

Micrium

Empowering Embedded Systems

μC/OS-II

μC/Probe

μC/TCP/IP

and the

NXP LH79520 Processor

(Using the LogicPD Zoom Development Kit with
a LH79520 Card Engine)

Read Me

www.Micrium.com

About Micrium

Micrium provides high-quality embedded software components in the industry by way of engineer-friendly source code, unsurpassed documentation, and customer support. The company's world-renowned real-time operating system, the Micrium **μC/OS-II**, features the highest-quality source code available for today's embedded market. Micrium delivers to the embedded marketplace a full portfolio of embedded software components that complement **μC/OS-II**. A TCP/IP stack, USB stack, CAN stack, File System (FS), Graphical User Interface (GUI), as well as many other high quality embedded components. Micrium's products consistently shorten time-to-market throughout all product development cycles. For additional information on Micrium, please visit www.micrium.com.

About μC/OS-II

Thank you for your interest in **μC/OS-II**. **μC/OS-II** is a preemptive, real-time, multitasking kernel. **μC/OS-II** has been ported to over 45 different CPU architectures and now, has been ported to NXP's LH79520 (ARM7) processor.

μC/OS-II is small yet provides all the services you'd expect from an RTOS: task management, time and timer management, semaphore and mutex, message mailboxes and queues, event flags an much more.

You will find that **μC/OS-II** delivers on all your expectations and you will be pleased by its ease of use.

About μC/TCP-IP

μC/TCP-IP is a compact, reliable, high performance TCP/IP protocol stack. Built from the ground up with Micrium's renowned quality, scalability and reliability, **μC/TCP-IP** enables the rapid configuration of required network options to minimize your time to market. **μC/TCP-IP** provides you with the highest quality source code in the industry.

μC/TCP-IP is a clean-room design and is not derived from publicly available Unix stacks, yet still maintains compatibility with the Berkeley 4.4 socket layer interface. As with all Micrium products, **μC/TCP-IP** is written in ANSI C enabling its usage with a wide array of best-of-class cross-development tools. **μC/TCP-IP** can be used on 16, 32 and even some 64-bit CPUs.

μC/TCP-IP was designed specifically for the demanding requirements of embedded systems. Critical sections were kept to a minimum and selected run-time validations can be disabled to enhance performance. **μC/TCP-IP** implements zero copy buffer management for highest efficiency.

μC/TCP-IP allows you to adjust the memory footprint based on your requirements. **μC/TCP-IP** can be configured to only include only those network modules absolutely required by your system. When a module is not used, it's not included in the build to save valuable memory space for resource limited embedded systems.

Licensing

μC/OS-II and μC/TCP-IP are provided in source form for **FREE** for educational use or for peaceful research.

If you plan on using μC/OS-II in a commercial product you can evaluate μC/OS-II and μC/TCP-IP for **FREE** for **45 days** and within that period, need to contact Micrium to properly license its use in your product.

We provide **ALL** the source code with this application note for your convenience and to help you experience μC/OS-II and μC/TCP-IP. The fact that the source is provided **DOES NOT** mean that you can use it without paying a licensing fee. Please help us continue to provide the Embedded community with the finest software available. Your honesty is greatly appreciated.

About µC/Probe

µC/Probe is a Windows application that allows a user to display the value (at run-time) of virtually any variable or memory location on a connected embedded target. The user simply populates µC/Probe's graphical environment with gauges, tables, graphs, and other components, and associates each of these with a variable or memory location. Once the application is loaded onto the target, the user can begin µC/Probe's data collection, which will update the screen with variable values fetched from the target.

µC/Probe retrieves the values of global variables from a connected embedded target and displays the values in a engineer-friendly format. The supported data-types are: booleans, integers, floats, and ASCII strings.

µC/Probe can have any number of 'data screens' where these variables are displayed. This allows to logically group different 'views' into a product.

A 30-day trial version of µC/Probe is available on the Micrium website:

<http://www.micrium.com/products/probe/probe.html>

The screenshot shows the Micrium µC/Probe application interface. The main window is titled "Micrium µC/Probe - OS-Probe-Workspace.wsp" and has a menu bar with "File", "Tools", and "Help". Below the menu bar is a toolbar with various icons. The main content area is divided into several tabs: "OS: Task CPU Usage", "OS: Task Stack Usage", "OS: Task Info", and "OS: General Info". The "OS: Task Info" tab is selected, displaying "Task Stack Information" and "General Task Information".

