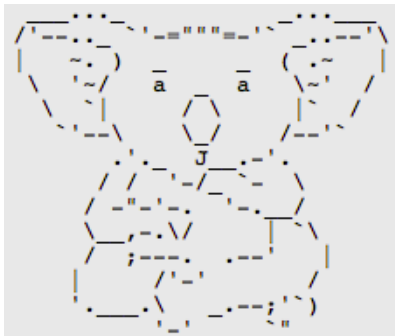


Manual passarel·la Java “To Yes”



Índex

| | |
|--|---|
| Manual passarel·la Java “2Yes” | 4 |
| Previ..... | 4 |
| 1 Utilització del programa | 4 |
| 2 Estructura | 5 |
| 2.1 Recollida de dades..... | 5 |
| 2.2 Comunicació amb el aplicatiu web..... | 5 |
| 3 Modificació del programa | 5 |
| 3.1 Previ..... | 5 |
| 3.2 Modificació del port del servidor..... | 5 |
| 3.3 Modificació del nom del fitxer de senyals..... | 5 |
| 3.4 Modificació de la ubicació del fitxer de senyals..... | 6 |
| 3.5 Modificació de la ubicació dels fitxers històrics..... | 6 |
| 3.6 Modificació de la forma d'execució de les classes..... | 6 |
| 3.6.1 Previ..... | 6 |
| 3.6.2 Modificacions..... | 6 |
| 4 Referències d'interès..... | 6 |

Índex de grafics

| | |
|---|---|
| Figura 1: Inici del programa ToYes..... | 4 |
| Figura 2: Esquema de la solució proposada | 5 |
| Figura 3: Recepció de dades i comunicació amb el aplicaitu YES..... | 6 |

Manual passarel·la Java “2Yes”

Previ

Aquest programa neix de la necessitat de intercomunicar el dispositiu ConnectportX2 de Digi, amb el sistema de recollida de dades del aplicatiu web YES (Your Energy System) del que disposa DsetSolutionsS.L.

1 Utilització del programa

Aquest programa no necessita cap requeriment per funcionar, sols cal executar-lo.

El programa automàticament es posara a l'espera de comunicacions per part del ConnectportX2, per altre banda anirà passant-li la informació al aplicatiu YES periòdicament (Fig .1).

```
0 [main] INFO root -
1 [main] INFO root -
1 [main] INFO root -
1 [main] INFO root -
1 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
2 [main] INFO root -
1672 [Timer-0] INFO root - Comença el procés: Tue Aug 14 09:32:32 CEST 2012
1674 [Timer-0] INFO root - Identificant instal·lació:
1674 [Timer-0] INFO root -   usuari: I585
1674 [Timer-0] INFO root -   password: *****
1814 [Timer-0] INFO root - Identificació OK
1814 [Timer-0] INFO root - Validant taula:
```

Fig 1: Inici del programa ToYes

2 Estructura

El projecte esta estructurat en dos grans parts :

- ◆ La recollida de dades del Connectport X2
- ◆ La comunicació amb el aplicatiu web YES

2.1 Recollida de dades

Aquesta part l'efectua la classe anomenada Bridge.java, aquest modul es comunica mitjançant “sockets” amb el ConnectportX2 i bàsicament el que fa és rebre i crear en el sistema, els fitxers “Signals.txt” i tots els històrics de les senyals creats pel ConnectportX2 (Fig2).

