

Manual EclipseARM

El contenido de este manual se ha probado, de momento, en las siguientes plataformas:

- Máquina virtual Windows 7 32 bits
- Windows 7 64 bits
- Máquina virtual Debian 6 32 bits
- Máquina virtual Windows XP 32 bits (de momento sólo simulador)

[1.1. Instalación en Windows](#)

[1.2. Instalación en Linux \(32 bits\)](#)

[2. Ejecución de proyecto en simulación](#)

[3. Ejecución de un proyecto en hardware](#)

[Preguntas frecuentes \(FAQs\)](#)

[Aunque tengo el PATH con la ruta del compilador cruzado de Sourcery, eclipse parece no encontrarlo ...](#)

[¿Cómo puedo saber que la memoria flash está configurada correctamente?](#)

1.1. Instalación en Windows

Para instalar el entorno nos descargamos el archivo EclipseARM.exe de la siguiente URL:

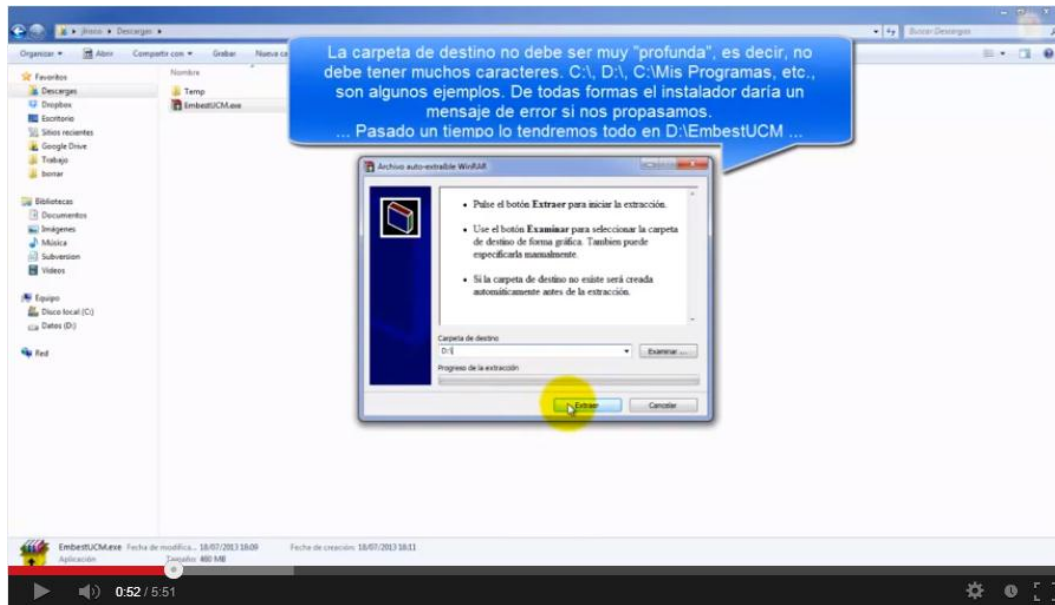
- [EclipseARM](#)

Después lo ejecutamos y lo instalamos. Ojo, la carpeta de destino no debe ser muy extensa (en número total de caracteres) ya que el extraíble tiene una profundidad máxima de 230 caracteres aproximadamente (EclipseARM/eclipse/plugins/org.eclipse....) y Windows no soporta más de 260 caracteres en una ruta de archivo. A esta carpeta la llamaremos TARGET.

Ahora tenemos que hacer dos cosas:

1. Añadir al PATH del sistema la ruta:
TARGET\EclipseARM\sourcery-g++-lite-arm-2011.03\bin
2. **Sólo para depuración hardware:** Enchufar ARM y OLIMEX a sendos puertos USBs del PC, y actualizar los drivers del OLIMEX, en la ruta:
TARGET\EclipseARM\Repo\DRIVERS-(libusb-1.2.2.0-CDM20808)
3. **Sólo para depuración hardware:** Configurar en eclipse el OpenOCD (sólo se hace una vez).

Todos los pasos a seguir quedan reflejados en el siguiente vídeo:



[Vídeo: Instalación EclipseARM](#)

1.2. Instalación en Linux (32 bits)

Se proporcionan todas las fuentes ([EclipseARM/Linux](#)) para la instalación en SO Linux 32 bits. No obstante, convendría intentar instalar previamente el `openocd` que venga con la distribución, lo que simplifica en gran medida la instalación manual.

