

ERIKA Enterprise Tutorial

per la piattaforma Microchip dsPIC (R) DSC

versione: 1.1.6
27 maggio 2009



A proposito di Evidence S.r.l.

Evidence è una spin-off del ReTiS Lab della Scuola Superiore S. Anna in Pisa. Siamo esperti nel dominio dei sistemi embedded e real-time, con una conoscenza profonda del processo di progetto e specifica del software embedded. Le nostre metodologie ed i nostri tool forniscono un passo in avanti nel campo dei sistemi dedicati di prossima generazione, con particolare riferimento ai sistemi multicore su singolo chip, hardware riconfigurabile, scheduling dinamico ed altro ancora!

Come contattarci

Indirizzo:

Evidence Srl,

Via Carducci 64/A

Località Ghezzano

56010 S.Giuliano Terme

Pisa - Italy

Tel: +39 050 991 1122, +39 050 991 1224

Fax: +39 050 991 0812, +39 050 991 0855

Per informazione sui prodotti Evidence, scrivete a: info@evidence.eu.com.

Altre informazioni riguardo i prodotti di Evidence possono essere trovate all'indirizzo:

<http://www.evidence.eu.com>.



This document is Copyright 2005-2008 Evidence S.r.l.

Information and images contained within this document are copyright and the property of Evidence S.r.l. All trademarks are hereby acknowledged to be the properties of their respective owners. The information, text and graphics contained in this document are provided for information purposes only by Evidence S.r.l. Evidence S.r.l. does not warrant the accuracy, or completeness of the information, text, and other items contained in this document. Matlab, Simulink, Mathworks are registered trademarks of Matworks Inc. Microsoft, Windows are registered trademarks of Microsoft Inc. Java is a registered trademark of Sun Microsystems. OSEK is a registered trademark of Siemens AG. The Microchip Name and Logo, and Microchip In Control are registered trademarks or trademarks of Microchip Technology Inc. in the USA. and other countries, and are used under license. All other trademarks used are properties of their respective owners. This document has been written using LaTeX and LyX.

Indice

1	Introduzione a RT-Druid e Erika Enterprise per Microchip dsPIC (R) DSC	4
2	Installazione di Erika Enterprise e RT-Druid	5
3	Prima esecuzione e configurazione di RT-Druid	8
4	Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC	12
5	Storia del documento	20

1 Introduzione a RT-Druid e Erika Enterprise per Microchip dsPIC (R) DSC

Questa guida descrive una serie di passi necessari per installare e compilare un'applicazione di esempio che mostra le caratteristiche principali di Erika Enterprise e di RT-Druid per la piattaforma dsPIC (R) DSC di Microchip.

Questo tutorial è stato realizzato usando la scheda FLEX prodotta da Evidence ed Embedded Solutions, e la scheda di sviluppo Microchip Explorer 16 prodotta da Microchip.

In questo documento supponiamo che il Lettore sia familiare con l'utilizzo del tool MPLAB IDE fornito da Microchip.

2 Installazione di Erika Enterprise e RT-Druid

Questo capitolo intende guidare lo sviluppatore nella procedura di installazione per Microsoft Windows di Erika Enterprise e RT-Druid per la piattaforma Microchip dsPIC (R) DSC.

Attenzione: Il percorso di installazione di Cygwin, di Erika Enterprise ed i percorsi utilizzati per i workspace di RT-Druid (un workspace è una cartella che contiene i progetti software realizzati dall'utente) NON DEVONO contenere spazi bianchi. Cartelle con nomi contenenti spazi bianchi possono provocare il malfunzionamento di Erika Enterprise.

Ad esempio, `C:\MyApplications\Evidence\` e `C:\MyApplications\Cygwin\` sono nomi corretti, mentre OCCORRE EVITARE nomi come `C:\My Applications\Evidence\` o `C:\My Applications\Cygwin\`.

La problematica relativa agli spazi nei nomi di file verrà risolta a breve nelle prossime versioni di Erika Enterprise.

L'installazione di Erika Enterprise e RT-Druid è composta dalle seguenti parti:

- L'ambiente di sviluppo Eclipse, utilizzato da RT-Druid per fornire l'ambiente di sviluppo per le applicazioni basate su Erika Enterprise.
- Una versione del Java Runtime Environment (JRE) che deve essere installato sulla macchina su cui occorre installare RT-Druid (JRE è necessario in quanto Eclipse è stato completamente scritto utilizzando Java).
- I plugins di RT-Druid che forniscono il supporto per la generazione di codice per Erika Enterprise.
- Il codice sorgente di Erika Enterprise.
- Il pacchetto Microchip MPLAB IDE.
- Il compilatore Microchip C30.
- Una versione del compilatore Microchip C30 ricompilata dai sorgenti GCC messi a disposizione da Microchip, che permette di compilare codice C senza la necessità di acquisire una licenza del compilatore C30 di Microchip.
- Un insieme di esempi per la piattaforma Microchip dsPIC (R) DSC che possono essere utilizzati per compilare un primo insieme di applicazioni per la scheda FLEX prodotta da Evidence/Embedded Solutions, la scheda Explorer 16 prodotta da Microchip, ed altre schede.

- Un sottoinsieme dell'ambiente Cygwin [1], che include un insieme di programmi di utilità come `make` e `gawk`, utilizzate nel processo di compilazione di una applicazione basata su Erika Enterprise.

Per installare il software, è necessario seguire i seguenti passi:

1. Installate il vostro Java Runtime Environment (JRE) preferito, necessario per l'esecuzione di Eclipse e per l'esecuzione dei plugin di RT-Druid.
2. Installate l'ultima versione dell'ambiente Microchip MPLAB IDE. Selezionate tranquillamente la directory di installazione predefinita. Alla fine del processo di installazione, è necessario effettuare il reboot del sistema come richiesto dall'installer.
3. Installate il compilatore Microchip C30, disponibile dal sito web Microchip. Anche in questo caso, può essere utilizzata la directory di installazione predefinita. Durante l'installazione potrà venire richiesto di cambiare il "default environment". Accettate il cambiamento.
4. Eseguite il pacchetto di installazione di Erika Enterprise e RT-Druid.
5. Il pacchetto richiederà la selezione di una lista di pacchetti da installare. Selezionate i pacchetti da installare e proseguite con la procedura di installazione (vedere Figura 2.1).

