

# MPE xARM/Cortex

---

## First steps – Nuvoton NuMicro-SDK



---

# First steps – Nuvoton NuMicro-SDK

---

## Book name

Manual revision 7.10

Date 08 July 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:

Sequoia Technology

---

# Contents

---

<b>1 Setting up</b>	<b>1</b>
AIDE tools.....	1
AIDE PowerTerm.....	3
What you did.....	4
<b>2 Compiling and testing</b>	<b>5</b>
Compile and download.....	5
Adding application code.....	7

---

# 1

## Setting up

---

We assume that you are using a supported board. For the Nuvoton NUC1xx family, this is the Nuvoton NuMicro-SDK board with an NUC140VE3AN device. The board also comes with a NuLink JTAG unit, which we assume you will be using. The process for the Nuvoton NuTiny-120 board is very similar.

If you have not already got it, install the Nuvoton ICP programmer and drivers from the CD that came with your board. This allows you to program the Flash on the chip through the NuLink unit. The latest versions are available from

<http://www.nuvoton.com/>

Find the directory containing the MPE control file

NUC140.ctl

and note it down. It should be in

<xArmCortex>/Cortex/Hardware/LPC17xx

### AIDE tools

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

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

To use the MPE control file

NuSdk140.ctl

set the start directory to the containing NuSdk140.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\CM01.bmp

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

Command tail: -ide /ide /pauseoff

Start dir: C:\buildkit.dev\software\ROM\Cortex\Hardware\NUC1xx

Keybd Strokes: include NuSdk140.ctl

All four checkboxes in the "Captured Tool Options" box are checked. See the following picture for my complete settings.

The screenshot shows the 'Add/Configure Command Line Tool Option' dialog box. It is divided into three main sections: IDE Parameters, Tool Execution Parameters, and Captured Tool Options. In the IDE Parameters section, the Tool Name is 'NuSDK140' and the Tool Bitmap is 'c:\buildkit.dev\software\Aide\CM01.bmp'. There is a checkbox for 'On Help menu' which is unchecked. In the Tool Execution Parameters section, the Program/File is 'c:\buildkit.dev\software\compiler\xArmCortexDev.exe', the Command tail is '-ide /ide /pauseoff', and the Start Directory is 'C:\buildkit.dev\software\ROM\Cortex\Hardware\NUC1xx'. In the Captured Tool Options section, the Keyboard Strokes are 'include NuSdk140.ctf'. There are four checkboxes: 'Capture Output to IDE' (checked), 'Notify on finish' (checked), 'Act on commands' (checked), and 'Clear screen' (checked). At the bottom, there are buttons for '< Prev', 'Next >', 'Add Tool', 'Remove Tool', 'Copy Tool', 'Exit', and 'Save'.

IDE Parameters

Tool Name: NuSDK140

Tool Bitmap: c:\buildkit.dev\software\Aide\CM01.bmp

☐ On Help menu

Tool Execution Parameters

Program/File: c:\buildkit.dev\software\compiler\xArmCortexDev.exe

Command tail: -ide /ide /pauseoff

Start Directory: C:\buildkit.dev\software\ROM\Cortex\Hardware\NUC1xx

Captured Tool Options

Keyboard Strokes: include NuSdk140.ctf

☒ Capture Output to IDE ☒ Notify on finish ☒ Act on commands ☒ Clear screen

< Prev Next > Add Tool Remove Tool Copy Tool Exit Save

Add the Nuvoton ICP programmer as an external tool

The screenshot shows the 'Add/Configure Command Line Tool Option' dialog box. It is divided into three main sections: IDE Parameters, Tool Execution Parameters, and Captured Tool Options. In the IDE Parameters section, the Tool Name is 'NuMicro ICP' and the Tool Bitmap is 'C:\buildkit.dev\software\aide\CM01.BMP'. There is a checkbox for 'On Help menu' which is unchecked. In the Tool Execution Parameters section, the Program/File is '"F:\Apps\Nuvoton\ICPTool\NuMicro ICP Programming Tool.exe"', the Command tail is empty, and the Start Directory is 'F:\Apps\Nuvoton\ICPTool'. In the Captured Tool Options section, the Keyboard Strokes are empty. There are four checkboxes: 'Capture Output to IDE' (unchecked), 'Notify on finish' (unchecked), 'Act on commands' (unchecked), and 'Clear screen' (unchecked). At the bottom, there are buttons for '< Prev', 'Next >', 'Add Tool', 'Remove Tool', 'Copy Tool', 'Exit', and 'Save'.

