

Intalar las herramientas de desarrollo Qt en el RPI

El primer paso es instalar las herramientas de desarrollo Qt en el RPI. El ultimo comando del codigo siguiente instala una suite completa de herramientas que necesita entre 60 y 200 MB:

```
apt-cache search qt5
sudo apt install qt5-default
```

Podemos comprobar la version de la instalacion usando lo siguiente:

```
qmake -version
```

A continuacion haremos una prueba que mostrara por pantalla una venta con el texto “Hola mundo!”.

El codigo es el siguiente:

```
#include <Qapplication>
#include <Qlabel>

int main(int argc, char *argv[]){
    Qaplication app(argc, argv);
    QLabel label(“Hola mundo!”);
    label.resize(200, 100);
    label.show();
    return app.exec();
}
```

Lo guardamos como simpleQt.cpp, este archivo sera el unico obligatorio en el directorioantes de que los pasos siguientes tengan lugar. El generador makefile multiplataforma qmake se puede usar entonces para crear un proyecto predeterminado.

```
~/exploringrpi/chap14/simpleQt $ ls
simpleQt.cpp
~/exploringrpi/chap14/simpleQt $ qmake -project
~/exploringrpi/chap14/simpleQt $ ls
simpleQt.cpp simpleQt.pro
~/exploringrpi/chap14/simpleQt $ more simpleQt.pro
```

// copiar aquí el .pro

Este archivo de proyecto describe los ajustes de proyecto y, si es necesario, se puede editar para añadir dependencias adicionales. En este caso, la linea:

```
QT += widgets
```

se debe añadir al archivo .pro (por ejemplo, entre las lineas TEMPLATE y TARGET), ya que, de otro modo, las librerias necesarias para mostrar los widgets de la interfaz grafica de usuario no se enlazarian correctamente. El generador qmake Makefile se puede ejecutar de nuevo, solo que esta vez sin el argumento -project :

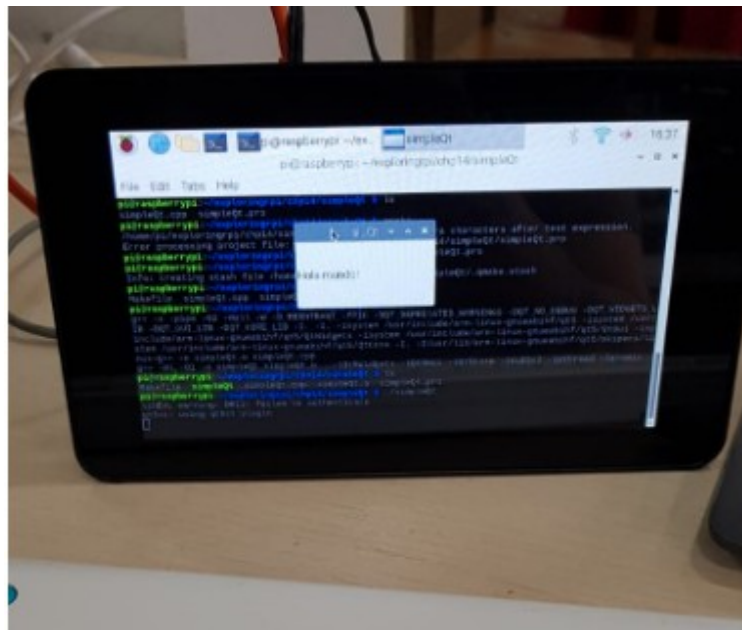
```
~/exploringrpi/chap14/simpleQt $ qmake
~/exploringrpi/chap14/simpleQt $ ls
Makefile simpleQt.cpp simpleQt.pro
```

Este paso tiene como resultado la creación de un archivo Makefile en el directorio actual, que permie al ejecutable ser compilado usando una llamada al programa make, que, a su vez, usa g++ para compilar la aplicación final:

```
~/exploringrpi/chap14/simpleQt $ make
g++ -c -pipe -O2 -Wall -W -D_REENTRANT -fPIE -DQT_NO_DEBUG ...
```

El ejecutable esta ahora presente en el directorio y se puede ejecutar del siguiente modo:

```
~/exploringrpi/chap14/simpleQt $ ls
Makefile simpleQt simpleQt.cpp simpleQt.pro
~/exploringrpi/chap14/simpleQt $ ./simpleQt
```



2- Aunque quizá, con lo hecho hasta aquí ya tendríamos para desarrollar las interfaces y programario para el tratado representado de la inflacionario recogida por los sensores, realmente seria mucho mas practico y eficiente utilizar la compilación cruzada.

Se llama compilación cruzada, cuando la compilación de la aplicación se compila en un sistema de características distintas al sistema en que dicha aplicación correrá, en nuestro caso, que utilizaremos como base de todo el sistema una Raspberry Pi 3 b+, que aunque potente, es un sistema limitado, seria mucho mas practico, rápido y deseable, hacer dicha compilación en nuestro ordenador habitual utilizado para programar ya que es mucho mas potente y capaz que la Raspberry.