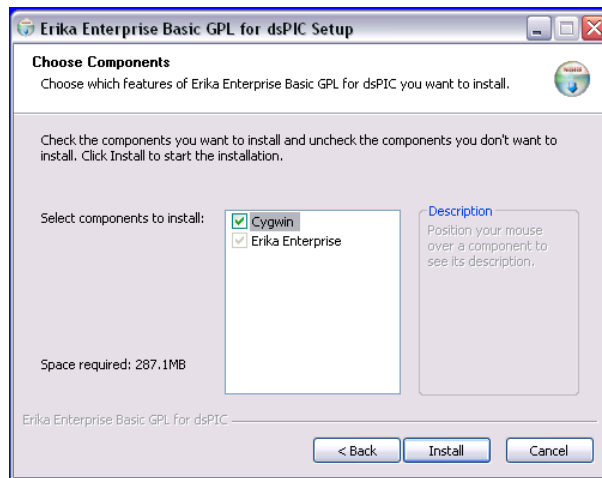


Figura 2.1: Questa figura visualizza la dialog box contenente i pacchetti di installazione disponibili.

Nota Bene: Il pacchetto di installazione di **Erika Enterprise** fornisce una versione del compilatore C30 di Microchip ricompilato dai sorgenti GCC resi disponibili da Microchip. Anche se tale compilatore può compilare applicazioni C basate su **Erika Enterprise** esso non include le librerie delle periferiche dei dispositivi Microchip, che sono distribuite solo con la versione “ufficiale” del compilatore C30.

6. Una volta terminata l'installazione, controllate la *prima* linea del file che si trova alla locazione `evidencedir\bin\mymake_cygwin.bat` (dove `evidencedir` è la cartella scelta per l'installazione del sistema). Per esempio, se Cygwin è stato installato nella cartella `C:\cygwin`, allora la prima linea di tale file dovrà essere la seguente:

```
@set EE_BASH_PATH=C:\cygwin\bin\bash
```

...ovvero, la linea deve contenere il percorso corretto verso il file `bash.exe` contenuto nella installazione di Cygwin. Se avete accettato le cartelle di installazione predefinite nell'installer, la cartella corretta deve essere `C:\cygwin\bin\bash` come specificato nell'esempio precedente.

Nota Bene: Vi chiediamo di effettuare questa verifica in quanto in alcune versioni di Windows l'installer di Cygwin non imposta correttamente le chiavi di registro che poi vengono successivamente utilizzate dall'installer di **Erika Enterprise** per la creazione del file.

La restante parte di questo tutorial suppone che Microchip MPLAB IDE sia installato nella cartella `C:\Programmi\Microchip` e che lo GNU Assembler per dsPIC (R) DSC sia installato nella cartella `C:\Programmi\Microchip\MPLAB ASM30\ Suite\bin`. Notate che questi valori possono essere differenti da quelli presenti sul vostro PC.

3 Prima esecuzione e configurazione di RT-Druid

Dopo aver installato tutti i pacchetti richiesti dal sistema, siete pronti per eseguire per la prima volta RT-Druid. Eseguite i seguenti passi:

1. Come primo passo, lanciate la Eclipse IDE dal menu Evidence all'interno del menu Start, scegliendo Start/Programmi/Evidence/RT-Druid.
2. Una dialog box apparirà richiedendo la scelta del workspace (vedere Figura 3.1). confermate la directory di default, e procedete premendo "OK".

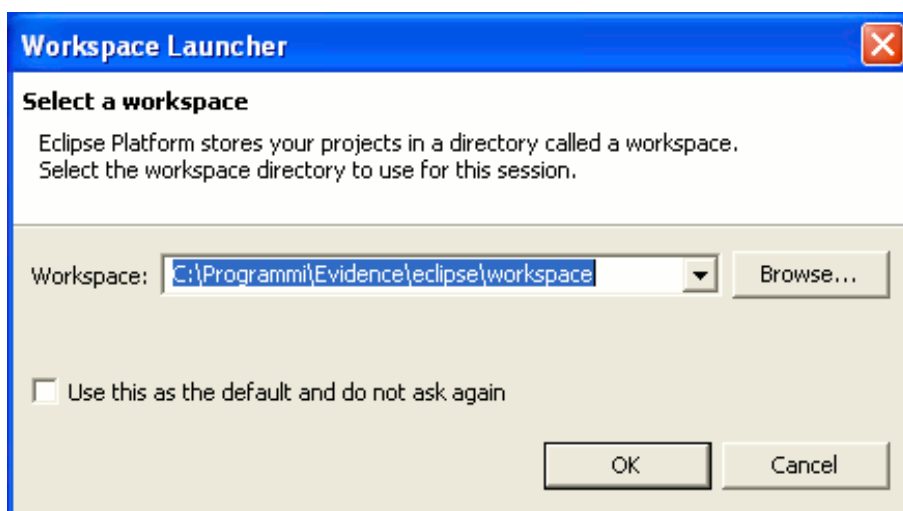


Figura 3.1: Questa figura mostra la dialog box che propone la scelta del workspace.

Attenzione: Il nome della cartella del workspace NON DEVE contenere spazi, altrimenti Erika Enterprise e RT-Druid potrebbero non funzionare correttamente.

3. A questo punto, apparirà la finestra di benvenuto di Eclipse, come in Figura 3.2.
4. Per essere in grado di compilare correttamente un applicativo, è necessario indicare correttamente all'interno di RT-Druid la posizione del compilatore Microchip C30 e dell'assemblatore Microchip ASM30 fornito con MPLABIDE.