Task Stack Information

Name	Stack Pointer	Stack Usage		Stack	
		Maximum	Current	Starts @	Ends @
uC/OS-II Idle	0x0201BC00	1376/2048	1316/2048	0x0201C124	0x0201B924
uC/OS-II Stat	0x0201ABEC	1436/2048	1176/2048	0x0201B084	0x0201A884
uC/OS-II Tmr	0x0201B460	1420/2048	1196/2048	0x0201B90C	0x0201B10C
Start Task	0x020141C0	1448/4096	1188/4096	0x02014664	0x02013664
uC/Probe OS	0x02019C54	1460/2048	1236/2048	0x0201A128	0x02019928
Probe RS-232	0x020193B4	1512/2048	1200/2048	0x02019864	
LED1	0x0201491C	1412/2048	1352/2048	0x02014E64	
LED2	0x020151B4	1412/2048	1200/2048	0x02015664	
LED3	0x020159B4	1412/2048	1200/2048	0x02015E64	
LED4	0x020161B4	1412/2048	1200/2048	0x02016664	
LED5	0x020169B4	1412/2048	1200/2048	0x02016E64	
LED6	0x020171B4	1424/2048	1200/2048	0x02017664	
LED7	0x020179B4	1424/2048	1200/2048	0x02017E64	
LED8	0x020181B4	1424/2048	1200/2048	0x02018664	
Probe Str	0x02018928	1404/2048	1344/2048	0x02018E68	

General Task Information

Name	ID	Priority	Task Status		
			State	Delay	Waiting On
uC/OS-II Idle	65535	31	Ready	----	
uC/OS-II Stat	65534	30	Delay	95	
uC/OS-II Tmr	65533	29	Semaphore	----	

Target Output

```
String Tx #00041
String Tx #00042
String Tx #00043
String Tx #00044
String Tx #00045
String Tx #00046
String Tx #00047
String Tx #00048
String Tx #00049
String Tx #00050
String Tx #00051
String Tx #00052
String Tx #00053
String Tx #00054
String Tx #00055
String Tx #00056
String Tx #00057
```

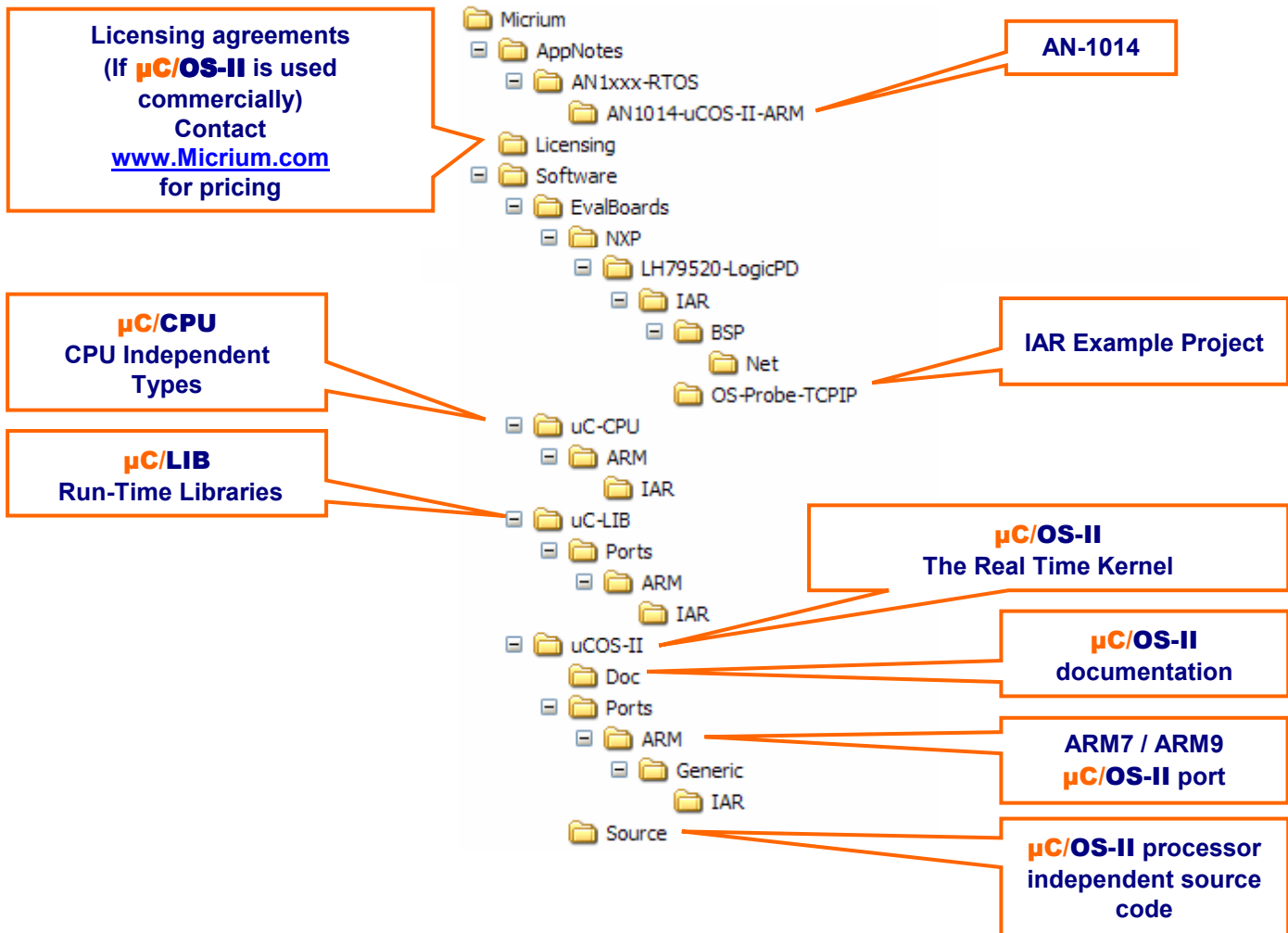
The status bar at the bottom of the application shows "Running" and "RS-232 38400 COM1 1183 bytes/sec".

