

# MPE xARM/Cortex

---

## First steps – Freescale Kinetis K60 Tower



---

# First steps – Kinetis K60 Tower

---

## Book name

Manual revision 7.10

Date 25 November 2011

## Software

Software version 7.10

<b>Package</b>	<b>Number:</b>	XARMCTX/71
----------------	----------------	------------

## For technical support

Please contact your supplier

## For further information

MicroProcessor Engineering Limited  
133 Hill Lane, Southampton  
SO15 5AF, UK  
Tel: +44 (0)23 8063 1441  
Fax: +44 (0)23 8033 9691

e-mail: [mpe@mpeforth.com](mailto:mpe@mpeforth.com)  
[tech-support@mpeforth.com](mailto:tech-support@mpeforth.com)

web: [www.mpeforth.com](http://www.mpeforth.com)

---

## **Acknowledgements**

MPE would like to thank the following people for their involvement in the production of this product:

Winfried Clemens

---

# Contents

---

<b>MPE xARM/Cortex</b>	<b>2</b>
<b>First steps – Kinetis K60 Tower</b>	<b>3</b>
Book name.....	3
Software.....	3
For technical support.....	3
For further information.....	3
Acknowledgements.....	4
<b>Contents</b>	<b>v</b>
<b>1 Setting up</b>	<b>1</b>
AIDE tools.....	1
AIDE PowerTerm.....	2
What you did.....	3
<b>2 Compiling and testing</b>	<b>5</b>
Compile and download.....	5
Adding application code.....	6

---

# 1

## Setting up

---

We assume that you are using a supported board. For the K60 family, this is the Freescale Tower system with a TWR-K60N512 CPU card and a TWR-SER board.

Find the directory containing the MPE control file

K60twr.ctl

and note it down. It should be in

<xArmCortex>/Cortex/Hardware/Kinetis

### AIDE tools

AIDE is a program that provides manages external tools and provides a terminal emulator. Make sure that everything is set up correctly.

- Run AIDE
- In AIDE, use  
IDE -> Cofigure Edit/Locate  
to set up your favourite editor.
- If the K60 is not on AIDE's toolbar, add a new tool using:  
IDE -> External Tools

To use the MPE control file

K60twr.ctl

set the start directory to the containing K60twr.ctl. You will need quotation marks if there are spaces in the path. Some people prefer to uninstall and reinstall to a directory with no spaces in the pathname. The set up in my development system is:

BMP file: c:\buildkit.dev\software\Aide\CM4.bmp

Compiler: c:\buildkit.dev\software\compiler\xArmCortexDev.exe

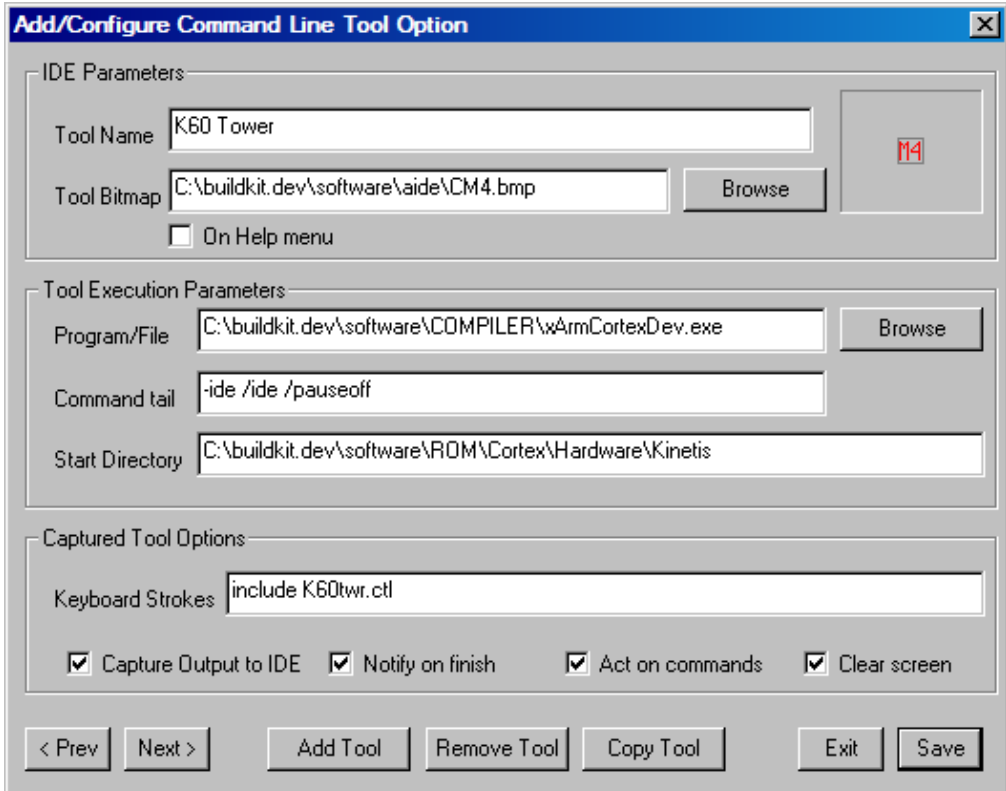
Command tail: -ide /ide /pauseoff

Start dir: C:\buildkit.dev\software\ROM\Cortex\Hardware\Kinetis

Keybd Strokes: include K60twr.ctl

All four checkboxes in the "Captured Tool Options" box are checked. See the following picture for my complete settings. If you are using the Stamp edition of the compiler, change the compiler line:

Compiler: c:\buildkit.dev\software\compiler\xArmCortexStamp.exe



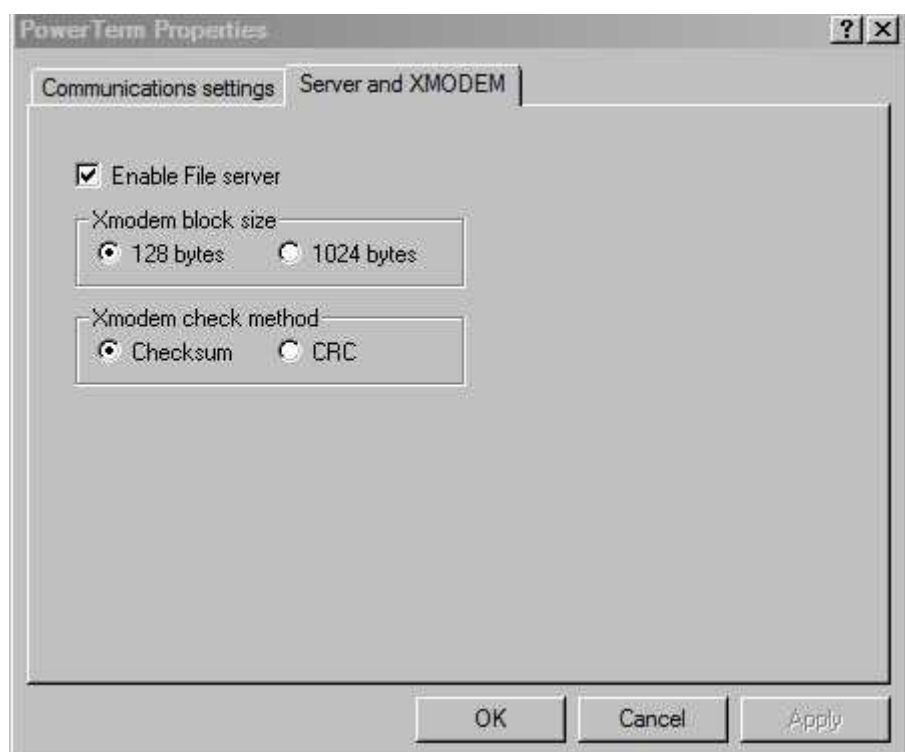
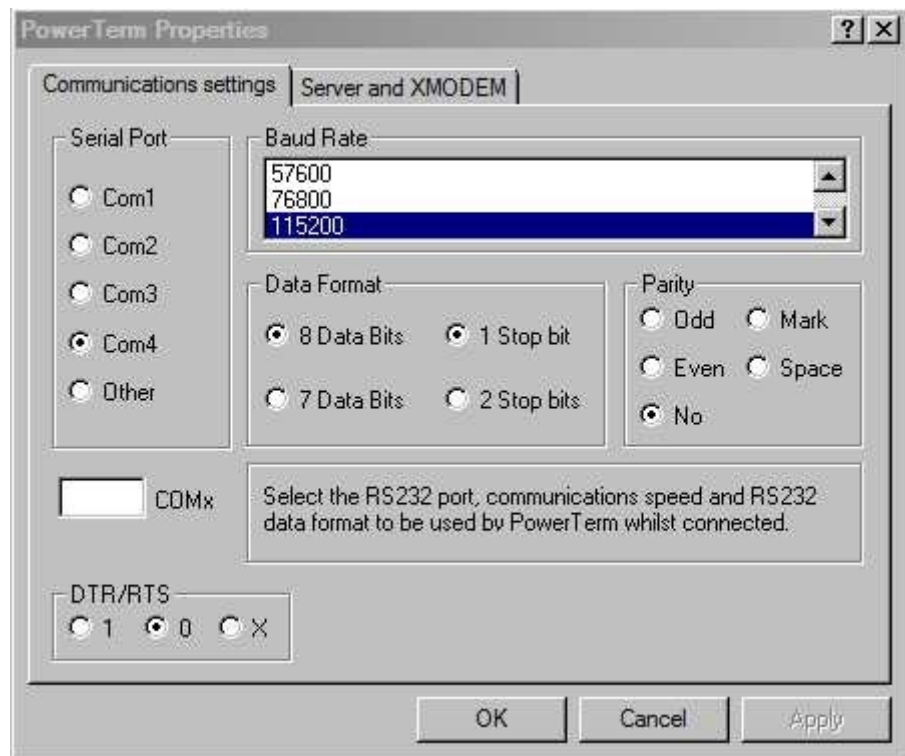
The dialog box is titled "Add/Configure Command Line Tool Option". It is divided into three main sections: "IDE Parameters", "Tool Execution Parameters", and "Captured Tool Options".

