

Sistemas Embebidos de Internet de las cosas (IoT)

INTRODUCCIÓN A LINUX EMBEBIDO SOBRE INTEL GALILEO

PONTIFICIA UNIVERSIDAD JAVERIANA

NOVIEMBRE DE 2015, BOGOTÁ COLOMBIA

INGENIERÍA ELECTRÓNICA



Tabla de contenido

1. CONFIGURACIÓN SD BOOTABLE WINDOWS	2
1.1 Descarga y extracción de la imagen yocto.....	2
1.2 Grabar imagen Yocto en la SD.....	2
2. CONFIGURACIÓN RED LOCAL ENTRE GALILEO Y PC	3
3. INSTALACIÓN INTEL XDK IoT EDITION	3
1.1 Instalación driver puerto serial	4
1.2 Instalación de Bonjour (Sólo para Windows)	4
1.3 Crear nuevo proyecto	5
4. DESCARGAR Y EJECUTAR PROGRAMA EN LA TARJETA	7
5. TUTORIAL NODE JS	7
5.1 Comandos básicos	7
5.1.1 Estructura de un programa	8
5.2 Librería mraa.....	8
5.3 Grove temperatura sensor	10
5.4 Librería UPM	11
5.5 Librería http	11
5.6 Librería express.....	11
5.7 Librería Socket.io	12
6. HTML	12

1. CONFIGURACIÓN SD BOOTABLE WINDOWS

Además de la programación mediante el Arduino IDE, la tarjeta de desarrollo se puede arrancar utilizando una micro SD que contenga una distribución de Linux, para este caso se utilizará la distribución YOCTO. Algunas de las funcionalidades que le agrega, es la capacidad de ejecutar programas de Python, Node.js, C++, OpenCV, entre otros.

1.1 Descarga y extracción de la imagen yocto

Para descargar la última versión de la imagen yocto, se debe ingresar a la siguiente página WEB: <https://software.intel.com/en-us/iot/hardware/galileo/downloads> y dirigirse a la sección: **Intel® Galileo Board microSD Card Linux* Operating System Image -> EGLIBC microSD card image.**

Una vez descargada la imagen, se recomienda extraer el archivo .bz2 mediante el programa 7-Zip, el cual se encuentre en el siguiente link: <http://www.7-zip.org/>

1.2 Grabar imagen Yocto en la SD

Para grabar la imagen en la SD, se utilizará el programa Win32, el cual se puede descargar mediante el siguiente link: <http://sourceforge.net/projects/win32diskimager/>. Se debe hacer clic en Download, he instantáneamente empezará la descarga de la versión más reciente del programa. Posteriormente, seguir las instrucciones del asistente de instalación.

Al finalizar la instalación, ejecutar el programa y hacer clic en la carpeta de color azul que se encuentra al costado derecho del programa (Figura 1).

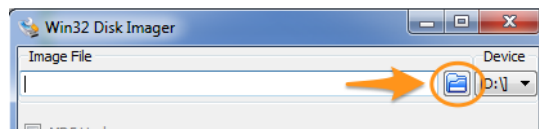


Figura 1. Explorador de archivos

Se desplegará el explorador de archivos y se busca la ubicación de la imagen de Yocto. Para que aparezca el archivo .direct, se selecciona *.* y de esta manera se muestran todos los archivos.



Figura 2. Visualización de todos los archivos

Se inserta la SD en el PC y se selecciona el dispositivo, como se aprecia en la siguiente imagen.

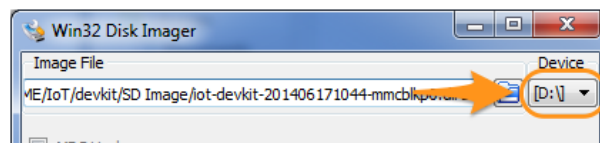


Figura 3. Selección del dispositivo

Por último, damos clic en la opción Write y esperamos a que termine el proceso. Puede tardar alrededor de 3 a 5 minutos. Al finalizar se puede extraer la SD e insertarla en la Galileo.