Installing the Micrium Software

The source code for **μC/OS-II** and **μC/TCPIP** is provided in source form along with application code that allows you to run **μC/OS-II** and **μC/TCPIP** on the NXP LH79520 (ARM7) processor. To install the software, simply run the self-extracting executable.

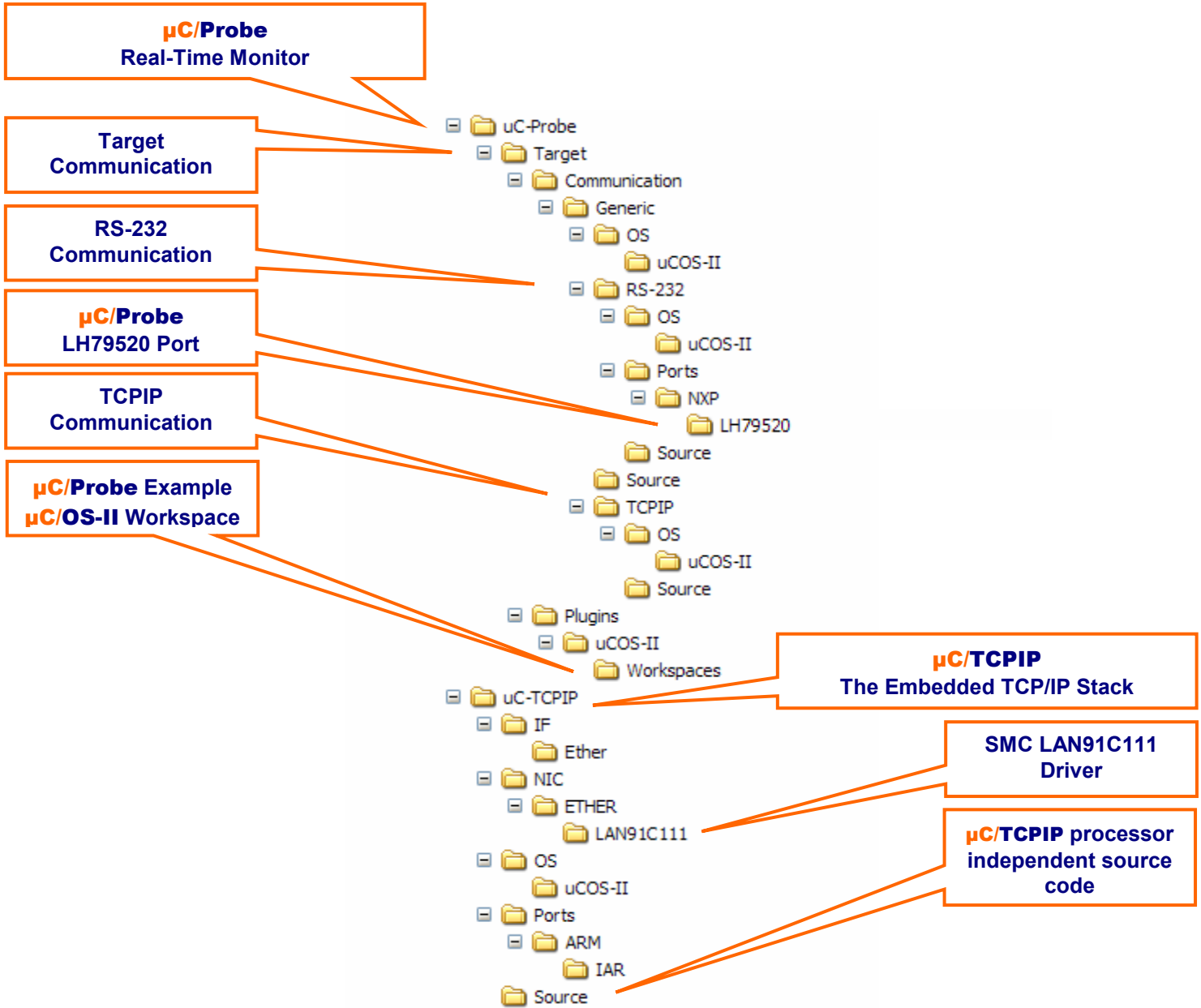
The self-extracting executable is called *Micrium-NXP-uCOS-II-TCPIP-LH79520-LogicPD.exe*.