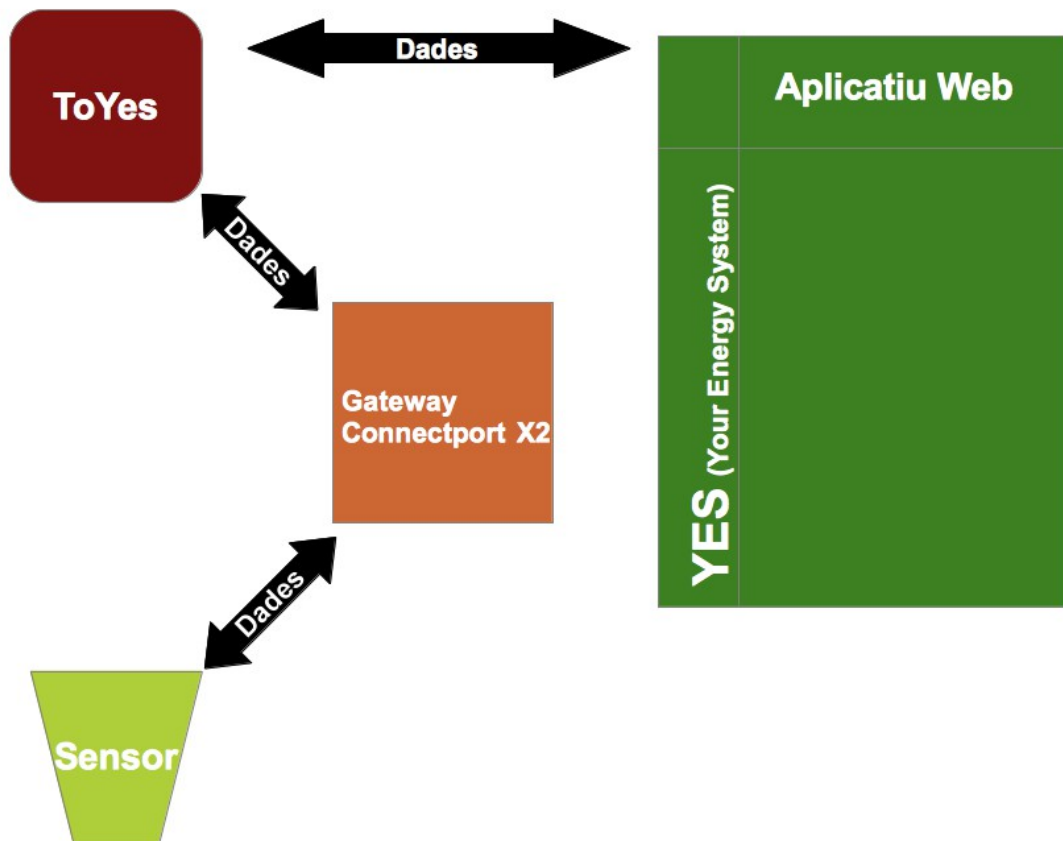


Fig 2: Esquema de la solució proposada

2.2 Comunicació amb el aplicatiu web

Aquesta part es un conjunt de mòduls pre-dissenyats pels enginyers de DsetSolutions S.L per establir una connexió web service amb el aplicatiu YES de l'empresa.

En aquest punt cal remarcar que la classe modificada per la implementació del projecte es la “*FileData.java*”, que es l'encarregada de adquirir la informació dels fitxers creats per la classe “*Bridge.java*” i enviar-la amb el format correcte al aplicatiu YES.

```
1800152 [Timer-0] INFO root - Validació taula OK
1800152 [Timer-0] INFO root - Ultims valors:
1800402 [Timer-0] INFO root - Últim valor del Codi senyal: 69, Mon Aug 27 23:41:50 CEST 2012 UNIX TIME 1346103710000 53.0
1800402 [Timer-0] INFO root - Data actual amb format UNIX : 1346103710000

1800402 [Timer-0] INFO root - Volcat històrics de la senyal 69:
1800949 [Timer-0] INFO root - Volcat històrics de la senyal 69 OK
1800949 [Timer-0] INFO root - S'ha completat correctament el volcat de dades a l'aplicatiu YES
1800949 [Timer-0] INFO root - *****Finalitza el procés: Wed Aug 29 12:19:39 CEST 2012*****
1800949 [Timer-0] INFO root - *****
1808589 [Thread-0] INFO root - ***** Atencio : Connexio entrant al port 10102 *****
1808605 [Thread-0] INFO root - S'estan rebent dades del captador de dades ...
1808605 [Thread-0] INFO root - S'esta rebent el fitxer historial
1826822 [Thread-0] INFO root - *****Finalització del proces de comunicacio amb l aplicatiu YESWed Aug 29 12:20:05 CEST 2012*****
1826822 [Thread-0] INFO root -
*****
```

Fig 3: Recepció de dades i comunicació amb el aplicatiu YES

3 Modificació del programa

3.1 Previ

El projecte esta dissenyat amb Netbeans 7.1 i programat amb Java.

