

---

# Kostengünstige Low-Power Prototyping Lösung

für MSP430-Embedded Systeme

---

## Inhaltsverzeichnis

<b>1 EINLEITUNG</b> .....	<b>2</b>
<b>2 AUFSETZEN DER MSP430 ENTWICKLUNGSUMGEBUNG MIT ECLIPSE</b> .....	<b>3</b>
2.1 CYGWIN.....	3
2.2 MSP GNU TOOLCHAIN .....	3
2.3 OLIMEX TREIBER .....	3
2.4 ECLIPSE .....	4
2.4.1 <i>Workspace Einstellungen</i> .....	5
2.4.2 <i>Projekt Build Einstellungen</i> .....	6
2.4.3 <i>Debug Einstellungen</i> .....	8
2.4.4 <i>Benötigte Externe Programme</i> .....	10
<b>3 UND SCHLIESSLICH</b> .....	<b>11</b>
<b>4 EINKAUFSLISTE</b> .....	<b>12</b>

### 1 Einleitung

Für Labor-Anwendungen, bei denen Embedded-Hardware ohne großen Entwicklungsaufwand angesprochen werden soll, bieten sich oft sogenannte Development Boards als erste Prototypenlösung an.

Die bulgarische Firma Olimex hat sich in den letzten Jahren auf diesem Gebiet einen Namen gemacht. Sie stellt für die MSP430 Prozessorfamilie eine Reihe von Evaluation- und Prototyping Boards bereit, die bei experimentierfreudigen Entwicklungsingenieuren kaum Wünsche offen lassen. Dazu gehört auch das MSP430-169LCD Starter-Kit. Mit dem MSP430F169 RISC Prozessor als Kern, einem pixelorientierten, kleinen LC Display und einem kleinen Joystick für die Menüsteuerung oder graphische Navigation ausgestattet, ist das Board gut geeignet für die Ansteuerung kleiner Messaufgaben mit beschränkter Ausgabe.

Der MSP430F169 ist Mitglied der MSP430 Signalprozessorfamilie von TI™. Die Familie zeichnet sich aus durch eine saubere 16-Bit Architektur, einen linearen Adressraum und ihre Low-Power Fähigkeiten. Bei einer Versorgungsspannung von 3.3V kann der Stromverbrauch des Prozessors im Standby-Modus auf ca. 0.1µA reduziert werden, so dass die Familie sich anbietet für batteriebetriebene Anwendungen. Das genannte Starter-Kit von Olimex kann daher allein durch die JTAG-Schnittstelle gespeist werden. Eine zusätzliche Spannungsversorgung ist ohne externe Beschaltung nicht notwendig.

Mit ein wenig Konfigurationsaufwand kann man zum Programmieren des Boards unter Windows XP® den GNU™ mspgcc Compiler für C und die inzwischen sehr komfortable Eclipse™ / CDT Entwicklungsumgebung nutzen. (Der einzige Wehrmutstropfen ist die sehr beschränkte Anzahl der Breakpoints im GNU-Debugger). Als Konfigurationsmanagement-Tool bietet sich, sofern keine anderen Vorgaben existieren, Subversion® mit dem Eclipse-Plugin Subclipse an. Die notwendigen Softwarewerkzeuge sind allesamt kostenlos im Internet erhältlich.

Die benötigten Hard- und Softwarekomponenten und deren Konfiguration werden im Folgenden beschrieben.

## 2 Aufsetzen der MSP430 Entwicklungsumgebung mit Eclipse

### 2.1 Cygwin

Um die GNU-Toolchain nutzen zu können, müssen bestimmte UNIX®-basierte Werkzeuge, wie z.B. das make-Programm installiert sein. Es gibt zwei Wege, dies bereitzustellen:

1. Das GPL-lizenzierte Cygwin® Paket ( [www.cygwin.com](http://www.cygwin.com) )
2. Das ebenfalls frei lizenzierte MinGW Paket (<http://www.mingw.org/> )

Beide Pakete sind für den hier beschriebenen Zweck geeignet. Im Folgenden wird von einer Cygwin Installation ausgegangen.

Zunächst muss das Cygwin Release 1.5.23-2 oder höher installiert werden. Wichtig: Beim Installieren müssen die Standardeinstellungen für die Development Tools geändert werden, diese müssen komplett installiert werden. Die Installation erfolgt am besten in das Defaultverzeichnis [C:\CYGWIN](#).

### 2.2 MSP GNU Toolchain

Nun kann man die msp-gcc Tools herunterladen und installieren. Die Installation erfolgt z.B. in das Verzeichnis C:\Programme\mspgcc.

Nun gibt es sowohl im mspgcc-Verzeichnis als auch im cygwin Verzeichnis die Datei cygwin1.dll. Das geht nicht. Hoffentlich ist die Datei im Cygwin Verzeichnis jünger. Dann die cygwin1.dll im mspgcc Verzeichnis einfach löschen. Das reine Umbenennen (z.B. in cygwin1.dll.old) nützt übrigens nichts.