You will be prompted to accept the simple terms of the licensing agreement. If you answer 'Yes', the software will be installed on your PC under the *\Micrium* directory from the root:



Micrium

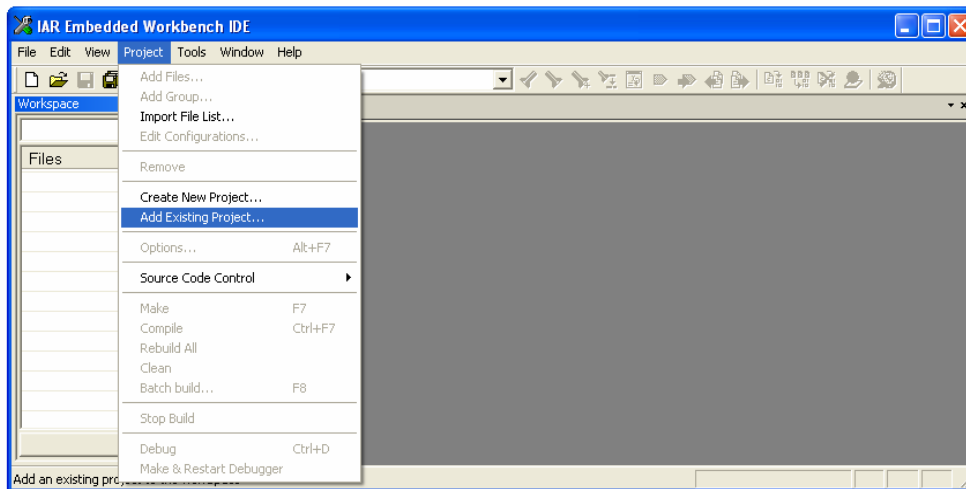
µC/OS-II, µC/Probe & µC/TCPIP for the
NXP LH79520



Using the Example Project

You should read AN-1014 if you are interested in further information about the port for ARM processors.

To view the example project, start an instance of IAR Embedded Workbench, and open either the project file **LH79520-OS-Probe-TCPIP.ewp** (if you are using IAR EW v4.4x) or **LH79520-OS-Probe-TCPIP-v5.ewp** (if you are using IAR EW v5.1x) located in the directory marked “IAR Example Project” in the tree above. To do this, use the *Add Existing Project...* menu command under the *Project* menu:



The µC/OS-II Kernel Awareness plugin will allow you to examine information about system objects while using the C-Spy debugger. To gain access to this feature, enable the plug-in by right-clicking on the project name in the work space browser and choosing *Options...* Then, select the “Debugger” entry in the list box, and the “Plugins” tab pane. Find the µC/OS-II entry in the list and, finally, select the check box beside the entry.