3.2 Modificació del port del servidor

Per poder canviar el port d'escolta del servidor cal dirigir-se a la classe “*Bridge.java*” i en al línia

➤ `ServerSocket servidor = new ServerSocket(10102);`

canviar el numero 10102 (port actual) pel port que es desitgi.

3.3 Modificació del nom del fitxer de senyals

Per modificar el nom del fitxer cal situar-se dins la classe “*Bridge.java*” i variar el nom de a la línia :

➤ `File fileSignals = new File("SignalList.txt");`

un exemple de com quedaria la línia modificada :

➤ `File fileSignals = new File("LlistaSenyals.tif");`

3.4 Modificació de la ubicació del fitxer de senyals

Per modificar la ubicació cal situar-se dins la classe “*Bridge.java*” e indicar una nova ruta en la línia :

➤ `File fileSignals = new File("SignalList.txt");`

un exemple de com quedaria la línia modificada :

➤ `File fileSignals = new File("/Users/Test/SignalList.txt");`

3.5 Modificació de la ubicació dels fitxers històrics

Per modificar d'ubicació cal situar-se dins la classe “*Brigde.java*” e indicar una nova ruta en la línia :

➤ `File fileHisto = new File(inputLine + ".txt");`

un exemple de com quedaria la línia seria :

➤ `File fileHisto = new File("/Users/Test" + inputLine + ".txt");`

3.6 Modificació de la forma d'execució de les classes

3.6.1 Previ

El programa en la classe “*ClientWs.java*” crea dos fills i els executa en paral·lel, el primer fill que es la classe “*Bridge.java*” disposa de un bucle “*While*” el qual el fa restar a l'espera fins l'entrada de noves peticions de “*socket*”, per altre banda, el segon fill crida la classe “*MainProces.java*” que es la comunicació amb el servidor la qual esta regulada per un “*Timer*” amb una durada per defecte de 5 minuts.

3.6.2 Modificacions

Per modificar aquests dos estàndards cal anar a la classe “*ClientWs.Java*” i realitzar les modificacions pertinents.

4 Referencies d'interès

Digi Your M2M Solutions Expert, Digi
www.digi.com

Digi Your M2M Solutions Expert – Digi international Support Forum, Digi
<http://forums.digi.com/>

The Java EE 5 Tutorial, Oracle
<http://docs.oracle.com/javaee/5/tutorial/doc/>

How To SLOTHX2



Pere Comellas Pujols

comellaspere@gmail.com

Índex de contingut

| | |
|---|---|
| How to SlothX4..... | 3 |
| Previ..... | 3 |
| Característiques del Programa..... | 3 |
| Utilització del programa..... | 4 |
| Previ..... | 4 |
| Accedir al dispositiu..... | 4 |
| Servei Web de iDigi..... | 4 |
| Telnet..... | 5 |
| Modificació del Programa..... | 6 |
| Modificació de senyals..... | 6 |
| Inserció/modificació de variables Modbus..... | 6 |
| Inserció..... | 6 |
| Modificació..... | 6 |
| Modificació Ip/port de la passarel·la “Bridge”..... | 7 |
| Compilació i enviament del programa en el dispositiu..... | 7 |
| ESP DIGI..... | 7 |
| Manualment..... | 7 |
| Visualització/Obtenció de fitxers..... | 8 |
| Inicialització manual del programa..... | 8 |
| Bibliografia..... | 9 |

Índex d'il·lustracions

| | |
|--|---|
| Figura 2: Web loggin de iDigi..... | 5 |
| Figura 3: Panell de control www.idigi.com..... | 6 |
| Figura 4: Opcions Telnet Connectport X2..... | 6 |
| Figura 5: Segment de codi on introduir els canvis..... | 7 |
| Figura 6: Segment de codi mostrant "host" i "port" | 8 |
| Figura 7: Pàgina de propietats del dispositiu..... | 9 |

How to SlothX4

1 Previ

SlothX4 és una aplicació creada amb l'IDE(Entorn de desenvolupament integrat) “Digi ESP for Python 1.4.0” utilitzant el llenguatge de programació “Python 2.4.0” que és el que suporta el dispositiu objectiu del programa, el ConnectportX2 de DIGI.