Bei Verwendung eines parallelen JTAG-Adapters muss das Programm Giveio einmalig gestartet werden, um den JTAG Adapter an den Parallel-Port anschließen zu können. Wenn benötigt, dann als Kommandozeileneingabe

```
C:\Programme\mspgcc\giveio\install_giveio.bat
```

aufrufen.

Für die USB-Variante muss dies nicht durchgeführt werden.

### 2.3 Olimex Treiber

Nur bei Verwendung eines USB JTAG Adapters müssen jetzt die Olimex-USB Treiber heruntergeladen und entpackt werden. Bei Verwendung des Paralleladapters kann dieser Schritt entfallen.

Zur Installation den JTAG Adapter an einen USB-Port anschließen. Windows möchte dann einen Treiber installieren. Als Quelle das eben ausgepackte Verzeichnis angeben.

Aus diesem Verzeichnis nun die Dateien SiUSBXp.dll, FTD2XX.dll und MSP430.dll in das Verzeichnis C:\Programme\mspgcc\bin (oder entsprechend) kopieren, dabei überschreibt man die von mspgcc mitgelieferte MSP430.dll, das ist in Ordnung so. Olimex HIL.dll NICHT mit kopieren.

Nun muss für den USB-JTAG Adapter ein Update des TI-FET gemacht werden: Die Kommandozeileneingabe starten und dann

```
msp430-gdbproxy -port=3333 msp430 --update-usb-fet TIUSB
```

eingeben. Abwarten.

