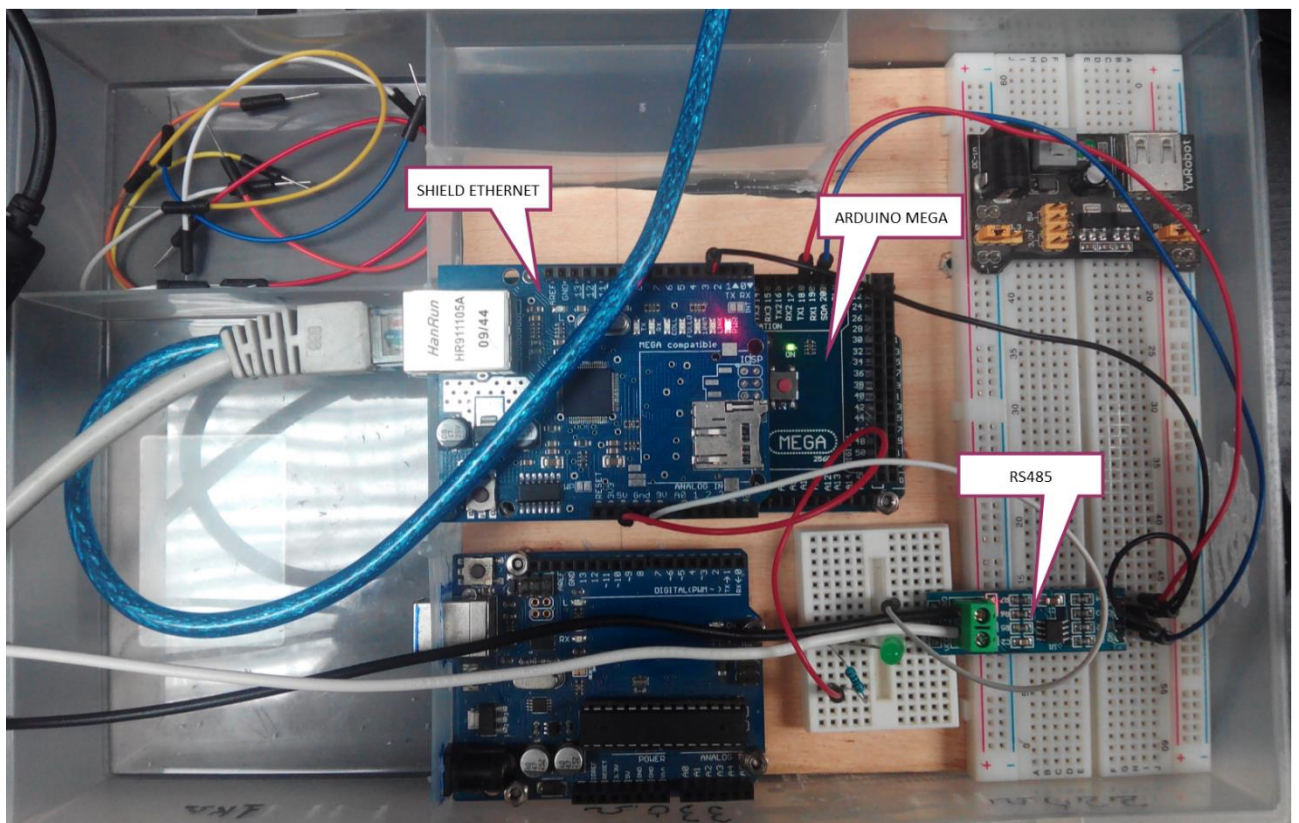
Com es pot observar a la següent imatge gràcies al mètode de fabricació d'Arduino, l'ensamblatge del conjunt es força senzill:



