



Asociación Civil para la Investigación,
Promoción y Desarrollo de los
Sistemas Electrónicos Embebidos

Seminario de Electrónica Sistemas Embebidos



Laboratorio de
Sistemas Embebidos

LPCXpresso - Introducción



Ing. Juan Manuel Cruz (jmcruz@hasar.com)

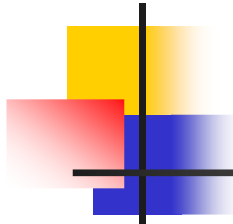
Gerente de Ingeniería de Cia. Hasar SAIC



Profesor Asociado Ordinario - Técnicas Digitales II TN-FRBA

Profesor Adjunto Interino - Sistemas Embebidos FIUBA

Buenos Aires, 18 de Agosto de 2016

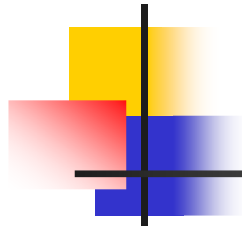


Temario

- | | |
|-------------------------------|--------------------------------|
| ■ Prácticas de Laboratorio | Plataforma de Trabajo |
| ■ LPC1769 LPCXpresso board | LPCXpresso/MBED Baseboard |
| ■ EDU-CIAA-NXP | OpenOCD - LPC-Link2 |
| ■ LPCXpresso IDE v8.2.0 | Proveedor: NXP |
| ■ Instalación / Ejecución IDE | Importar Librerías |
| ■ Genera un Nuevo Proyecto | Abrir main.c |
| ■ Tratamiento del Proyecto | |
| ■ Clean/Build/Debug | Tool bar/Debug & Console wind. |
| ■ LPCXpresso User Guide | Primer Actividad Práctica |

Link: <http://campus.fi.uba.ar/enrol/index.php?id=648>

Ingresar como: **Invitado** c/clave: **658** y **matricularse**

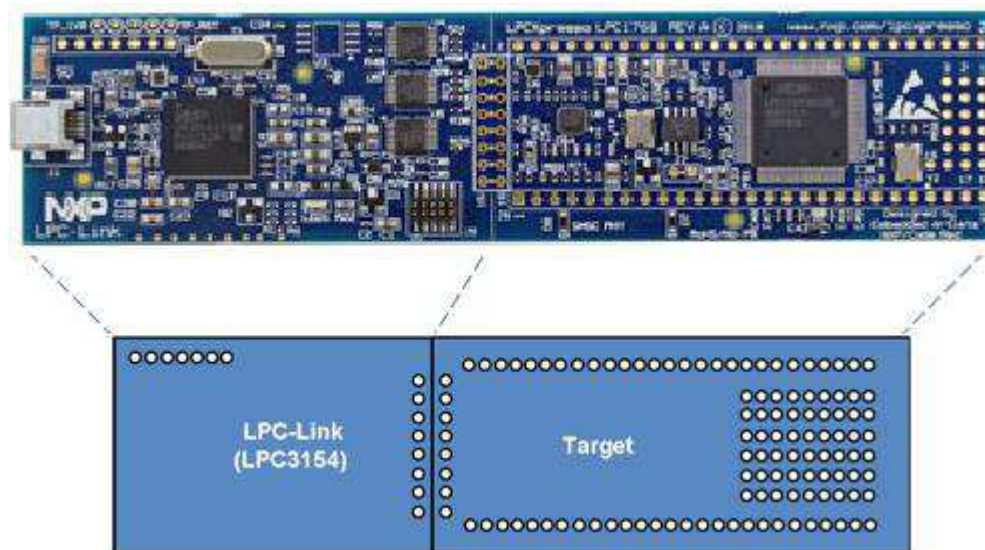


Prácticas de Laboratorio

- Objetivo
 - Brindar al alumno un suave y sencillo acercamiento a la arquitectura, tecnología, técnicas y herramientas que le ayuden en la concreción del trabajo práctico final
- Metodología
 - Presentación y aplicación de temas teóricos y técnicas
 - Presentación y uso de herramientas
 - Ejercitación básica e integradora
- Se pretende que el alumno consolide los conocimientos teóricos con la ejercitación debida y que las herramientas de trabajo empleadas en el laboratorio puedan ser utilizadas por el alumno durante el proceso de consolidación de la enseñanza en su casa

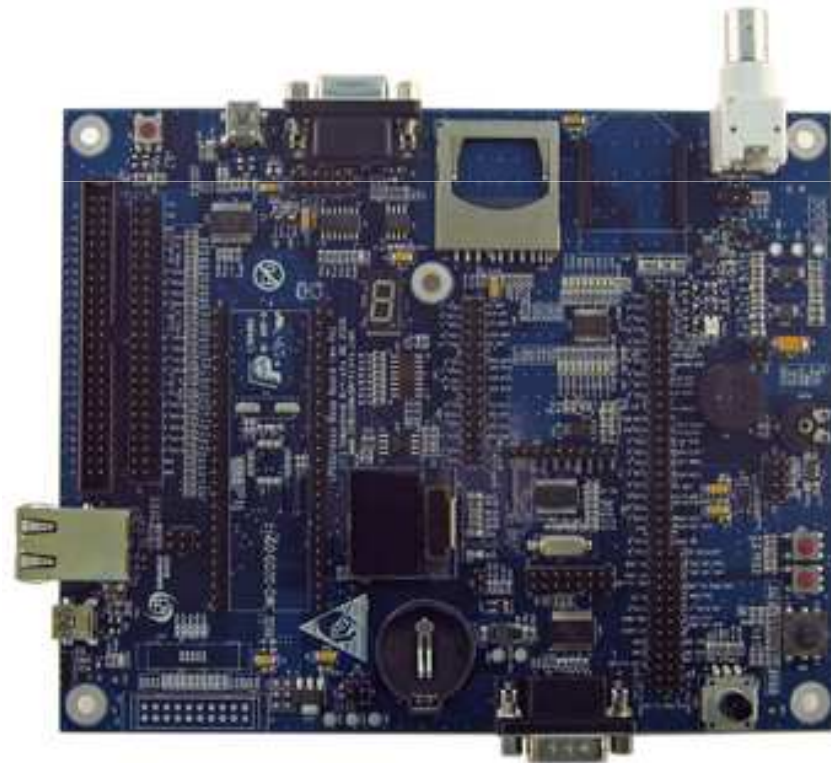
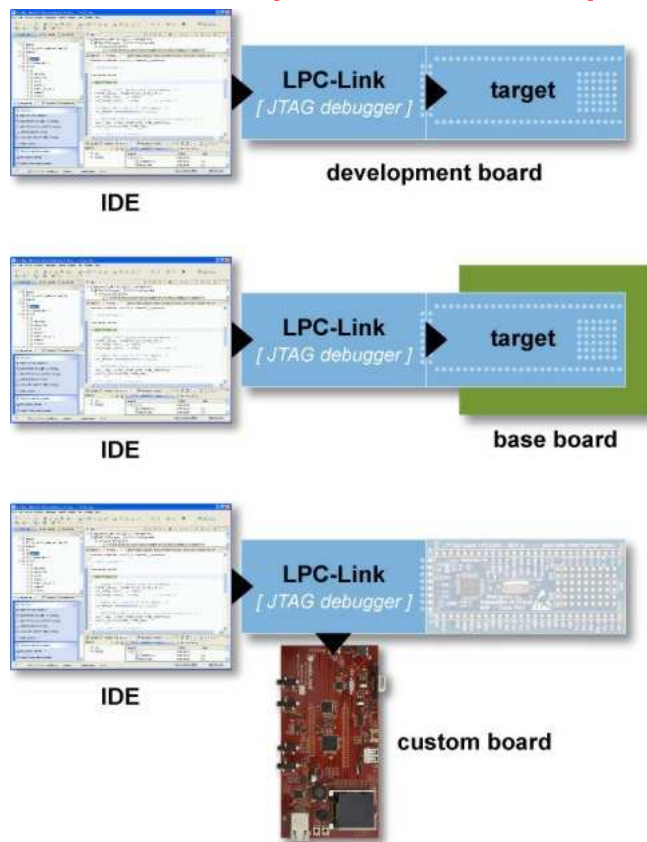
Plataforma de Trabajo

- Microcontrolador de 32 bits: **ARM Cortex M3 – NXP**
 - **LPC1769 LPCXpresso board**
www.lpc tools.com/lpc1768.lpcxpresso.aspx



LPCXpresso/MBED Base Board

- www.lpc tools.com/lpcxpresso.mbed.base.board.aspx



Plataforma de Trabajo

- Microcontrolador de 32 bits: **ARM Cortex M4/M0 – NXP**
 - **EDU-CIAA-NXP board (LPC4337 M4/M0)**
www.proyecto-ciaa.com.ar/devwiki/doku.php?id=desarrollo:edu-ciaa:edu-ciaa-nxp
 - **OpenOCD 0.9.0**
<https://github.com/gnuarmclipse/openocd/releases/tag/gae0.9.020150519>
 - **LPC-LINK 2** <http://embeddedartists.com/products/lpcxpresso/lpclink2.php>
www.lpcware.com/lpclink2



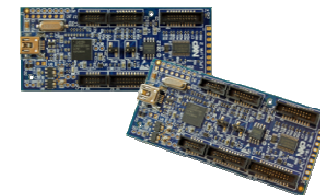
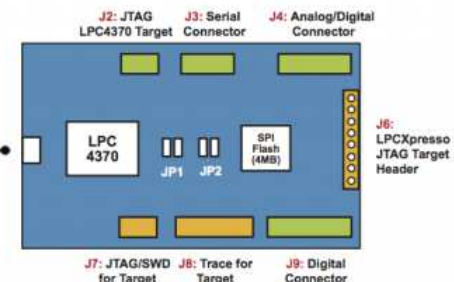
18 de Agosto de 2016

Ing. Juan Manuel Cruz



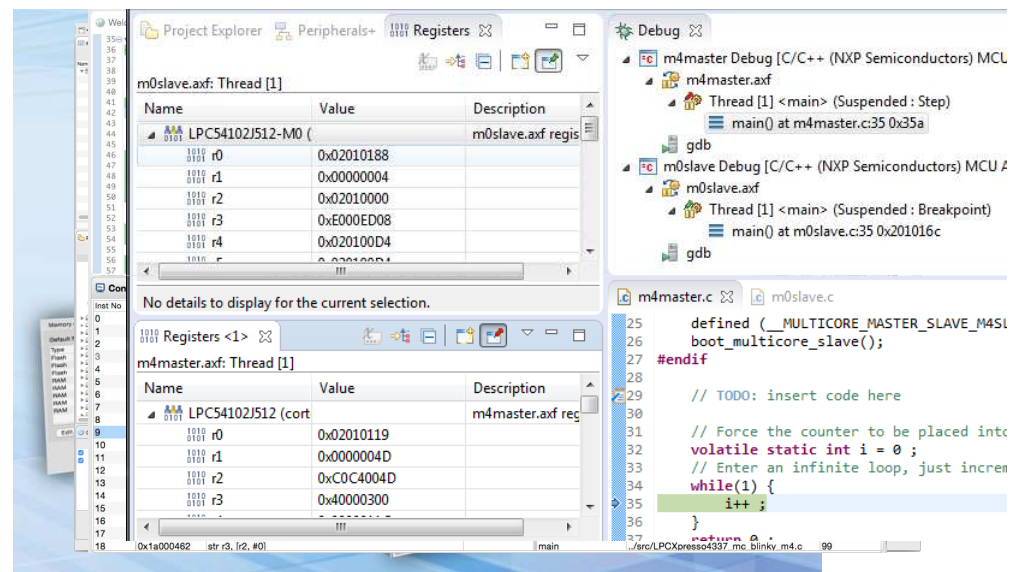
Debugging Target ☐
JP2:
Open: Target is self-powered
Closed: Target powered by LPC-Link 2

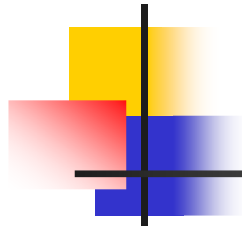
Debugging LPC4370 ☐
JP1:
Open: Boot from USB DFW
Closed: Boot from SPI Flash