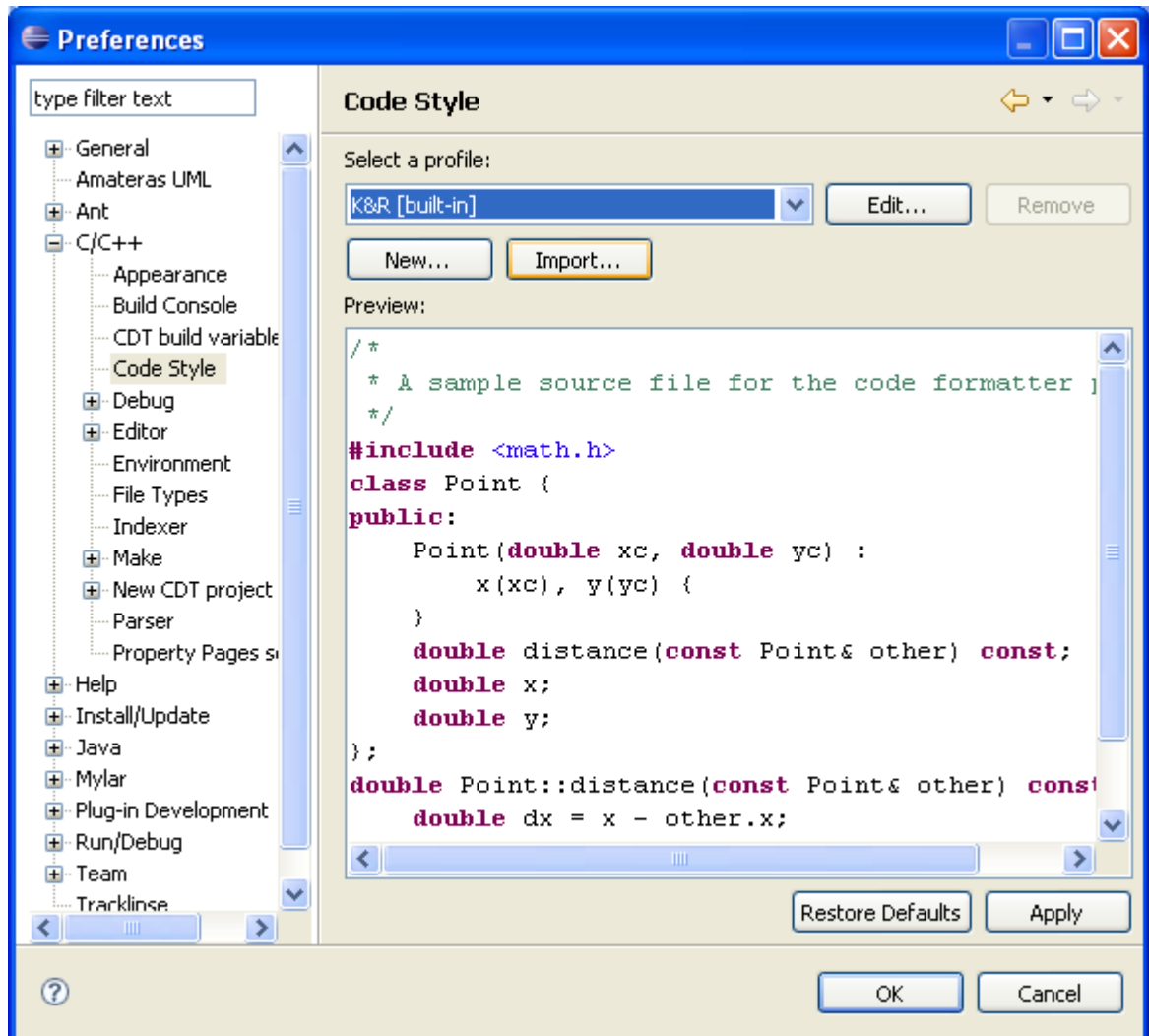
## 2.4 Eclipse

- Eclipse Release installieren: Eclipse herunterladen und z.B. nach C:\Programme\eclipse entpacken.
- Dann alle Plugins in der in der Tabelle angegebenen Reihenfolge installieren: Alle Eclipse Plugins können entpackt und dann über den Update-Mechanismus „Help – Softwareupdates - Find and Install – search for new features“ ( Für lokale Pakete „Local Update Site“, bei direktem Download „Remote Update Site“) eingespielt werden. **Ausnahme:** Das mspgcc-Plugin wird entpackt und die entpackten Dateien manuell in das Eclipse\plugins Verzeichnis (z.B. C:\Programme\eclipse\plugins) kopiert.

Achtung: Das neue Eclipse Europa Release 3.3 und das freigegebene CDT-Plugin Version 4.0.0 hat ein Format, das nicht mit dem mspgcc-Plugin kompatibel ist. Daher müssen bis auf weiteres die Ständer der Eclipse-Version 3.2 oder die Prerelease-Stände der Eclipse 3.3 Version und der Plug-ins verwendet werden (siehe Einkaufsliste weiter unten).

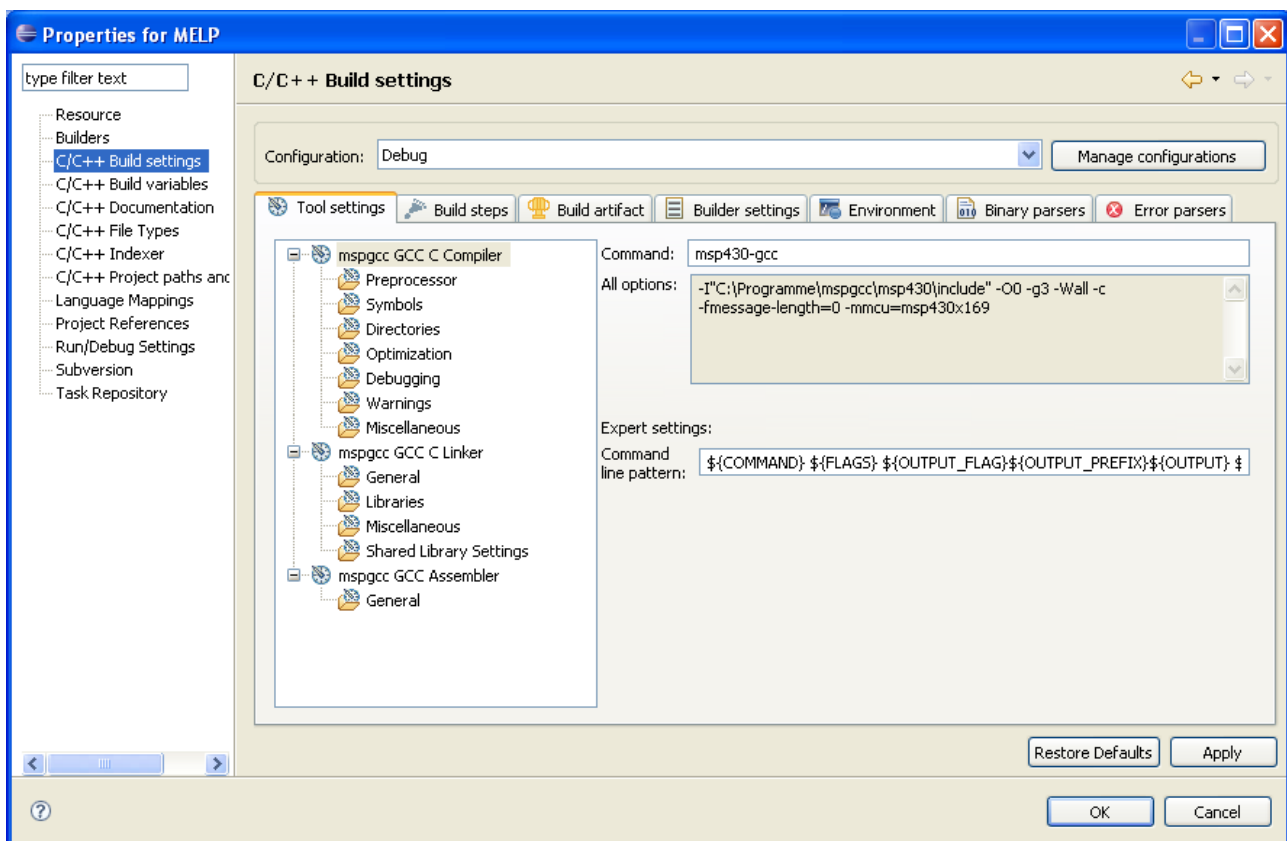
### 2.4.1 Workspace Einstellungen

- Subclipse konfigurieren:  
Unter Window – Preferences – Team-SVN JavaHL als SVN Adapter auswählen.
- Unter Window – Preferences-C/C++ auf Wunsch einen CodeStyle adaptieren oder importieren