2 Característiques del Programa

El programa en la seva configuració inicial enquesta a un esclau que utilitza la tecnologia de comunicacions “Modbus” amb una **freqüència d'una hora**.

Cal remarcar que hi ha la possibilitat d'afegir nous esclaus i noves variables per esclau però tot i així **la freqüència de mostreig serà sempre la mateixa**.

El programa s'anomena SlothX, aquest, envia les dades a un servidor extern on hi ha carregat un programa anomenat “ToYes” el qual és l' encarregat d'enviar les dades al servidor web YES (Your Energy System).

Per més informació sobre l'aplicatiu “ToYes” cal consultar el “Manual d'us de l'aplicatiu ToYes”

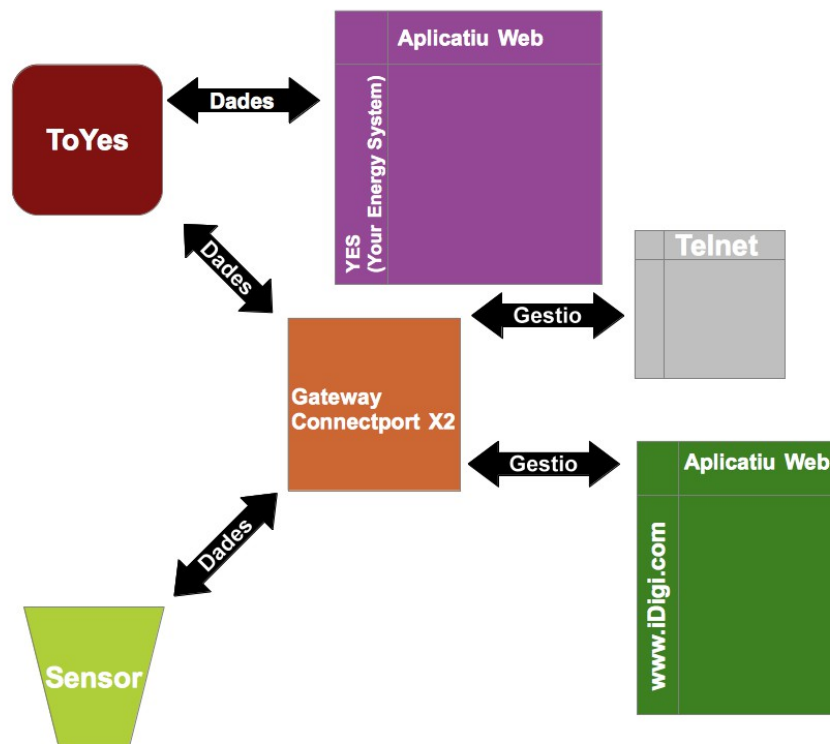


Fig 1: Esquema de la solució proposada

3 Utilització del programa

3.1 Previ

El programa està dissenyat de tal manera que només encendre el dispositiu ja es posa en marxa automàticament. Aquest, cada quinze minuts s'encarrega de recollir les dades dels sensors i les envia a la passarel·la ToYes. Tota la informació de les accions del programa la recull en un fitxer tipus "LOG" situat a "WEB/PYTHON/log".

3.2 Accedir al dispositiu

Es pot accedir al dispositiu de dos maneres, mitjançant el servei web de iDIGI o via Telnet.

3.3 Servei Web de iDIGI

L'empresa DIGI proporciona per alguns dels seus dispositius un servei WEB on es pot veure la informació d'aquest i accedir a les dades, així com modificar alguns paràmetres en concret del dispositiu. Per accedir via web cal entrar a www.idigi.com e introduir l'usuari i contrasenya "perakus" "dset" (Aquest usuari i contrasenya son temporals).