3 Prima esecuzione e configurazione di RT-Druid

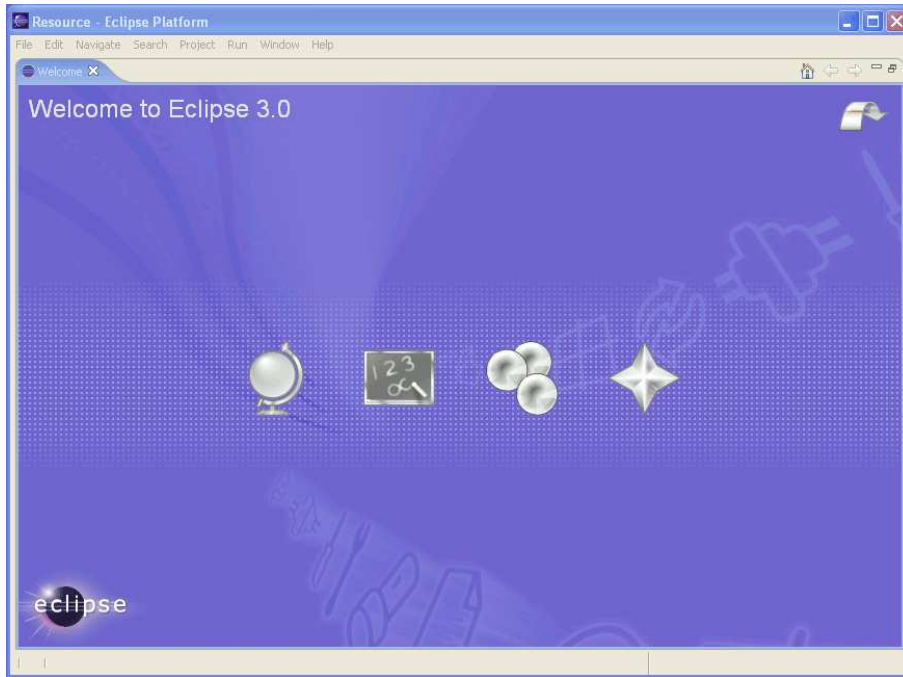


Figura 3.2: La finestra di benvenuto di Eclipse.

Dalla menu “Preference”, come mostrato in Figura 3.3 cercate la scheda “RT-Druid/Oil/PIC30 Configurator” come mostrato in Figura 3.4. La prima casella di testo, chiamata `Gcc path`, fa riferimento alla cartella di installazione del compilatore Microchip C30. La seconda casella di testo, chiamata `Asm path`, fa riferimento alla cartella di installazione dell’assemblatore ASM30 fornito con MPLAB IDE.

Attenzione: Le cartelle di installazione specificate nelle cartelle di testo in Figura 3.4 *non* includono la cartella `bin`!

Ovvero, `c:\Programmi\Microchip\MPLAB C30` è un nome di cartella corretto, mentre `c:\Programmi\Microchip\MPLAB C30\bin` non lo è.

Attenzione: La cartella di installazione dell’assemblatore fa riferimento all’assemblatore ASM30 fornito con MPLAB IDE e *non* all’assemblatore fornito con il compilatore C30. La ragione della scelta è che la cartella dell’assemblatore è utilizzata sia per richiamare l’assemblatore sia per copiare il file `crt0.s` nella directory corrente. Deve essere fornita la cartella di ASM30 in quanto la posizione del file `crt0.s` è diversa nelle due versioni di ASM30 fornite da Microchip (una in MPLAB IDE, una all’interno del compilatore C30).

3 Prima esecuzione e configurazione di RT-Druid

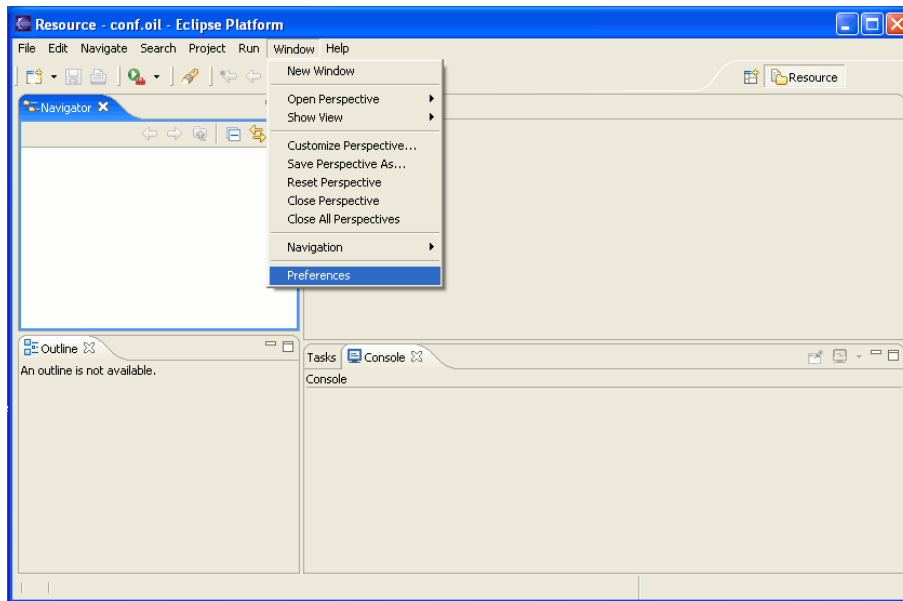


Figura 3.3: Andate al menu "Preference".

5. Prima di creare e compilare la vostra applicazione, deselectionate la voce "Build Automatically" nel "Project" menu, come visualizzato in Figura 3.5.

