



Chipsmall Limited consists of a professional team with an average of over 10 year of expertise in the distribution of electronic components. Based in Hongkong, we have already established firm and mutual-benefit business relationships with customers from,Europe,America and south Asia,supplying obsolete and hard-to-find components to meet their specific needs.

With the principle of “Quality Parts,Customers Priority,Honest Operation,and Considerate Service”,our business mainly focus on the distribution of electronic components. Line cards we deal with include Microchip,ALPS,ROHM,Xilinx,Pulse,ON,Everlight and Freescale. Main products comprise IC,Modules,Potentiometer,IC Socket,Relay,Connector.Our parts cover such applications as commercial,industrial, and automotives areas.

We are looking forward to setting up business relationship with you and hope to provide you with the best service and solution. Let us make a better world for our industry!



Contact us

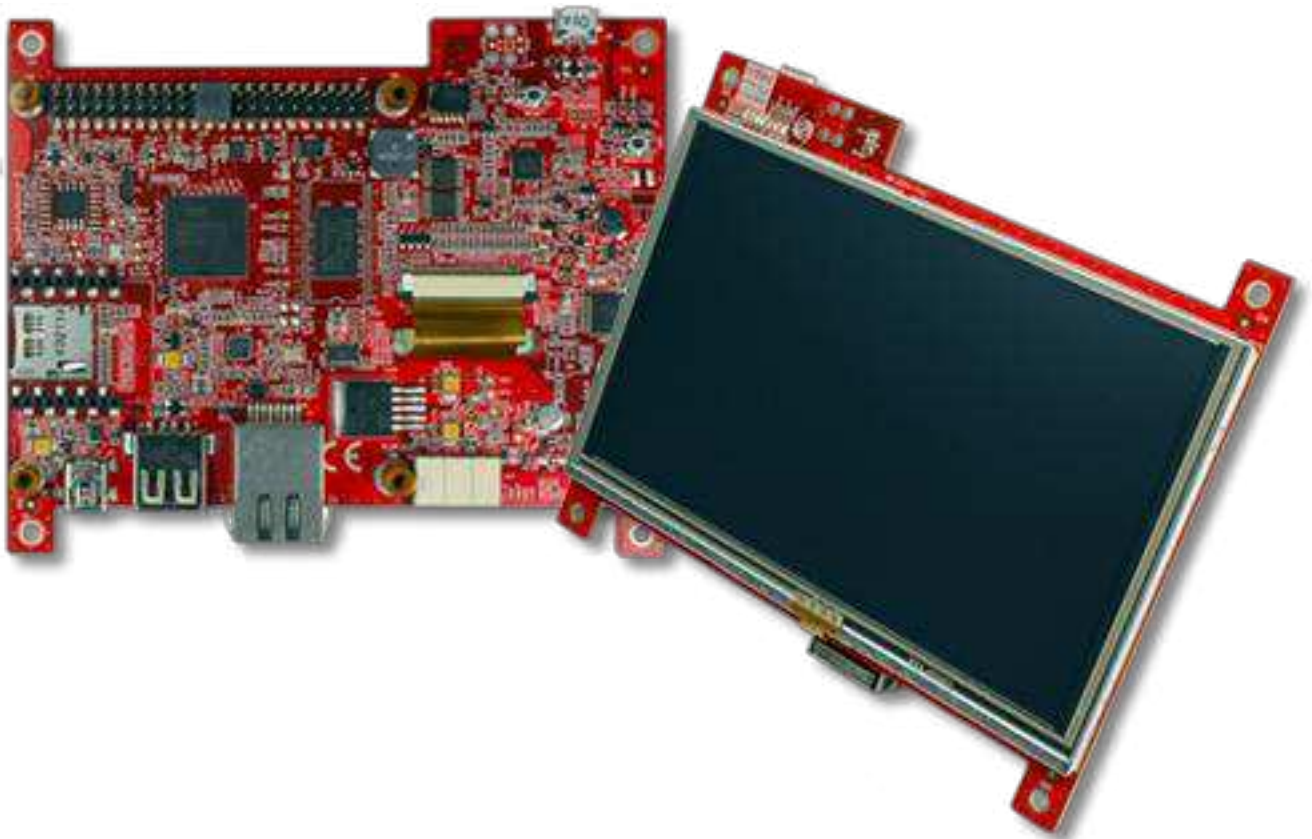
Tel: +86-755-8981 8866 Fax: +86-755-8427 6832

Email & Skype: info@chipsmall.com Web: www.chipsmall.com

Address: A1208, Overseas Decoration Building, #122 Zhenhua RD., Futian, Shenzhen, China



LPC4088 Display Module User's Guide



*Get Up-and-Running Quickly and
Start Developing Your Application On Day 1!*

Embedded Artists AB

Davidshallsgatan 16
211 45 Malmö
Sweden

info@EmbeddedArtists.com
<http://www.EmbeddedArtists.com>

Copyright 2015 © Embedded Artists AB. All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of Embedded Artists AB.

Disclaimer

Embedded Artists AB makes no representation or warranties with respect to the contents hereof and specifically disclaim any implied warranties or merchantability or fitness for any particular purpose. Information in this publication is subject to change without notice and does not represent a commitment on the part of Embedded Artists AB.

Feedback

We appreciate any feedback you may have for improvements on this document. Please send your comments to support@EmbeddedArtists.com.

Trademarks

All brand and product names mentioned herein are trademarks, services marks, registered trademarks, or registered service marks of their respective owners and should be treated as such.

Table of Contents

1	Document Revision History	5
2	Introduction	6
2.1	Features	6
2.2	ESD Precaution	7
2.3	Warranty Hints and General Handling Care	7
2.4	Code Read Protection	8
2.5	CE Assessment	8
2.6	Other Products from Embedded Artists	8
2.6.1	Design and Production Services	8
2.6.2	OEM / LPCXpresso / QuickStart Boards and Kits	8
3	Getting Started	9
3.1	Usage Models	9
3.2	Hardware Overview	11
3.3	Powering	12
3.4	Demo/Test Application	13
3.5	Software Development	13
3.5.1	mbed Framework	13
3.5.2	LPCOpen Framework	14
3.5.3	Setup Own Framework	14
3.6	Program Download	15
3.7	How to Debug	18
4	Connectors	19
4.1	Main Interface Connector, J11	19
4.2	mbed micro-B USB Connector, J9	22
4.3	Debug Interface, J7	23
4.4	Tag-Connect Debug Interface, J6	23
4.5	RJ45 Ethernet Connector, J1	24
4.6	USB Host/Device Connectors, J5/J4	24
4.7	RF Module Interface Connector, J2	24
4.8	Memory Card Interface Connector, J3	26
4.9	Expansion Connector, J10	26
5	Expansion	27
5.1	General Design Checklist	30
6	How to Embed / Integrate the Display Module	33
6.1	Powering	33
6.1.1	+5V Power supply example	33
6.2	Optional VBAT Powering	34
6.3	Interface to Environment	34

6.4	Mechanical Integration	34
6.5	CE Declaration of Conformity	35
6.6	Contact Embedded Artists	35
7	Custom Specific Versions	36
8	Future Proof Design - BIOS	37
9	Things to Note	38
9.1	Current Consumption and Limits of USB Ports	38
9.2	CMSIS-DAP program download time	38
9.3	HDK USB Mass-storage does not work on Mac - LPC4088 locks up	38
9.4	Cannot download binary to LPC4088 via Cortex Debug Connector	38
9.5	Updating the HDK	38
9.5.1	Update Procedure	39
10	Troubleshooting	40
10.1	Powering	40
10.2	Program Download to LPC4088	41
10.3	Using Test Application to Verify Correct Operation	41
11	Appendix: Rev A Board Differences	44
12	Further Information	45

1 Document Revision History

<i>Revision</i>	<i>Date</i>	<i>Description</i>
PA1	2014-12-10	First version.
PA2	2015-01-26	Update all sections for rev B boards. Appendix added to cover rev A boards.
PA3	2015-01-29	Moved <i>mbed Getting Started Tour</i> to SW manual.