LPCXpresso IDE v8.2.0: NXP

- LPCXpresso IDE v8.2.0
- Debugging
- Instruction Trace
- Multicore Debugging
- Jumpstarting your NXP® MCU project development
- Rapid Embedded Development with LPCXpresso





Proveedor

- En el mercado local consultar a **Semak SA** (www.semak.com.ar/) :
 - **LPC1769 LPCxpresso board** (código NXP OM13000)
www.lpc tools.com/lpc1768.lpcxpresso.aspx
 - **LPCXpresso/MBED Base Board** (código NXP OM11083)
www.lpc tools.com/lpcxpresso.mbed.base.board.aspx
 - **LPC-LINK 2** (código NXP OM13054)
www.lpcware.com/lpclink2
- Contacto:
 - Sr. Jorge Esteche (jorge.esteche@semak.com)
 - Av. Belgrano 1580 - Piso 5 (CP1093) - Ciudad Autónoma de Buenos Aires - Argentina - Teléfono: (+54-11) 4381-2108 - Fax: (+54-11) 4383-7420
 - Horario: de Lunes a Viernes de 9h a 17h

Instalación / Ejecución IDE

- Descargar la última versión disponible para Windows, Linux y Mac OS X (en el proceso le pedirá que se registre como usuario)
- Ejecutar el instalador descargado (al concluir la instalación seguir instrucciones p/activar la **edición libre** del IDE)
- Ejecutar LPCXpresso
- Descargar, instalar y agregar al IDE el plug-in: OpenOCD 0.9.0 (ojo que es una versión vieja)
- Descargar Configuration Tool de LPC-Link 2
- Links:
 - **LPCWare:** www.lpcware.com/
 - **Devices:** www.lpcware.com/?order=title&sort=asc&tid=60
 - **LPCXpresso :** www.nxp.com/products/software-and-tools/software-development-tools/software-tools/lpc-microcontroller-utilities/lpcxpresso-ide-v8.2.0:LPCXPRESSO
 - **OpenOCD 0.9.0:** <https://github.com/gnuarmclipse/openocd/releases/tag/gae0.9.020150519>





Instalación / Ejecución IDE

- **LPCXpresso: toolchain** completo para **evaluación y desarrollo** de aplicaciones para **microcontroladores** de **NXP**, compuesto por:
 - **LPCXpresso IDE** (Integrated Development Environment)
 - IDE basado en **Eclipse**
 - **Compilador & Linker** GNU
 - GDB **Debugger** más muchas otras **herramientas** (**Eclipse plug-ins**)
 - **LPC1769 LPCXpresso board & LPCXpresso/MBED baseboard**
 - **o EDU-CIAA-NXP & c/OpenOCD 0.9.0 o c/LPC-Link 2**
- LPCXpresso IDE permite definir:
 - **Workspace**: Contenedor de proyectos (aplicaciones y/o bibliotecas). Además almacena la configuración del entorno (portable e/computadoras)
 - **Proyecto**: Del tipo **Biblioteca estática** o una **Aplicación ejecutable**. Contiene archivos de **código fuente C (.c)**, **encabezado (.h)** y del tipo que sea necesario incorporar al proyecto



Instalación / Ejecución IDE

- **Workspace:**

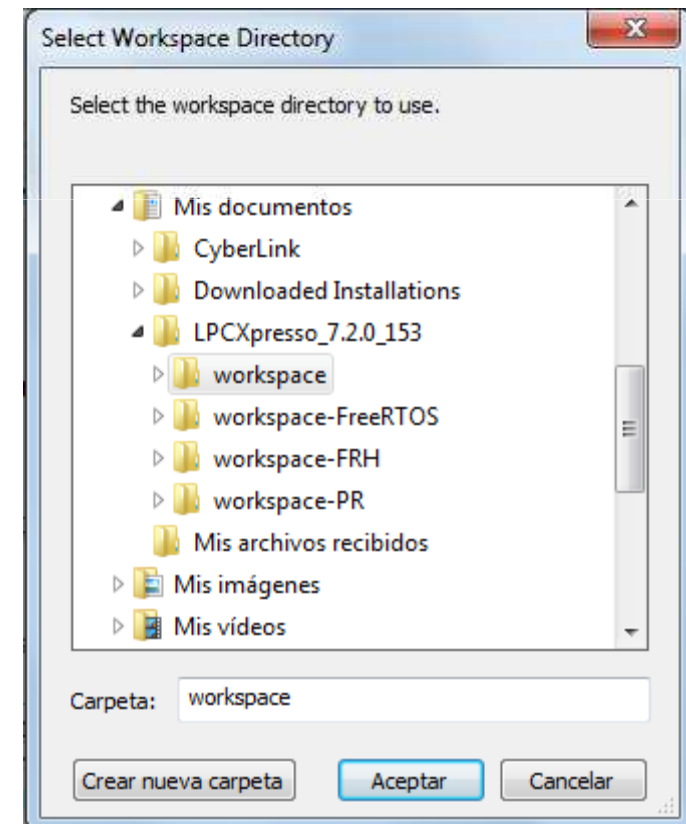
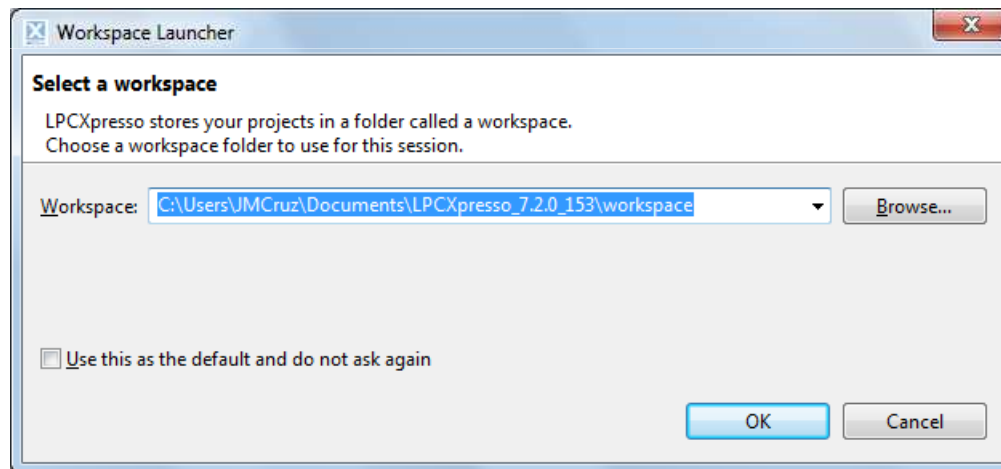
- Los usaremos para **almacenar** e **intercambiar proyectos** ya que contienen **aplicaciones** & **bibliotecas estáticas** & **configuraciones**

- **Proyectos:**

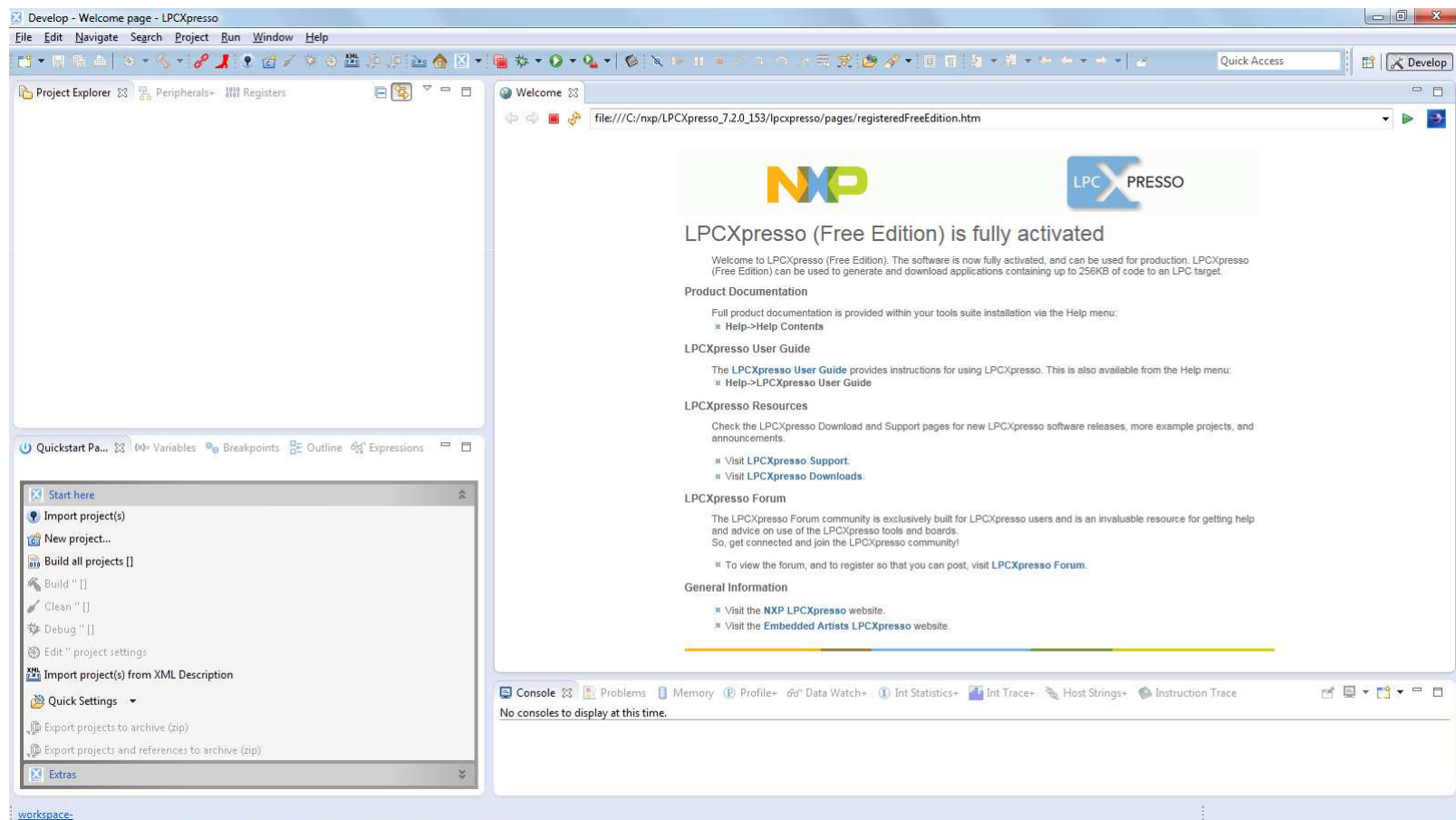
- **Aplicación** (desarrollaremos varias):
 - Contiene al **main()** (donde se ejecutará nuestro programa)
 - **Edita, compilar, linkear** y **descargar** directamente al **target** (uC), para luego **debuggear** la aplicación
- **Biblioteca estática** (desarrollaremos al menos una):
 - No tiene **main()**, sus **funciones** pueden ser **llamadas** desde los proyectos de **aplicación**
 - Se **compila** y **no se puede descargar** directamente al **target** (uC)
 - **NXP** provee alguna que utilizaremos

Instalación / Ejecución IDE

- => Browser => Seleccionar dónde quiero almacenar el Workspace y qué nombre quiero asignarle