fritzing

Imatge 1: muntatge del conjunt

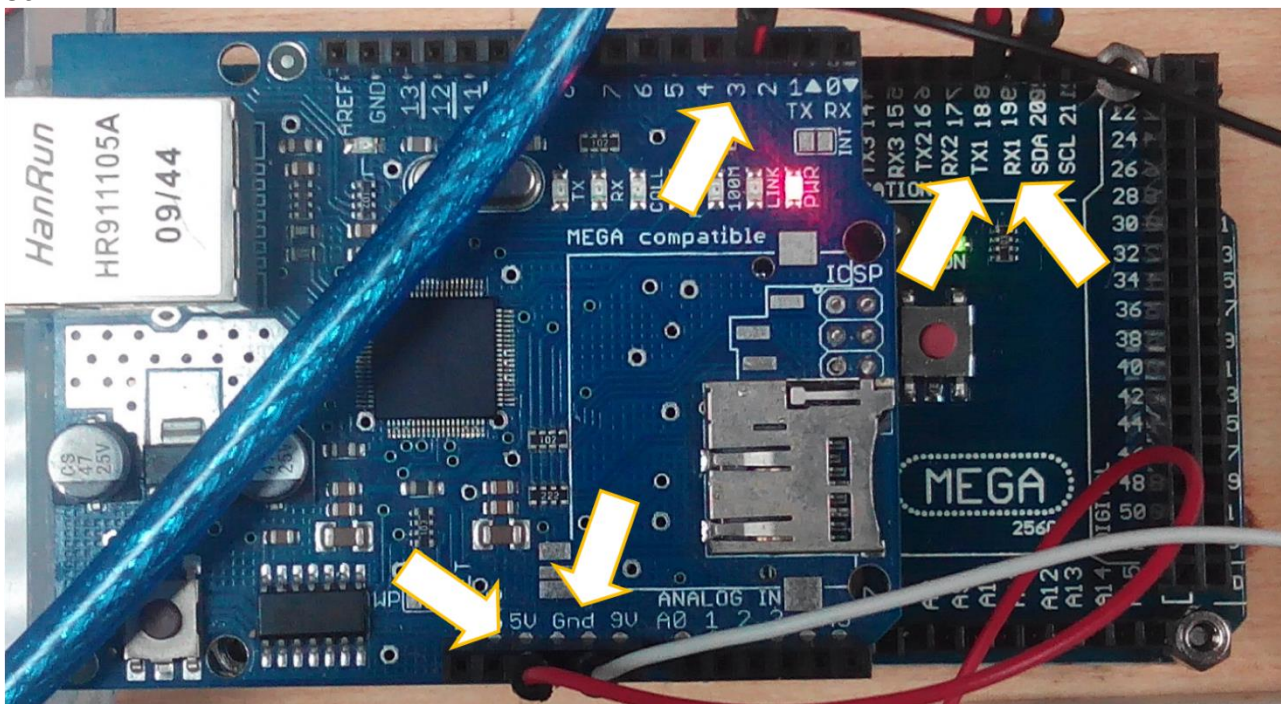
La imatge 11 mostra un esquema de com seria la connexió dels equips abans anomenats, el muntatge real seria així:



Imatge 2: muntatge final

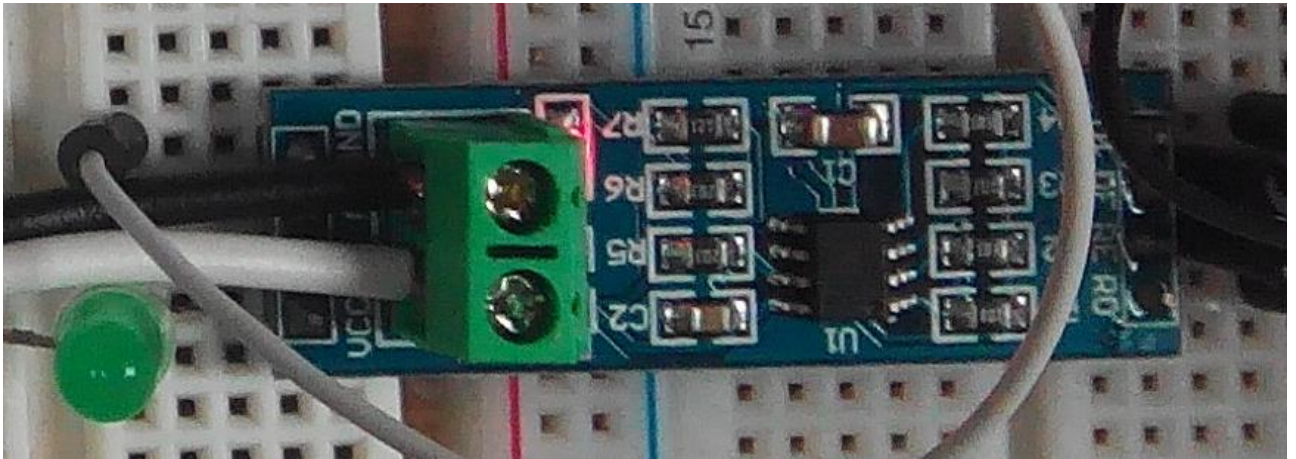
Com es pot observar el sistema ha sigut muntat sobre unes protoboards per a poder realitzar les proves de manera més eficient. Una vegada el prototip es doni per finalitzat es fabricarà el circuit imprès pertinent.