Sin embargo, hemos detectado que por ejemplo, en Debian 6 el `openocd` no soporta las `libftdi` usadas con el ARM. La instalación completamente manual consiste en los siguientes pasos:

- Crear una carpeta `/home/usuario/EclipseARM`, por ejemplo
- Descomprimir `eclipse` en `EclipseARM/`
- Descomprimir `jre` en `eclipse/`
- Ejecutar `./eclipse` como usuario (desde shell) e instalar los plugins recomendados en las diapositivas de Luis, Christian, Juan Carlos, etc.
- Descomprimir `arm-2011.03-42-arm-none-eabi-i686-pc-linux-gnu` en `EclipseARM/`
- Añadir `/home/usuario/EclipseARM/arm-2011.03/bin` al `PATH` del sistema (modificando `.bashrc`, por ejemplo)
- Instalar `libusb` (tanto librerías como `-dev`) (`apt-get ...`)
- Instalar `cmake` (`apt-get ...`)
- Descomprimir `libftdi1-1.0` (donde se quiera)
 - `mkdir build`
 - `cd build`
 - `cmake -DCMAKE_INSTALL_PREFIX="/usr" ../`
 - `make`
 - (como root) `make install`
- Descomprimir `openocd-0.7.0`
 - Copiar los script de Christian (`arm-fdi-ucm.cfg`, `embest_s3cev40.cfg` y `samsung_s3c44b0x.cfg`) a `/test`, `/board` y `/target`, respectivamente, dentro de `tcl`
 - `./configure --enable-ft2232_libftdi`

- Cuidado aquí: Observar los mensajes de salida. Seguramente haya que crear enlaces simbólicos `/usr/include/ftdi.h -> /usr/include/libftdi1/ftdi.h`, `/usr/lib/libftdi.a -> /usr/lib/libftdi-1.a`, etc.

- make
- (como root) make install

Con todo esto, se aplican los mismos pasos que en el vídeo anterior para OpenOCD, pero en este caso, el ejecutable está en `/usr/local/bin/openocd` y el directorio de trabajo es `/usr/local/share/openocd/scripts` (que en windows se quedaba vacío)

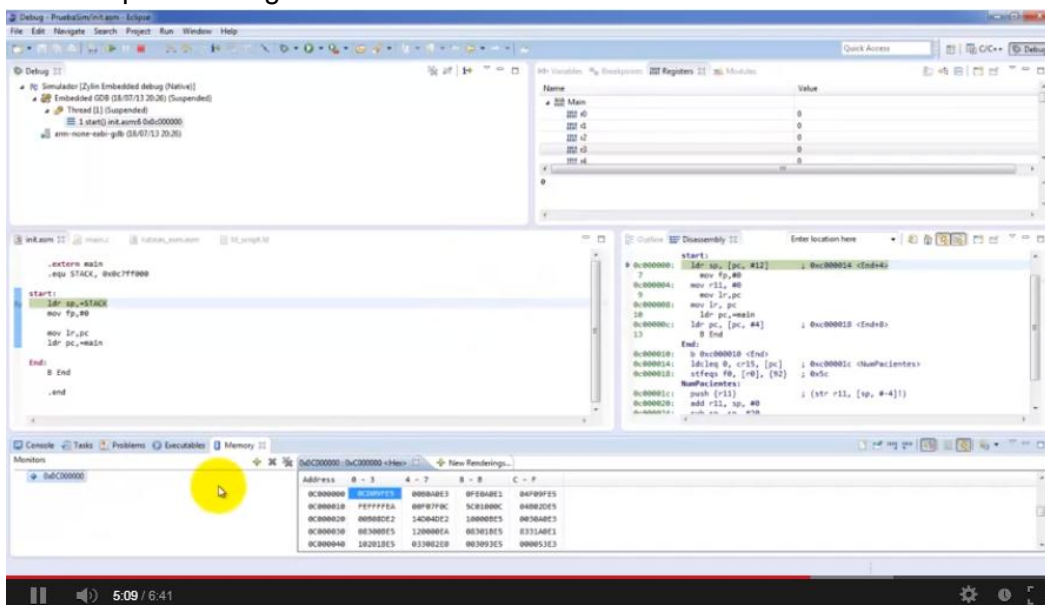
Finalmente, para poder dar permisos al OpenOCD para que use los puertos USB, podemos crear un archivo en `/etc/udev/rules.d/libusb.rules` que contenga:

`SUBSYSTEM=="usb", ENV{DEVTYPE}=="usb_device", MODE="0666"`

Alternativamente, se puede proporcionar una máquina virtual Debian 6, con todo instalado. Tenéis que pedírsela a José L. o a Christian.