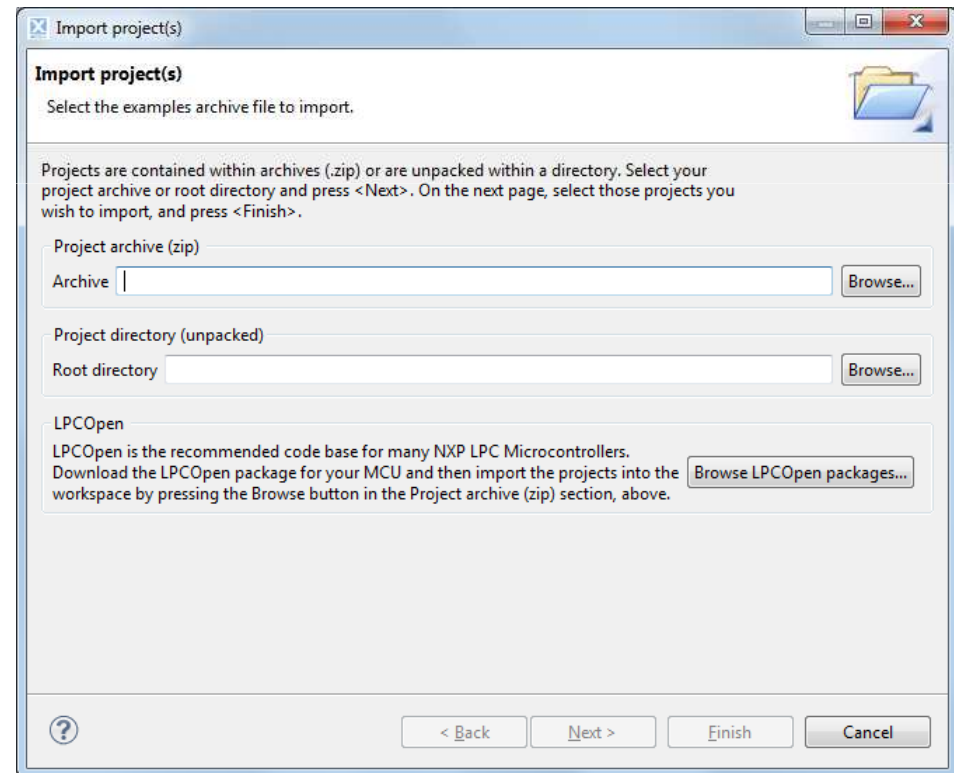
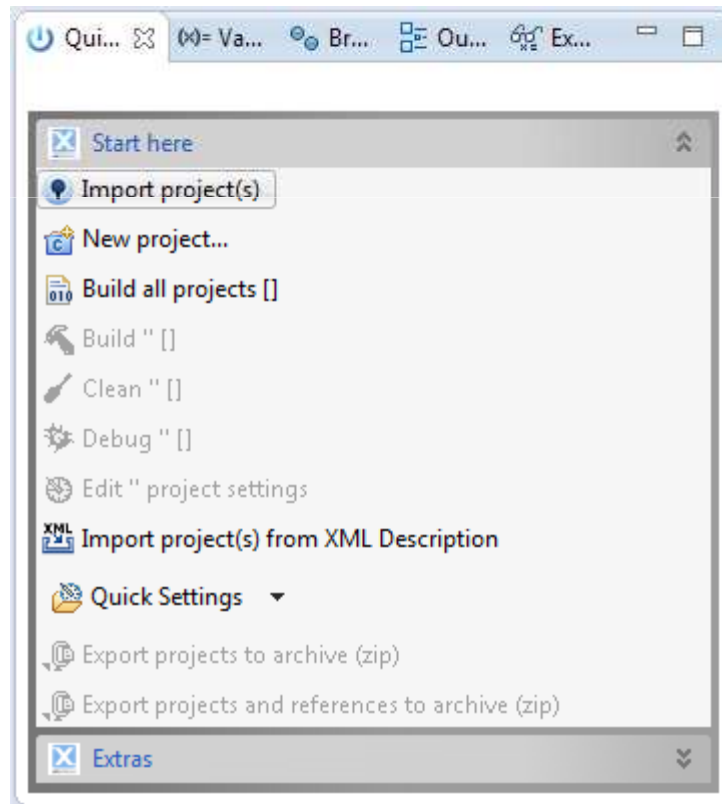


Instalación / Ejecución IDE



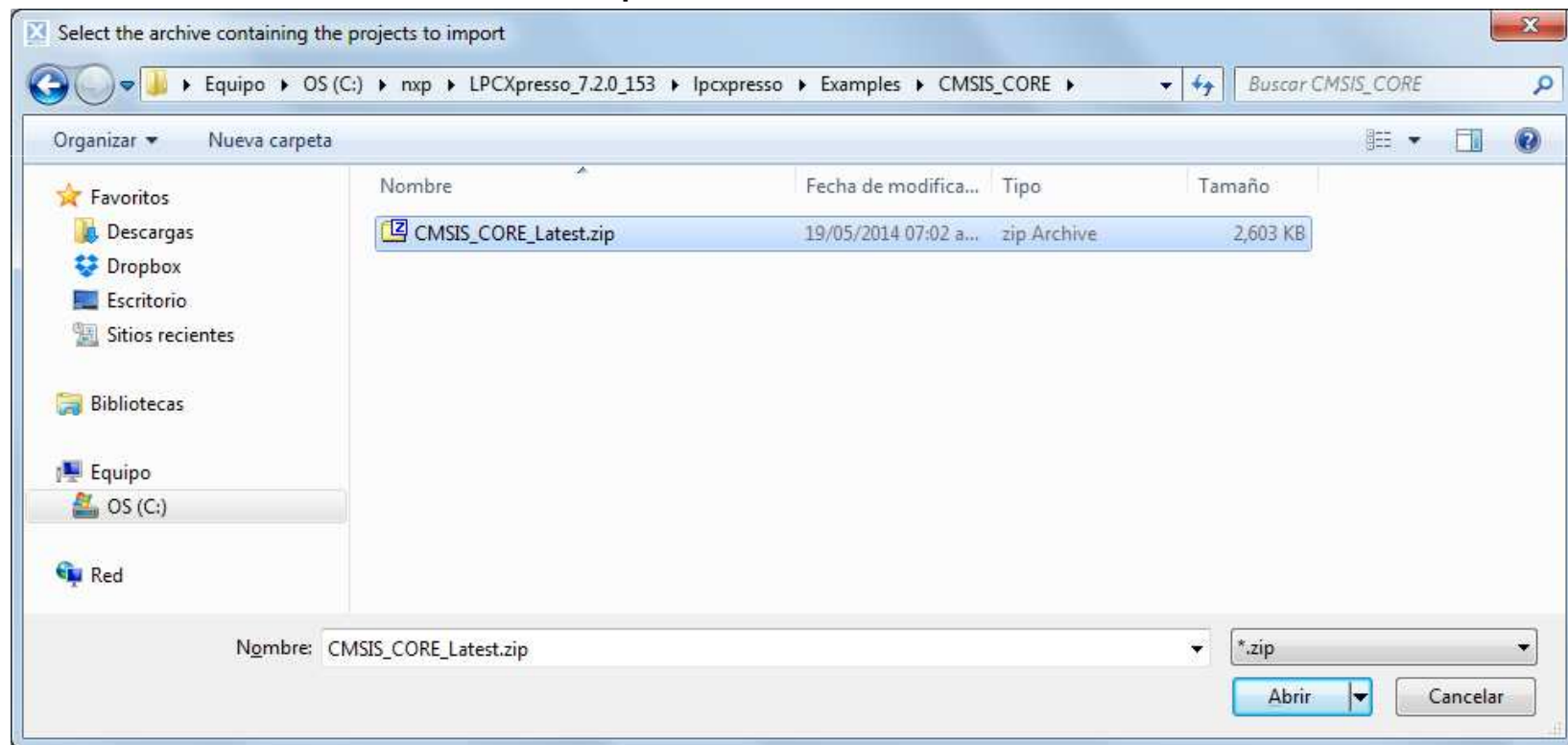
Importar Librerías

- Import Porject(s) => Archive => Browse...



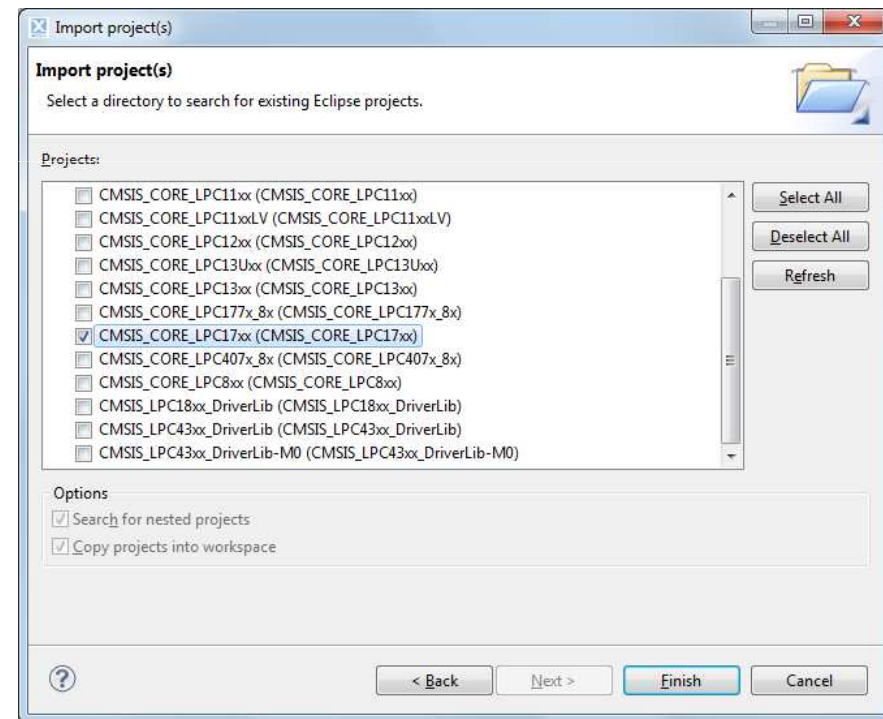
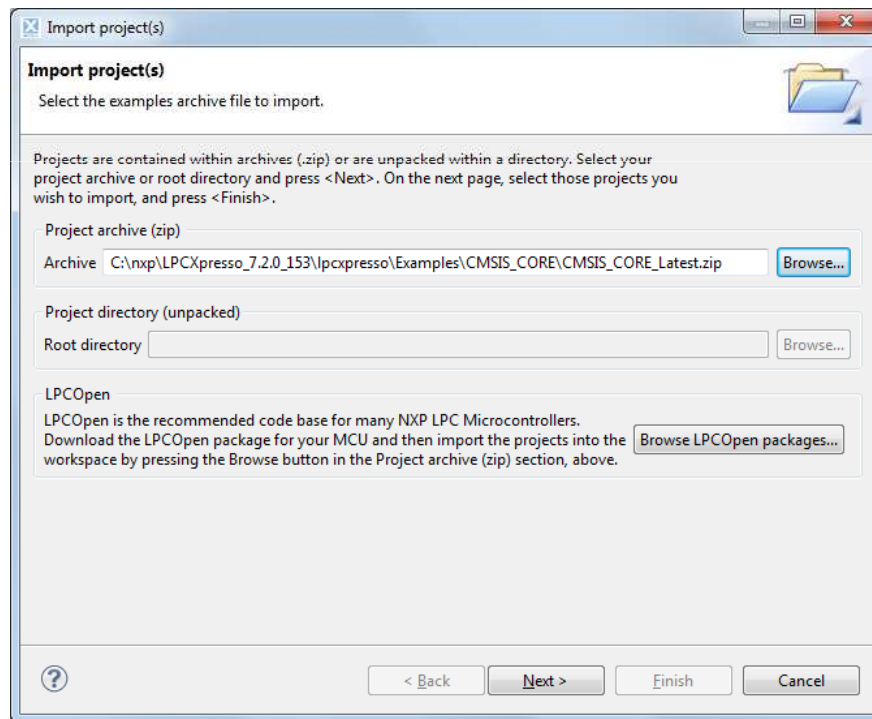
Importar Librerías

- C:\npx\LPCXpresso_7.8.0_426\lpcxpresso\Examples\CMSIS_CORE\CMSIS_CORE_Latest.zip => Abrir