Les connexions dels diferents dispositius són:



Imatge 3: connexions Arduino

PIN	ARDUINO MEGA	RS485
CONTROL	PIN 3	DE + RE
ESCRITURA	SERIAL 1 TX1 – PIN 18	DI
LECTURA	SERIAL 1 RX1 – PIN 19	RO
VCC	5V	VCC
MASA	Gnd	GND



Imatge 4: connexions RS485

I finalment només queden disponibles al mòdul RS485 els pins A i B, que es connectaran als pins de l'esclau corresponent.

Per a poder realitzar les proves i poder fer un debug correcte de les comunicacions realitzades vàrem utilitzar un adaptador USB-RS485.

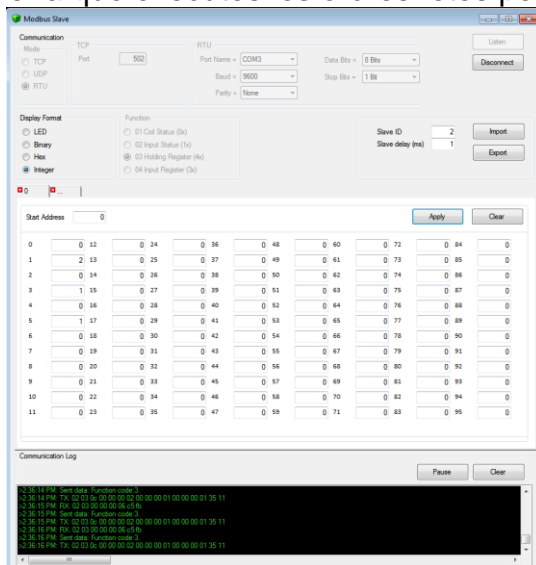


Imatge 5: adaptador USB cap a RS485

Aquest adaptador permet transformar un port USB en una línia serial RS485, d'aquesta forma i amb el programari adequat ens permetria veure quines peticions estaven realitzant. Aquest adaptador te un cost d'uns pocs euros i és una eina força útil. La instal·lació del dispositiu no presenta cap problema, en Windows 7 s'instal·la automàticament, mentre que per a versions posterior requereix de divers específics. En el nostre cas l'instalarem en Windows 8.1, per lo que tinguérem que cercar els divers adequats, que es poden descarregar del http://www.prolific.com.tw/us/showproduct.aspx?p_id=225&pcid=41.

Per a un correcte debug del programa emprarem tres utilitats. Un software que es comportava com un mestre i feia les peticions Modbus, el monitor sèrie del IDE de Arduino i un programa que simula un esclau Modbus.