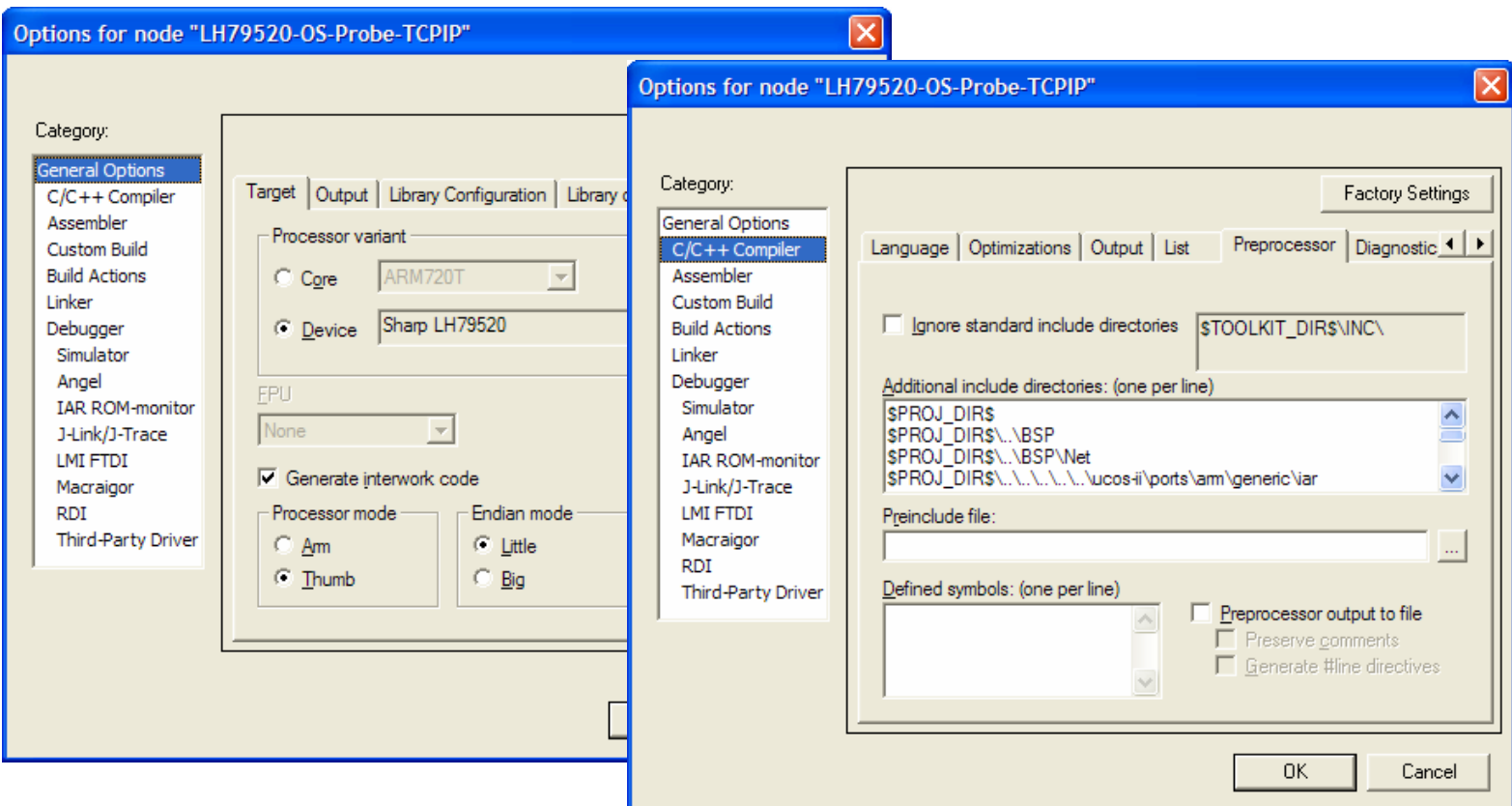
IAR EWARM Versions

Be certain to open the proper project for your version of EWARM. IAR EWARM v4.4x will NOT open a v5.1x project. And though IAR EWARM v5.1x will open a v4.4x project, many errors will be generated upon compilation.

If you have opened the correct project file, but IAR gives errors because of project file compatibility problems, verify that the following project settings (or equivalent) are in place:

1. The target device "Sharp LH79520" is selected.
2. The following "C/C++ Compiler" → "Preprocessor" → "Additional Include Directories" are present:

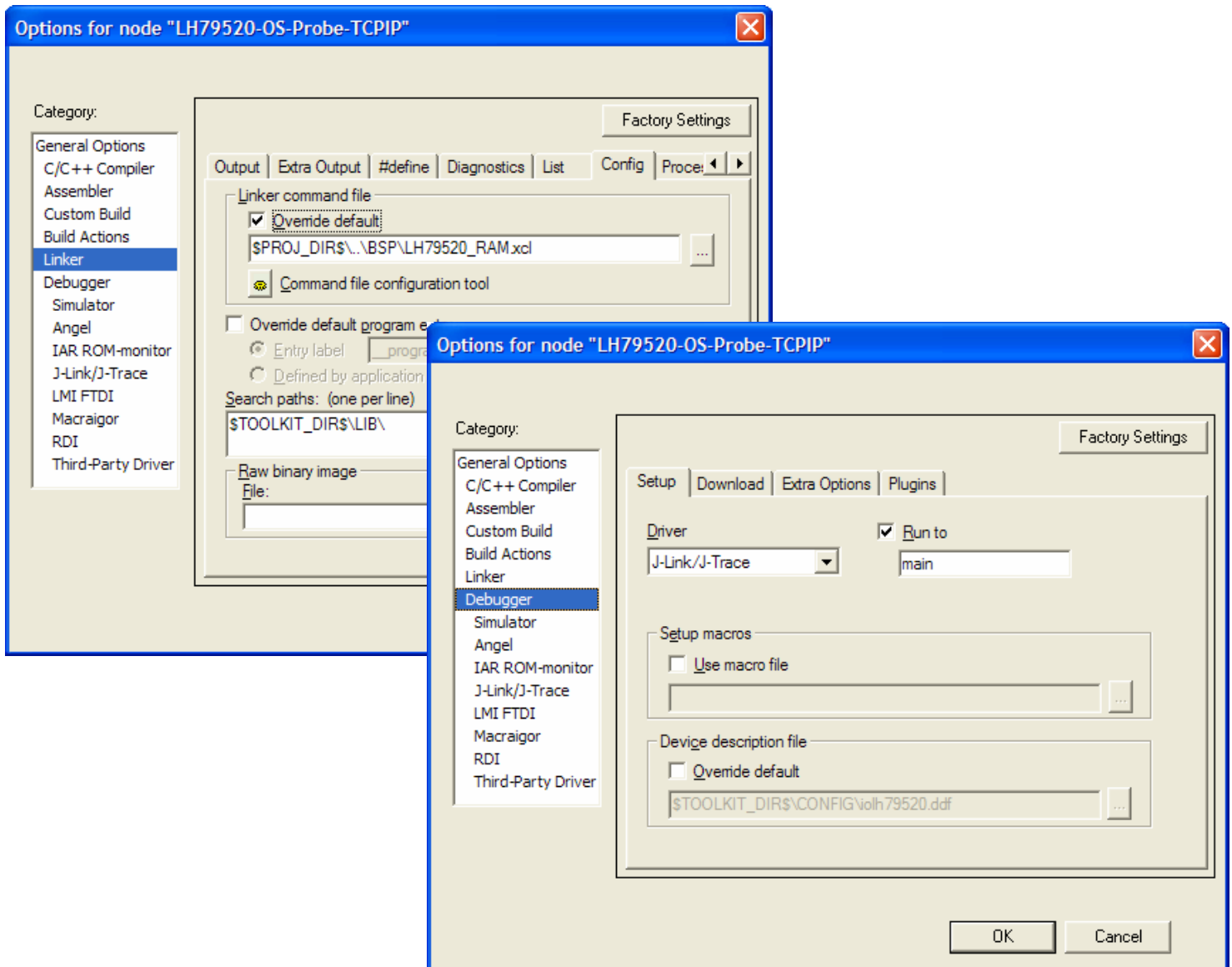
```
$PROJ_DIR$\..\BSP
$PROJ_DIR$\..\BSP\Net
$PROJ_DIR$\..\..\..\..\ucos-ii\ports\arm\generic\iar
$PROJ_DIR$\..\..\..\..\ucos-ii\source
$PROJ_DIR$\..\..\..\..\uc-lib
$PROJ_DIR$\..\..\..\..\uc-cpu
$PROJ_DIR$\..\..\..\..\uc-cpu\arm\IAR
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Plugins\uCOS-II
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Communication
    \Generic\RS-232\Source
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Communication
    \Generic\RS-232\Ports\NXP\LH79520
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Communication
    \Generic\Source
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Communication
    \Generic\TCPIP\Source
$PROJ_DIR$\..\..\..\..\uC-Probe\Target\Plugins\uCOS-II
$PROJ_DIR$\..\..\..\..\uC-TCPIP\OS\uCOS-I I
$PROJ_DIR$\..\..\..\..\uC-TCPIP\NIC
$PROJ_DIR$\..\..\..\..\uC-TCPIP\IF
$PROJ_DIR$\..\..\..\..\uC-TCPIP\IF\Ether
$PROJ_DIR$\..\..\..\..\uC-TCPIP\NIC\ETHER\LAN91C111
$PROJ_DIR$\..\..\..\..\uC-TCPIP\Source
$PROJ_DIR$\..\..\..\..\uC-TCPIP\NIC\ETHER\PHY
```



Micrium

µC/OS-II, µC/Probe & µC/TCPIP for the
NXP LH79520

3. The appropriate "Linker" → "Config" → "Linker configuration file" is chosen. For v4.4x, this should be `$PROJ_DIR$\..\BSP\LH79520_RAM.xcl`. For v5.1x, this should be `$PROJ_DIR$\..\BSP\LH79520_RAM.icf`.
4. In "Debugger" → "Setup" → "Setup macros", no file should be selected.