2. CONFIGURACIÓN RED LOCAL ENTRE GALILEO Y PC

Para crear una red local y compartir internet a la Galileo, se debe conectar un cable RJ45 al puerto Ethernet de la tarjeta y otro extremo en el PC. Posteriormente, ir a **Panel de control -> centro de redes y recursos compartidos -> cambiar configuración del adaptador**. Damos clic derecho en la conexión de red inalámbrica e ir a propiedades.

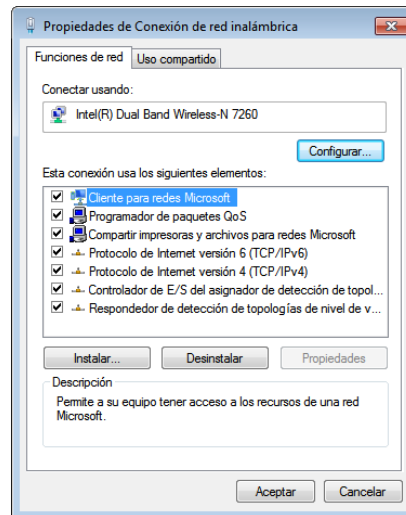


Figura 4. Propiedades de conexión de red inalámbrica

Entramos a la pestaña Uso Compartido y activamos la opción “Permitir que los usuarios de otras redes se conecten a través de la conexión a internet de este equipo”. En el menú desplegable de conexión de red doméstica, seleccionamos conexión de área local. Por ultimo damos clic en aceptar.

3. INSTALACIÓN INTEL XDK IoT EDITION

El XDK de Intel es un software que permite crear y probar aplicaciones IoT en la tarjeta Galileo. También proporciona templates para interactuar con sensores y actuadores, mediante el sistema operativo Yocto instalado en la tarjeta SD.

Para descargar la versión más reciente de Intel XDK IoT Edition se debe ingresar a la siguiente página WEB: <https://software.intel.com/en-us/iot/software/ide>. Para iniciar la descarga nos remitimos a la sección Intel XDK IoT Edition y damos clic en “Download for Windows”, posteriormente iniciará la descarga del programa.

Al ejecutar el programa de instalación, siga las instrucciones en pantalla.



Figura 5. Instalación Intel XDK IoT Edition

1.1 Instalación driver puerto serial

Para utilizar el cable USB a RS232 (el cual se utilizará más adelante), se deben instalar los drivers correspondientes, los cuales se encuentran en el siguiente link: <http://www.ftdichip.com/Drivers/D2XX.htm> . Buscamos el sistema operativo y la arquitectura para la descarga.

Una vez descargado el archivo llamado CMD vx.x.x.zip, extraer y ejecutar el archivo dpinst-amd64.exe. Seguir los pasos de la instalación de los drivers.



Figura 6. Cable USB a RS232

1.2 Instalación de Bonjour (Sólo para Windows)

Este programa sirve para que el XDK detecte automáticamente dispositivos IoT instalados en la red. Para la descarga del programa debemos ir al siguiente link: https://support.apple.com/kb/DL999?locale=es_ES y realizar la descarga. Seguir las instrucciones del asistente de instalación para instalar Bonjour.



Figura 7. Instalación Bonjour

Ya finalizado la descarga de los dos programas se procede a ejecutar el XDK de Intel y se debe seguir las instrucciones para la creación e inicio de sesión de cuenta de Intel XDK.

1.3 Crear nuevo proyecto

En el menú seleccionar Templates.