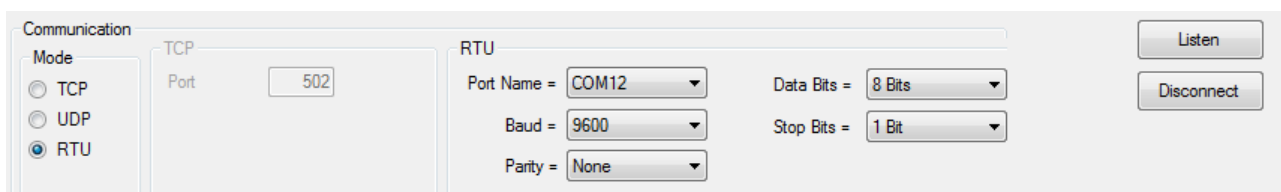
El programa que simula l'esclau Modbus s'anomena Modbus Slave²⁸. Aquest software forma part d'un paquet gratuït que ofereix dues eines, un simulador d'un mestre Modbus i un simulador d'un esclau Modbus. Nosaltres hem emprat únicament l'esclau, per a que executés les ordres fetes per un mestre.



Imatge 6: captura de Modbus Slave

La utilització del programa és força simple, una vegada instal·lat, el configurem per que accepti peticions Modbus.

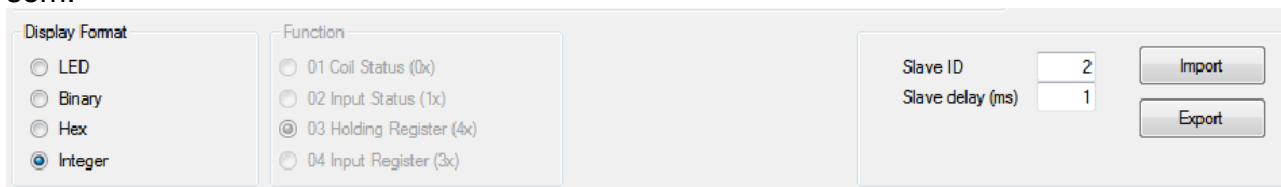
Per a fer açò, una vegada obert el Modbus Slave, en l'apartat de comunicacions seleccionem RTU. Ja seleccionat se'ns desbloquegen les opcions de configuració del RTU. Les deixem en aquests valors:



Imatge 7: valors de configuració

Fet açò és molt important introduir dades als registres que vulguem llegir del Modbus Slave per a que el nostre fals esclau ens torni dades.

Els paràmetres de lectura són: el valor de resposta (enters en el nostre cas), la funció (Read Holding Registers) i com no el nombre que identifica quin esclau som:



Els nom del port variarà depenent del vostre equip, es on estigui instal·lat el vostre adaptador. Una vegada fet açò, punxem en "Listen" i el programa es queda a l'escolta de rebre peticions.

²⁸ <https://modbustool.codeplex.com/releases/view/101146>

Per a saber si estem treballant correctament, canviarem els valors de les adreces de memòria, per a que una vegada resolta la petició, puguem verificar que efectivament funciona.

Start Address	0	12	24	36	48	60	72	84	96	108
0	0	0	0	0	0	0	0	0	0	0
1	2	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0
3	1	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0
5	1	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0

Imatge 8: valors dels registres 1,3 i 5 modificats

Polsem “Apply”.

Ja tenim l’esclau configurat, ara ens queda configurar al mestre. Per a simular el mestre hem emprat el software Indusoft, ja que ens interessava saber si la nostra passarel·la funcionaria igual que les passarel·les comercials, i per a això res millor que provar-ho en un entorn el més real possible.