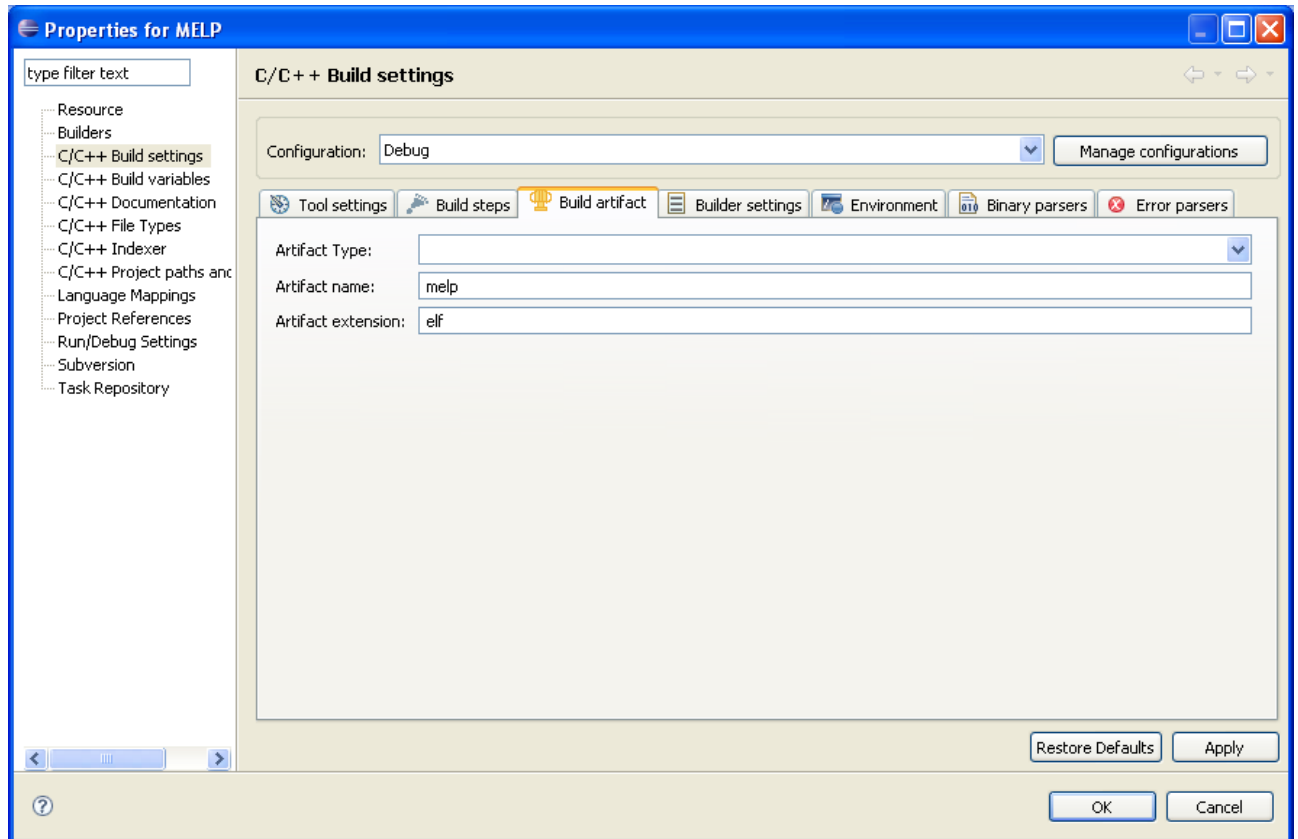


## 2.4.2 Projekt Build Einstellungen

- Projekt erstellen oder aus Subversion laden. „Build Automatically“ ausschalten, das ist für C-Projekte nicht geeignet.  
Binary Parser (Project-Properties): Nur ELF-Parser aktivieren.
- Projekt - Properties wählen. C/C++ Build Settings:
  - Reiter Tool-Settings:
    - msp430-gcc als Compiler und auch Linker Command angeben. Assembler-Command ist msp430-as
    - Directories: `C:\Programme\mspgcc\msp430\include` als externen Pfad angeben
    - Compiler-Miscellaneous, Linker – General, Assembler - General: CPU Type setzen, in diesem Fall msp430x169



- Reiter Build Artifact
  - Artifact Namen angeben (das ist der Name des zu erzeugenden Binaries), z.B. melp
  - Artifact Extension ist elf



- Reiter „Binary Parser“: Nur den ELF-Parser anklicken, alle anderen sind deaktiviert.