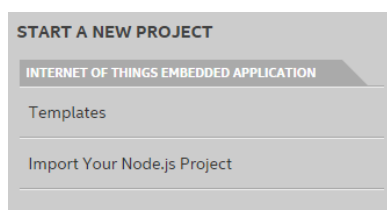


Figura 8. Crear nuevo proyecto

En la lista seleccionar Onboard LED Blink y dar click en Continue

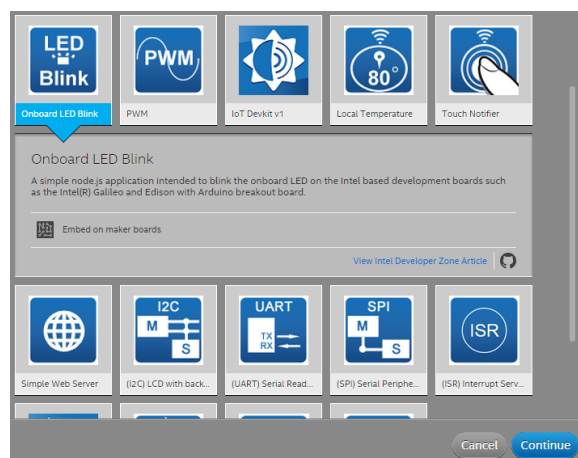


Figura 9. Template Blink

Colocar el nombre y la ubicación del nuevo proyecto. Dar clic en crear.



Figura 10. Guardar proyecto

Ya creado el proyecto, conectamos el cable USB a RS232 y nos dirigimos a la pestaña “Serial terminal”. Seleccionamos el puerto serial que en este caso es el COM13 y dejamos los demás parámetros por defecto. Damos clic en Connect.

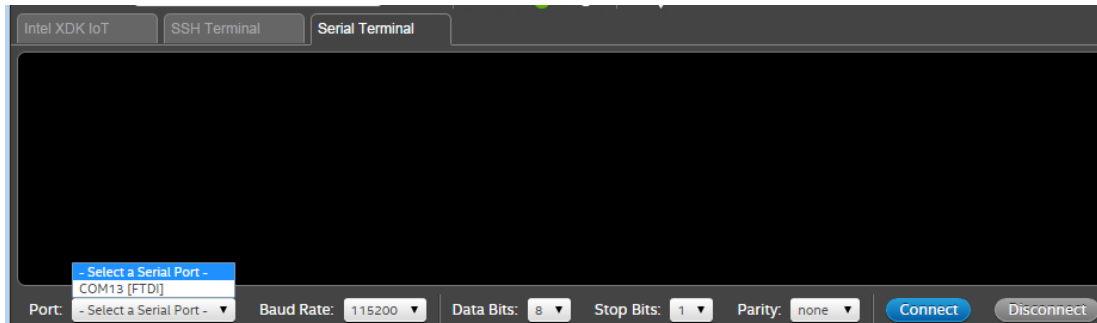


Figura 11. Terminal serial

Realizada la configuración del puerto serial, se procede a conectar el cable USB to RS232, cable de alimentación, cable Ethernet e insertamos la SD a la galileo.

En el Serial Terminal veremos la inicialización de la consola de Linux que tiene la tarjeta. Cuando termine de cargar nos pedirá el “galileo login”, el cual es root.

Verificamos la conexión de red con el pc mediante el comando ifconfig, en donde veremos la IP que nos asignó el pc. De la siguiente manera:

```
root@galileo:~# ifconfig
enp0s20f6 Link encap:Ethernet HWaddr 98:4F:EE:00:22:5B
          inet addr:192.168.137.117 Bcast:192.168.137.255 Mask:255.255.255.0
          inet6 addr: fe80::3041:eeff:fe00:225b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:26 errors:0 dropped:0 overruns:0 frame:0
          TX packets:72 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:3146 (3.0 KiB) TX bytes:10450 (10.2 KiB)
          Interrupt:44

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:164 errors:0 dropped:0 overruns:0 frame:0
          TX packets:164 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:15124 (14.7 KiB) TX bytes:15124 (14.7 KiB)
```

Figura 6. IP de la Galileo

Para acceder a internet desde la Galileo se debe configurar el DNS. Para esto, se debe ingresar el siguiente comando: `vi /etc/resolv.conf`. Con este comando se accede al archivo `resolv.conf` y se debe agregar la siguiente línea de código: `nameserver 8.8.8.8`. Para editarlo se deben tener en cuenta los siguientes comandos del editor `vi`:

comando	descripción
i	insertar antes del cursor
a	añadir detrás del cursor
o	añadir una línea en blanco
x	borrar un carácter
j	borrar el final de línea (une dos líneas)
dd	borra la línea completa
u	deshacer la última edición
:q	salir
:q!	salir sin guardar
:w	guardar
:wq	guardar y salir
:set nu	muestra números de línea
:set nonu	oculta números de línea
:n copy m	copia la línea n detrás de la línea m
:n1, n2 copy m	copia desde la línea n1 hasta la línea n2 detrás de la línea m