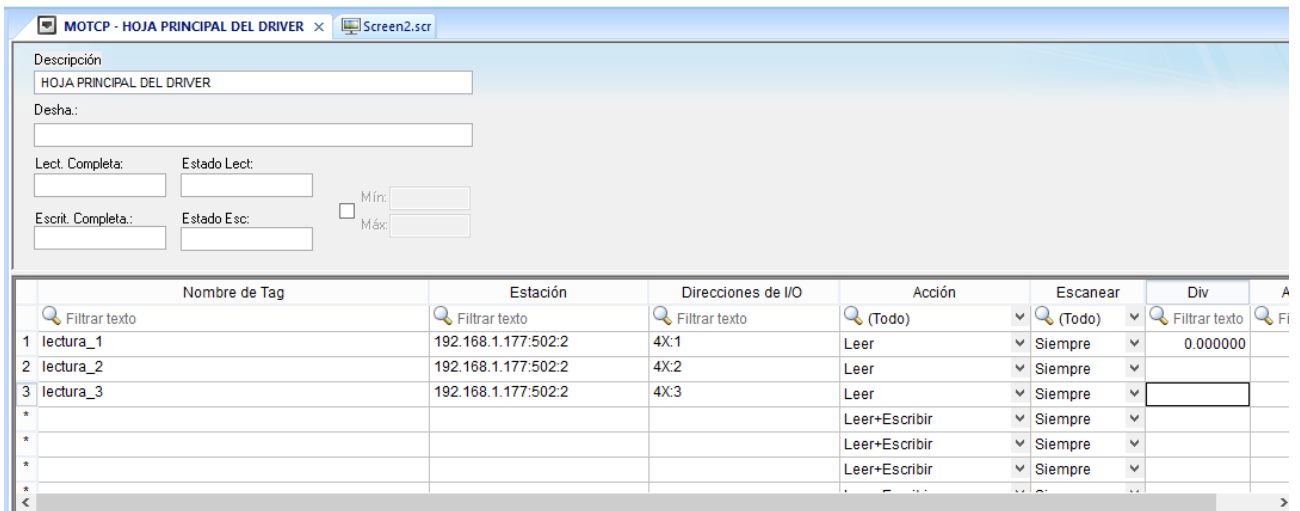
Es pot descarregar²⁹ una versió de prova amb 40 hores de desenvolupament gratuït que es més que suficient per a fer les proves. Per a poder descarregar el programa cal registrar-se a la web. Una vegada descarregat e instal·lat el programa generarem un nou projecte.

Generar un nou projecte es força senzill, de fet el projecte que anem a fer serà una petita pantalla que obtindrà tres valors d’un esclau Modbus, les peticions seran constants, per lo que no caldrà generar events ni alarmes, ja que l’única funció del projecte serà el debug.

Una vegada creat el nou projecte crearem els Tags on es guardaran els valors que anem a llegir, en aquest cas hem emprat tres noms prou clars: lectura_1, lectura_2 i lectura_3.

El següent pas es incloure el driver de comunicació necessari per a poder comunicar i obtenir la informació.