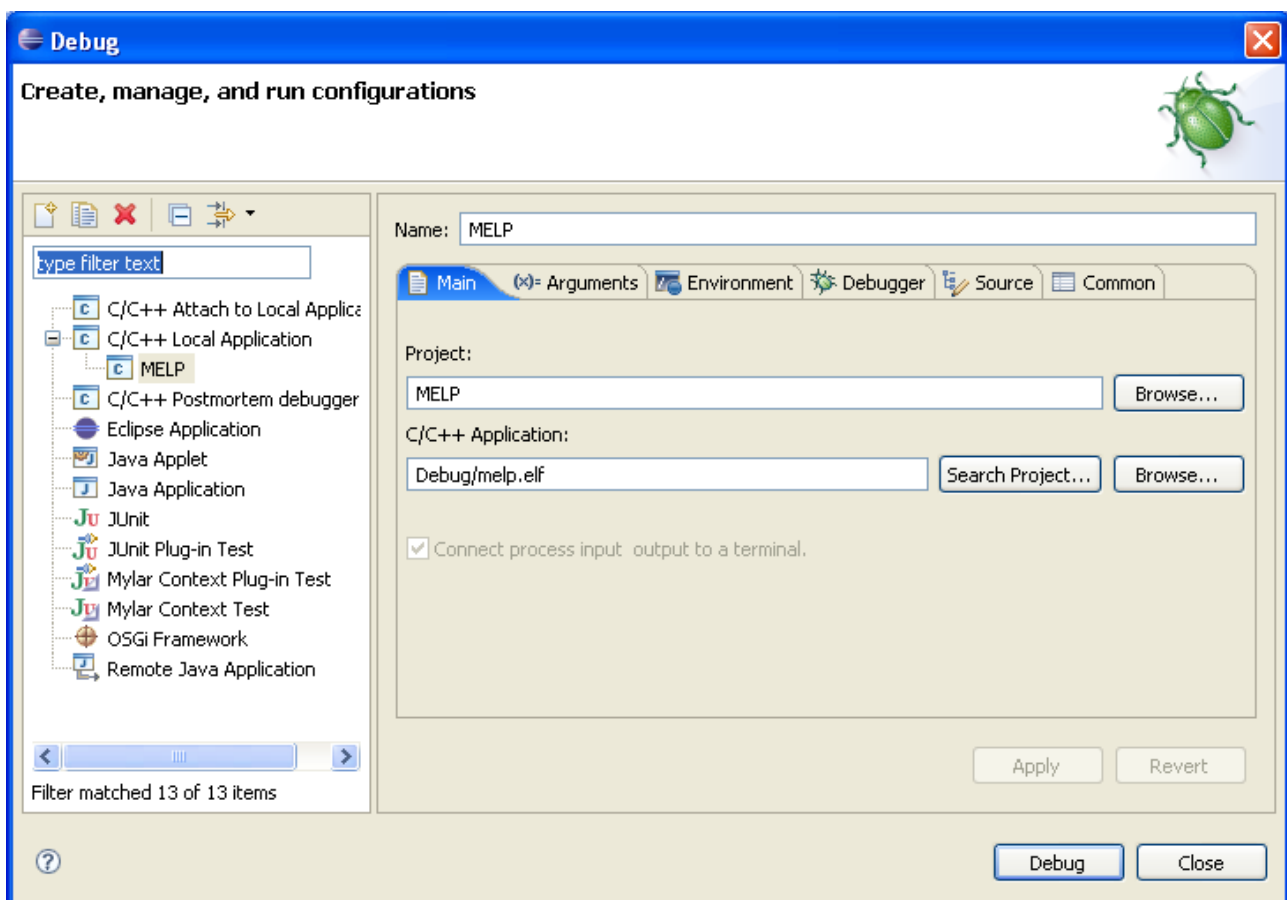
### 2.4.3 Debug Einstellungen

Im Root-Verzeichnis des Projektes eine Textdatei gdb-target.ini anlegen. Folgende Zeilen hinein kopieren:

```
set remoteaddresssize 64
set remotetimeout 999999
target remote localhost:3333
set download-write-size 2000
set remote memory-write-packet-size 2000
set remote memory-write-packet-size fixed
set remote memory-read-packet-size 2000
set remote memory-read-packet-size fixed
monitor erase all
load Debug\melp.elf
```

- Hierbei muss Debug\melp.elf durch das jeweilige Target-Executable ersetzt werden.
- Neue Debug-Konfigurationen erstellen (Local Application!), und die unten gezeigten Einstellungen vornehmen.

Main-Reiter:

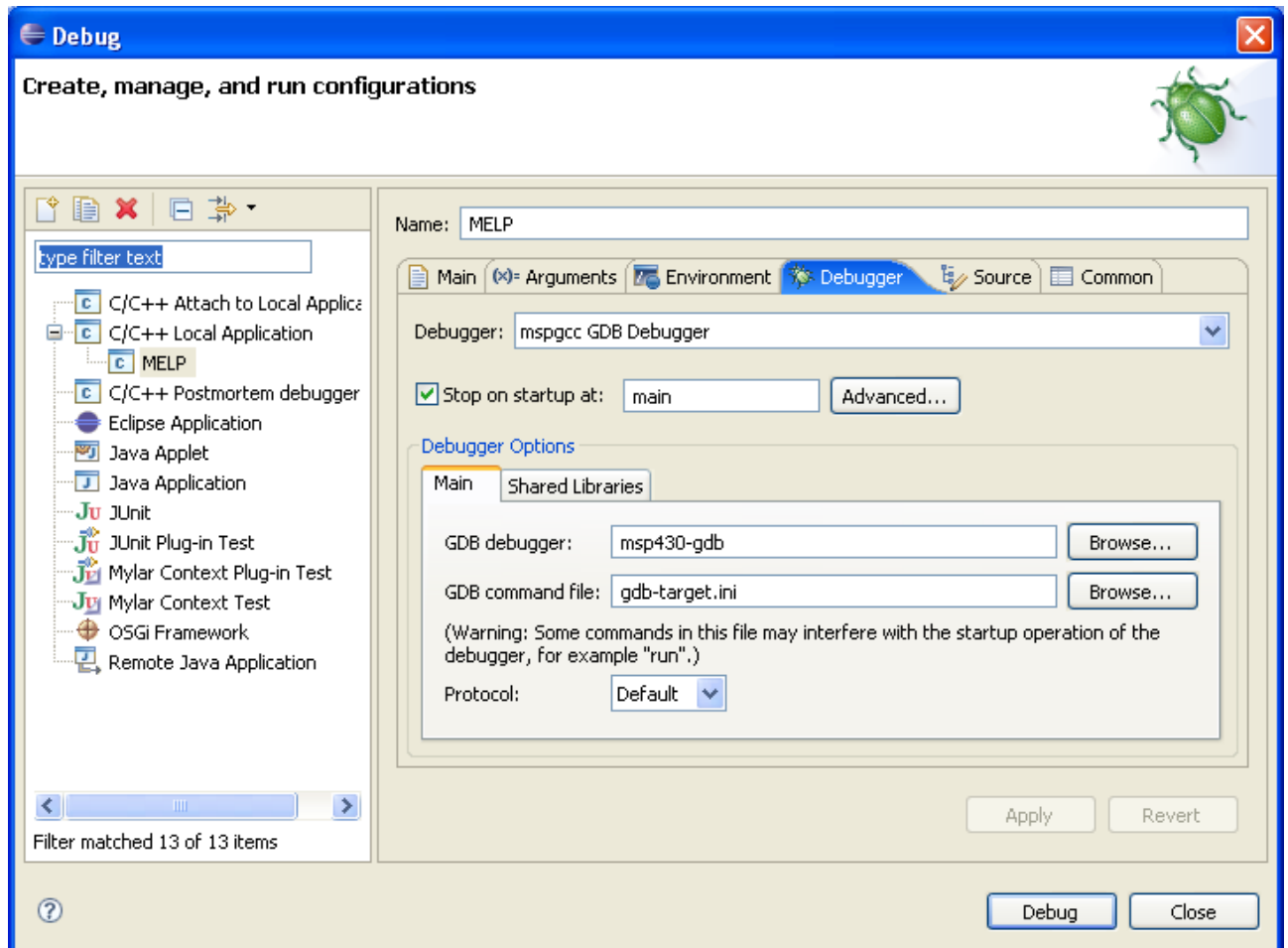


Name: <Project file name>

Project: <Project name>

C/CC++ Application: <Muss der Name des oben angegebenen ELF files sein, so dass die Debug-Info gefunden wird.>

Debugger-Reiter:



Debugger: mspgcc GDB Debugger

GDB debugger: "msp430-gdb"