Figura 7¹. Comandos editor vi

Posteriormente se reiniciará el puerto Ethernet, para esto se debe ejecutar los siguientes comandos: `ifconfig enp0s20f6 down` , y `ifconfig enp0s20f6 up`. Con esto ya tendremos conexión a internet desde la Galileo. (Opcional) Para verificar la conexión a internet se puede ingresar el siguiente comando: `ping google.com`

Para finalizar la configuración de la tarjeta Galileo, nos dirigimos a la pestaña Intel XDK IoT y posteriormente en el menú desplegable “- Select a Device-”, seleccionamos Add Manual Connection, insertamos la dirección IP que anteriormente consultamos y damos clic en Connect.

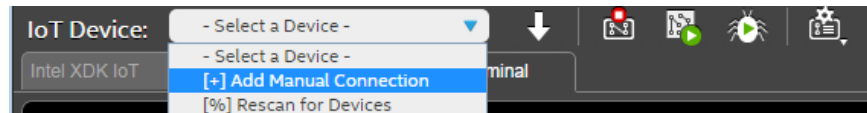


Figura 8. Cable USB a RS232

4. DESCARGAR Y EJECUTAR PROGRAMA EN LA TARJETA

Una vez realizada la configuración de la tarjeta y del programa, se puede proceder a descargar el programa que se desea ejecutar. Para esto lo primero que debemos hacer es hacer clic en la Flecha de color blanco y para ejecutarlo se debe hacer clic en el círculo verde con un triángulo blanco.

5. TUTORIAL NODE JS

NodeJS es básicamente un framework para implementar operaciones de entrada y salida. Está basado en eventos, streams y construido encima del motor de Javascript V8, que es con el que funciona el Javascript de Google Chrome.

5.1 Comandos básicos

En el lenguaje de programación de Nodejs existen algunos comandos básicos, como por ejemplo:

Comando	Descripción
<code>var variable;</code>	Declara una variable o constante, opcional inicializarla en un

¹ Imagen tomada de: http://www.mclibre.org/consultar/linux/modotexto/editor_vi.html

<code>const value;</code>	valor, entre los cuales pueden ser: true or false (boolean), un número (independiente de ser int, float), un string ("hola mundo"), un objeto, un arreglo ([1,2,3,4])
<code>var objeto.metodo();</code>	Los métodos son comandos que están dentro de un objeto.
<code>console.log("texto" + variable);</code>	Imprime en pantalla, se puede imprimir texto y una variable separado por un "+" o sólo alguna de las dos
<code>require('libreria');</code>	Llamado de una librería
<pre>if(value1 == value2){ } else { }</pre>	<pre>== igual que != no igual que < menor que > mayor que <= menor o igual que >= mayor o igual que</pre>
<pre>for(i=0,i<value,i++){ } While(condición){ }</pre>	bucle
<pre>function hacer(variables){ ... return; }</pre>	Funciones en Javascript
<pre>function Hello_World() { setTimeout(Hello_World,1000); }</pre>	Función que se llama a sí misma cada segundo
<code>Math.log();</code>	Función de logaritmo

Tabla 1. Comandos básicos

5.1.1 Estructura de un programa

En Nodejs no se cuenta con un main, setup o loop. Por ende los programas solo se ejecutan una vez. Si se desea que el programa se ejecute varias veces se debe crear una función que se llame a sí misma, de la siguiente manera:

```

Hello_World();
function Hello_World()
{
  Console.log("HELLO WORLD");
  setTimeout(Hello_World,1000);
}
//Primer llamado de la función
//Función Hello_World
//Se guarda el estado actual del boton
//Se auto-llama la función cada segundo
```

5.2 Librería mraa

La librería mraa es la que permite manejar los periféricos de entrada y de salida de la Galileo. El llamado de la librería se hace de la siguiente manera: `var mraa=require('mraa');`. De esta manera se crea un objeto llamado mraa.