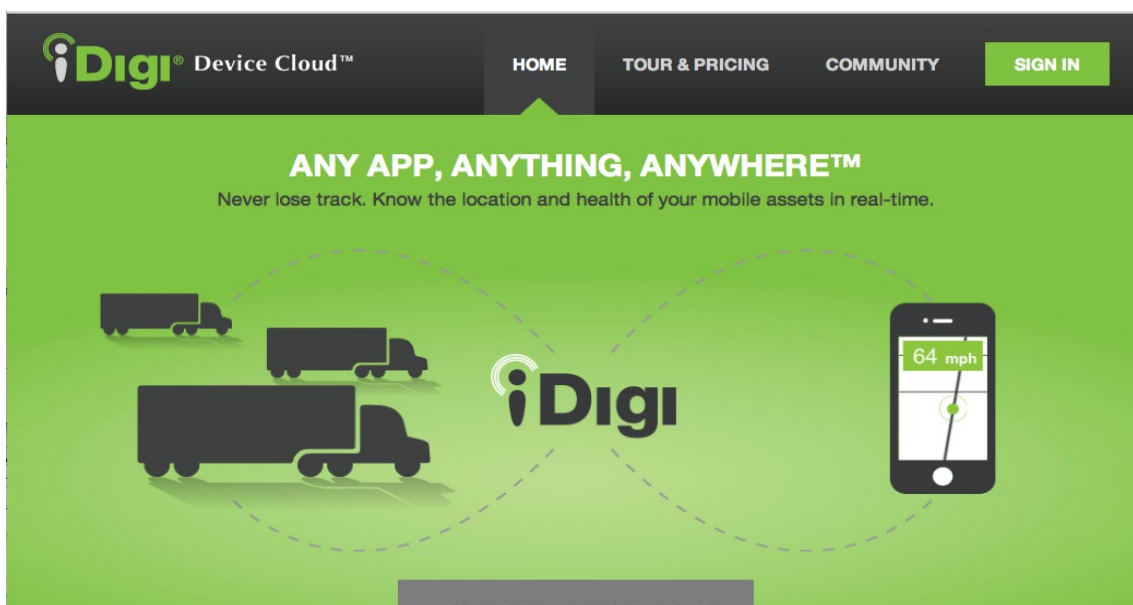


Fig 2: Web loggin de www.idigi.com

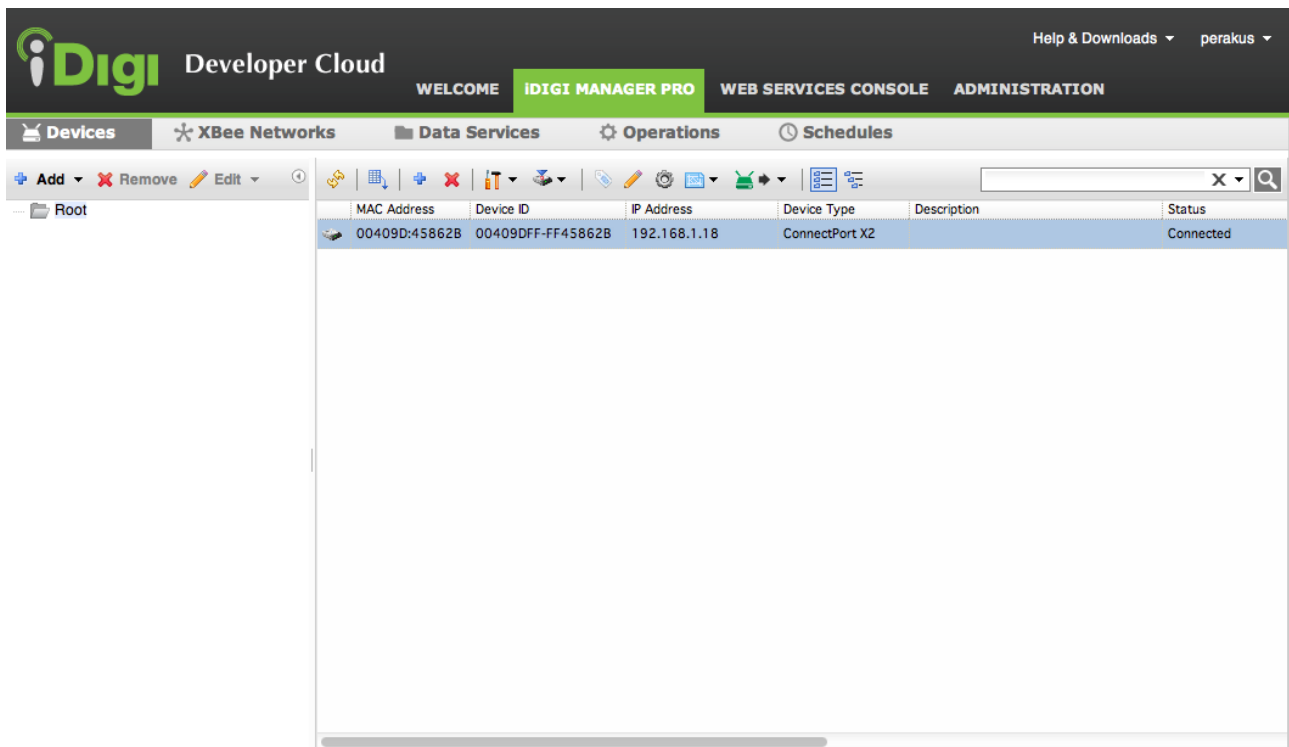


Figura 3 : Panell de control del web www.idigi.com

3.4 Telnet

Per accedir via Telnet, és necessari el nom d'usuari i la clau per defecte del dispositiu, aquests són “root” i “dbps”.

Un cop dins, cal prémer la tecla “?”, tot seguit apareixeran les diferents opcions del dispositiu.

```

PeRaKuS — telnet — 80x24
Last login: Wed Aug 8 01:34:54 on ttys001
admins-MacBook:~ PeRaKuS$ telnet 192.168.1.18
Trying 192.168.1.18...
Connected to 192.168.1.18.
Escape character is '^]'.

login: root
password:
#>

backup    boot      certmgmt  close     display   exit      help
info     kill      newpass   ping      python    quit      revert
set      show     status    who       xbee     ?

#>

```

Figura 4 : Opcions Telnet de ConnectportX2

4 Modificació del Programa

La única forma existent de modificar el funcionament del programa és la escriptura/modificació del codi, es recomana realitzar una còpia de seguretat abans de qualsevol modificació.

4.1 Modificació de senyals

Per inserir nous “Signals” és necessari utilitzar la funció :

- `AddSignal(self,SetSlavesDef,CodiSenyal,NomSenyal,NomFitxerHistoric)`

aquesta funció s'ha d'utilitzar sobre l'objecte “*ObjecteLlista*” ja que cal adherir-la dins la llista de “*Signals*” del sistema, aquesta acció es realitzarà de la següent forma:

- `ObjecteLlista.AddSignal(self,SetSlavesDef,CodiSenyal,NomSenyal,NomFitxerHistoric)`

cal remarcar que aquesta línia de codi s'afegirà després de la línia

- `ObjecteLlista = SignalList()`

i abans de la línia

- `LlistaSignals = ObjecteLlista.ReturnList()`

a sota es mostra una captura de pantalla per fer-ho més entenedor.

```
44 #ObjecteLlista = IniParameters()
45 #LlistaEsclaus = ObjecteLlista.GetInitParameters() #Constructor defecte sense res ??
46
47 #Creacio de llista de senyals
48 ObjecteLlista = SignalList()
49 #Insercio Singals = dispositius a la llista
50 ObjecteLlista.AddSignal(SetSlaveDef("192.168.125.69",502,0,1,1,0,0,5),69,"ModBusDset","HModbusDset")
51 #Si volem inserir mes Singals fer-ho aqui a sota
52
53
54
55 #
56 #Creem una llista de senyals
57 LlistaSignals = ObjecteLlista.ReturnList()
```

Fig 5: Segment de codi on introduir els canvis

4.2 Inserció/modificació de variables Modbus

4.2.1 Inserció

Per poder inserir variables caldrà afegir un nou “*Signal*” amb la mateixa “*IP*”, per aconseguir-ho s'afegirà la línia :

- `ObjecteLlista.AddSignal(self,SetSlavesDef,CodiSenyal,NomSenyal,NomFitxerHistoric)`

cal remarcar que el paràmetre “*SetSlavesDef*” és on es té que fer el canvi de variable, a sota mostrem un exemple de la línia.