3 Prima esecuzione e configurazione di RT-Druid

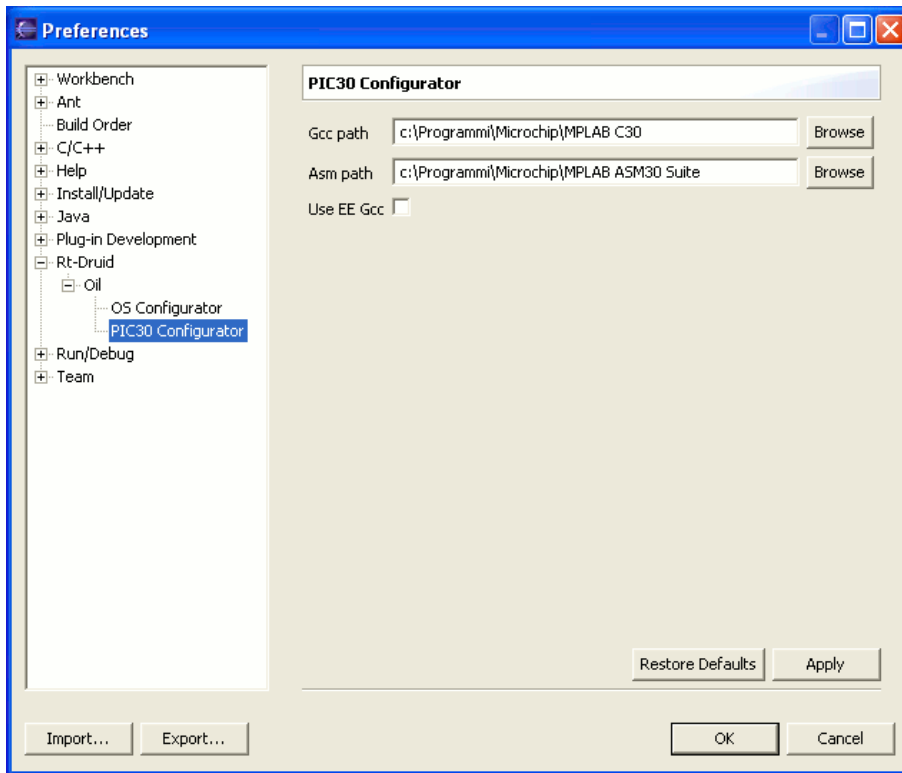


Figura 3.4: Specificate le cartelle di installazione per il compilatore C30 e per l'assemblatore ASM30 fornito con MPLABIDE.

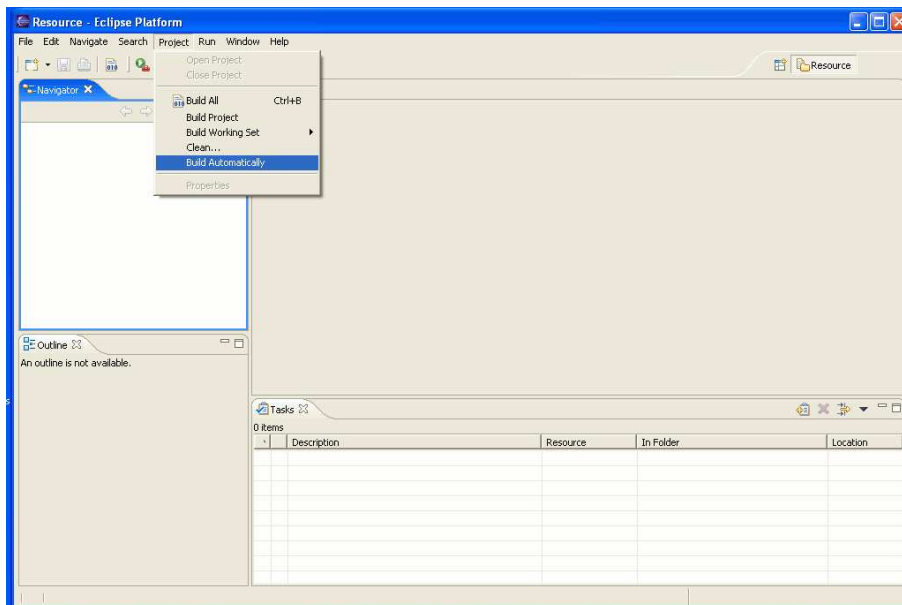


Figura 3.5: Deselezionate la voce “Build Automatically” nel “Project” menu.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

Siamo a questo punto pronti per compilare la vostra prima applicazione Erika Enterprise. Per fare ciò, seguite i seguenti passi:

1. Selezionate “New Project”, poi “RT-Druid Oil and c/c++ Project” dal “File menu”, come in Figura 4.1.

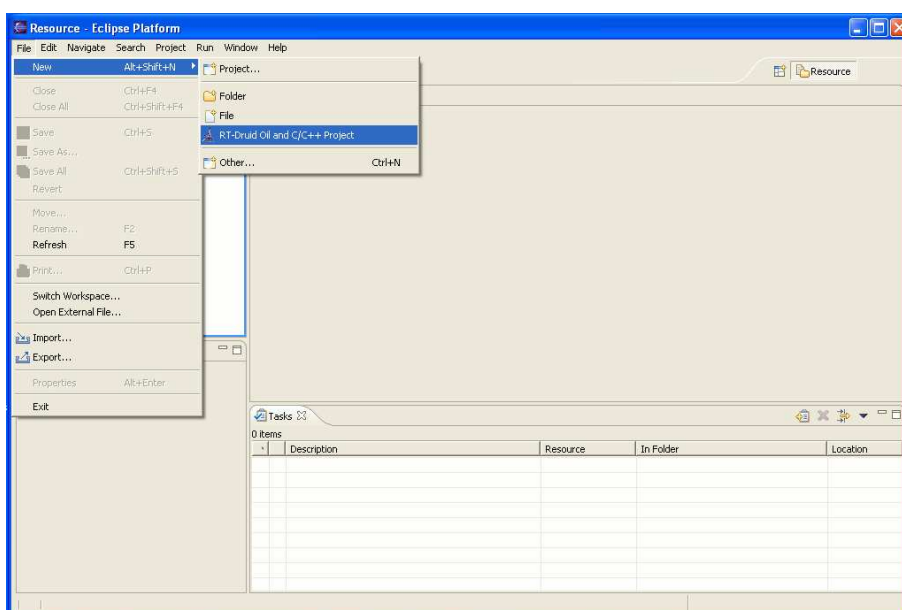


Figura 4.1: Selezionate “New project” dal “File” menu.