Comandos de la librería:

Método	Descripción
Gpio(pin, owner, raw); Retorna un número.	Pin =número del pin digital que se usará. Figura 9 Owner = Sirve para cerrar la instrucción (por defecto se debe colocar en false) Raw =Indica que se escoge la enumeración de los pines del kernel (true). Figura 9
dir(dir);	Asigna si el puerto es de entrada o de salida. dir= mraa.DIR_OUT ó dir mraa.DIR_IN
read(); retorna un valor	Lee un valor del Gpio.
write(value);	Escribe un valor en el puerto Gpio.
Aio(pin); Retorna un número	Se define el número del pin análogo que se usará
read(); retorna un valor	Retorna un valor leído del puerto análogo de 10 bits
Pwm(Pin); Retorna un número	Pin =número del pin usado en la tarjeta.
write(porcentaje); (PWM)	Escribe en el puerto PWM un valor entre 0 a 1 que corresponde del 0 al 100% de ciclo útil.
read(); (PWM) retorna un valor	Retorna un valor de ciclo útil entre 0 a 1
period_us(periodo);	Se configura el periodo de la señal en micro segundos
enable(true);	Hablit el PWM.

Tabla 2. Comandos librería mraa.

En la siguiente imagen, los gpioxx que se encuentran en la parte izquierda, indican las conexiones físicas del Quark X 1000 de Intel, mientras que los números del lado derecho, indican la enumeración de los puertos para arduino.

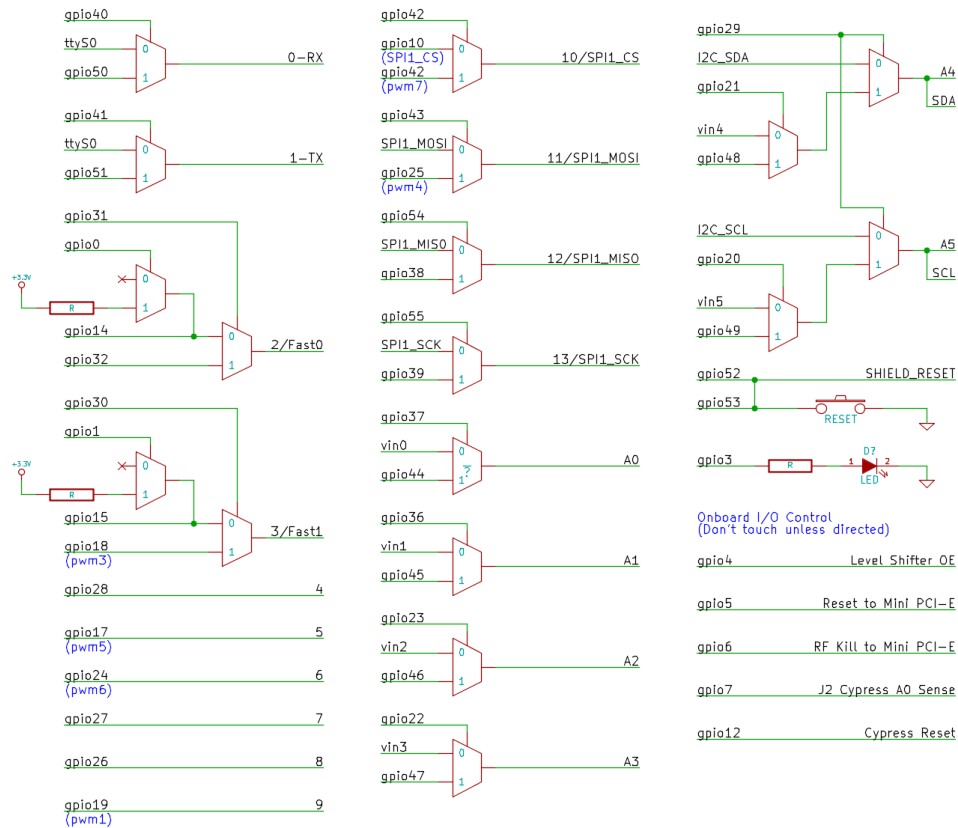


Figura 9. Esquemático arquitectura Gpio

5.3 Grove temperatura sensor

El sensor de temperatura es análogo, por lo tanto para realizar la medición cuenta con un termistor (sensor resistivo de temperatura), lo que indica que con un cambio en la temperatura, cambia su resistencia de forma logarítmica. A continuación se muestra la curva característica del sensor:

1. Zero-power Resistance of Thermistor: R
 $R = R_0 \exp B (1/T - 1/T_0)$ (1)
 R: Resistance in ambient temperature T (K)
 (K: absolute temperature)
 R₀: Resistance in ambient temperature T₀ (K)
 B: B-Constant of Thermistor
2. B-Constant
 as (1) formula
 $B = \ln (R/R_0) / (1/T - 1/T_0)$ (2)
3. Thermal Dissipation Constant
 When electric power P (mW) is spent in ambient temperature T₁ and thermistor temperature rises T₂, there is a formula as follows
 $P = C (T_2 - T_1)$ (3)
 C: Thermal dissipation constant (mW/°C)
 Thermal dissipation constant is varied with dimensions, measurement conditions, etc.

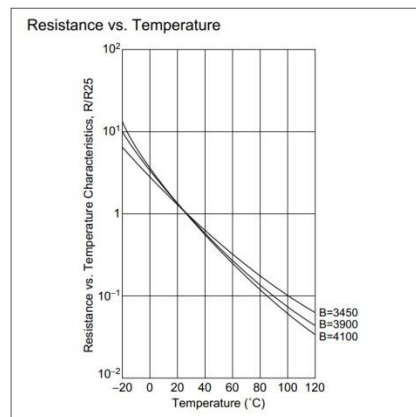


Figura 10. Curva característica del sensor

Según la hoja de especificaciones y la Figura anterior se tiene las siguientes ecuaciones para la medida de la temperatura en Celsius.

Constante del termistor B=3975

$$Resistencia = \frac{(1023 - Valor_{leido}) * 10000}{Valor_{leido}}$$

$$Temperatura = \frac{1}{\frac{\log\left(\frac{Resistencia}{10000}\right)}{B} + \frac{1}{298.15}} - 273.15$$

5.4 Librería UPM

La librería UPM, es la que contiene algunas funciones de los sensores de Grove starter kit. Para utilizar correctamente el display LCD (JHD1313M1), el cual cuenta con dos direcciones I2C, una para controlar la visualización y la otra para controlar la luz de fondo. El voltaje de operación es de 5V. Para más información dirigirse a la librería upm que se encuentra en las referencias.

5.5 Librería http

La librería http, es la que permite crear un servidor de manera sencilla. El llamado de la librería se hace de la siguiente manera: `var http=require('http');`. De esta manera se crea un objeto llamado http.

Método	Descripción
<code>createServer(function (req, res){ }).listen(puerto);</code>	req =requerimientos del cliente res =respuesta del servidor Puerto = Puerto donde se escucharán las solicitudes
<code>res.writeHead(200, { "Content-Type": "text/html" });</code>	Se define la respuesta del servidor en el lenguaje HTML
<code>res.write("comando HTML");</code>	Se escribe en el puerto desde el servidor
<code>res.end();</code>	Se termina el envío de parámetros HTML.
<code>req.url</code>	Se requiere la URL del cliente

Tabla 3. Comandos librería http.

5.6 Librería express

La librería http, es la que permite crear un servidor de manera sencilla. El llamado de la librería se hace de la siguiente manera: `var express=require('express');`. De esta manera se crea un objeto llamado http.

Método	Descripción
<code>var app = express();</code>	Se guarda el objeto en una variable llamada app
<code>app.use(express.static(__dirname));</code>	Sirve para definir la dirección predeterminada del servidor
<code>app.listen(puerto);</code>	Se define el puerto del servidor

Tabla 4. Comandos librería express.

5.7 Librería Socket.io

La librería Socket.io, permite una comunicación bidireccional en tiempo real entre servidores y clientes. El llamado de la librería se hace de la siguiente manera: `var io=require('socket.io');`. De esta manera se crea un objeto llamado `io`.

Método	Descripción
<code>io.on('connection', function (socket) { });</code>	Función para la inicialización de la comunicación a través de socket
<code>socket.on('name', data);</code>	Name = nombre del socket que se recibirá Data = información recibida
<code>socket.emit('name', data);</code>	Name = nombre del socket que se enviará Data = información enviada

Tabla 5. Comandos librería socket.io.