2 Introduction

Thank you for buying Embedded Artists' *LPC4088 Display Module*, which is based on NXP's ARM Cortex-M4F LPC4088 microcontroller.

This document is a User's Guide that describes the *LPC4088 Display Module* mainly from a hardware point of view. Some basic software related issues are also addressed but there is also a separate software manual for the software framework that has been created.

2.1 Features

Embedded Artists' *LPC4088 Display Module* lets you get up-and-running quickly with graphical applications, the mbed-platform and Cortex-M4F programming in general. Communication interfaces, large on-board memories and LCD controller/display enables graphical user interface (GUI) and communication gateway designs. On-board connectors and industrial design makes the module **perfect for product integration**.

The features of the display module are:

- Several display options (like 4.3" color TFT LCD, 480x272 pixels, and 5" color TFT LCD, 800x480 pixels)
 - Resistive or capacitive touch panel with integrated controller to off-load the MCU
 - LCD backlight voltage generator, controlled via PWM signal
 - 16- and 24-bit color depth supported
- 120 MHz Cortex-M4F LPC4088 MCU from NXP
 - MCU core has floating point and DSP support in hardware
- On-module memories
 - 512 kB LPC4088 on-chip flash
 - 96 kB LPC4088 on-chip SRAM
 - 32 MB SDRAM (32-bit databus width for high bandwidth access)
 - 16 MB QSPI flash on SPIFI interface (for both program code and data)
 - 4032 Byte LPC4088 on-chip E2PROM (for parameter storage)
 - 1 Kbit E2PROM with EUI-48™ number (for Ethernet MAC address)
 - 8 Kbyte E2PROM with BIOS code - future proof design! (see chapter 8)
- Extensive Interfaces and Peripherals
 - 100/10 Mbps Ethernet with RJ45 connector
 - USB Host interface with USB-A connector
 - USB Device interface with mini-B connector
 - 20 pos XBee™ compatible connector for RF module add-on
 - Very simple expansion for custom specific I/O via 46 pos, 100 mil pitch pin header (second USB interface, UART, SPI, I2C, ADC, DAC, CAN, GPIO, timer, etc)
 - LPC4088 On-chip RTC, with super capacitor backup
 - uSD/transflash memory card interface for optional memory capacity
 - Reset generation, 4 LEDs, pushbutton on P2.10

- Debug interfaces
 - mbed-enabled via on-board HDK (CMSIS-DAP interface, USB drag-n-drop, Virtual COM port)
 - 10 position SWD/JTAG connector (ARM standard Cortex Debug Connector)
 - 6 position SWD Tag-Connect™ connector
- Powering
 - On-module +3.3V linear power supply
 - External +5V supply needed, either via USB or external +5V DC
 - Optional, external VBAT supply for RTC
- Dimensions
 - Compact size: 123 x 100 mm
- Other
 - Proper ESD protection on communication interfaces
 - Multilayer PCB design for best EMC performance, with controlled impedance and length matched routing for critical signals
 - EMC, EMI and ESD tested
 - Supported by the mbed SDK and online tools
 - Supported by professional tools and middleware
 - Wide/extended temperature range (-20 to +60/70 degrees Celsius, limited by LCD)
 - ISO 9001:2008 / ISO 14001:2004 produced
 - Production and shipping compensated for carbon dioxide emission

2.2 ESD Precaution

Please note that the *LPC4088 Display Module* come without any case/box and all components are exposed for finger touches – and therefore extra attention must be paid to ESD (electrostatic discharge) precaution. Only qualified personnel shall handle the product.

Make it a habit always to first touch the metal surface of one of the USB or Ethernet connectors for a few seconds with both hands before touching any other parts of the boards. That way, you will have the same potential as the board and therefore minimize the risk for ESD.



In general touch as little as possible on the boards in order to minimize the risk of ESD damage.

Note that Embedded Artists does not replace boards that have been damaged by ESD.

2.3 Warranty Hints and General Handling Care

Handle the *LPC4088 Display Module* with care. The board is not mounted in a protective case/box and is not designed for rough physical handling. Only qualified personnel shall handle the product. Connectors can wear out after excessive use.

Do not exercise excessive pressure on the LCD glass area. That will damage the display. Also, do not apply pressure on the flex cables connecting the LCD/touch screen. These are relatively sensitive and can be damaged if too much pressure is applied to them.

Embedded Artists products are subject to manufacturer's guarantee as long as the products are handled with adequate care and caution and in accordance to this manual.

For exact details about warranty, please read the *General Terms and Conditions* document that can be found at <http://www.embeddedartists.com/shop/>

The warranty does not cover defects caused by improper or inadequate installation, maintenance or handling by the user, unauthorized modification or misuse, operation outside the specification a non-compliance of this manual and related documents. In case of doubt, please contact Embedded Artists prior to intended activity.

Note that Embedded Artists does not replace boards where the LCD has been improperly handled. Also note that the warranty does also not cover any defects or damages of other equipment connected to the Embedded Artists product, faulty or not.

2.4 Code Read Protection

The LPC4088 microcontroller has a Code Read Protection function (specifically CRP3, see datasheet for details) that, if enabled, will make the LPC4088 impossible to reprogram (unless the user program has implemented such functionality in firmware).

Note that Embedded Artists does not replace LPC4088 Display Modules where the LPC4088 has CRP3 enabled. It's the user's responsibility to not invoke this mode by accident.

2.5 CE Assessment

The *LPC4088 Display Module* is CE marked. See separate *CE Declaration of Conformity* document.

The *LPC4088 Display Module* is a class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

EMC emission test has been performed on the *LPC4088 Experiment Bundle*. Standard interfaces like Ethernet, USB, serial have been in use. General expansion connectors where internal signals are made available (for example processor pins) have been left unconnected. Connecting other devices to the product via the general expansion connectors may alter EMC emission. It is the user's responsibility to make sure EMC emission limits are not exceeded when connecting other devices to the general expansion connectors of the *LPC4088 Display Module*.

2.6 Other Products from Embedded Artists

Embedded Artists have a broad range of microcontroller based boards that are very low cost and developed for prototyping / development as well as for OEM applications (i.e., for integration). Modifications for OEM applications can be done easily, even for modest production volumes. Contact Embedded Artists for further information about design and production services.

2.6.1 Design and Production Services

Embedded Artists provide design services for custom designs, either completely new or modification to existing boards. Specific peripherals and I/O can be added easily to different designs, for example, communication interfaces, specific analog or digital I/O, and power supplies. Embedded Artists has a broad, and long, experience in designing industrial electronics in general and with NXP's LPC microcontroller families in specific. Our competence also includes wireless and wired communication for embedded systems.

Contact us at info@EmbeddedArtists.com for a discussion about how we can help you!

2.6.2 OEM / LPCXpresso / QuickStart Boards and Kits

Visit Embedded Artists' web page, www.EmbeddedArtists.com, for information about other products.

3 Getting Started

This chapter contains information about how to get acquainted with the *LPC4088 Display Module*. **Please read this section first before you start using the board - it will be well spent time!**

Embedded Artists' has a family of *Display Modules* that are self-contained, intelligent modules with many usage scenarios. There are different display options in the family and new are easily created based on customer requirements.

There are many reasons to use a display module - all of them will make your life easier and less stressful and saves your company a lot of money!

- **Save development time and cost - focus on the application**
Being able to start with a proven design gets you started immediately.
 - **Extensive SW frameworks to start from!**
You can focus on your application instead of all infrastructure software.
 - **Easy to expand HW**
If needed, it is simple to add an expansion board that contains specific hardware that might be needed.
You no longer have to maintain a big platform. You can focus on maintaining the expansion board that contains your core interfaces.
- **Long term availability and Future safe design**
Embedded Artists offers long term availability of display modules along with a future safe design principle that allow changing displays on the modules without changing the application software.
- **Multiple displays options**
Your customers may have different requirements on your product and want different display options. Or maybe you want to have different product options, involving different display options.
Either way, the same HW/SW platform can drive many different displays (different sizes and resolution, different brightness and different touch panel solutions).
- **Pre-CE tested**
The display modules are designed for integration and have been EMC, EMI and ESD tested. This increase the likelihood for your final product to also pass tests.
- **On-board debugger**
There is an on-board debug interface that simplifies program download and debugging.