2. Apparirà una Dialog box. Selezionate un template per il nuovo progetto, come in Figura 4.2.
3. Premete il bottone “Next”.
4. Inserite il nome del nuovo progetto, ad esempio `myProject` (ovviamente potete utilizzare nomi diversi), come specificato in Figura 4.3. Premete il bottone “Finish”.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

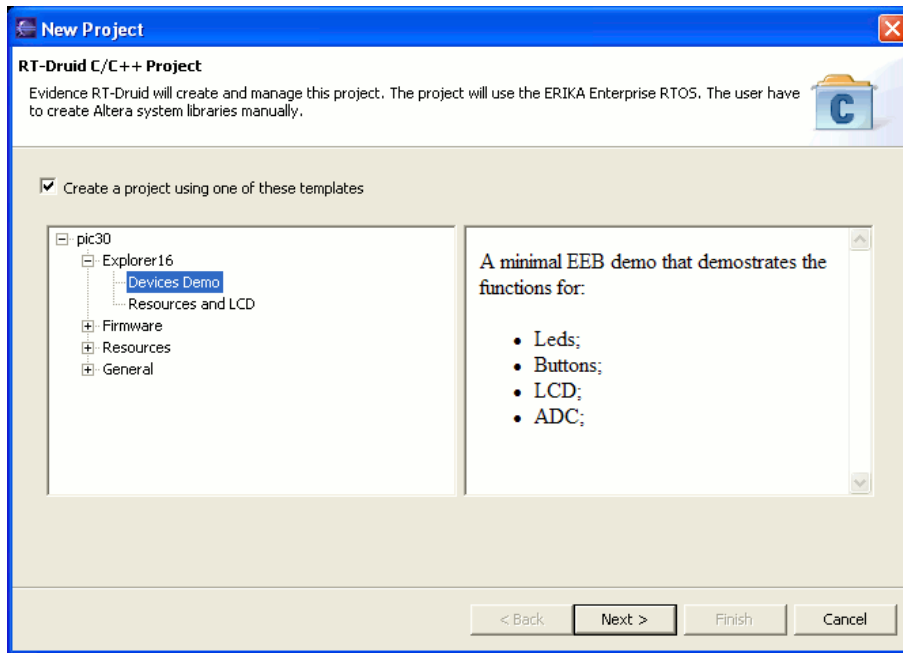


Figura 4.2: Selezionate “RT-Druid Oil and c/c++ Project”.

5. Siete a questo punto pronti per compilare l’applicazione. Premete il tasto destro del mouse sul nome del progetto nella navigation bar di Eclipse, e selezionate “Build Project”¹ (vedere Figura 4.4).
6. A questo punto, il processo di compilazione parte come visualizzato in Figura 4.5.

Nota Bene: Se appare l’errore visualizzato in Figura 4.6 (ovvero, `mymake_cygwin.bat` non può trovare un file), allora seguite attentamente le istruzioni all’ultimo punto del Capitolo 2.

7. Alla fine del processo di compilazione viene prodotto un file in formato ELF chiamato `pic30.elf` all’interno della directory `Debug` all’interno del progetto, come mostrato in Figura 4.7. Tale file è il programma eseguibile che può essere programmato sul device utilizzando MPLABIDE.
8. Il prossimo passo da eseguire riguarda la programmazione del file ELF generato durante la compilazione all’interno della scheda target. Per fare questo, occorre importare il file ELF all’interno dell’ambiente di sviluppo Microchip MPLAB IDE. Per fare questo, eseguite MPLAB IDE. Apparirà una finestra come quella mostrata in Figura 4.8.

¹ “Build Project” appare solo se l’opzione “Build Automatically” non è selezionata nel menu “Project”.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

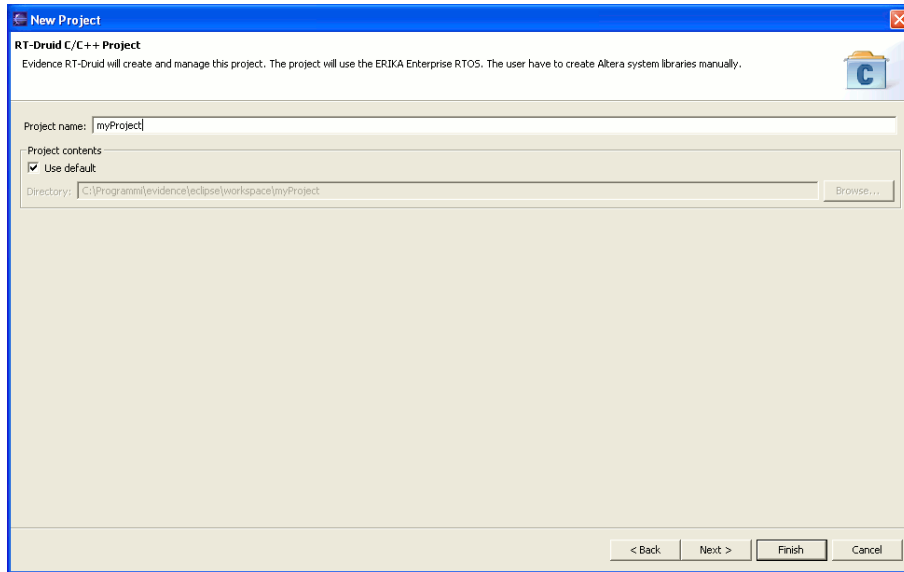


Figura 4.3: Specificate un nome per il nuovo progetto.