6. HTML

HTML es un lenguaje que hace posible presentar información en Internet. Lo que se visualiza en una página en Internet es la interpretación que hace el navegador del código HTML. Para ver el código HTML de una página sólo tienes que pinchar en la opción "Ver" de la barra de menús y elegir "Código fuente".

```
<html>                                     //Indica al navegador que
                                           //abra un archivo html
    <head>                                //Encabezado. Sección que no
                                           //se muestra en pantalla
    </head>
    <body>                                //Cuerpo del documento.
                                           //Sección que aparece en pantalla
    </body>
</HTML>
```

JavaScript es un lenguaje de programación, al igual que PHP, si bien tiene diferencias importantes con éste. JavaScript se utiliza principalmente del lado del cliente (es decir, se ejecuta en nuestro ordenador, no en el servidor) permitiendo crear efectos atractivos y dinámicos en las páginas web. Los navegadores modernos interpretan el código JavaScript integrado en las páginas web.²

Para incluir dichos archivos se coloca la siguiente etiqueta dentro del head

```
<script src="archivo.js"></script>
```

Por otro lado, existen archivos .css. CSS es un lenguaje de estilo que define la presentación de los documentos HTML. Por ejemplo, CSS abarca cuestiones relativas a fuentes, colores, márgenes, líneas, altura, anchura, imágenes de fondo, posicionamiento avanzado y muchos otros temas.

Para incluir dichos archivos se coloca la siguiente etiqueta dentro del head

² Tomado de: http://aprenderaprogramar.com/index.php?option=com_content&view=article&id=590:ique-es-y-para-que-sirve-javascript-embeber-javascript-en-html-ejercicio-ejemplo-basico-cu00731b&catid=69:tutorial-basico-programador-web-html-desde-cero&Itemid=192

<link rel="stylesheet" type="text/css" href="archivo.css"/>

REFERENCIAS

- Getting started with Intel XDK IoT Edition on Intel IoT platforms. Accedido: Noviembre 2015. Website:
 - <http://www.codeproject.com/Articles/874738/Getting-Started-with-Intel-XDK-IoT-Edition-on-Inte>
- Making a bootable micro SD Card with Windows: November 2015. Website:
 - <https://software.intel.com/en-us/programming-blank-sd-card-with-yocto-linux-image-windows>
- Librerías de Node.JS . Accedido: October 2015. Website:
 - <https://www.npmjs.com/>
- Librería mraa. Accedido: October 2015. Website:
 - <http://iotdk.intel.com/docs/master/mraa/node/>
- Librería upm. Accedido: October 2015. Website:
 - <http://iotdk.intel.com/docs/master/upm/>
- Hoja de especificaciones Sensor de temperatura. Accedido: October 2015. Website:
 - http://www.seeedstudio.com/wiki/Grove_-_Temperature_Sensor_V1.2
- Guía Javascript. Accedido: October 2015. Website:
 - <http://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Introduction>
- Node.js templates for the Intel XDK . Accedido: October 2015. Website:
 - https://software.intel.com/en-us/node-js-templates-for-intel-xdk-iot-edition#LCD_with_backlight_driver_and_demonstrator
- Tutorial Node.js . Accedido: October 2015. Website:
 - <https://github.com/graftlabs/node-workshop>
- Desarrollo web con express.js . Accedido: Noviembre 2015. Website:
 - <http://www.genbetadev.com/frameworks/introduccion-a-la-programacion-asincrona-con-nodejs-desarrollo-web-con-expressjs>
- WebSockets . Accedido: Noviembre 2015. Website:
 - <https://carlosazaustre.es/blog/websockets-como-utilizar-socket-io-en-tu-aplicacion-web/>
- Guía librería http. Accedido: October 2015. Website:
 - <http://www.genbetadev.com/frameworks/introduccion-a-la-programacion-asincrona-con-nodejs>
- Tutorial HTML. Accedido: October 2015. Website:
 - <http://www.virtualnauta.com/html-tutorial>