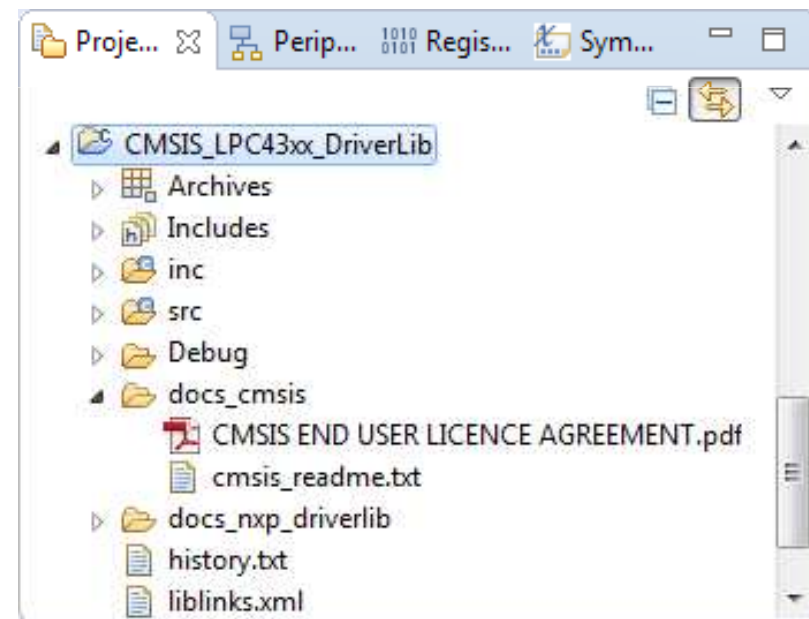
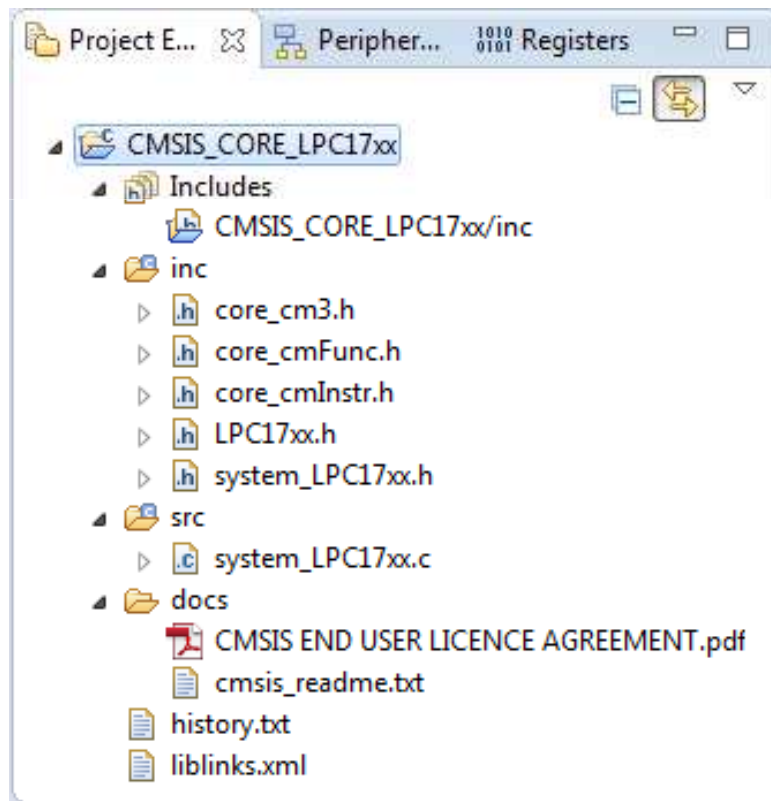
Importar Librerías

- Next > => Deselect All => CMSIS_CORE_LPC17xx (...) => Finish
○ CMSIS_LPC43xx_DriverLib (...)



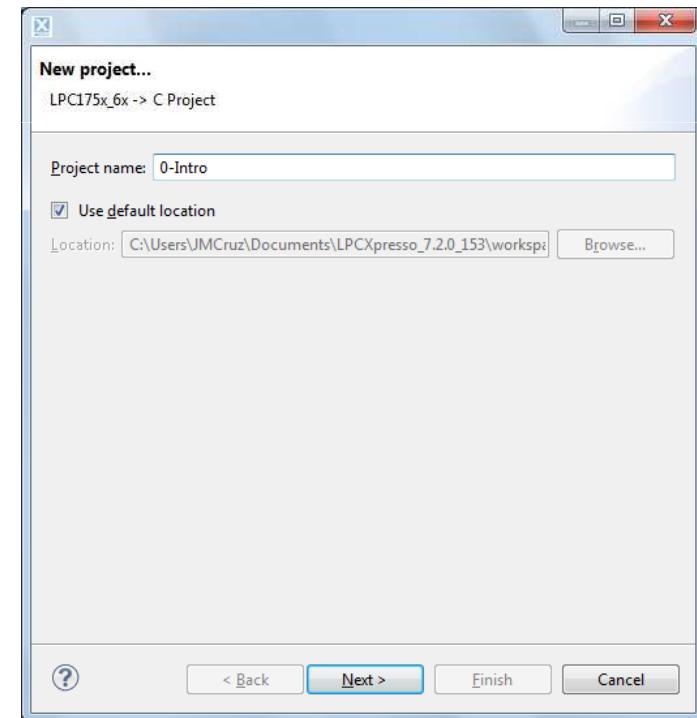
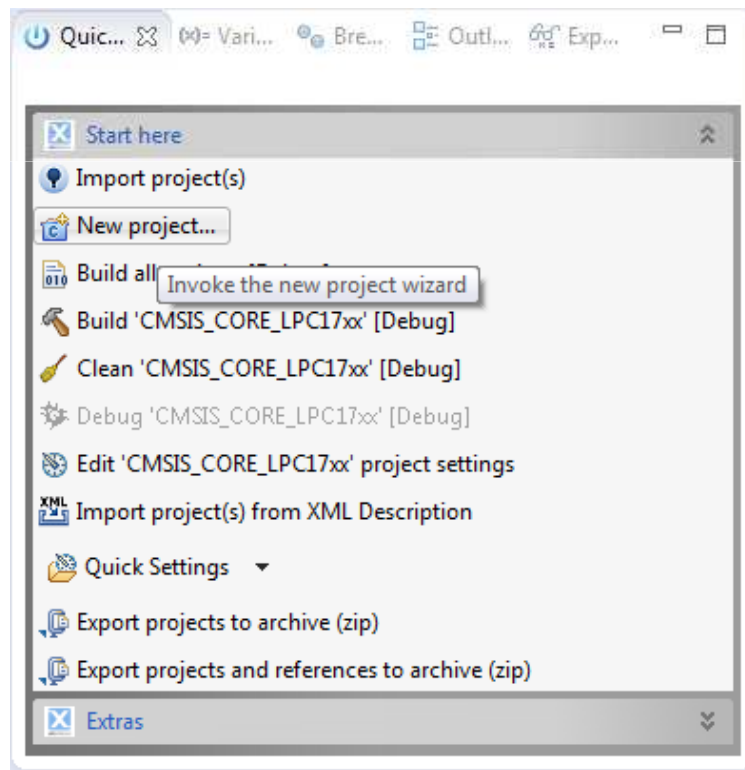
Importar Librerías

- => CMSIS_CORE_LPC17xx ○ CMSIS_LPC43xx_DriverLib



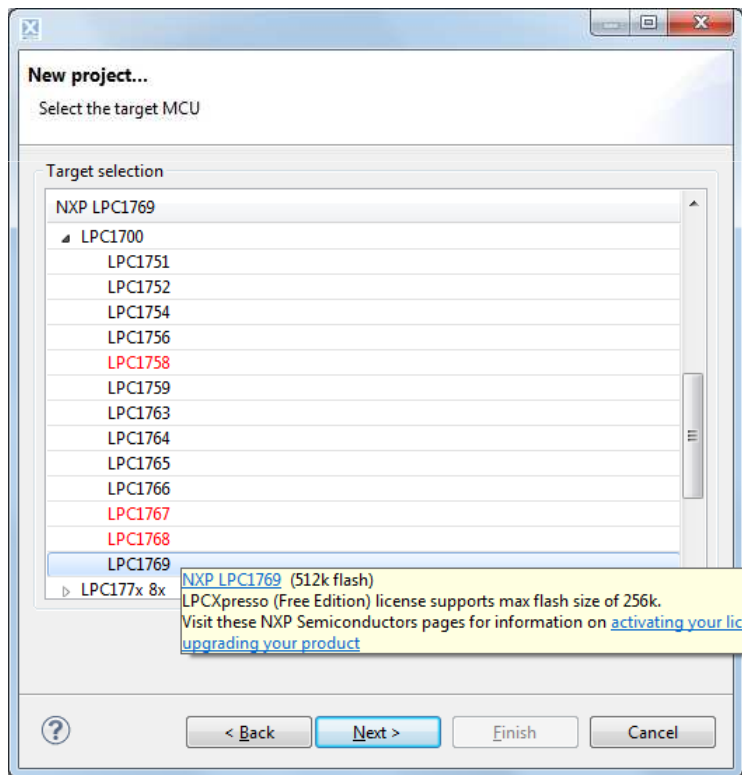
Genera un Nuevo Proyecto: LPC1769

- New Project... => LPC1700/LPC4000 => LPC175x_6x => C Project (semihosted) => Next > => 0-Intro => Next >

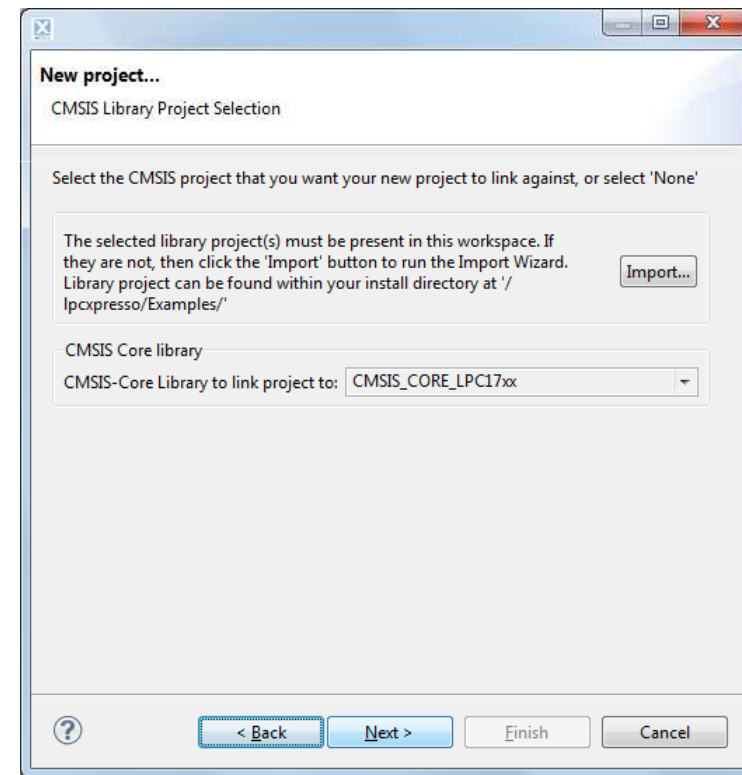


Genera un Nuevo Proyecto: LPC1769

- LPC1700 => LPC1769 => Next > => "C Project (Semihosted)" => CMSIS_CORE_LPC17xx / None => Next >

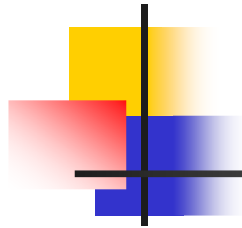


18 de Agosto de 2016



Ing. Juan Manuel Cruz

19



Genera un Nuevo Proyecto: LPC1769

- Next > ==> Next > ==> Finish

The image displays three sequential screenshots of the 'New project...' wizard in Keil uVision, illustrating the steps to create a new project for the LPC1769 microcontroller.