3.1 Usage Models

A common usage scenario is to **add a display module to an embedded system that already exist**. This embedded system typically has an old user interface that needs to be updated to a more modern look-and-feel. The display module acts as a standalone front panel that handles the new graphical Human Machine Interface (HMI). There are several benefits with this solution:

- No need for a complete redesign of the hardware and software of the existing embedded system.
- Different display sizes can be offered to customers.
- Retrofitting existing systems is possible (sold as an upgrade).
- The existing embedded system can exchange high-level commands with the display module when controlling the graphical elements.

The picture below illustrates this usage scenario.

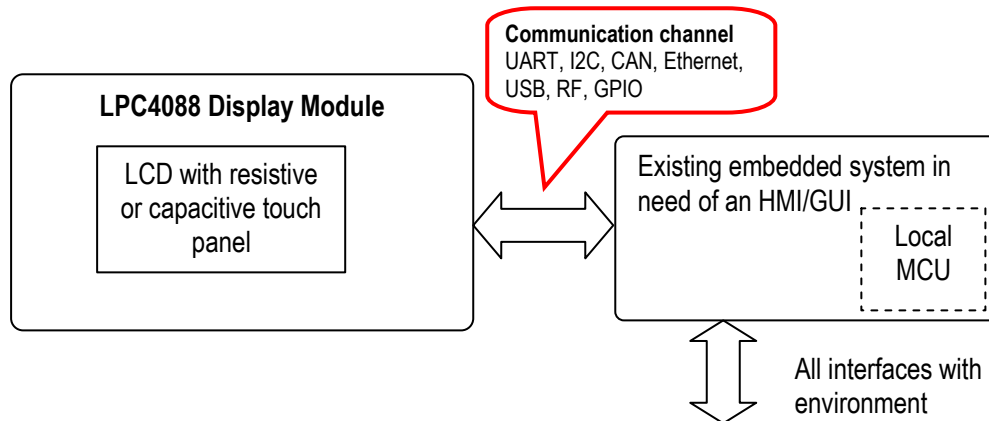


Figure 1 – LPC4088 Display Module HMI/GUI Usage Scenario

A second usage scenario is to **build the complete embedded system around the display module**. The *LPC4088 Display Module* is built around the LPC4088 MCU from NXP. It has a very capable Cortex-M4F core, running at up to 120MHz. There is lots of memory, communication interfaces and expansion possibilities on the module. There are several benefits with this solution:

- The core design has already been done. The only thing needed is a the expansion board design for custom specific interfaces and functions that the core design cannot implement.
 - Low total ownership cost - only need to maintain the expansion board
- Comprehensive software framework exist - the designer can start the real software development immediately.
- Different display sizes can be offered to customers.

The picture below illustrates this usage scenario. If the display module is a standalone solution, an expansion board is likely part of the design, but not always. In many cases, the existing interfaces on the display module are just what is needed. Chapter 5 covers how to design an expansion board

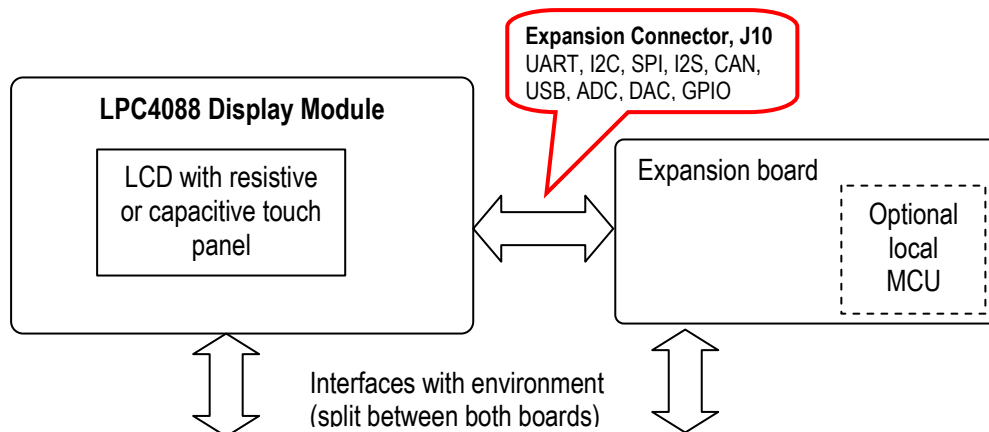


Figure 2 – LPC4088 Display Module Standalone Usage Scenario with Expansion Board

3.2 Hardware Overview

The block diagram below presents the major components of the LPC4088 Display Module.

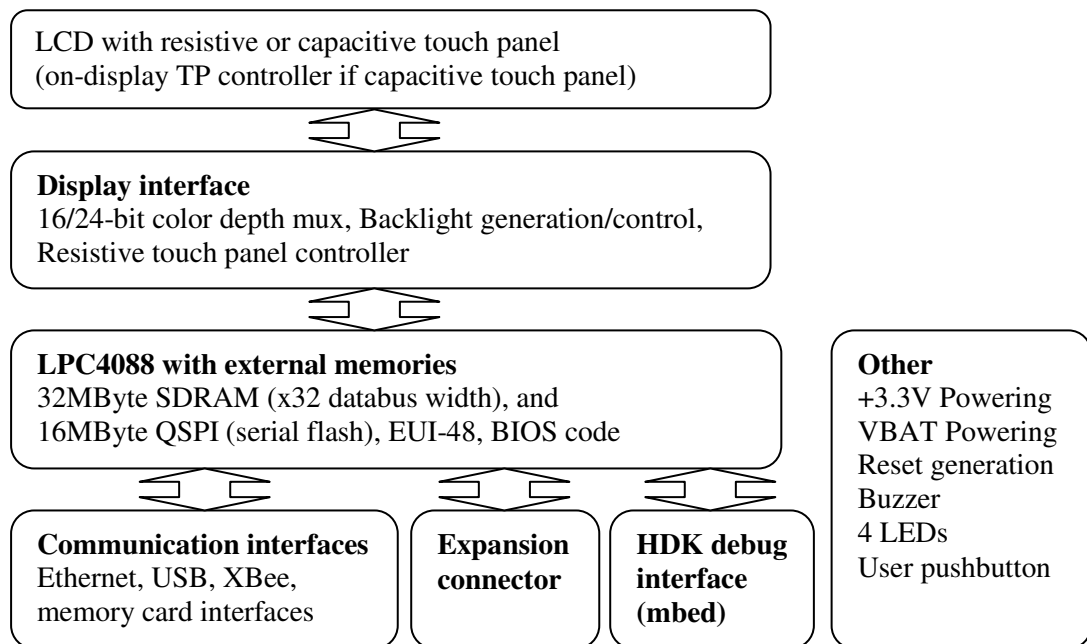


Figure 3 – LPC4088 Display Module Block Diagram

- LPC4088 with external memories**
 In the heart of the design there is the LPC4088 MCU from NXP with associated memory; 32MByte of SDRAM with 32-bit databus access, 16MByte of QSPI serial flash and an Ethernet MAC address (EUI-48). The amount of memory is more than enough for most applications. The LPC4088 runs at up to 120MHz and has a Cortex-M4F core.
- Display interface**
 The display interface is the glue between the LPC4088 on-chip LCD controller and the actual LCD. Both 16- and 24-bit color depth is supported (controlled via a single GPIO pin). 16-bit color depth supports the 5:6:5 RGB-format.
 There is a resistive touch panel controller, AR1021, that can handle the low-level sampling and filtering of the touch panel samples, hence off-loading the LPC4088 from this task.
 There is a DC/DC switched voltage converter that generates a constant current for the display backlight. The current level can be controlled via a PWM signal to allow software to implement backlight dimming.
- LCD**
 Different LCDs are supported. Both resistive and capacitive touch panels are supported. LCDs with capacitive touch panel has on-display touch panel controller.
- HDK debug interface**
 The **mbed Hardware Development Kit (HDK)** is included (which is part of what makes the display module an mbed-enabled module). The HDK offers the following features over a common USB connection (the HDK USB port):
 - USB drag-n-drop programming** of the LPC4088. A USB Mass Storage Device (MSD) is created and the binary image that the compiler creates can just be dragged and dropped on this drive on the PC-side. The image will then automatically be flashed to the LPC4088 on the board.