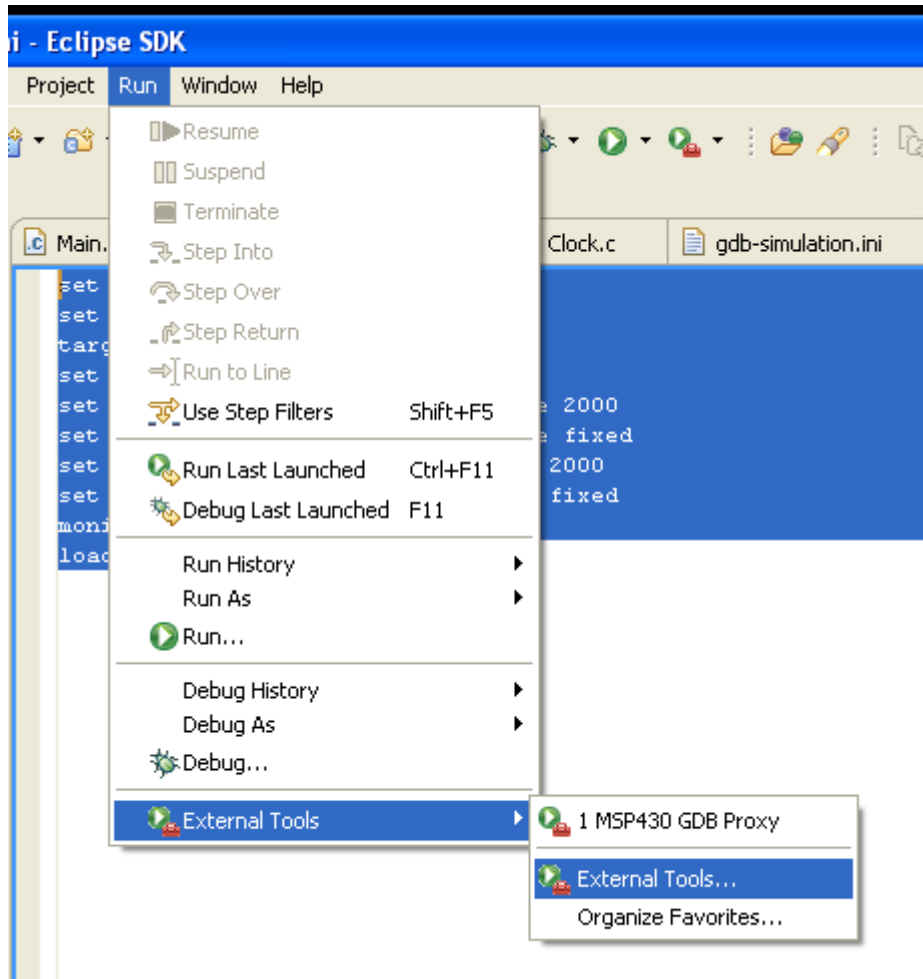
GDB command File: gdb-target.ini

Protocol: default

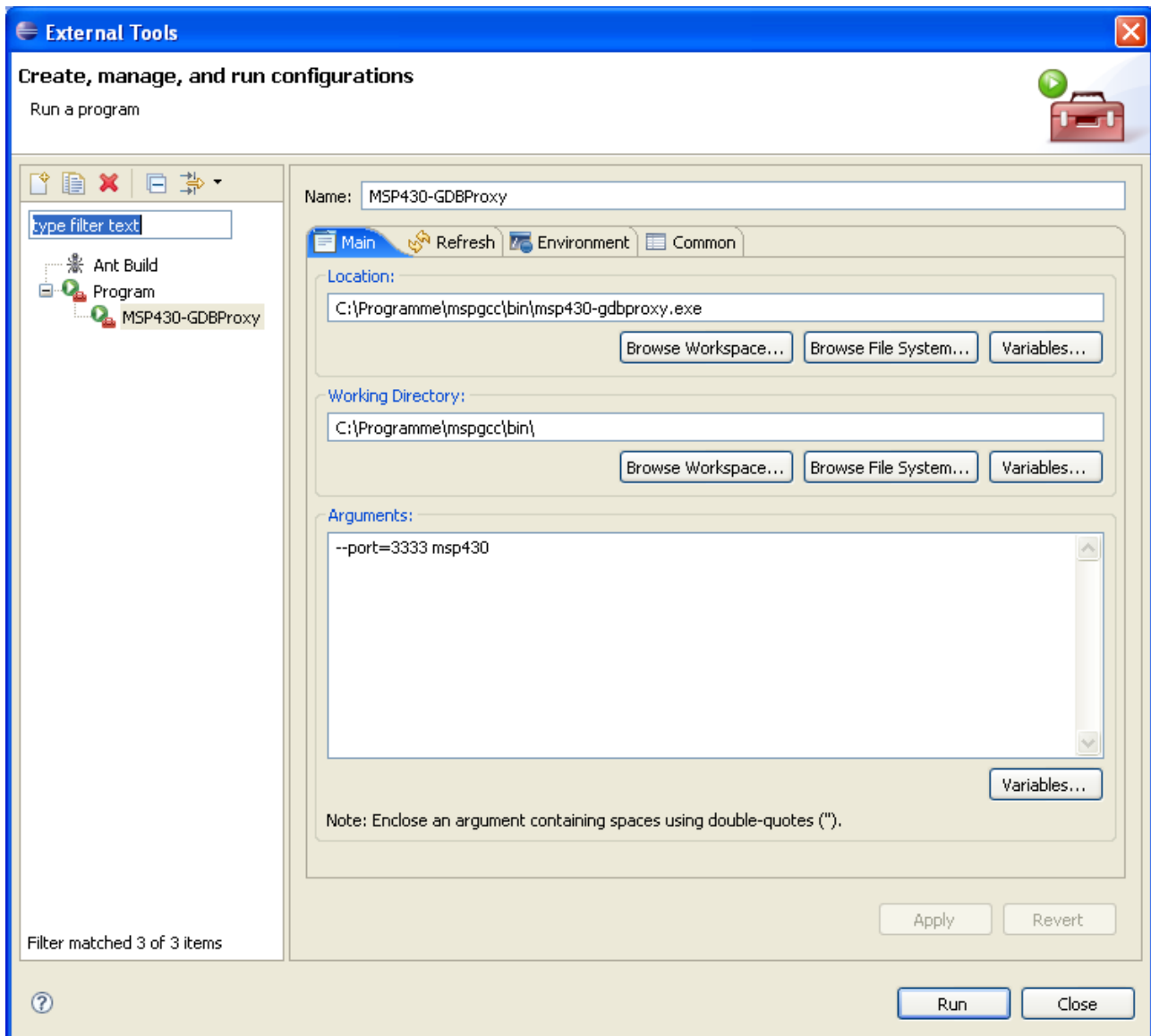
Common-Reiter: Auf Wunsch zum Favourites Menü hinzufügen (empfohlen)

## 2.4.4 Benötigte Externe Programme

Externes Programm hinzufügen:



Hier muss der GDB-Proxy Aufruf angelegt werden. (run-external tools-New program). Dieser Aufruf muss nach Eclipse-Start jeweils vor der ersten Debugsession 1x gestartet werden.