9. Scegliete “Import...” dal menu “File”, come mostrato in Figura 4.9.
10. Apparirà una dialog box. Selezionate il file `pic30.elf` prodotto dal processo di compilazione in Eclipse, come mostrato in Figura 4.10. Il file da importare si trova nella cartella del workspace che avete specificato all’inizio di questo documento in Figura 3.1. In questo esempio, il file è memorizzato nella directory `c:\Programmi\Evidence\ eclipse\workspace\ pic30_oo_mono\Debug`.
11. Una volta importato il file ELF all’interno di MPLAB IDE, siete pronti per effettuare una sessione di programmazione e debugging come normalmente avviene con il software fornito da Microchip. Occorre inoltre notare come non ci sia bisogno di creare un “progetto” di MPLAB IDE, in quanto l’intero processo di compilazione è gestito da Eclipse. Ad esempio, Figura 4.11 mostra la finestra “Disassembly Listing” e la finestra “Program Memory”. Notate come MPLAB IDE riconosca correttamente i simboli di debug del codice sorgente prodotto e compilato all’interno dell’ambiente Eclipse.
12. A questo punto è possibile far partire una sessione di debugging e programmazione per la vostra applicazione demo usando MPLAB IDE.

La Figura 4.12 mostra la scheda Microchip Explorer 16 con in esecuzione il demo presente nel template `pic30\explorer16\Devices Demo`. Il demo sfrutta i device a bordo della scheda Explorer 16 per visualizzare sul display LCD la temperatura ambiente letta dal sensore di temperatura.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

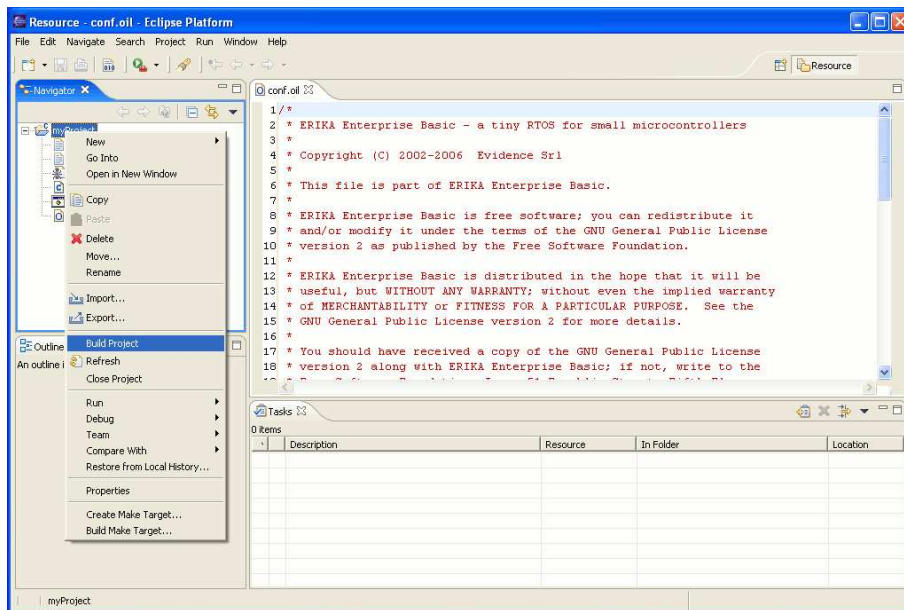


Figura 4.4: Adesso è possibile compilare il progetto.

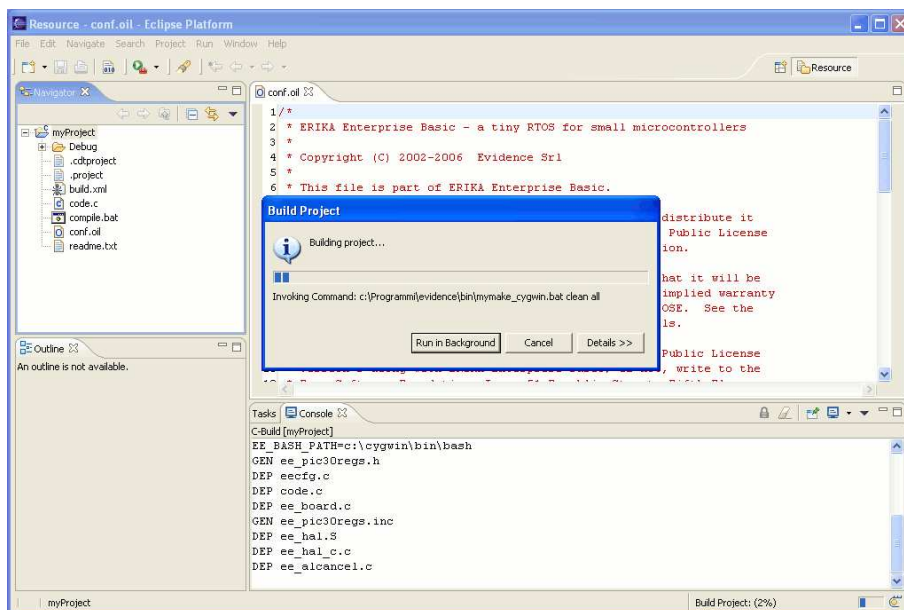


Figura 4.5: Il processo di compilazione.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

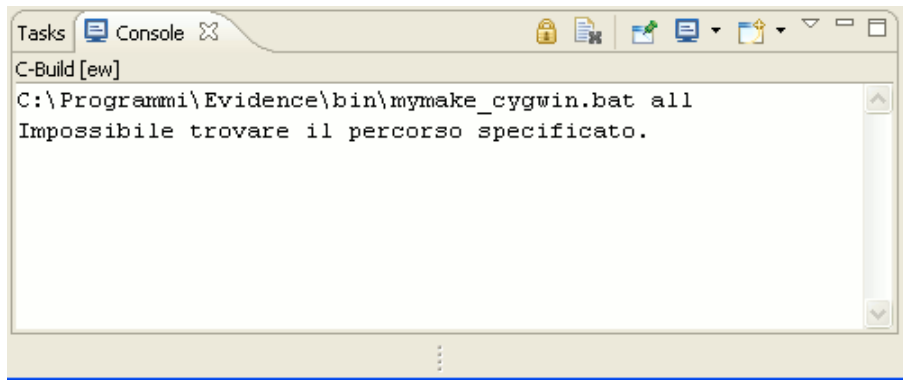


Figura 4.6: Un errore che si verifica su alcune macchine Windows. Per risolvere il problema, controllate il file `mymake_cygwin.bat` come indicato nell'ultimo punto del Capitolo 2.