Screen 1: CMSIS DSP Library Project Selection

- Title: New project...
- Section: CMSIS DSP Library Project Selection
- Text: If required, select the CMSIS DSP library project that you want your new project to link against.
- Text: The selected library project must be present in this workspace. If it is not, then click the 'Import' button to run the Import Wizard. CMSIS DSP library projects can be found in your LPCXpresso install directory at '/lpcxpresso/Examples/CMSIS_DSPLIB'
- Button: Import..
- Dropdown: CMSIS DSP Library to link project to: None
- Text: For more information on the CMSIS DSP Library, visit <http://www.lpcware.com/content/faq/lpcxpresso/cmsis-dsp-library>.
- Buttons: ? < Back Next > Finish Cancel

Screen 2: Other options

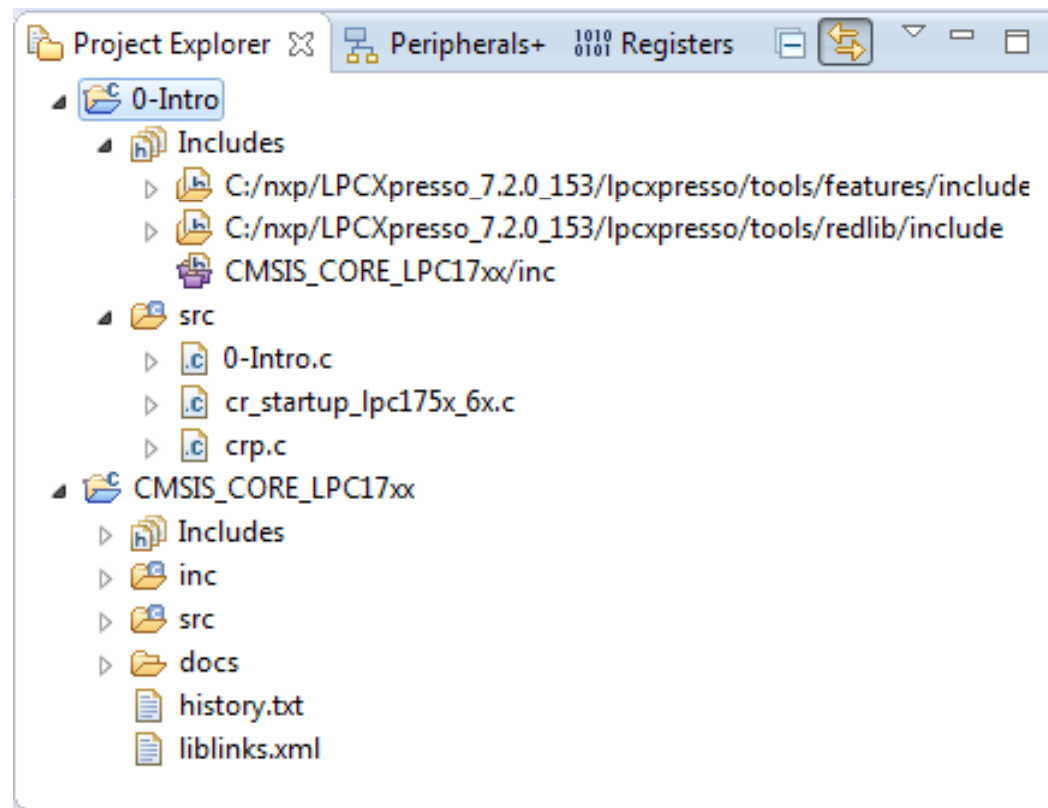
- Title: New project...
- Section: Other options
- Section: Code Read Protect (CRP)
 - Checkbox: Enable linker support for CRP
 - Text: See NXP documentation for your MCU at <http://www.nxp.com/products/microcontrollers> for more information on CRP.
- Section: Project Structure
 - Text: User source directory src
 - Checkbox: Create 'inc' directory and add to path
- Section: Compiler language dialect
 - Text: C Dialect Default
- Buttons: ? < Back Next > Finish Cancel

Screen 3: Printf options

- Title: New project...
- Section: Printf options
- Section: Redlib printf variants
 - Checkbox: Use non-floating point version of printf
 - Text: Using the non-floating point variants of printf family functions reduces code size, but means that you cannot output floating point values.
 - Checkbox: Use character rather than string based printf
 - Text: Using the 'character-by-character' printf provides slower semihosting output, but does not directly use the heap.
- Buttons: ? < Back Next > Finish Cancel

Genera un Nuevo Proyecto: LPC1769

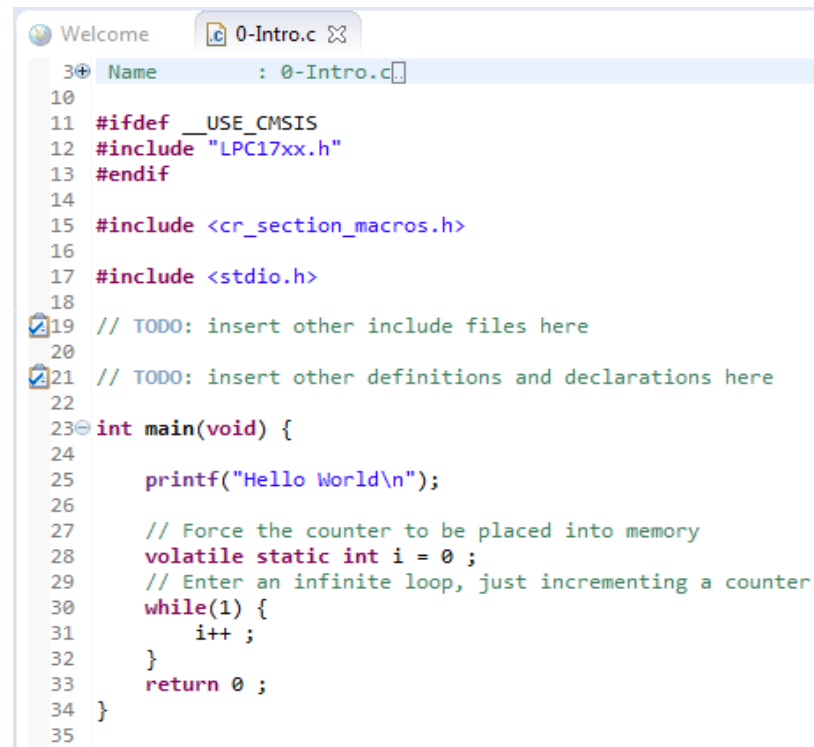
- => **0-Intro**





Abrir 0-Intro.c (LPC1769)

- => **0-Intro.c**

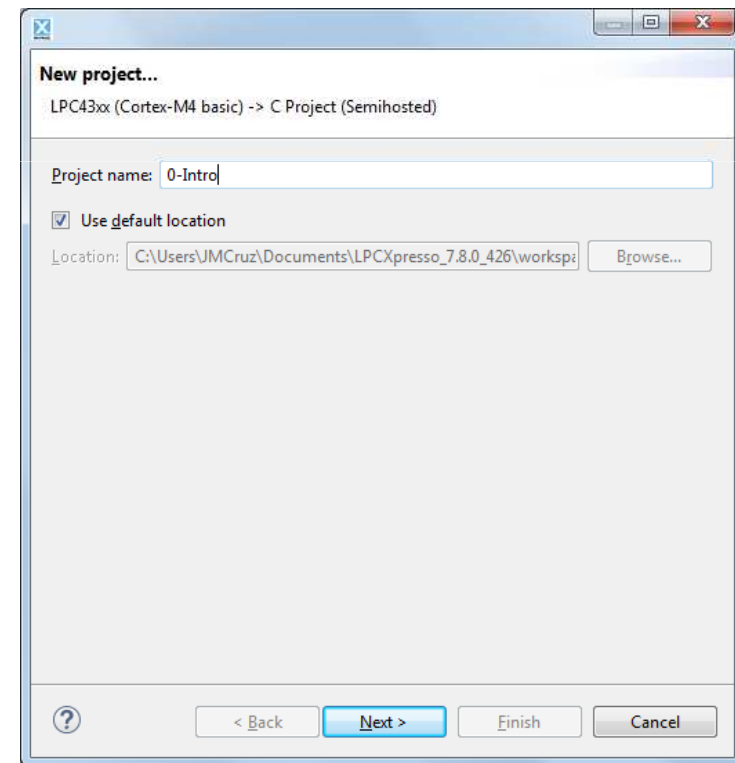
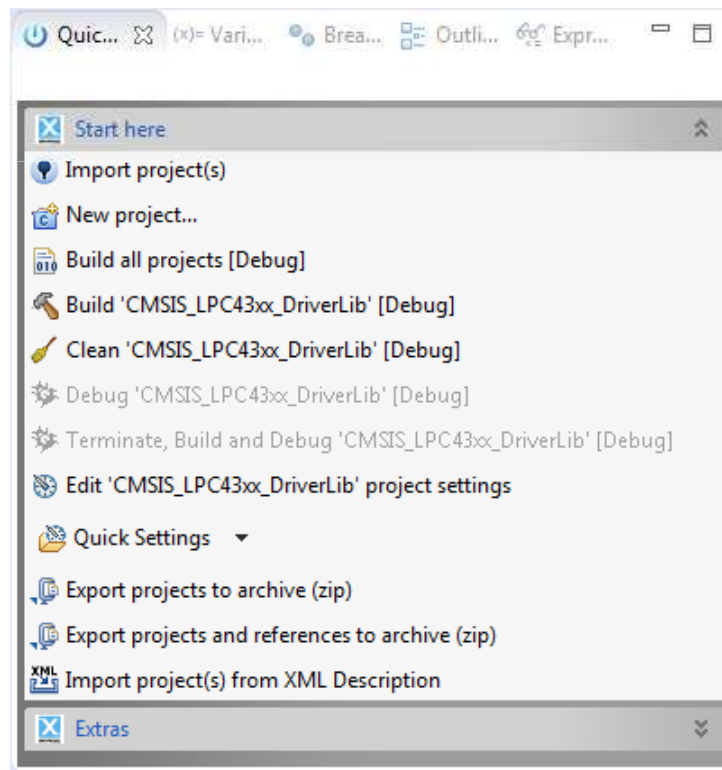


```
10
11 #ifndef __USE_CMSIS
12 #include "LPC17xx.h"
13 #endif
14
15 #include <cr_section_macros.h>
16
17 #include <stdio.h>
18
19 // TODO: insert other include files here
20
21 // TODO: insert other definitions and declarations here
22
23 int main(void) {
24     printf("Hello World\n");
25
26     // Force the counter to be placed into memory
27     volatile static int i = 0 ;
28     // Enter an infinite loop, just incrementing a counter
29     while(1) {
30         i++ ;
31     }
32     return 0 ;
33 }
34
35
```

- **Agregar éstas líneas a los fuentes para salvar un problema de Eclipse con la consola en Windows**
// http://wiki.eclipse.org/CDT/User/FAQ#Eclipse_console_does_not_show_output_on_Windows
setvbuf(stdout, NULL, _IONBF, 0);
setvbuf(stderr, NULL, _IONBF, 0);