2. **Virtual serial port**, which creates a virtual COM port on the PC-side. There is a serial (UART) channel on the *LPC4088 Display Module* that can be tunneled over to the PC-side for debug, logging, statistics or other reasons.
 3. **CMSIS-DAP debug interface**, which basically creates an SW debug interface to the LPC4088. The CMSIS-DAP interface is supported by major tool vendors.
- **Communication interfaces**

There are multiple communication interfaces, corresponding to LPC4088 key peripherals.

 - 100/10 Mbps Ethernet interface, including RJ45 connector and ESD protection.
 - USB Host and Device interface, with USB-A and USB-mini B connectors and ESD protection. Note that Host and Device cannot be active at the same time. For a second USB channel, use the expansion connector.
 - XBee compatible connector of RF module expansion.
 - uSD/transflash memory card connector for large memory requirements.
 - **Expansion connector**

Several interfaces are available on the easy-to-connect expansion connector. It is a 2x23 pos, 100 mil pitch pin header (male) that allow easy and quick prototyping on an expansion board.
 - **Other**

The display module also contains a 3.3V linear power supply, VBAT supply including a super capacitor, reset generation, buzzer, 4 LEDs and a user push-button.

3.3 Powering

Before start handling and working with the display module, read the general handling precautions addressed in chapter 2 , especially about ESD precaution.

There are three ways to power the display module, of which the first two are recommended to start with. An external +5V DC supply is needed. 1-3Amp, depending on what is connected to the display module (for example, USB Devices and RF modules that consume current).

- 1) **mbed HDK micro-B USB connector, J9**

This is the natural way to power the display module during program development and initial prototyping. The PC USB interface powers the board. Note that long USB cables can give problems because of too big voltage drops in the cable. Max 1 meter cables with 24 AWG power cords are preferred. Also note that some laptops have lower current limits on their USB host ports. An external, powered USB hub can always be used if there is a problem to directly power the display modules from a USB Host port.
- 2) **Pin 1/2 on main connector J11**

This is the natural way to power the display module when it has been deployed/integrated into the end-product.
- 3) **Power supply on expansion board, via expansion connector, J10**

The external 5V DC supply can be integrated on an expansion board to create a more compact and self-contained solution. The display module is then powered (+5V DC) via the expansion connector, J10.

The picture below illustrated where the three powering options can be found on the display module.

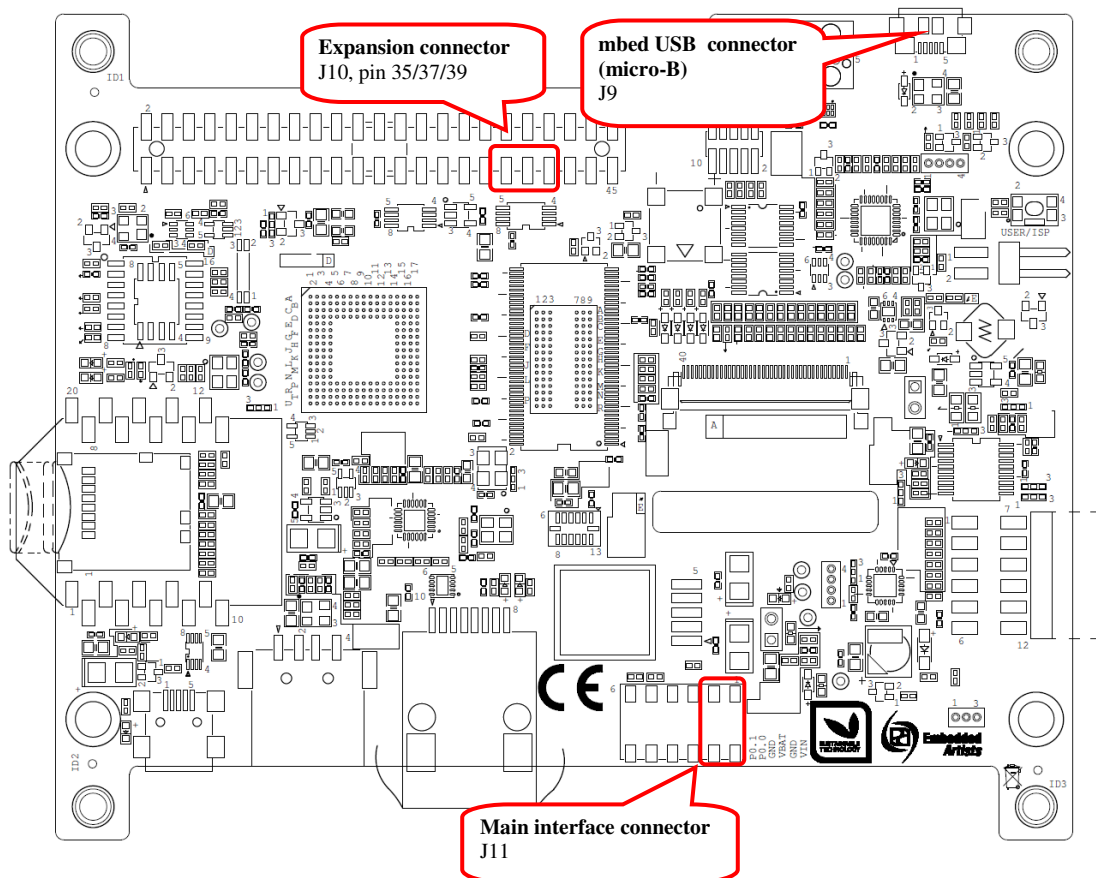


Figure 4 – LPC4088 Display Module Powering

3.4 Demo/Test Application

The *LPC4088 Display Module* is preprogrammed with a demo/test application has been created. Besides demonstrating the features and capabilities of the display module, it can also be used as a troubleshooting application.

3.5 Software Development

When starting program development two initial selections must be done - which program framework to start working in and which tool chain to use. Sometimes the choices are linked together.

Three different framework solutions for the *LPC4088 Display Module* are presented in the following subsections. The different tool chains are presented in each sub-section, since the choices are somewhat different between the frameworks

3.5.1 mbed Framework

The mbed framework[11] is a natural choice for the *LPC4088 Display Module* since it is an mbed-enabled board. The mbed framework is briefly described in this subsection. There is a separate *LPC4088 Display Module Software Framework* document describing all details. There are other software framework alternatives as well. The mbed framework has its strengths and weaknesses, just like any other solution.

The mbed site (<http://mbed.org>) points to four cornerstones in the framework:

- The **mbed Software Development Kit (SDK)**, which is an open source C/C++ microcontroller software platform designed for a suitable level of hardware abstraction to simplify microcontroller programming.

- The **mbed Compiler** is a powerful online IDE that is tightly integrated with the mbed SDK and Developer's Website (mbed.org).
- The **mbed Hardware Development Kit (HDK)**, which is an interface design that provides simple USB drag-n-drop programming and CMSIS-DAP debug interface for the LPC4088 microcontroller. The *LPC4088 Display Module* has an on-board HDK.
- The **mbed.org community**, with extensive documentation in the form of handbooks, cookbooks, project pages, User Forums for getting help and advice from other mbed users, etc.