²⁹ <http://www.indusoft.com/Products-Downloads>

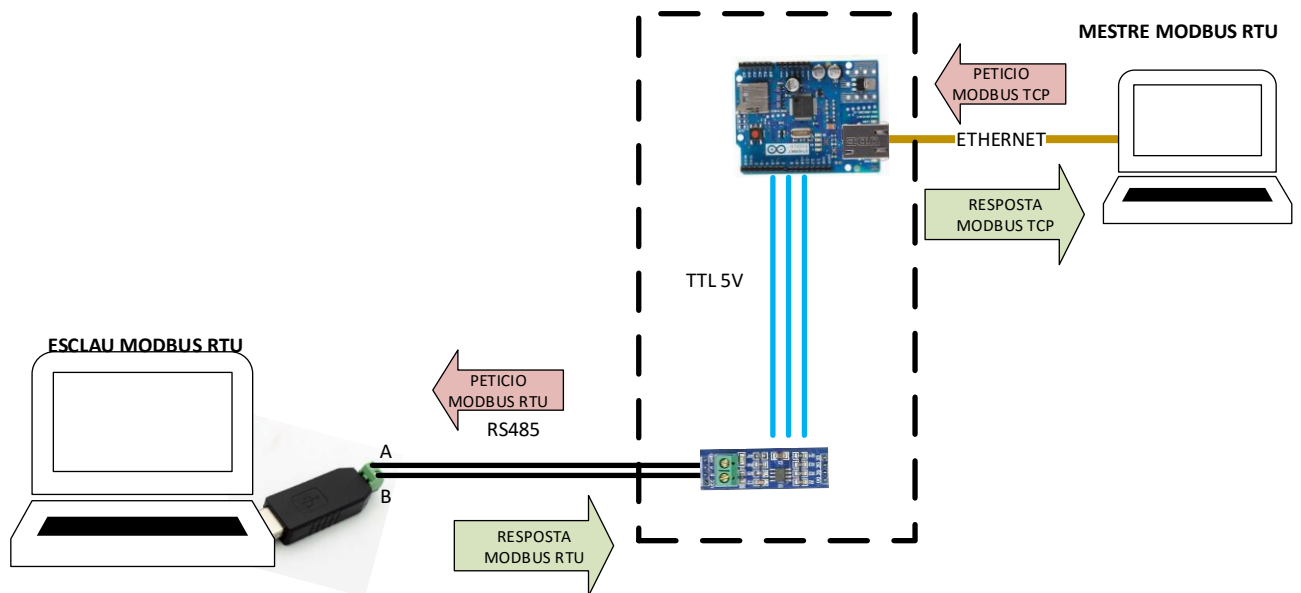


Imatge 9: fulla de drivers completada

Com a tercera part del nostre sistema de debug emprem el monitor sèrie del IDE d'Arduino, que ens permet veure en tot moment quina informació està entrant i sortint del Arduino, mitjançant missatges impresos dirigits al port sèrie.

Finalment unim mitjançant un parell de cables de coure l'adaptador USB – RS485 amb RS485 Serial Communicator de l'Arduino.

Aquesta connexió es simple, es connecta el cable A amb el A i el B amb B, no cal crear el cables.



Imatge 10: esquema de connexió del sistema

3.- Estructura dades MySQL

Taula 'users'.

Atribut	Tipus de dades	Longitud	Funció
user_id	Int	11	Identificador del usuari
email	Varchar	100	Dades d'usuari
password	Varchar	10	Dades d'usuari
nick	Varchar	100	Dades d'usuari

Taula 'listado_analizadores'.

Atribut	Tipus de dades	Longitud	Funció
'id'	Int	11	Identificador del analitzador
'id_esclavo'	int	11	Nombre d'esclau
'IP_esclavo'	Varchar	50	Adreça IP del esclau
'nombre'	Varchar	50	Nom de l'equip
'activado'	tinyInt	1	0= equip desactivat 1= equip activat
'enFallo'	tinyInt	1	0= equip OK 1= equip en fallo
'enAlarma'	tinyInt	1	0= equip OK 1= equip en alarma
'estado'	Int	5	0 = equip desactivat 1 = equip OK 5 = equip en alarma

Taula "analitzador_1"

Atribut	Tipus de dades	Longitud	Funció
'id_lectura'	Int	11	Identificador del analitzador
'dateTime'	timeStamp		Data i hora de la inserció
'tension_fase_1'	decimal	10,3	Dades lectura
'corriente_fase_1'	decimal	7,3	Dades lectura
'tension_fase_2'	decimal	10,3	Dades lectura
'corriente_fase_2'	decimal	7,3	Dades lectura
'tension_fase_3'	decimal	10,3	Dades lectura

'corriente_fase_3'	decimal	7,3	Dades lectura
'energiaActiva'	double		Dades lectura
'energiaReactivaInductiva'	double		Dades lectura
'energiaReactivaCapacitativa'	double		Dades lectura
'energiaAparente'	double		Dades lectura
'frecuencia'	decimal	10,1	Dades lectura
'tension_L1_L2'	decimal	10,1	Dades lectura
'tension_L2_L3'	decimal	10,1	Dades lectura
'tension_L3_L1'	decimal	10,1	Dades lectura
'THD_V_L1'	decimal	10,1	Dades lectura
'THD_V_L2'	decimal	10,1	Dades lectura
'THD_V_L3'	decimal	10,1	Dades lectura
`THD_A_L1`	decimal	10,1	Dades lectura
`THD_A_L2`	decimal	10,1	Dades lectura
`THD_A_L3`	decimal	10,1	Dades lectura
`potenciaActiva`	int	11	Dades lectura
`potenciaInductiva`	Int	11	Dades lectura
`potenciaCapacitativa`	Int	11	Dades lectura
`cosenoPhi`	decimal	7,2	Dades lectura
`factorPotencia`	decimal	7,2	Dades lectura