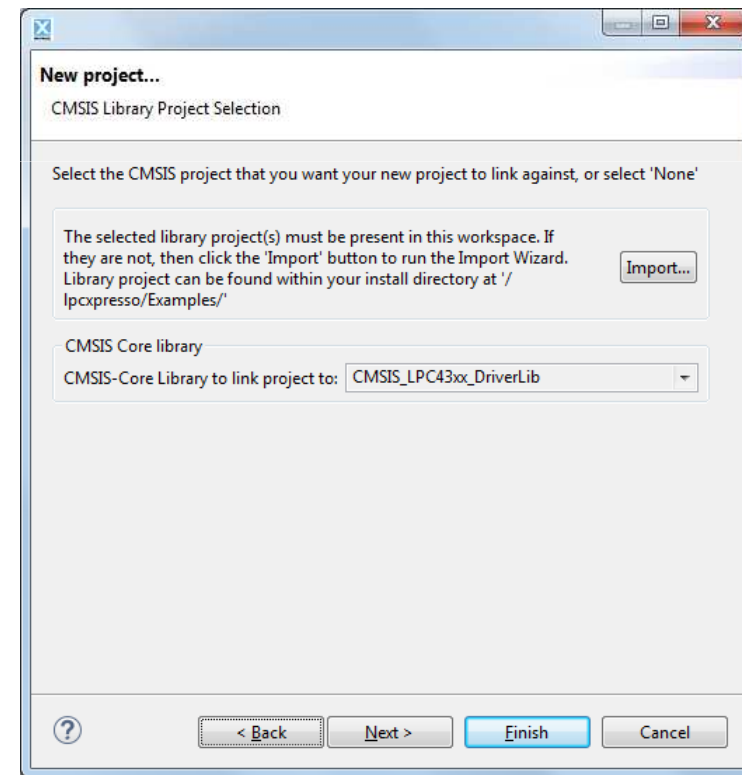
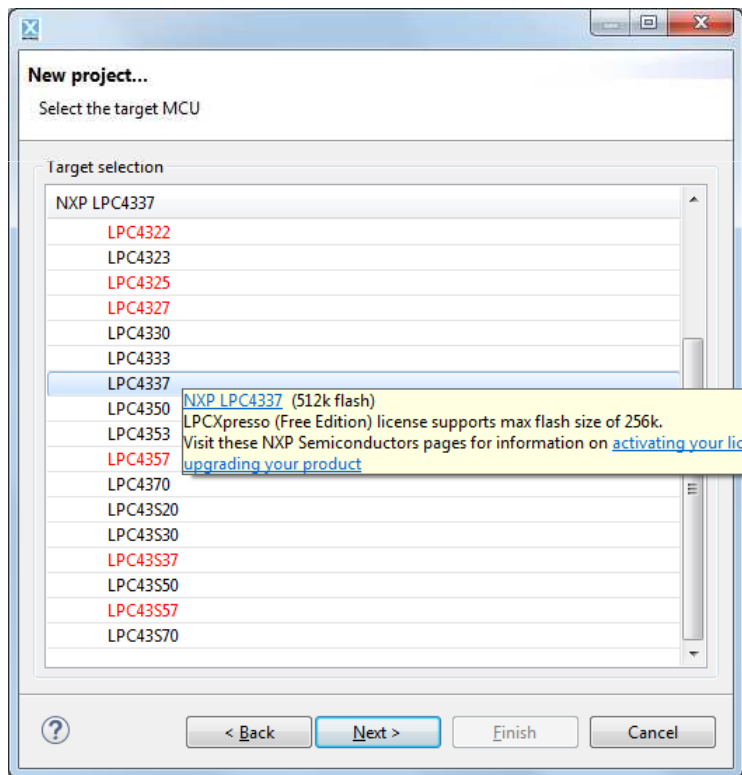
Genera un Nuevo Proyecto: LPC4337

- New Project... => LPC1800/LPC4300 => LPC43xx (Cortex M4 basic)
=> C Project (semihosted) => Next > => 0-Intro => Next >



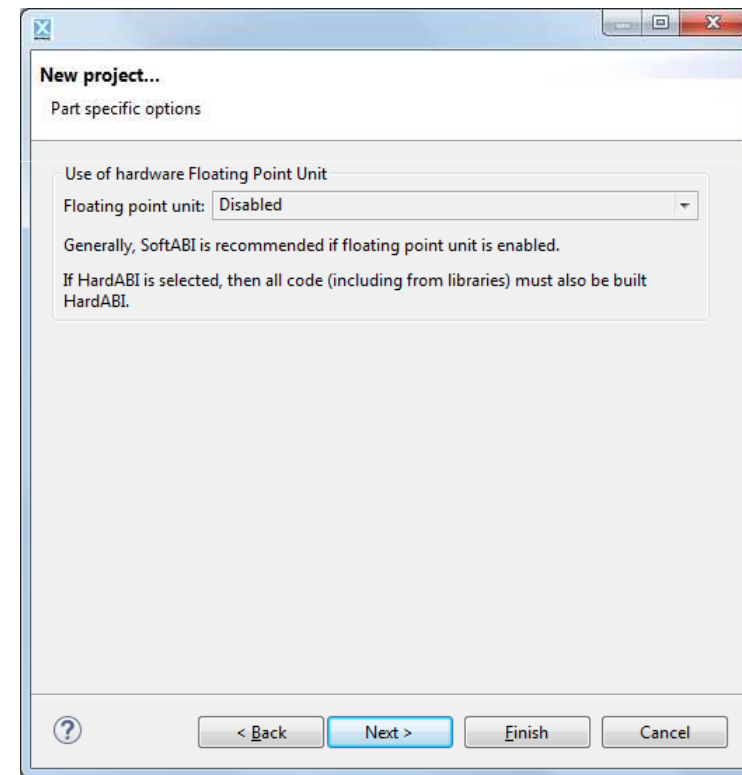
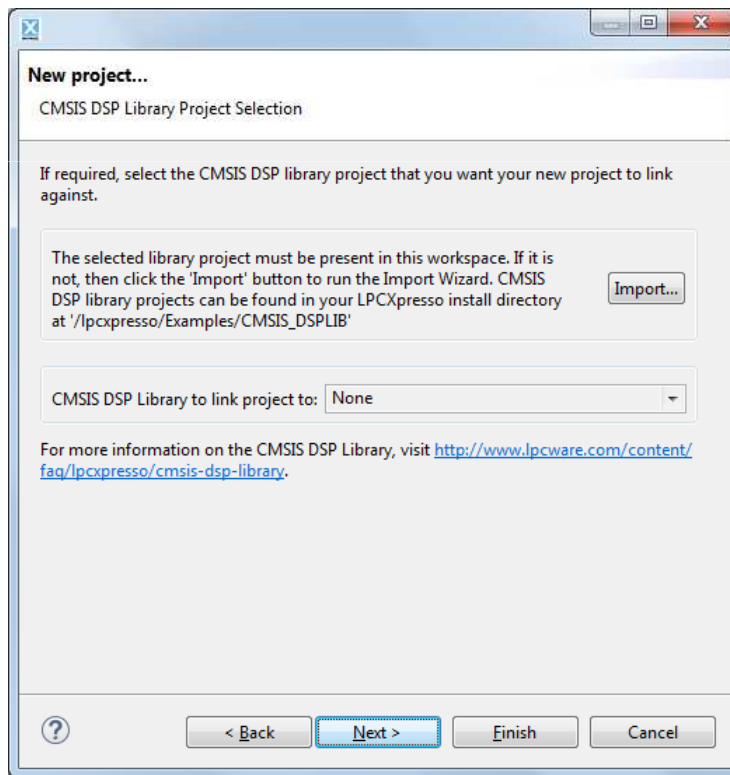
Genera un Nuevo Proyecto: LPC4337

- LPC4300 => LPC4337 => Next > => "C Project (Semihosted)" => CMSIS_LPC43xx_DriverLib / None => Next >



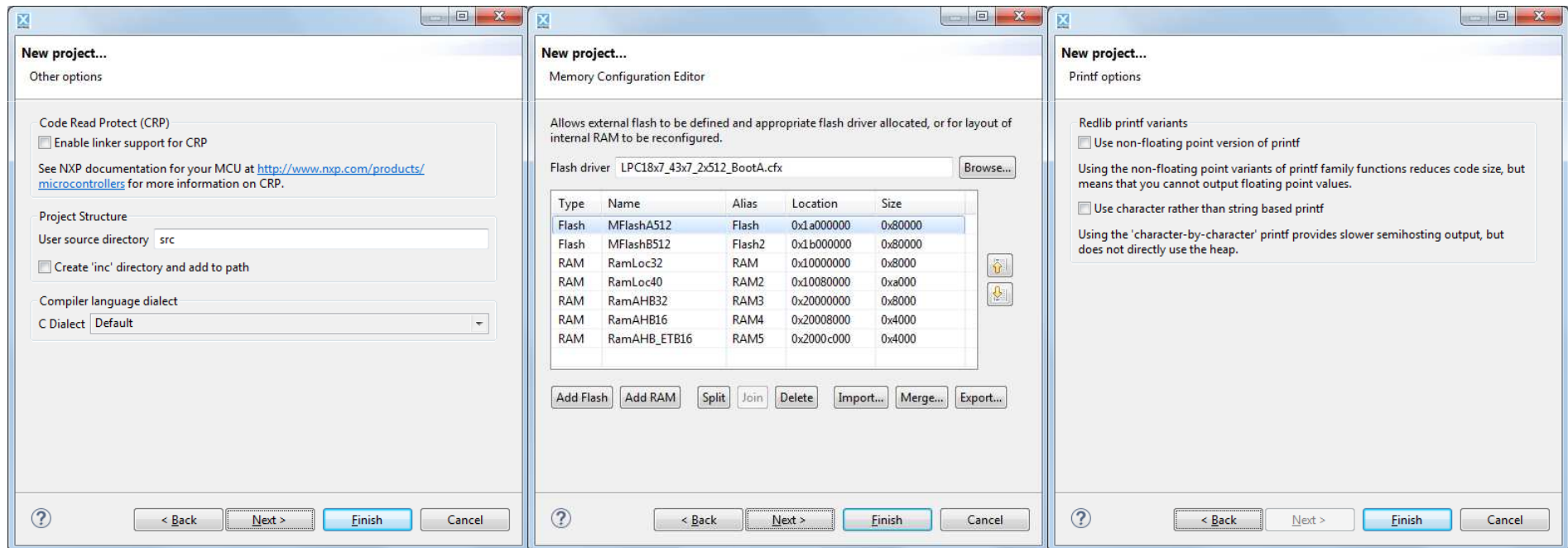
Genera un Nuevo Proyecto: LPC4337

- Next > => Floating point unit: Disabled => Next >



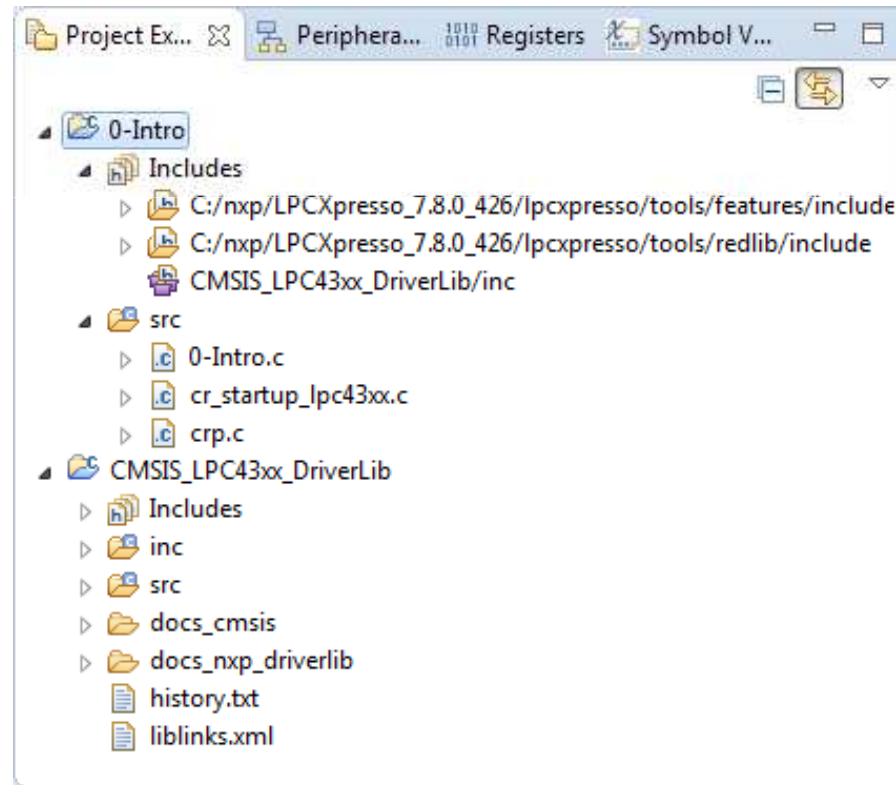
Genera un Nuevo Proyecto: LPC4337

- Next > ==> Next > ==> Finish



Genera un Nuevo Proyecto : LPC4337

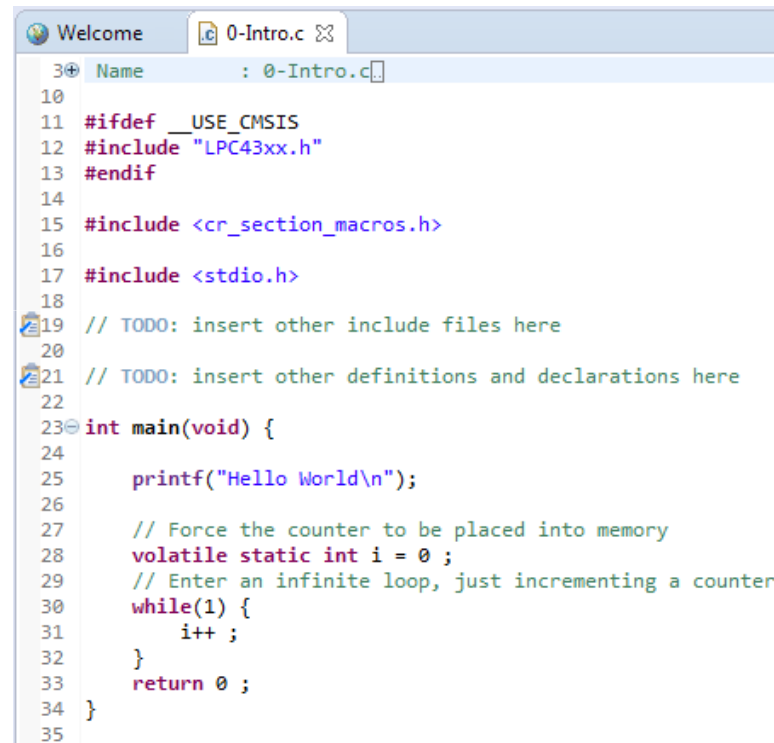
- => **0-Intro**





Abrir 0-Intro.c (LPC4337)