The mbed Compiler gives you instant access to lightweight C/C++ microcontroller development. It is an online web application IDE (Integrated Development Environment) that you can log in from anywhere and work on Windows, Mac, iOS, Android or Linux. For a quick guide how to write code, compile and download to the *LPC4088 Display Module*, see chapter **Fel! Hittar inte referenskölla..**

The mbed Compiler has some limitations, mainly when it comes to debugging. There are alternatives to the online compiler when more professional program development and debugging is needed. The mbed framework and the application code can be exported from the online compiler and imported into an offline tool chain. For more details, see here: <http://mbed.org/handbook/Exporting-to-offline-toolchains>

Currently there are guides for exporting to:

- uVision
- LPCXpresso IDE
- CodeSourcery
- GCC ARM Embedded
- IAR Embedded Workbench

Under these tool chains it is possible to perform professional debugging with breakpoints, inspecting variable values, dumping memory regions and even trace the program execution flow (provided the debug probe supports the feature). Remember that the *LPC4088 Display Module* has an on-board debug interface so an external debug probe is in general not needed. Most tool chains support the CMSIS-DAP standard now.

3.5.2 LPCOpen Framework

LPCOpen[10] has been created by NXP. It is a platform consisting of a large number of libraries and examples that allows you to get started quickly. The platform uses common APIs that work equally across all NXP microcontroller families. It supports the common development tool chains; uVision from Keil/ARM, Embedded Workbench from IAR, and LPCXpresso IDE from NXP.

There is a core driver library containing three important sections:

- Chip Driver Layer - Peripheral drivers for the microcontroller
- Board Layer - Board specific functions and low level setup code
- Examples - Chip family specific examples for peripherals, interfaces, libraries, etc.

LPCOpen also contains some important larger program packages/libraries, like FreeRTOS, LWIP (TCP/IP stack), USB driver and graphical library - all together with bigger project examples.

Embedded Artists has created an LPCOpen port for the *LPC4088 Display Module*.

3.5.3 Setup Own Framework

There is always the option to roll your own solution and start from scratch to build the needed framework. Most tool chains (LPCXpresso IDE from NXP, uVision from Keil/ARM, Embedded

Workbench from IAR, etc.) can create empty program skeletons. Besides an empty main()-function, only the most important startup code for initializing clocks and the very basic functions in the microcontroller is given.

This approach is definitively a possibility and might be a preferred option if for example there is legacy code to be ported to the new platform. The approach will require more reading, and understanding, of the LPC4088 User's Manual from NXP.

3.6 Program Download

This section describes some basic methods for downloading application code to the LPC4088. Note that this section does not describe how to create the application code (write the program and compile it). It is assumed that a binary file exist that represent the application program.

Most program development environments create files that also contains debug information (*.elf file or *.axf file). This way, the integrated debuggers have the information needed when debugging. When these files are stripped from the debug information and just the pure binary file is left, it is often also called hex-file, which is a file format that Intel created a long time ago. It can also be a pure binary file (which is typically called a bin-file).

Note that the on-line mbed compiler create pure binary files without debug information (since this environment does not support debugging in the traditional meaning).

There are four basic methods for program download (the top two methods are the recommended that will give a quick start since it is support by on-board functionality):

- **On-board HDK: USB drag-n-drop**
The on-board HDK (see section 4.2 for more information) creates a USB drag-n-drop programming function. A USB Mass Storage Device (MSD) is created and the binary image that the on-line mbed compiler creates can just be dragged and dropped on this drive on the PC-side. The image will then automatically be flashed to the LPC4088 on the board. Note that any binary file can be dragged and dropped - and programmed. It is not only the on-line mbed compiler that can generate binary files. Normally, however, the generated files also contains debug information. **These files must be stripped to generate a binary file** that can be used. See the documentation for the used program development environment for details.
- **On-board HDK: CMSIS-DAP Interface (SWD/JTAG)**
The on-board HDK (see section 4.2 for more information) also creates a CMSIS-DAP debug interface, which basically creates a debug interface to the LPC4088. The CMSIS-DAP interface is supported by major program development tool vendors. This interface is not the fastest available (due to limitation of the LPC1114, the MCU that implements the HDK), but it works very well for hobbyists and semi-professional users. For higher download speed, needed for professional use, an external debug probe is recommended. See next alternative for more details.
- **SWD/JTAG via External Debugger**
The *LPC4088 Display Module* also offers direct access to the Serial Wire (SW) debug interface of the LPC4088 MCU. Any external debug probe that supports the LPC4088 and the 10-pin SW interface, called the Cortex Debug Connector, can be connected to this interface. The cheap, 15 EUR, LPC-Link 2 debug probe directly supports the 10-pin connector creates a powerful, professional quality debug interface. The first picture below illustrates how LPC-Link2 is directly connected to the Cortex Debug Connector on the *LPC4088 Display Module* pcb. The second picture illustrates when the Tag-Connect™ pads are used instead. A special cable (from Tag-Connect) is used.