- `SetSlaveDef("192.168.125.69",502,0,1,1,0,0,5),69,"ModBusDset","HModbusDset")`

4.2.2 Modificació

Per poder modificar una variable cal canviar la línia que crea el seu “Signal” en l'apartat “*SetSlavesDef*” més concretament.

4.2.3 Modificació Ip/port de la passarel·la “ToYes”

Per poder canviar la IP per defecte del servidor on està allotjada la passarel·la YoYes , cal obrir la classe “*SocketConnections.py*” del projecte SlothX4 i modificar la “IP” de la variable “*host*” i per canviar el port caldrà modificar el port de la variable “*port*”.

```
23 |         #Dades server
24 |         host="192.168.1.11"
25 |         port =10102
?c
```

Fig 6: Segment de codi mostrant els camps “*host*” i “*port*”

4.2.4 Compilació i enviament del programa en el dispositiu

Per compilar i enviar el programa al dispositiu es pot utilitzar dos mètodes

4.2.4.1 Kit de desenvolupament “ESP DIGI”

Dins la suite de Digi existeix la opció d'enviar directament el programa al dispositiu. Per poder-ho fer cal configurar el dispositiu en la opció “Propietats del projecte”, i seguir els següents passos :

- ◆ Click secundari sobre “Propietats del projecte”, seguidament seleccionem propietats
- ◆ En el menú esquerra de navegació cal escollir l'opció "*Digi Python Properties*"
- ◆ Click sobre “*New*” , seguidament s'obrirà l'assistent de nou dispositiu remot.
 - ◆ En aquest assistent s'haurà de configurar les dades del dispositiu i les del compte de iDigi per tal d'establir un vincle entre l'IDE i el dispositiu.

4.2.5 Manualment

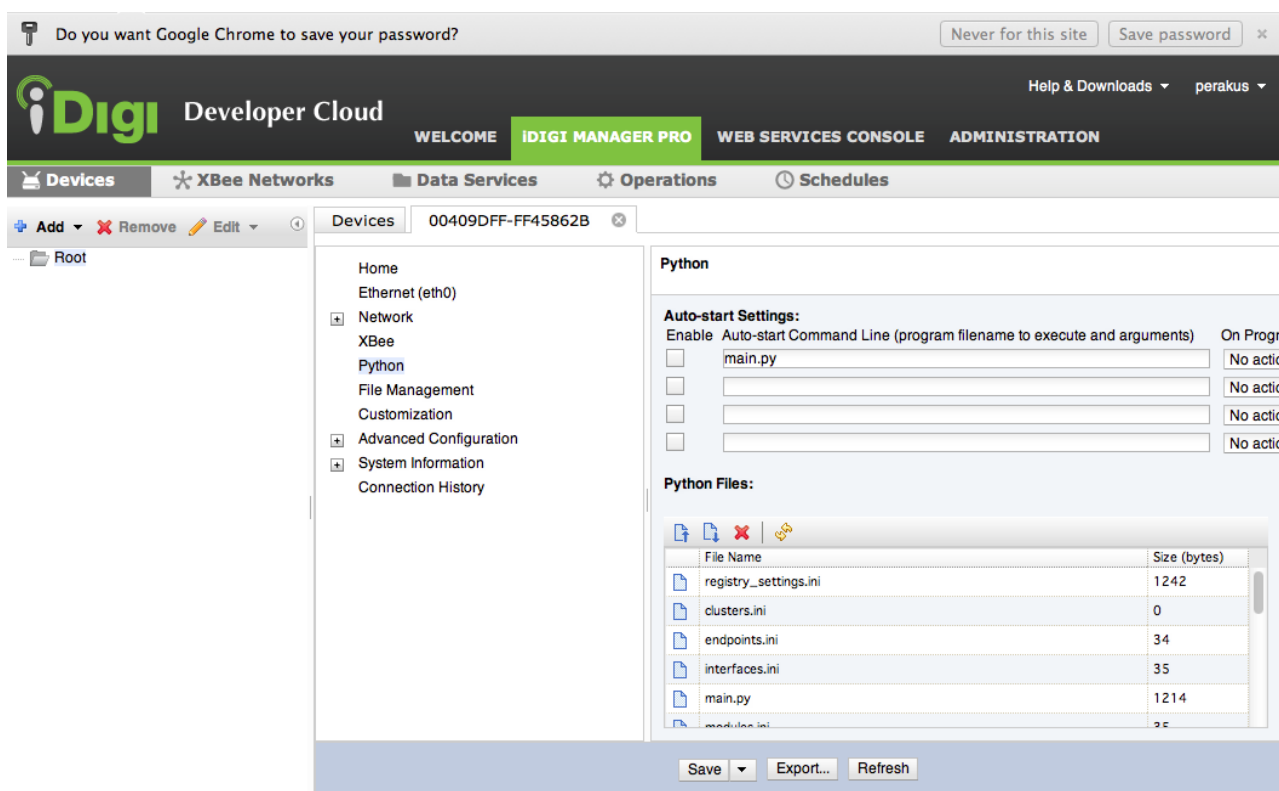
A la wiki de Digi apareix com compilar manualment i carregar el programa en el dispositiu, al no utilitzar-se cap cop no s'ha comprovat la fiabilitat de l'operació, a sota es mostra el link del procediment.