Using μC/Probe with the Example Project

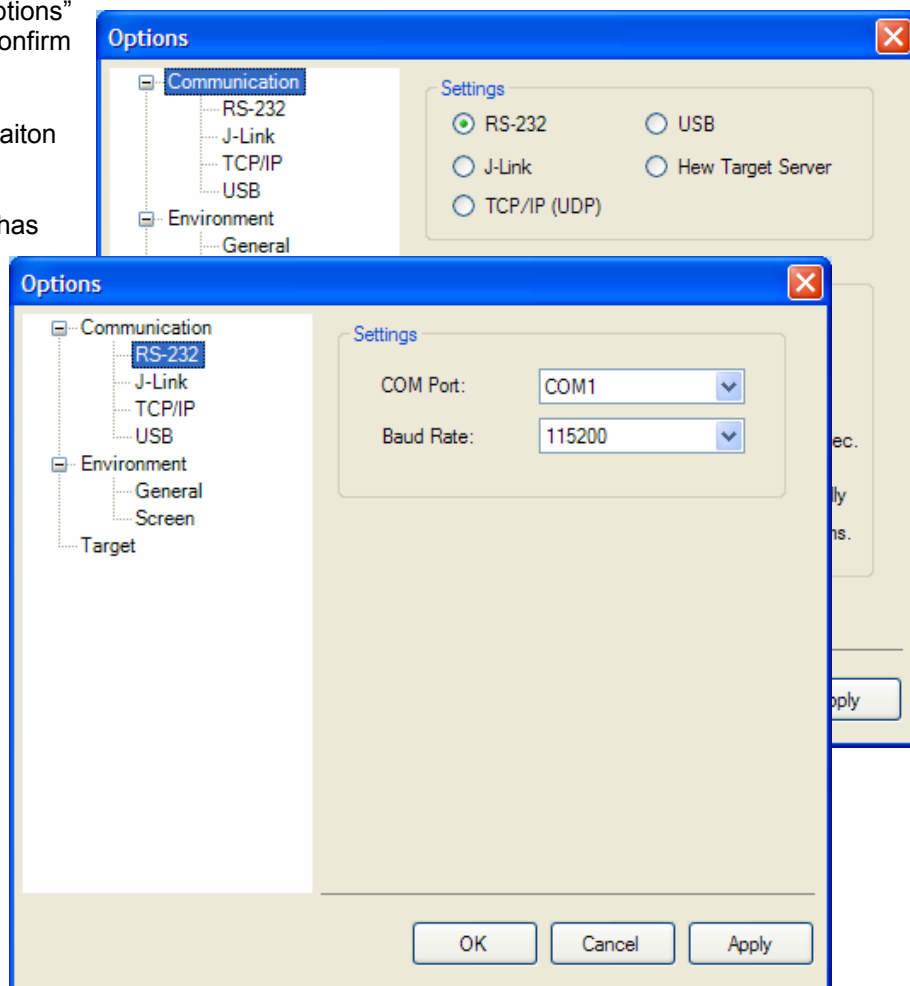
As configured, the project will use the LH79520's UART1, which is connected to the only RS-232 (DB9) connector on the LogicPD evaluation board. To use μC/Probe with the example project, perform the following steps:

1. Compile the project, load the code onto the board and begin running. The GPIO1 LED on the evaluation board should begin to blink.
2. Start μC/Probe and open the example workspace (*.wsp) located in the directory

\\Micrium\Software\uC-Probe\Plugins\uCOS-II\Workspace

3. Check the options (choose "Options" from the "Tools" menu) and confirm that

- a. The RS-232 communication method is selected;
- b. A baud rate of 115200 has been selected;
- c. The appropriate COM port (on your computer) is selected.



4. Load the ELF file into the Symbol Browser. This should be located in the directory

\\Micrium\Software\Evalboards\NXP\LH79520-LogicPD\IAR\OS-Probe-TCPIP\Flash\Exe

For IAR EW v4.4x, the file will have a *.elf extension; for IAR EW v5.1x, the file will have *.out extension.