Arguments für USB-JTAG: `--port=3333 msp430 TIUSB`

Arguments für Parallel-JTAG: `--port=3333 msp430`

### 3 Und schließlich...

... kann das Codieren beginnen.

Hilfreich sind natürlich alle Datenblätter für den Prozessor.

Die Dokumentation der Peripherie von Olimex beschränkt sich auf einen Schaltplan, aber es gibt ein kleines Olimex-Demoprogramm zur Ansteuerung des Boards. Der Code ist für einen anderen Compiler geschrieben worden, so dass kleinere Anpassungen, z.B. zur Ansteuerung der Ports mit dem mspgcc nötig sind. Eine gute Fingerübung, um sich mit dem Compiler und dem Prozessor vertraut zu machen.

## 4 Einkaufsliste

Hier finden Sie eine gute Kombination der Tool-Versionen und ihre Bezugsquellen. Obacht! Wie oben schon beschrieben, haben das neue Eclipse Europa Release 3.3 mit dem freigegebenen CDT-Plugin Version 4.0.0 ein Format, das nicht mit dem mspgcc-Plugin kompatibel ist. Daher müssen bis auf weiteres die Ständer der Eclipse-Version 3.2 oder die Prerelease-Stände der Eclipse 3.3 Version und der Plug-ins verwendet werden.

Hardware	Netto-Preis	Quelle
Olimex MSP430-169LCD Starter Kit	ca. 55 Euro	z.B. <ul style="list-style-type: none"> <li>Sander Elektronik <a href="http://www.sander-electronic.de">www.sander-electronic.de</a></li> <li>Elektronik-Laden <a href="http://elmicro.com/de/index.html">http://elmicro.com/de/index.html</a></li> <li>oder andere</li> </ul>
Olimex MSP430-JTAG Adapter	In verschiedenen Varianten erhältlich (alle Varianten sind mit dieser Tool-Chain kompatibel)  Je nach Variante ca. 20-75 €	z.B. <ul style="list-style-type: none"> <li>Sander Elektronik <a href="http://www.sander-electronic.de">www.sander-electronic.de</a></li> <li>Elektronik-Laden <a href="http://elmicro.com/de/index.html">http://elmicro.com/de/index.html</a></li> <li>oder andere</li> </ul>
Software-Tool	Version	Quelle
cygwin	Cygwin-1.5.23-2 oder höher	<a href="http://www.cygwin.com">www.cygwin.com</a>
Olimex USB-Treiber	OLIMEX MSP430 USB drivers 1-012.zip	<a href="http://www.olimex.com/">http://www.olimex.com/</a>
msp430-gcc	mspgcc-20061126.exe	<a href="http://mspgcc.sourceforge.net/">http://mspgcc.sourceforge.net/</a>
Eclipse	Eclipse 3.3M6	<a href="http://download.eclipse.org/eclipse/downloads/">http://download.eclipse.org/eclipse/downloads/</a>
<b>Eclipse Plugins:</b>		
<b>Für MSP430-C Entwicklung</b> <ul style="list-style-type: none"> <li>CDT</li> <li>MSPGCC</li> </ul>	cdt-4.0.0-I200704120500-win32.x86.zip  net.sf.mspgcc.cdt 0.02	<a href="http://download.eclipse.org/tools/cdt/builds/4.0.0/">http://download.eclipse.org/tools/cdt/builds/4.0.0/</a>  <a href="http://homepage.hispeed.ch/py430/mspgcc/">http://homepage.hispeed.ch/py430/mspgcc/</a>

<b>Subversion:</b>		
<b>Für Subversion Anbindung</b>		
<ul style="list-style-type: none"> <li>● Subversion</li> <li>● Subclipse</li> </ul>	Subversion 1.4.4 Subclipse 1.2.0	<a href="http://subversion.tigris.org/project_packages.html">http://subversion.tigris.org/project_packages.html</a> <a href="http://subclipse.tigris.org/update_1.2.x">http://subclipse.tigris.org/update_1.2.x</a>
<b>Demo-Software</b>		
<b>Demo-Code von Olimex</b>		<a href="http://www.olimex.com/">http://www.olimex.com/</a>

Alle verwendeten Namen dienen ausschließlich der Beschreibung und sind teilweise eingetragene Warenzeichen. Alle verwendeten fremden Warenzeichen sind Eigentum der jeweiligen Unternehmen.  
 Eclipse™ ist ein Warenzeichen der Eclipse Foundation, Inc. TI™ ist ein Warenzeichen der Texas Instruments Incorporated. Subversion® ist ein eingetragenes Warenzeichen der Subversion Corporation. Cygwin® ist ein eingetragenes Warenzeichen der Red Hat, Inc. GNU™ ist ein Warenzeichen der Free Software Foundation. Windows® XP ist ein eingetragenes Warenzeichen der Microsoft Corporation. UNIX® ist ein eingetragenes Warenzeichen der Open Group.