- => **0-Intro.c**

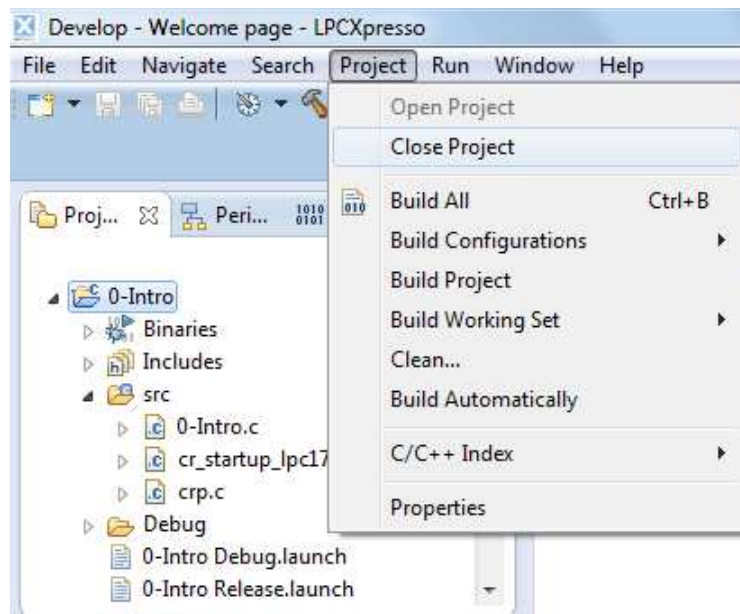


```
10
11 #ifndef __USE_CMSIS
12 #include "LPC43xx.h"
13 #endif
14
15 #include <cr_section_macros.h>
16
17 #include <stdio.h>
18
19 // TODO: insert other include files here
20
21 // TODO: insert other definitions and declarations here
22
23 int main(void) {
24     printf("Hello World\n");
25
26     // Force the counter to be placed into memory
27     volatile static int i = 0 ;
28     // Enter an infinite loop, just incrementing a counter
29     while(1) {
30         i++ ;
31     }
32     return 0 ;
33 }
34
35
```

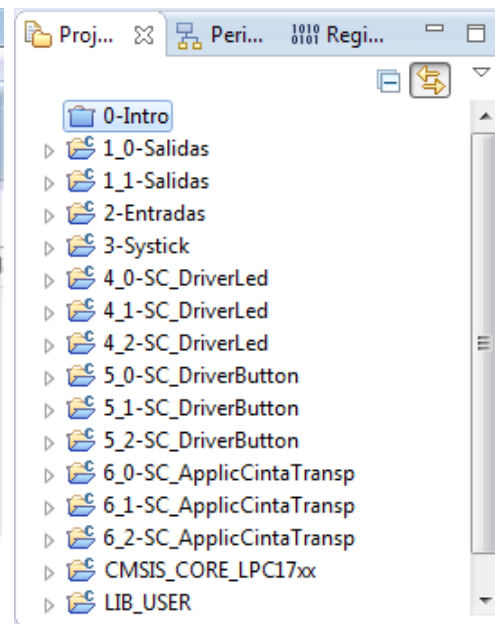
- **Agregar éstas líneas a los fuentes para salvar un problema de Eclipse con la consola en Windows**
// http://wiki.eclipse.org/CDT/User/FAQ#Eclipse_console_does_not_show_output_on_Windows
setvbuf(stdout, NULL, _IONBF, 0);
setvbuf(stderr, NULL, _IONBF, 0);

Tratamiento del Proyecto

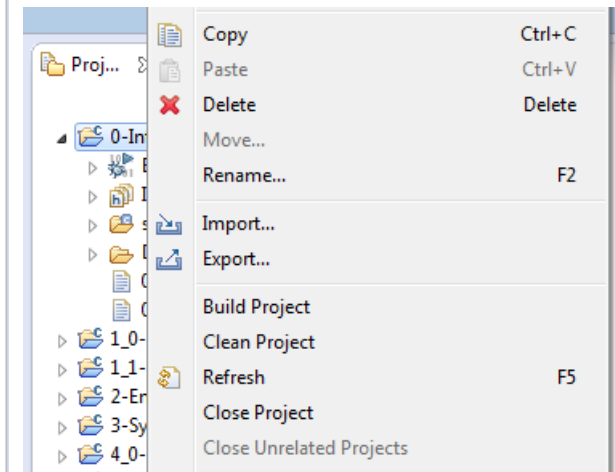
- Un proyecto se puede **Cerrar** o **Abrir** a voluntad
- También se puede **Copiar**, **Pegar**, **Borrar** (sólo del Workspace o del Workspace y del Disco), **Renombrar**, **Importar** (lo borrado del Workspace) y **Exportar** (no recomendable, hacerlo con export to zip)



18 de Agosto de 2016



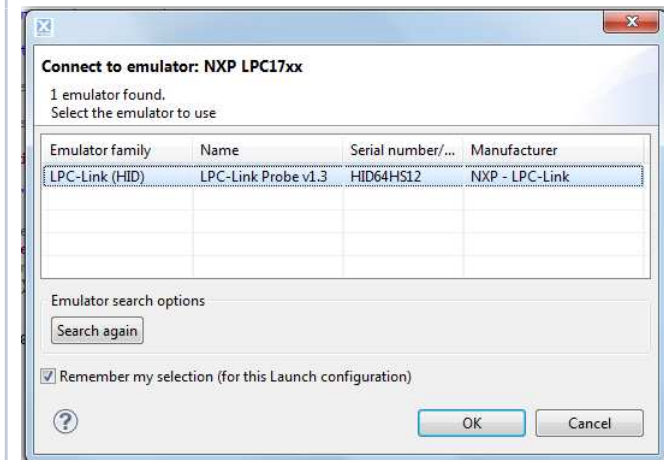
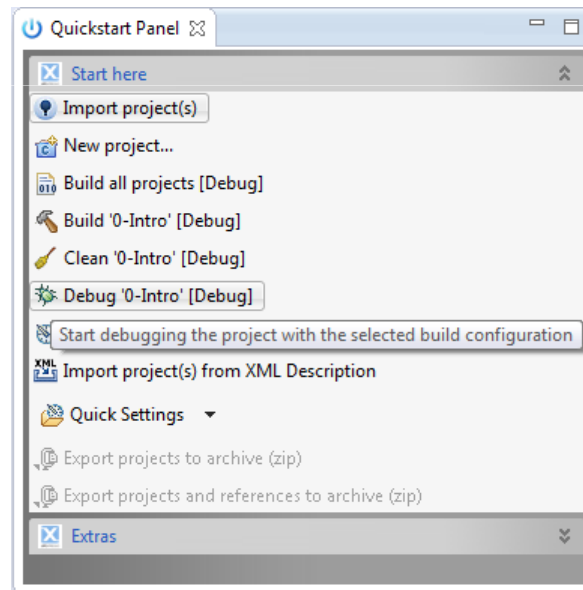
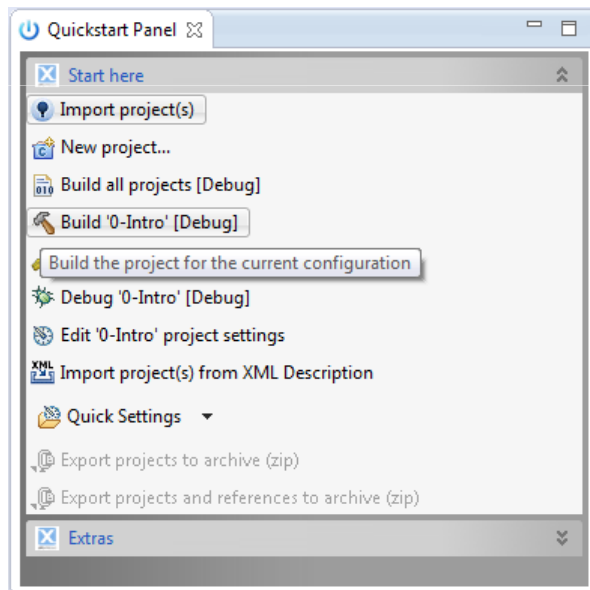
Ing. Juan Manuel Cruz



29

Clean/Build/Debug

- => Clean => Build => **Debug** (conectar LPC1769 LPCXpresso board ● LPC-Link 2 & EDU-CIAA-NXP)

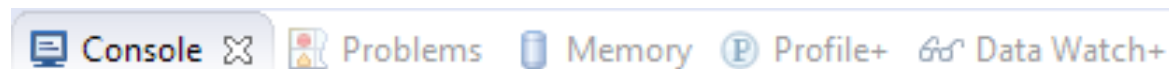
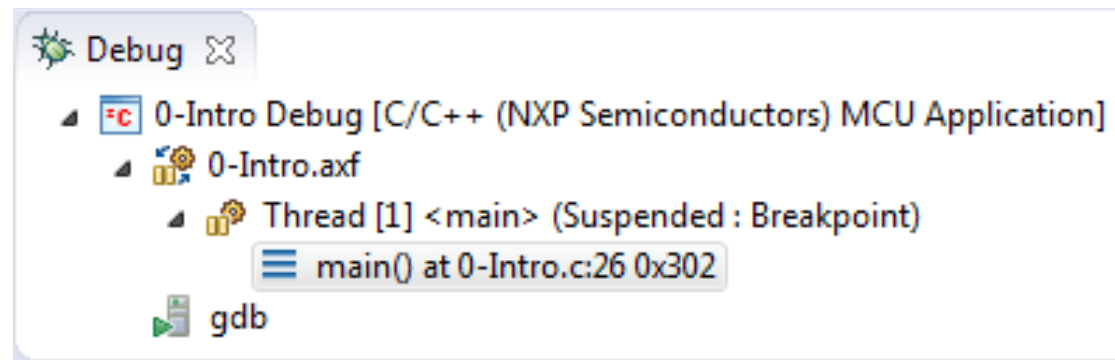