IDE Parameters

Tool Name: NuMicro ICP

Tool Bitmap: C:\buildkit.dev\software\aide\CM01.BMP

☐ On Help menu

Tool Execution Parameters

Program/File: "F:\Apps\Nuvoton\ICPTool\NuMicro ICP Programming Tool.exe"

Command tail:

Start Directory: F:\Apps\Nuvoton\ICPTool

Captured Tool Options

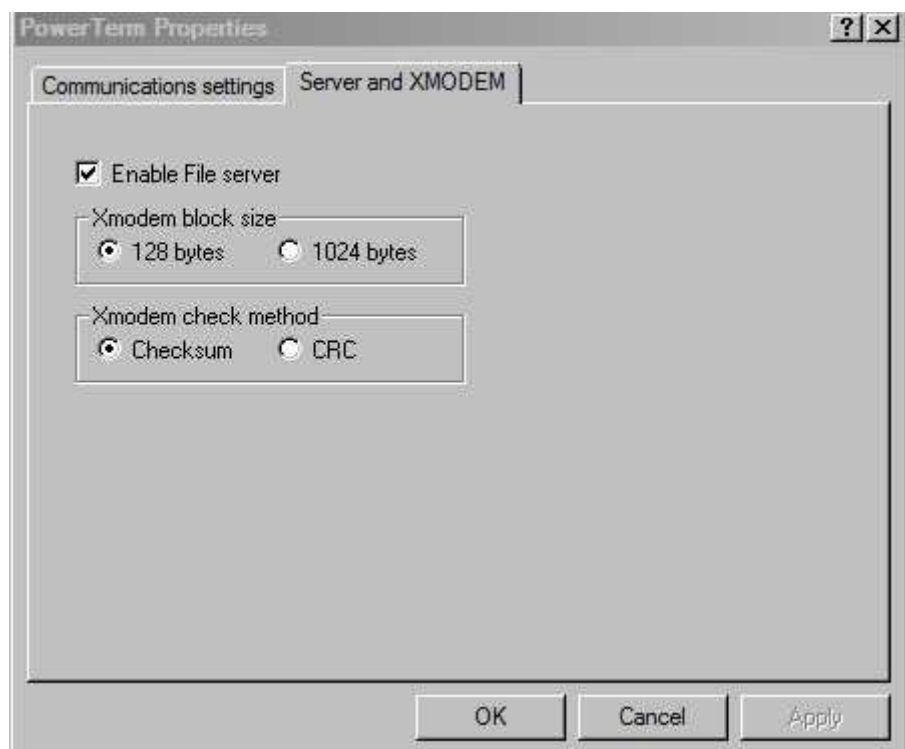
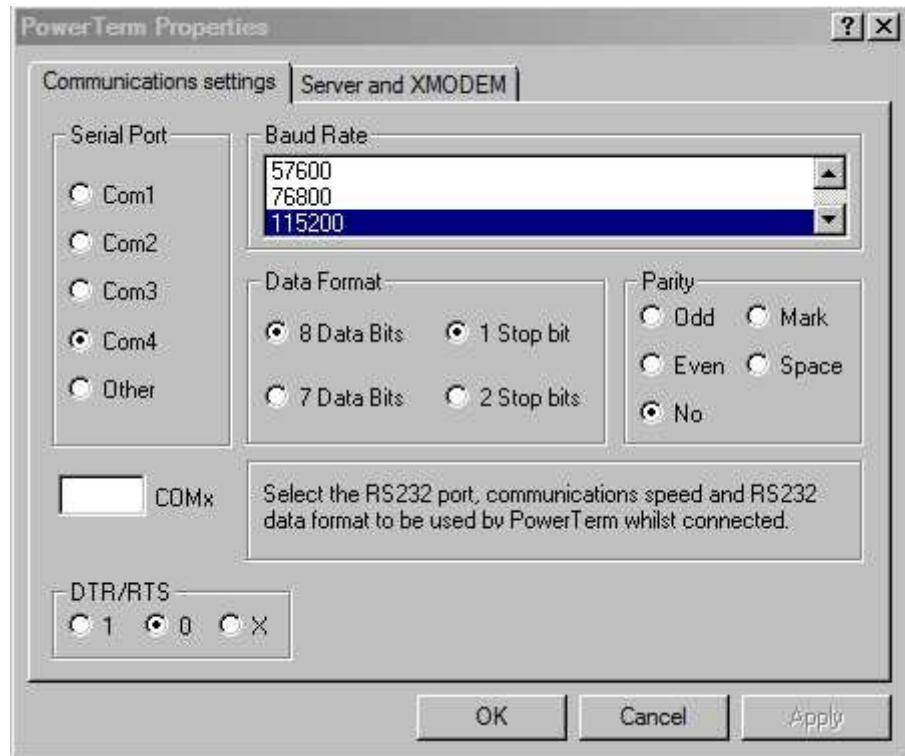
Keyboard Strokes:

☐ Capture Output to IDE ☐ Notify on finish ☐ Act on commands ☐ Clear screen

< Prev Next > Add Tool Remove Tool Copy Tool Exit 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 two tools set up, the Forth cross compiler and the Nuvoton ICP programmer. There will be a corresponding buttons on the toolbar.

When you click the compiler button the file

`NuSdk140.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.

When you click the Nuvoton ICP button, the Nuvoton ICP runs. Because the tool capture checkboxes are unchecked, it runs in an independent 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

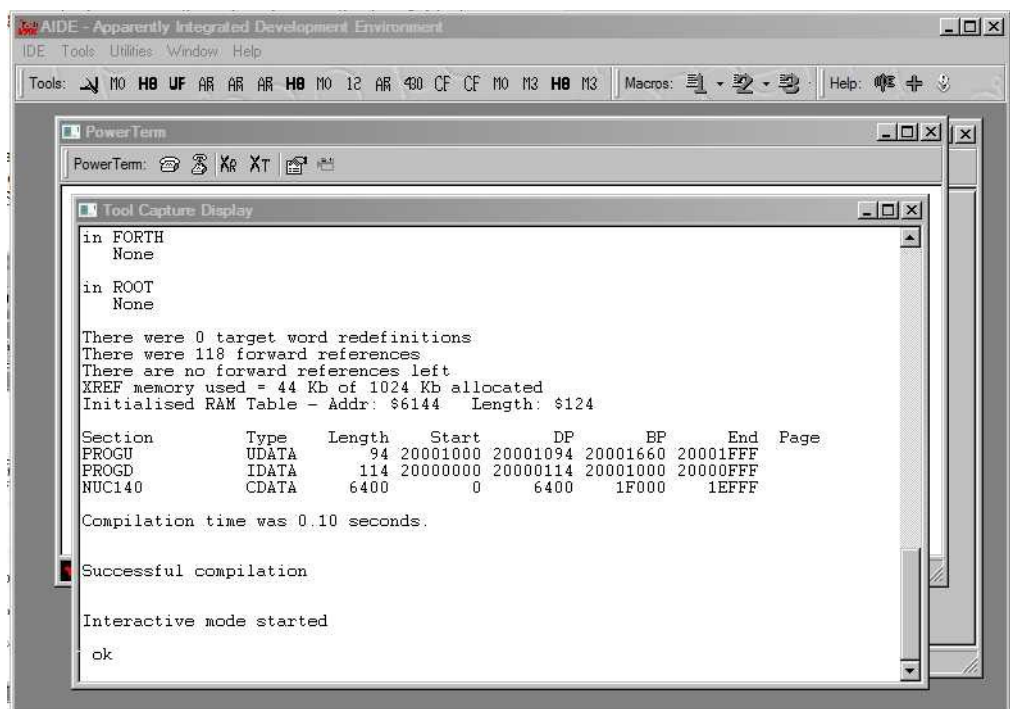
NuSdk140.ctl

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 NUC1xx folder

NUC140.img

NUC140.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 external tools such as the Nuvoton ICP.