- IDE Parameters:** Contains a "Tool Name" field with the text "K60 Tower", a "Tool Bitmap" field with the path "C:\buildkit.dev\software\aide\CM4.bmp" and a "Browse" button, and an unchecked checkbox labeled "On Help menu".
- Tool Execution Parameters:** Contains a "Program/File" field with the path "C:\buildkit.dev\software\COMPILER\wArmCortexDev.exe" and a "Browse" button, a "Command tail" field with the text "-ide /ide /pauseoff", and a "Start Directory" field with the path "C:\buildkit.dev\software\ROM\Cortex\Hardware\Kinetis".
- Captured Tool Options:** Contains a "Keyboard Strokes" field with the text "include K60twr.ctl" and four checked checkboxes: "Capture Output to IDE", "Notify on finish", "Act on commands", and "Clear screen".

At the bottom of the dialog, there are six buttons: "< Prev", "Next >", "Add Tool", "Remove Tool", "Copy Tool", "Exit", and "Save".

## AIDE PowerTerm

Configure PowerTerm for 115200 baud, N81, COMx using the Properties button that is second from the right on the PowerTerm toolbar. Switch to the "Server and XMODEM" page, and check "Enable File Server" with 128 byte and Checksum Xmodem selected.



## What you did

You now have the Forth cross compiler tool set up. There will be a corresponding button on the toolbar.

When you click the compiler button the file

K60twr.ctf

is included by the cross-compiler. This file is a control file. It tells the cross compiler how to compile the target and what to compile. Because the tool capture checkboxes were checked when the tool was set up, the compiler runs in the “Tool Capture” window.



## 2 Compiling and testing

### Compile and download

You now have a tool set up. There will be a corresponding button on the toolbar. When you click it, the file