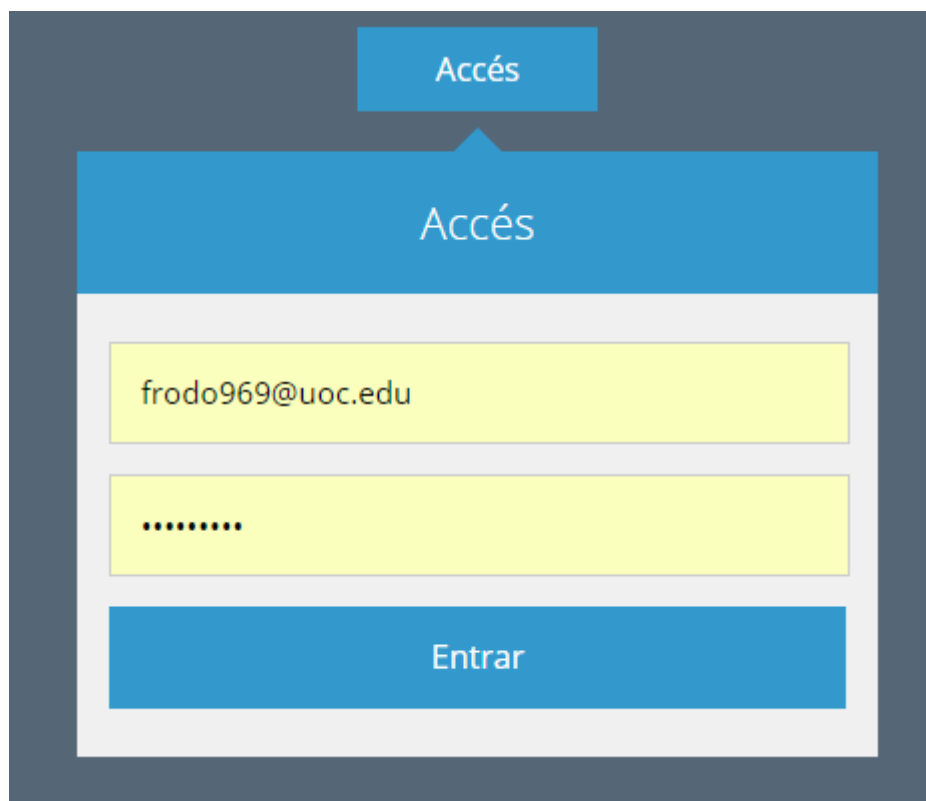
4.- Manual d'usuari

En primer lloc cal inicialitzar el programa a la Raspberry per a que comenci la captura de dades. Iniciem sessió be de forma presencial o si hem configurat l'accés remot, en el terminal de la Raspberry escriguem lo següent:

```
python python_programicas/sion2.py
```

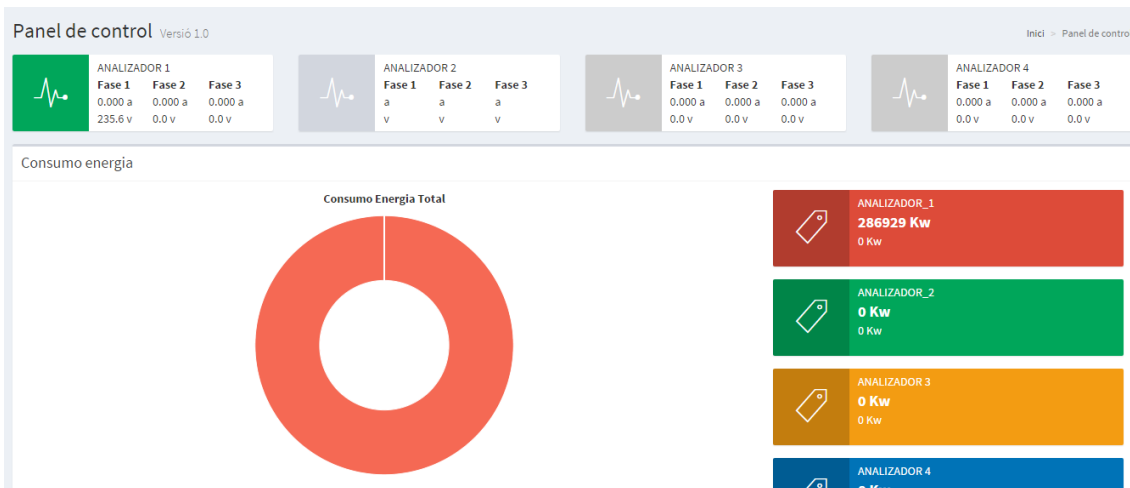
En eixe moment el programa s'executa.