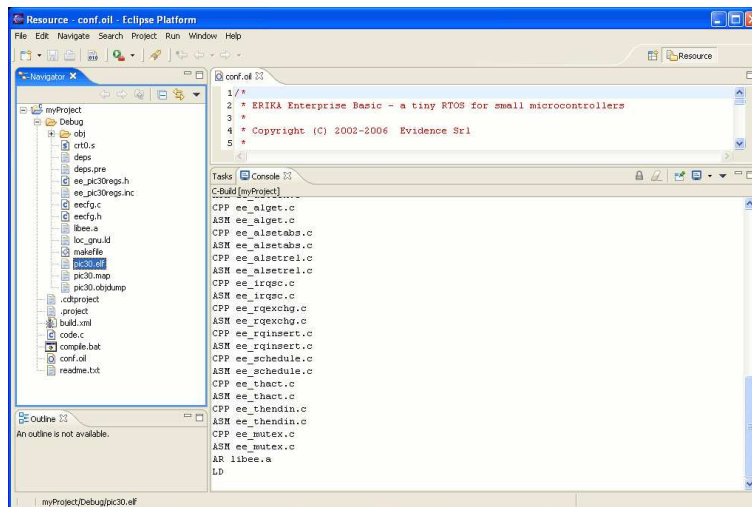


Figura 4.7: `pic30.elf` è il file di output che è pronto per essere programmato sulla scheda target.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

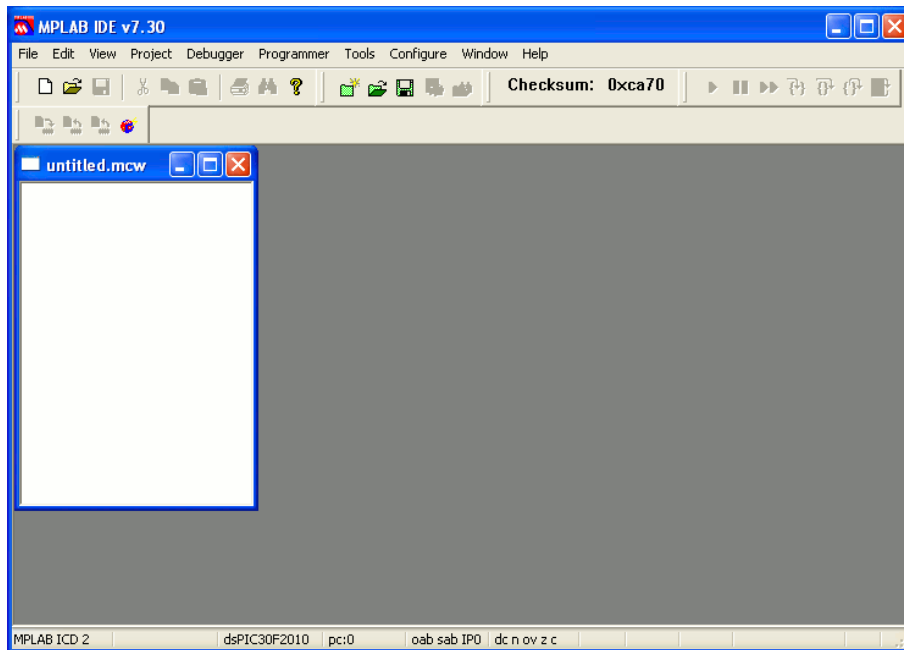


Figura 4.8: L'ambiente di sviluppo Microchip MPLAB IDE.

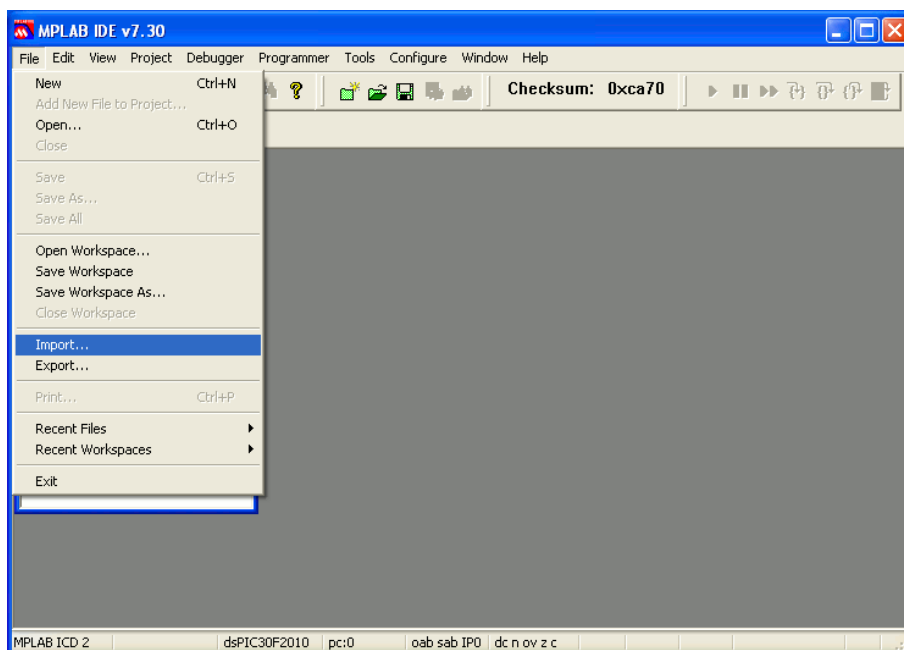


Figura 4.9: Scegliete "Import..." dal menu "File" per importare il file ELF prodotto da Eclipse.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