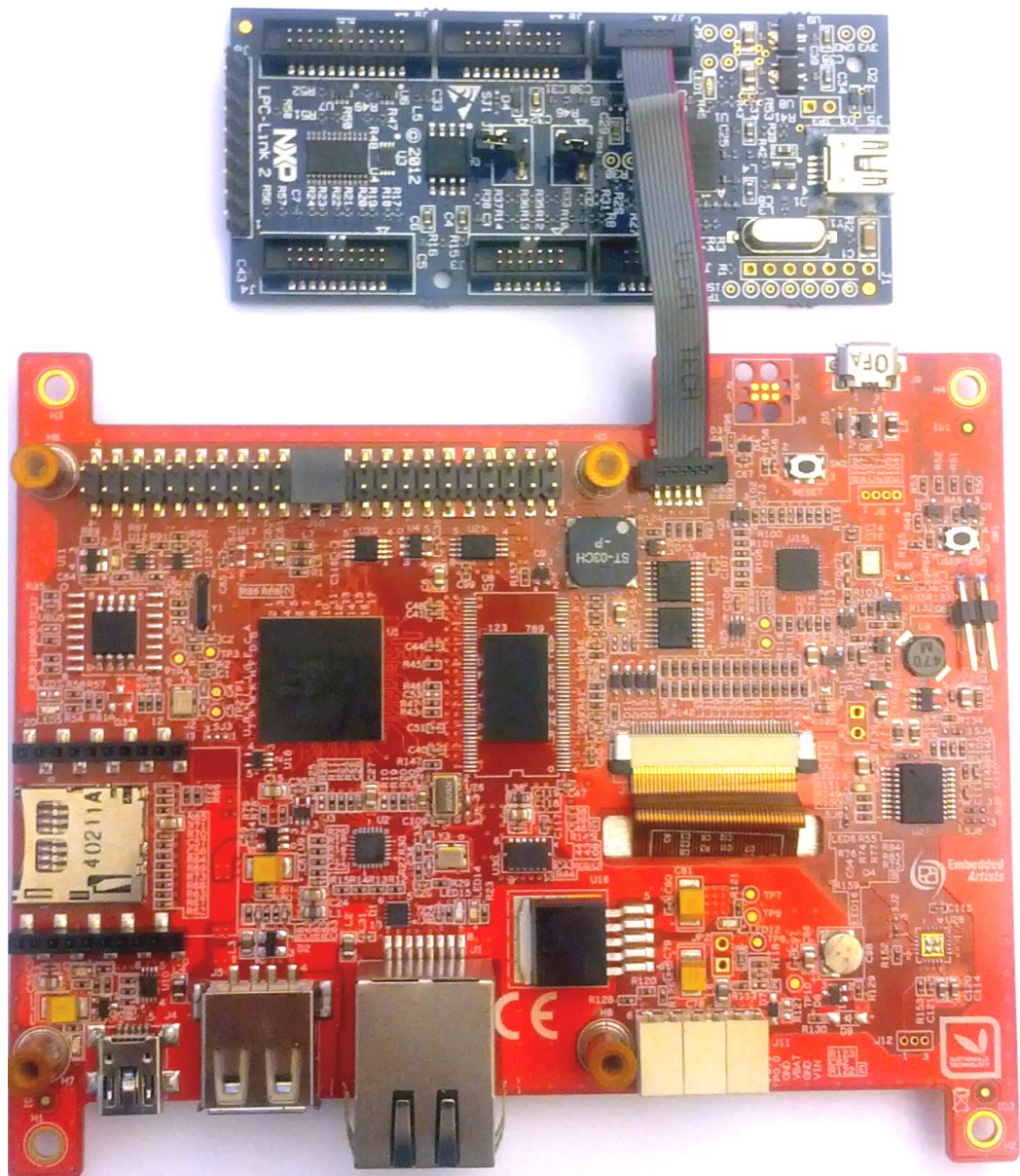


Figure 5 – LPC-Link2 Connected To LPC4088 Display Module

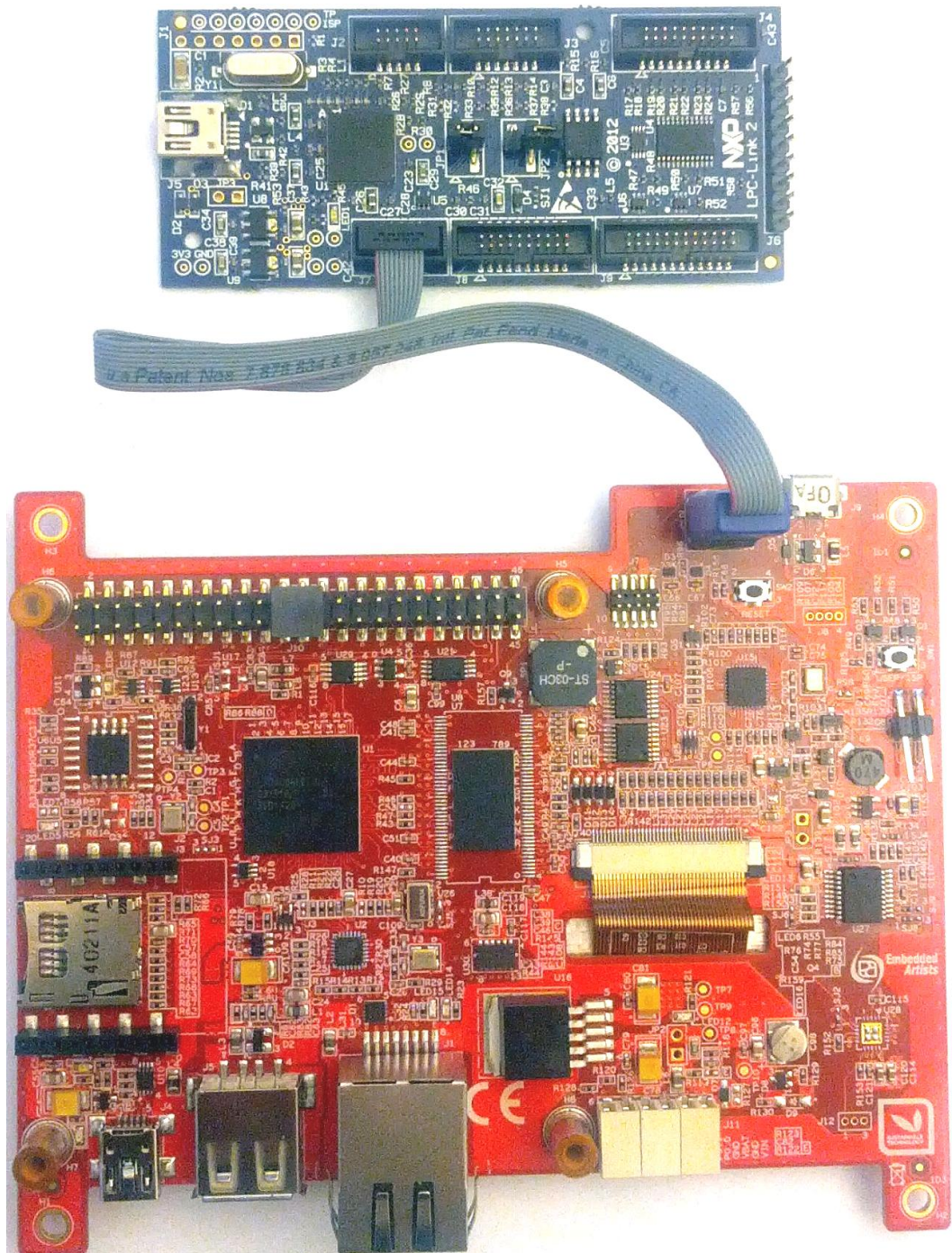


Figure 6 – LPC-Link2 Connected To LPC4088 Display Module via Tag-Connect Cable

There are many other SWD/JTAG interfaces on the market, for example ULINK and ULINK-Pro from Keil, JLINK from Segger, I-Jet from IAR. There is also OpenOCD, which is an open source project. Consult the respective manual for the SWD/JTAG interface used to get instructions how to download a hex/binary file via SWD/JTAG.

Using this method, the application can be downloaded to (internal) SRAM, to (internal) flash memory or (external) qspi flash. Note that the debugger needs special download functionality in order to download to external qspi flash memory. Check the supported functions in the program development environment/IDE that is used.

- **ISP over UART**

ISP is short for In-System Programming. The LPC4088 contains a bootloader in ROM that is activated by pulling pin P2.10 low during reset. This is easily accomplished by pressing the user pushbutton during reset/power-up. UART0 pins P0.2 (U0_TXD) and P0.3 (U0_RXD) are then used to download the program. An application is needed on the PC for downloading the application code, for example Flash Magic (<http://www.flashmagictool.com/>).

This method is only included to make the list of options complete, but it is **not recommended** since the other options are simpler and faster alternatives. Also note that the ISP bootloader cannot program the on-board qspi flash.

It is possible to use the HDK virtual COM port to communicate with the UART channel on P0.2/P0.3 (called HDK_MBED_TX and HDK_MBED_RX, respectively, in the schematic).

3.7 How to Debug

The debug options are given by the development environment selected. If mbed is selected as framework and the on-line compiler is used, the debug options are very few - basically only printf()-style and flashing LED debugging. This is not acceptable for professional users. If the mbed project is exported to an external development environment, for example the LPCXpresso IDE, then the full debug options are available (like breakpoints, watch-points, etc).

The on-board HDK implements a CMSIS-DAP debug interface that is supported by most development tool vendors. It is not as fast as an external debug probe would be but still acceptable for many applications. For more information, see section 3.6 .

4 Connectors

This chapter contains information about all connectors on the display module. The picture below locates the connectors that are described in the subsequent sections.