En el navegador web escrivim l'adreça web de la nostra Raspberry Pi, en este cas 192.168.2.79. Automàticament apareixerà la pàgina de control d'accés:



The image shows a web browser window displaying a login page. At the top, there is a blue button labeled 'Accés'. Below it, the page has a blue header with the word 'Accés' in white. The main content area is white and contains two yellow input fields. The first field contains the email address 'frodo969@uoc.edu'. The second field contains a series of dots representing a password. Below the input fields is a blue button labeled 'Entrar'.

Introduïm el nom d'usuari i el mot de pas per a tenir accés a l'aplicació. Si les dades són correctes apareixerà la pantalla d'inici.



En la part superior tenim les dades dels analitzadors existents a l'aplicació. En aquest cas l'aplicació disposa de 4 analitzadors. En concret el primer analitzador es troba en funcionament, i la resta estan desactivats. En concret estos són els estats que pot mostrar cada analitzador:

Analitzador en Marxa	Analitzador Desactivat	Analitzador en Alarma

A la part de la dreta es poden llegir les dades de cada analitzador en referència al consum (energia activa consumida) i la potència activa. Cada analitzador es presentat a la gràfica mitjançant un color. La suma de tots els consums es representa a la gràfica de dònut.



Per a accedir a totes les dades de l'analitzador cal polsar sobre la icona del analitzador que desitgem. La següent pàgina que ens obri són les dades de l'analitzador.



Com es pot observar hem pulsat sobre un analitzador que està en Alarma. La línia roja en la part superior indica tal situació. Eixa línia tindrà el color del estat de l'equip en concordància amb els estats de la taula anterior. Desde aquesta pàgina podem obtenir la gràfica de la corrent, ompliguent les dades del següents camps:

Fecha Inicio

dd/mm/aaaa

--:--

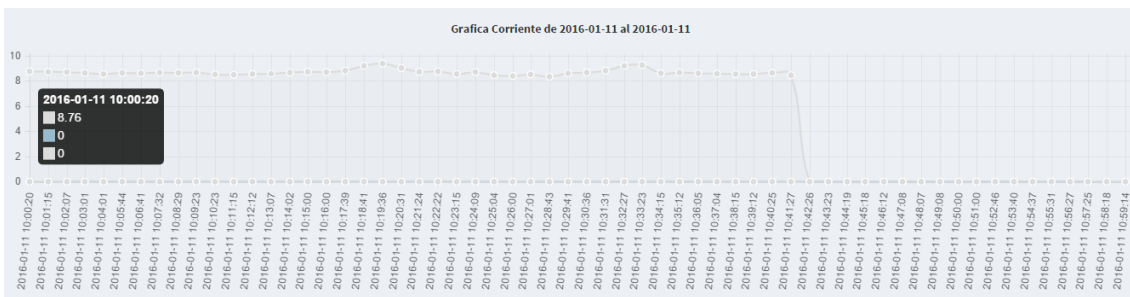
Fecha Fin

dd/mm/aaaa

--:--

Gràfica Corriente

Si el interval de temps es superior a 1 hora mostrarà dades de cada hora, mentres que si el interval es menor les dades seran de cada minut.



També podem modificar les dades de connexió, així com desconnectar el analitzador.

Ajustes

ID esclavo

1

IP esclavo

192.168.2.177

Nombre Analizador

analizador_1

Encendido Permitir Cambios Guardar Cambios

Per a poder realitzar canvis en aquest apartat cal premer el botó de permetir canvis. Aquest botó existeix per a no realitzar canvis per error.