5. Press the "Run" button on the toolbar (a green triangle).

About the Example Application

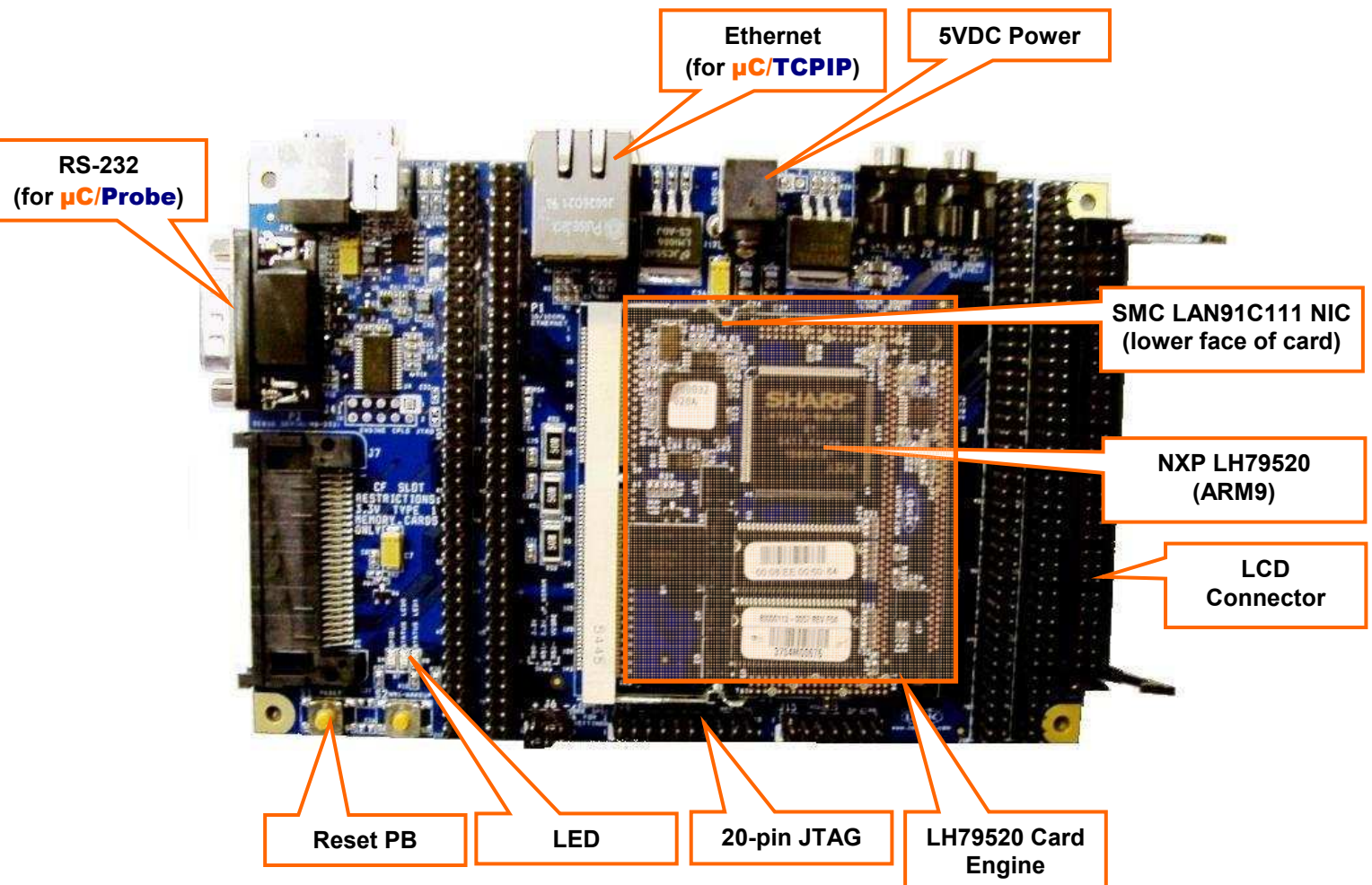
The example project includes a basic demonstration of μ C/OS-II, μ C/Probe and μ C/TCPIP. The evaluation board components are labeled in the figure below. After you load the evaluation board with the sample project, the LED will start to blink. The RS-232 port (at 115200 baud) is used for μ C/Probe, which allows you to view (in real-time) the value of static / global variables in the target system. A 30-day trial version of this program can be downloaded from the Micrium website.

The Ethernet port is used for μ C/TCPIP, which uses the external LAN91C111 NIC on the LH79520 card engine. The project, as configured, uses the static IP address 10.10.1.161, the network mask 255.255.255.0, and the gateway 10.10.10.1. If these are not appropriate for your network, you can correct them by modifying the values used in the function `AppInitTCPIP()` in the code file `app.c`, which is located in

/Micrium/Software/Evalboards/LH79520-LogicPD/IAR/OS-Probe-TCPIP

Once you have configured your IP address, compiled the code, and loaded your evaluation board, you can then ping your target from a PC that is on the same network. (μ C/Probe can also be used with the target via TCP/IP—just configure the options in the Windows program.)

See the other LH79520 projects on the Micrium for additional information about this processor and examples applications for this evaluation board.



Erratas

If you find any errors in the documentation or code provided, please send those corrections to Support@Micrium.com. Be sure to specify the processor, version of μC/OS-II and any other pertinent information about the error being reported.

Contacts

Micrium
949 Crestview Circle
Weston, FL 33327
USA

+1 954 217 2036
+1 954 217 2037 (FAX)

e-mail: Sales@Micrium.com
WEB: www.Micrium.com