2. Ejecución de proyecto en simulación

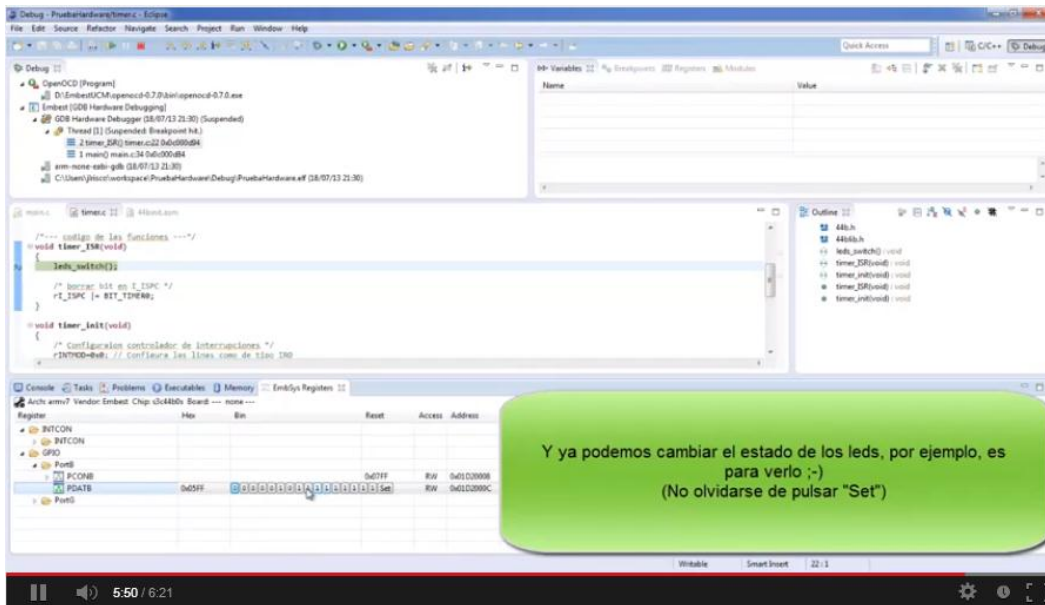
La configuración de un proyecto para simulación es sumamente fácil. El siguiente vídeo ilustra cada uno de los pasos a seguir.



[Vídeo: Simulación Proyecto ARM](#)

3. Ejecución de un proyecto en hardware

Ya para terminar, mostramos un vídeo de como ejecutar un proyecto en hardware. Además mostramos la vista que permite ver los registros del controlador y en general cualquier dispositivo de E/S. Ya envió Christian detalles de cómo configurar este visor por correo electrónico. Incorporaremos aquellas instrucciones en este manual cuando sea posible.



[Vídeo: Proyecto ARM en hardware](#)

Preguntas frecuentes (FAQs)

Aunque tengo el PATH con la ruta del compilador cruzado de Sourcery, eclipse parece no encontrarlo ...

Este problema puede deberse a dos razones: (1) Has actualizado el PATH con eclipse abierto, por lo que debes cerrarlo y volver a abrirlo, o (2) has puesto algún espacio en blanco en la variable PATH, tipo "Ruta1;_RutaSourcery", donde "_" sería un espacio en blanco.

¿Cómo puedo saber que la memoria flash está configurada correctamente?

Al encender el maletín, se observa en la pantalla del mismo un menú para probar la configuración de los distintos componentes del maletín. Si este menú no aparece seguramente es que la flash no es la correcta. Para restaurarla a su versión original, tenemos un archivo .BAT en el directorio Repo, que podremos ejecutar con el maletín conectado.