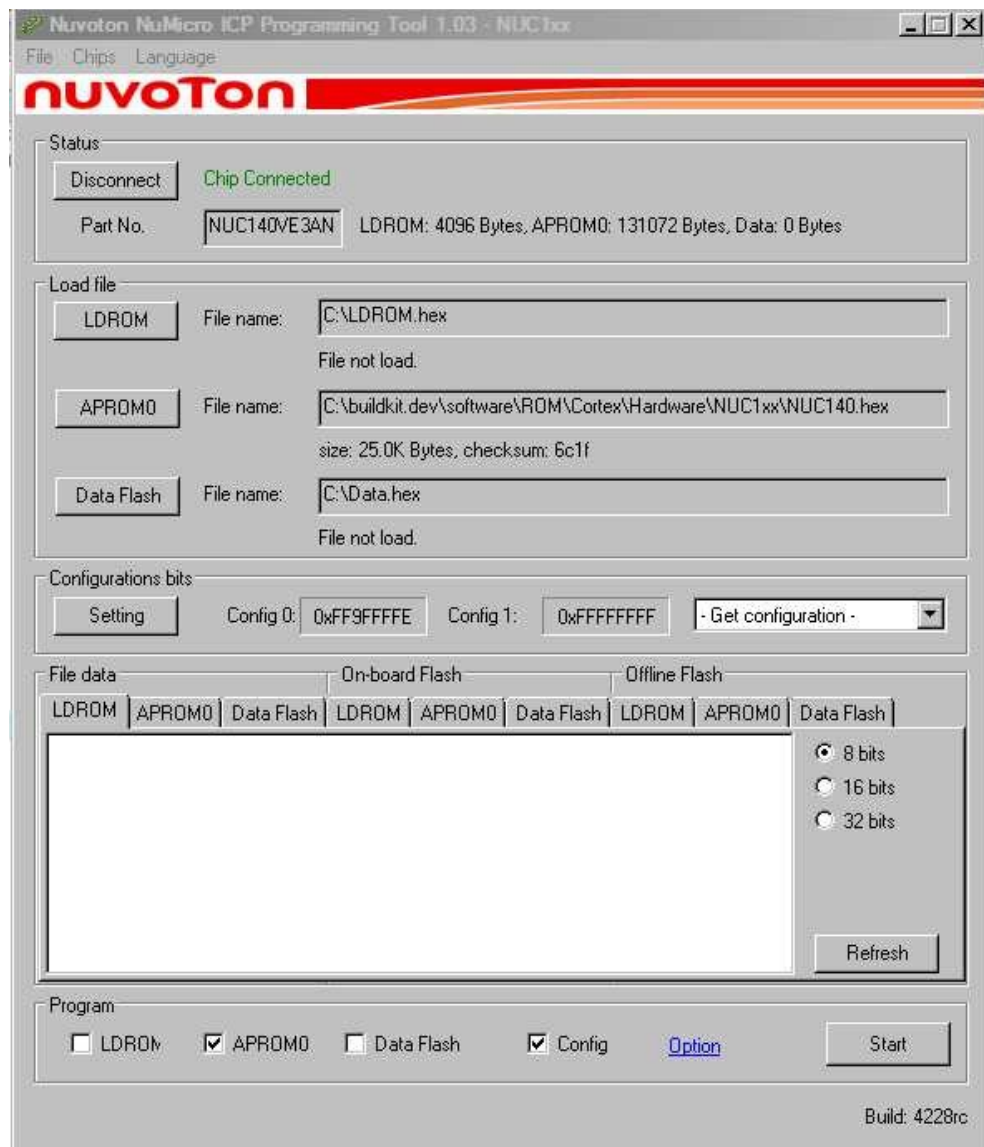


AIDE's Tool Capture window should be visible - the compiler is still alive so that you can disassemble words and use the cross reference tools such as XREF and LOCATE. Make sure that PowerTerm is also visible, but is disconnected.

We will now program the Flash on the Nuvoton board.

- Connect the Nu-Link to a PC USB port and the board. This should power the board and an LED will light on the board.
- Launch the Nuvoton ICP.

- Check that the configuration is correct – ours is illustrated below.
- Get the latest hex file using the Refresh button.
- Program the Flash using the Start button.



Connect the board's serial port to the PC. Check the connections with a meter or oscilloscope – they may not be as you expect. Just use DB-9 pins 2, 3 and 5 (ground). The pinouts are described in detail at

[http://pinouts.ru/SerialPorts/Serial9\\_pinout.shtml](http://pinouts.ru/SerialPorts/Serial9_pinout.shtml)

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, it is usually quicker just to recompile and reflash.

## 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 NuSdk140.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) MyNUC140.ctl. Also copy over nusdk140.no and rename it as (say) mynuc140.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 around line 272. Now use your new button to recompile and then reflash the new image.