Tool bar/Debug & Console windows

- Restart /~~Drop To Frame~~/~~Use Step Filters~~/ Step Return

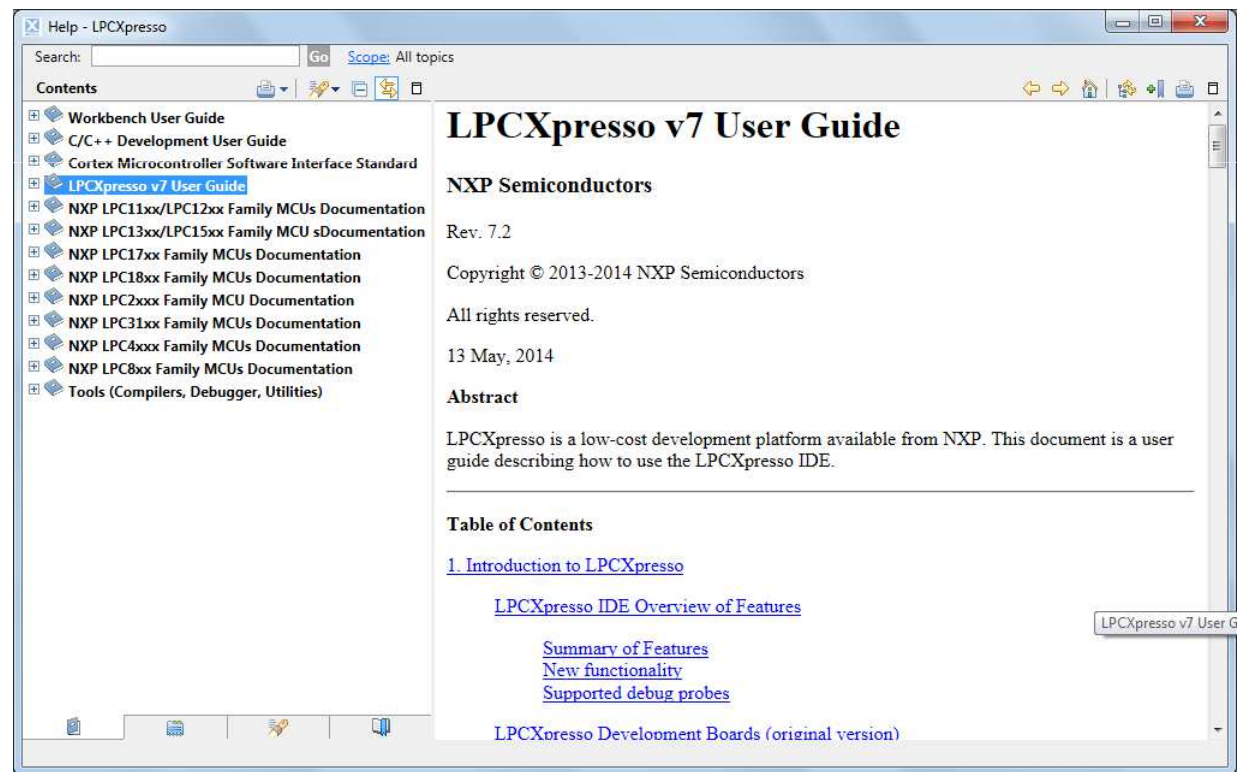
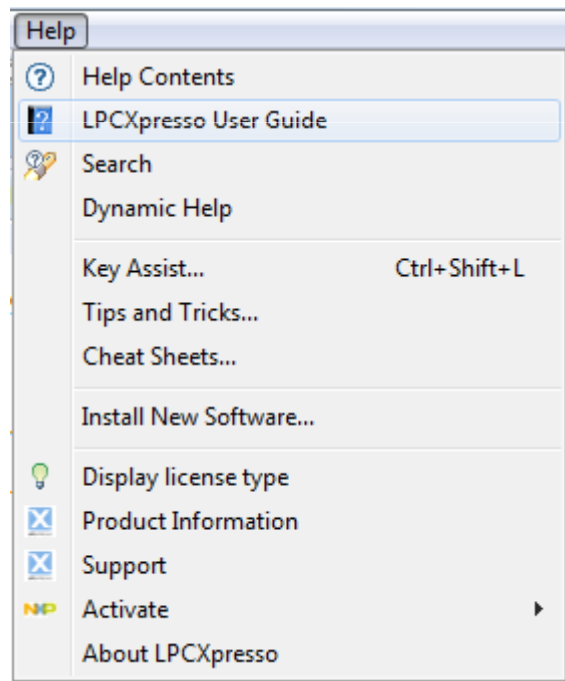


- Step Over /Step Into /Resume /Suspend /~~TerminateConsole~~/~~Disconnect~~



LPCXpresso User Guide

- => Help => **LPCXpresso User Guide**





Primer Actividad Práctica

- Se recomienda probar variantes de **printf()** y **getchar()**

<www.codingunit.com/printf-format-specifiers-format-conversions-and-formatted-output>

```
printf("The color: %s\n", "blue");
printf("First number: %d\n", 12345);
printf("Second number: %04d\n", 25);
printf("Third number: %i\n", 1234);
printf("Float number: %3.2f\n", 3.14159);
printf("Hexadecimal: %x\n", 255);
printf("Octal: %o\n", 255);
printf("Unsigned value: %u\n", 150);
printf("Just print the percentage sign %%\n", 10);
```

```
printf(":%s:\n", "Hello, world!");
printf(":%15s:\n", "Hello, world!");
printf(":%.10s:\n", "Hello, world!");
printf(":%-10s:\n", "Hello, world!");
printf(":%-15s:\n", "Hello, world!");
printf(":%.15s:\n", "Hello, world!");
printf(":%15.10s:\n", "Hello, world!");
printf(":%-15.10s:\n", "Hello, world!");
```

```
int a,b;
float c,d;

a = 15;
b = a / 2;
printf("%d\n",b);
printf("%3d\n",b);
printf("%03d\n",b);

c = 15.3;
d = c / 3;
printf("%3.2f\n",d);
```

```
int Fahrenheit;

for (Fahrenheit = 0; Fahrenheit <= 300; Fahrenheit = Fahrenheit + 20)
    printf("%3d %06.3f\n", Fahrenheit, (5.0/9.0)*(Fahrenheit-32));
```

```
char letra;
letra = getchar( );
printf("La letra tecleada es: %c", letra);
```

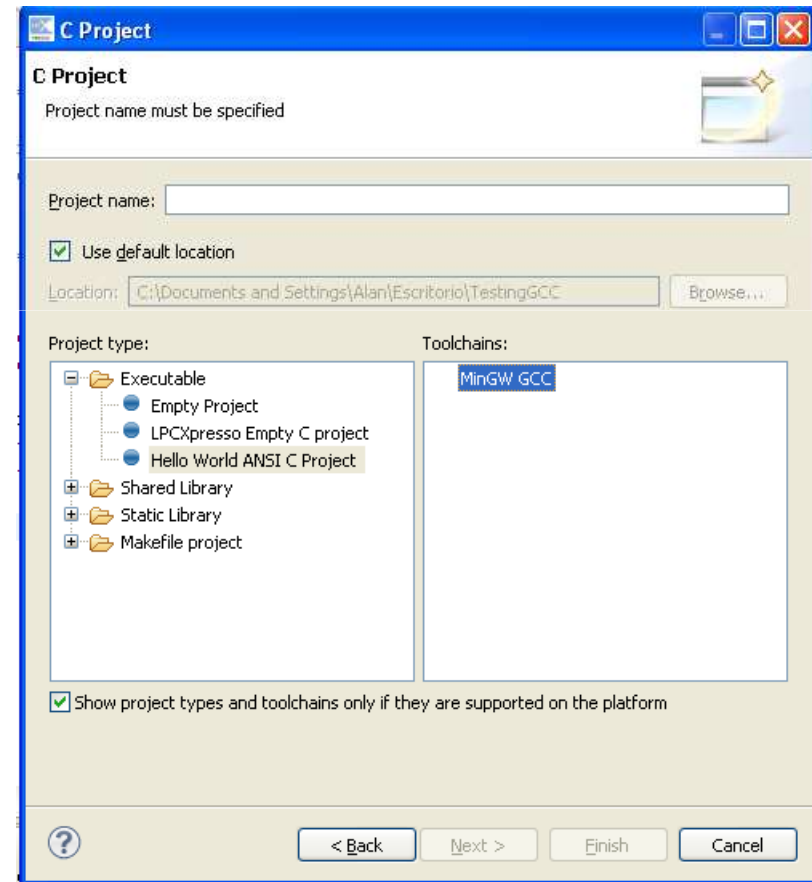


Primer Actividad Práctica

- Para ésta primer actividad Ud. no requiere del hardware, sólo se necesita el entorno del LPCXpresso IDE y compilar para Windows (utilizar el compilador GCC estándar)
- Un entorno recomendable para instalar el GCC en su computadora y así poder compilar es el **MinGW** (www.mingw.org/)
 - Esta instalación hace que el Eclipse (y en este caso en particular el LPCXpresso) vean directamente como instalado al compilador cosa que nos facilita el trabajo
- Los pasos a seguir para que esto funcione son:
 - Descargar el MinGW desde el link arriba indicado
 - Instalarlo con todas las opciones por defecto (es decir: next, next, next, ..., finish)

Primer Actividad Práctica

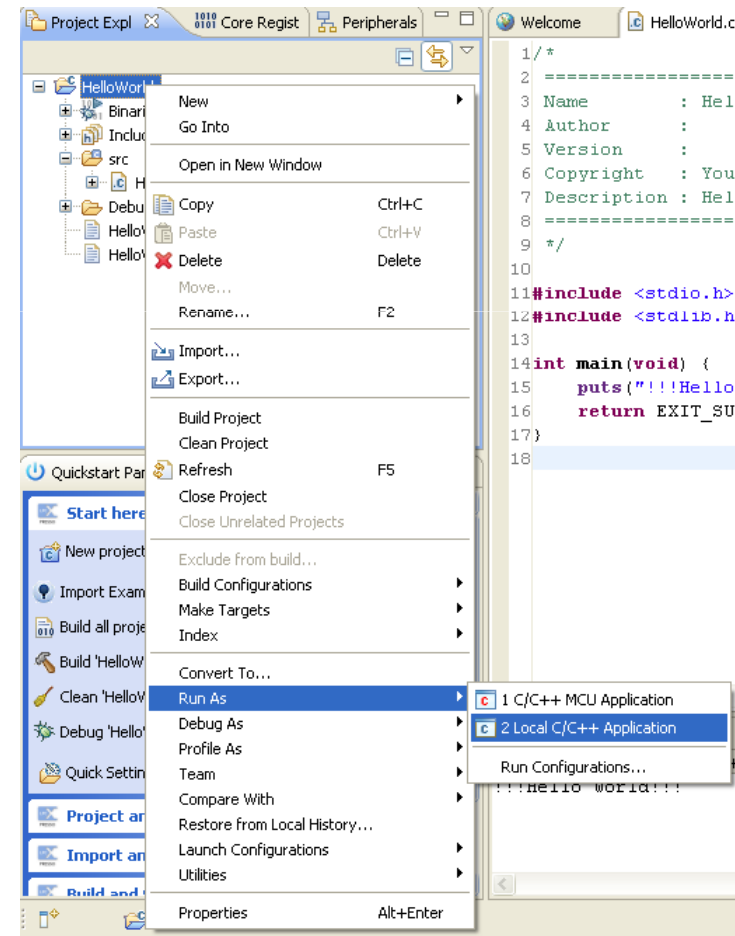
- Para **crear** un **proyecto nuevo** dentro del **LPCXpresso IDE**:
 - File -> New -> Project
- Si el entorno **MinGW** se instaló correctamente al crear un proyecto nuevo y seleccionar "**C Project**" se deberá ver algo similar a la figura
- Se recomienda utilizar el "**Hello world ANSI C Project**" pues el mismo viene configurado con lo necesario para ejecutarse en Windows así como un archivo **main.c** donde escribir el código



Primer Actividad Práctica

- Para **ejecutar** el **proyecto nuevo**:
 - Build
 - Run As -> Local C/C++
- Veremos que el **stdout** se redireccionará a la consola de abajo así también como el **stdin**
- **IMPORTANTE. Eclipse tiene un problema de refresco con la consola y** debemos incluir un par de líneas al comienzo de nuestro programa:

```
setvbuf (stdout , NULL , _IONBF , 0);  
setvbuf (stderr , NULL , _IONBF , 0);
```



Primer Actividad Práctica

- Para correr el **debugger** se deberá hacer algo similar, la primera vez aparece un cuadro de dialogo en el que debemos elegir la opción:
 - **"Standard Create ..."**
 - **"OK"**
- Y luego el Eclipse entra en modo **"Debug"**