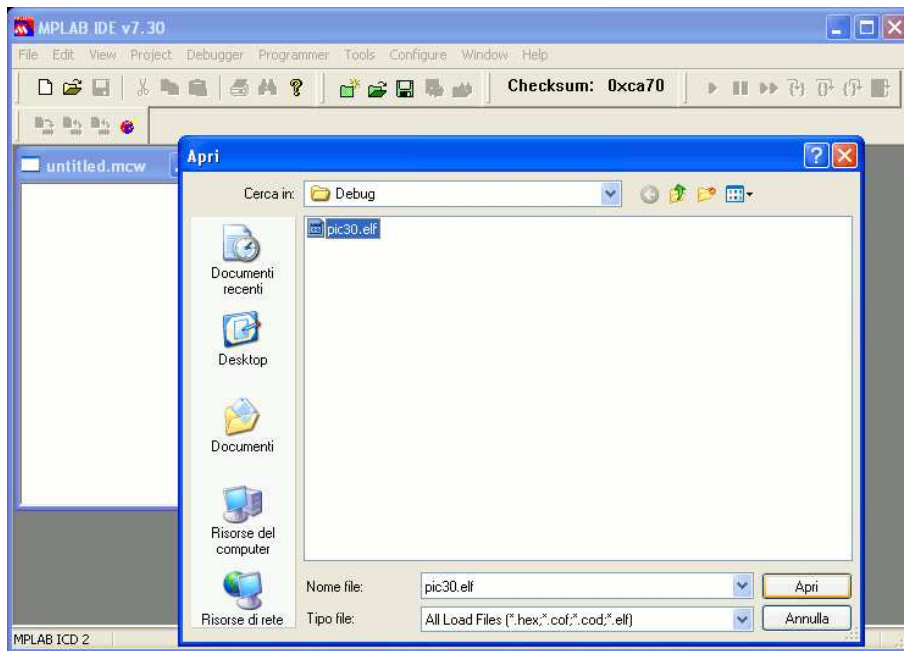


Figura 4.10: Selezionate il file ELF prodotto nel processo di compilazione.

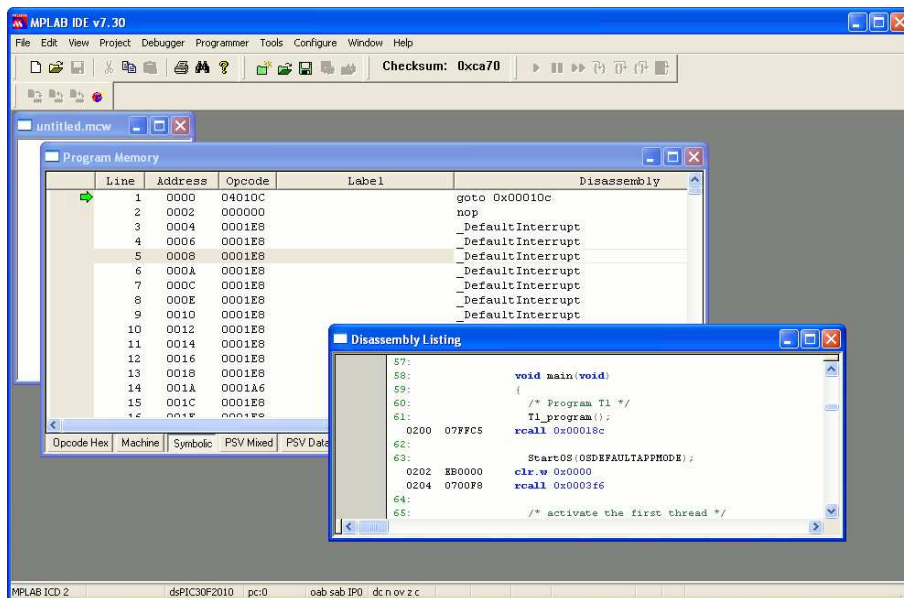


Figura 4.11: I simboli di debug sono correttamente riconosciuti da MPLAB IDE.

4 Compilare la vostra prima applicazione Erika Enterprise per dsPIC (R) DSC

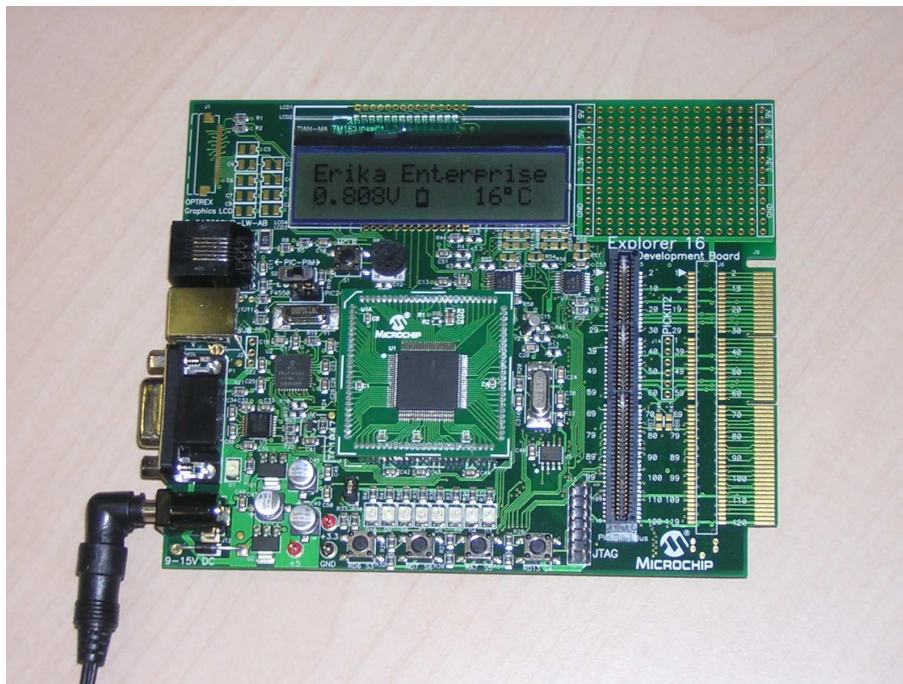


Figura 4.12: La scheda Explorer 16 in esecuzione con il programma demo.

5 Storia del documento

Version	Comment
1.1.6	Traduzione della corrispondente versione inglese del documento.

Bibliografia

- [1] Cygwin. The cygwin project. <http://www.cygwin.com>, 2005.