- http://www.digi.com/wiki/developer/index.php/Digi_Python_Programmer%27s_Guide
- La informació apareix en el punt “*Loading Python Programs onto a Digi Device*)

5 Visualització/Obtenció de fitxers

Per poder visualitzar o aconseguir els fitxers que crea el dispositiu caldrà que es realitzi mitjançant el Web de iDigi. Per aconseguir-ho es seguiran aquests passos :

- ◆ Entrar en el servei web www.idigi.com
- ◆ Entrar a les propietats del dispositiu que es desitgi ja sigui fent dos clics a sobre o seleccionant les propietats d'aquest amb el click secundari del ratolí.
- ◆ Seleccionar del menú de navegació esquerra la opció “Python”
 - ◆ Seleccionar el fitxer desitjat i seguidament clicar a “Save”



The screenshot shows the iDigi Developer Cloud interface. The top navigation bar includes 'WELCOME', 'IDIGI MANAGER PRO', 'WEB SERVICES CONSOLE', and 'ADMINISTRATION'. The left sidebar shows a tree view with 'Python' selected. The main content area displays the 'Python' configuration page for device '00409DFF-FF45862B'. It includes 'Auto-start Settings' with a table for enabling programs and 'Python Files' with a table listing files and their sizes.

| File Name | Size (bytes) |
|-----------------------|--------------|
| registry_settings.ini | 1242 |
| clusters.ini | 0 |
| endpoints.ini | 34 |
| interfaces.ini | 35 |
| main.py | 1214 |
| modules.ini | 25 |

Fig 7: Pàgina de propietats del dispositiu

6 Inicialització manual del programa

Per inicialitzar el programa manualment caldrà entrar en el dispositiu via “Telnet”. , un cop a dins cal teclejar

- `python dpdsrv.py`, automàticament s'executarà el programa SlothX2

7 Enllaços d'interès

Digi Your M2M Solutions Expert, Digi
www.digi.com

Digi Your M2M Solutions Expert – Digi international Support Forum, Digi
<http://forums.digi.com/>

Digi Wiki for Developers, Digi
http://www.digi.com/wiki/developer/index.php/Main_Page

Online Support Request, Digi
www.digi.support/eservice

The Python Wiki, Python Software Foundation
<http://wiki.python.org/moin/>

Python Documentation, Python Software Foundation
<http://docs.python.org/>