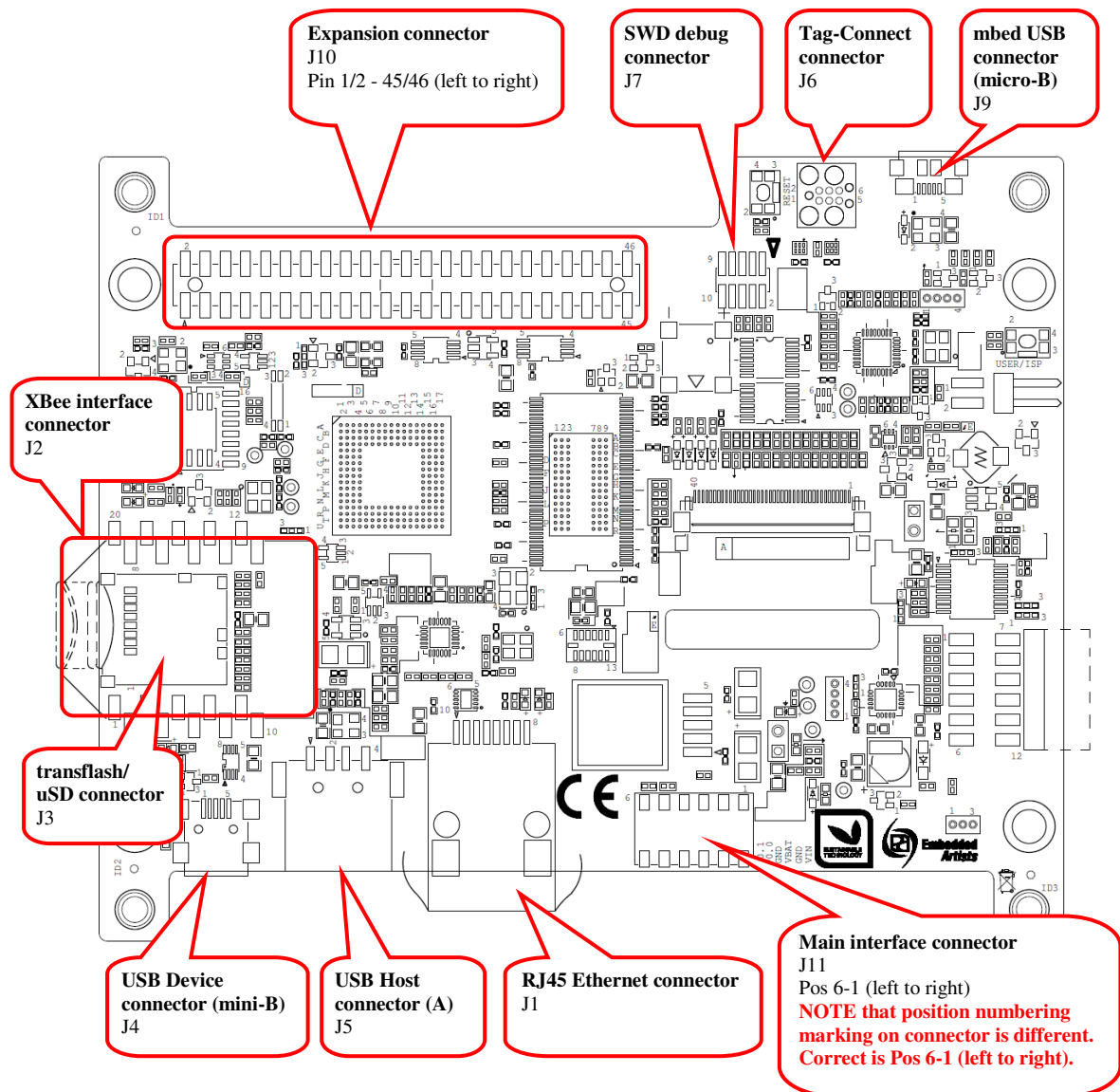


Figure 7 – LPC4088 Display Module Connectors

4.1 Main Interface Connector, J11

This is the main interface connector for the board. It has six positions according to table below. **NOTE that position numbering marking on connector J11 is not correct. See Figure 11 above for correct numbering.**

Position	Usage
1	Vin - external +5V supply. Consult the display module datasheet for details about valid voltage range, current consumption and allowed voltage ripple/noise. Depending on configuration and application, current consumption can vary between

	1-3 Amp. Expansion board, LCD backlight, RF module, Ethernet and USB Host interfaces can all make significant contribution to the current consumption. A Schottky series diode on the display module protects against reverse polarity.
2	GND - ground for Vin supply
3	VBAT - optional external supply for LPC4088 real-time clock (RTC). Consult the display module datasheet for details about valid voltage range and current consumption. If no external supply exist, leave position open. Note that an external supply is not (trickle) charged from the board internal.
4	GND - ground for VBAT supply
5	P0.0 - connected via 100 ohm series resistor to pin P0.0 on the LPC4088 MCU. 0-3.3V logic, 5V tolerant. Functionality controlled by firmware running on LPC4088. <ul style="list-style-type: none"> • Can be UART-TXD (external transceiver needed for RS232 compatibility). • Can be CAN-RD (external CAN transceiver needed). • Can be I2C-SDA (on-board 5K pull-up resistor). • Can be general GPIO.
6 (closest to RJ45 Ethernet connector, J1)	P0.1 - connected via 100 ohm series resistor to pin P0.1 on the LPC4088 MCU. 0-3.3V logic, 5V tolerant. Functionality controlled by firmware running on LPC4088. <ul style="list-style-type: none"> • Can be UART-RXD (external transceiver needed for RS232 compatibility). • Can be CAN-TD (external CAN transceiver needed). • Can be I2C-SCL (on-board 5K pull-up resistor). • Can be general GPIO.

J11 is a quick and reliable wire-to-board connector from AVX, series 9276 (<http://www.avx.com/docs/Catalogs/9276.pdf>). Solid or stranded wires between 18-24 AWG are inserted directly into the connector. The table below lists the 18-24 AWG equivalents in inch, mm and mm².

AWG	Diameter in inch	Diameter in mm	Area in mm²
18	0.040 inch	1.02 mm	0.823 mm ²
20	0.032 inch	0.81 mm	0.512 mm ²
22	0.025 inch	0.64 mm	0.326 mm ²
24	0.020 inch	0.51 mm	0.205 mm ²

The absolute maximum diameter of a wire is 1.2 mm. The maximum diameter for the insulator is 2.1 mm.

The picture below (from the AVX connector's datasheet) illustrates how to insert a wire. The strip length (removed insulation) shall be 4-5mm. Stranded wires shall be twisted together before insertion.

Series 9276

WIRE ASSEMBLY

FOR FURTHER DETAILS REFER TO APPLICATION NOTES 201-01-127

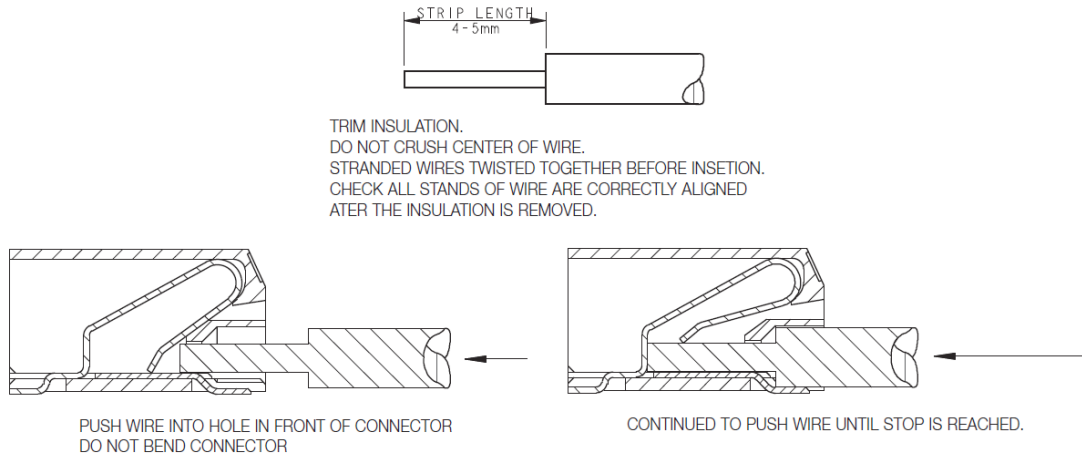


Figure 8 – AVX series 9276 Wire Assembly

When removing a wire, a small bladed extraction tool (1.6 mm wide and 0.5 mm thick) is inserted in the slot above the cable to release it. AVX sells this tool with part number: 06-9276-7001-01-000. The picture below illustrates the process of wire extraction.

Note that the AVX connector is design for up to 5 insertions/extractions per wire position.

WIRE EXTRACTION

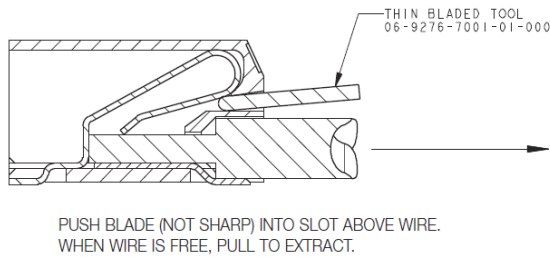


Figure 9 – AVX series 9276 Wire Extraction

4.2 mbed micro-B USB Connector, J9

J9 is a micro-B USB connector to the on-board **mbed Hardware Development Kit (HDK)**. This is part of what makes the display module an *mbed-enabled*. The HDK offers the following features over a common USB connection (J9):

4. **USB drag-n-drop programming** of the target processor (LPC4088). A USB Mass Storage Device (MSD) is created and the binary image that the compiler creates can just be dragged and dropped on this drive on the PC-side. The image will then automatically be flashed to the LPC4088 on the board.
5. **Virtual serial port**, which creates a virtual COM port on the PC-side. There is a serial (UART) channel on the *LPC4088 QuickStart Board* that can be tunneled over to the PC-side for debug, logging, statistics or other reasons.
6. **CMSIS-DAP debug interface**, which basically creates an SW debug interface to the LPC4088. The CMSIS-DAP interface is supported by major tool vendors.