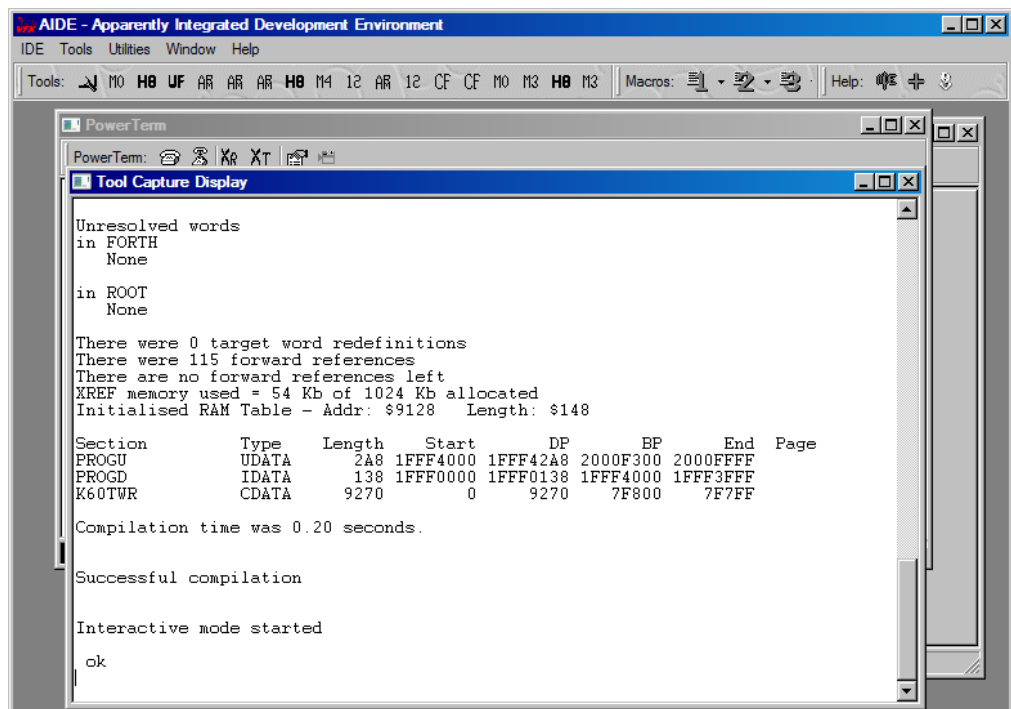
K60twr.ctf

is included by the cross-compiler. This file is a control file. It tells the cross compiler how to compile the target and what to compile. When the compiler has finished, there will be two output files in your Kinetis folder

K60TWR.img

K60TWR.hex

The cross compiler always produces binary image files with a ".img" extension. The control file tells the compiler also to produce an Intel Hex file with a ".hex" extension. The binary files are used by MPE tools, the hex files by some external tools for Flash programming.



AIDE's Tool Capture window should be visible - the compiler is still alive so that you can disassemble words and use the cross reference tools such as XREF and LOCATE. Make sure that PowerTerm is also visible, but is disconnected.

You now need to program one of the output files into the K60 Flash. Production versions of the MPE compiler will have support for various P&E BDM units, including the one integrated into the K60 board. If you downloaded the evaluation version of CodeWarrior, you can also use this.

There is a full description of the required jumper settings for the board in [Cortex/Manual/K60Code.pdf](#).

Once the board is programmed and powered, press the PowerTerm connect button (the telephone) and then reset the board. You should now get the MPE PowerForth sign on message. This is the target Forth on the board. You can use this as a normal Forth, and you can even compile code on it. However, since the cross compiler takes a fraction of a second, and the Forth has facilities to reflash itself, it is quicker just to recompile and reflash. To use the Forth facilities, just type

REFLASH

AIDE will put up a box asking for an image file. Navigate to the Kinetis folder and select K60twr.img. AIDE will download this to the target and reflash the target. If the process has no errors, at the end you will be offered R to reboot. The new target image will then run.

You just repeat the recompile and reflash cycle as required.

If the REFLASH operation fails, you will have to use the BDM/JTAG tools to put the image or hex file into the target again.

## Adding application code

Control files and text macros are important in MPE Forth systems. They are documented in the main cross compiler manual. Please read these sections of the manual.

Although you can just add new lines to the MPE control files, we do not recommend this. You run the risk of losing your changes when you update the compiler. Make your own project. You do this by making a copy of K60twr.ctl and using it as the basis of your project. MPE suggests that you leave our code alone and make a new project that is **not** in the cross compiler folder.

To do this, make a new folder and put the renamed control file in it, (say) MyK60.ctl. Also copy over K60twr.no and rename it as (say) myK60.no. Use AIDE's "External Tools" "Copy Tool" button to make a new project button and edit the start directory and include file name.

Edit the control file's text macros that start at line 41 so that they point to the right folders. Edit the build file name on line 320. Now use your new button to recompile and then reflash the new image.