The HDK functionality is implemented in a separate microcontroller (LPC11U35). It communicates with the PC over a USB interface and with the LPC4088 via the (Cortex) debug interface. The picture below illustrates the connections and also the physical place of the HDK microcontroller.

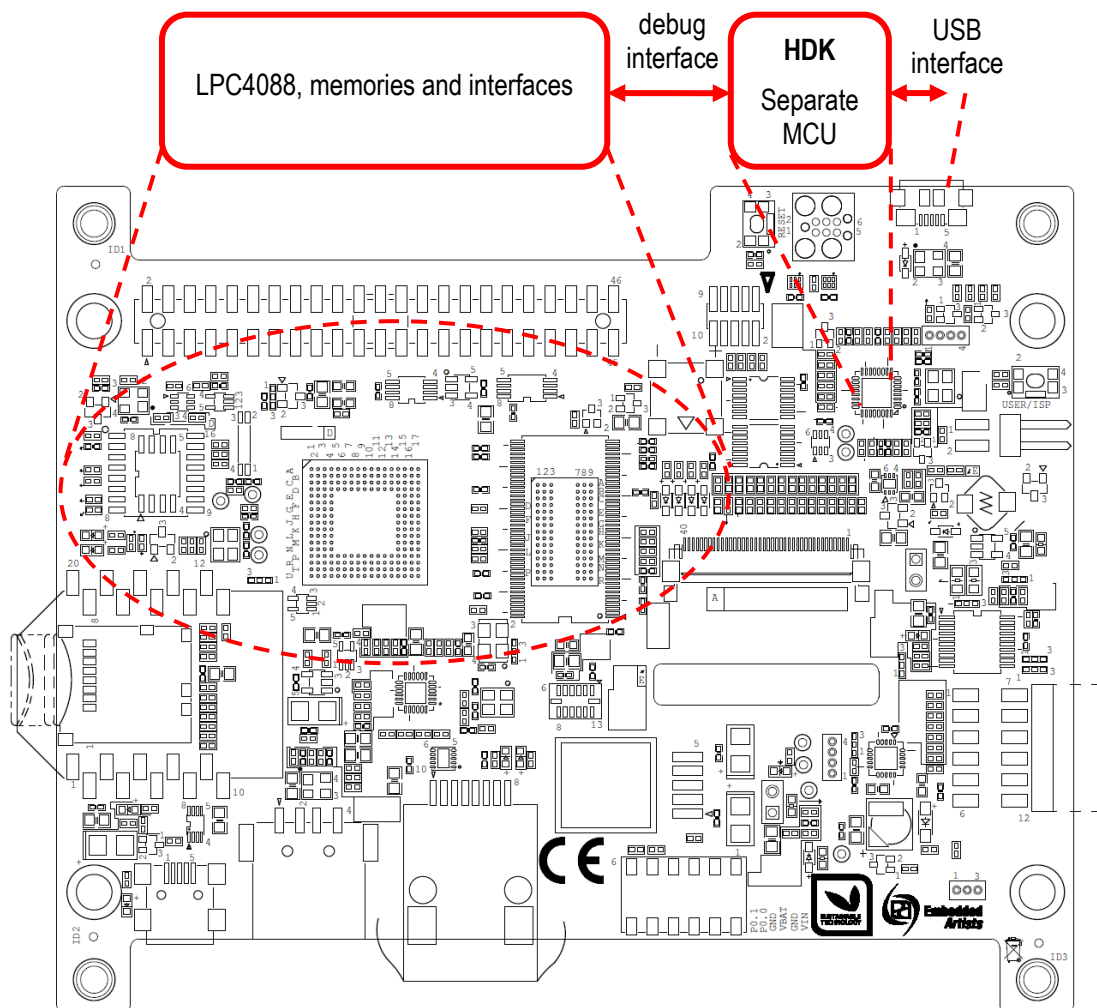


Figure 10 – LPC4088 Display Module and HDK

The micro-B USB connector signals are ESD protected.

Note that the mbed USB connector can be used for powering during development but it is **not suitable for being the permanent powering method** for integrated display modules.

4.3 Debug Interface, J7

J7 is a 2x5 position, 50 mil pitch, standard Cortex Debug Connector as defined by ARM. Debugging with the LPC4088 defaults to JTAG. Once in the JTAG debug mode, the debug tool can switch to Serial Wire Debug mode. For details about the debug and trace capabilities of the LPC4088, see the chip user manual from NXP.

Note that the on-board debug interface (mbed HDK) must not be powered when using the debug interface (J7) for accessing the LPC4088 MCU. This is accomplished by removing resistor R98.

The table below lists the pin functions on the 2x5 pos connector.

Pin	SWD mode	JTAG mode	Notes
1	VCC	VCC	3.3V logic
2	SWDIO	TMS	33K ohm pull-up resistor and ESD protected.
3	GND	GND	
4	SWCLK	TCK	33K ohm pull-down resistor and ESD protected.
5	GND	GND	
6	SWO	TDO	33K ohm pull-up resistor and ESD protected.
7	Key - NC	(RTCK)	This position is sometimes a key-position, meaning that the pin shall not exist on the connector.
8	NC	TDI	33K ohm pull-up resistor and ESD protected.
9	GND detect	GND detect	Can be used by debug probe to detect that an MCU is connected.
10	NRST	NRST	15K ohm pull-up resistor and ESD protected.

Any debugger supporting the LPC4088 MCU and is compatible with the ARM's Cortex Debug connector standard will work. One example is the low-cost LPC-Link2 debug interface.

4.4 Tag-Connect Debug Interface, J6

J6 is a 6 position Tag-Connect pad according to the TC2030-CTX Plug-of-Nail™ format. For details see here: <http://www.tag-connect.com/TC2030-CTX>

The cable connects to the display module pcb (J6 pads) in one end and the other end has a 2x5 pos, 50 pitch, female IDC-style connector. Many debuggers has this interface connector, for example the low-cost LPC-Link2 debug interface and ULINK2 from Keil/ARM.

Note that the on-board debug interface (mbed HDK) must not be powered when using the Tag-Connect debug interface (J6) for accessing the LPC4088 MCU. This is accomplished by removing resistor R98.

J6 is connected in parallel with the debug interface connector, J7, see table below.

TC2030 pad	Signal	Debug Connector, J7 pin
1	VCC	1
2	SWDIO/TMS	2
3	NRST	10
4	SWCLK/TCK	4
5	GND	3, 5, 9
6	SWO/TDO	6

4.5 RJ45 Ethernet Connector, J1

J1 is an 8 position RJ45 Ethernet connector with tab direction; down. The connector does not have any LEDs. These are instead located on the pcb, just behind the connector.

The Ethernet interface supports 10/100 Base-T applications, i.e., 10/100 Mbps speeds over twister pair cables with RJ45 termination in both ends.

The Ethernet interface has ESD protection.

4.6 USB Host/Device Connectors, J5/J4

The *LPC4088 Display Module* has a USB interface that can be configured as a USB Host or USB device interface. Each interface has a separate connector; J5 for USB Host (upright USB-A connector) and J4 for USB Device (mini-B connector). Only one of the can be used at a time because they are connected in parallel over the data and VBUS signals.

The USB interfaces have ESD protection.

4.7 RF Module Interface Connector, J2

J2 is an XBee™ compatible interface connector (dual 10 pos, 2.0mm pitch female connectors). 7 signals are directly connected to the LPC4088 MCU. 2 LEDs connect directly to the connector for status signaling.

The XBee form factor is used on many RF-modules with radio technologies including (but not limited to): ZigBee, WiFi, BT, BTLE, general mesh networks, general ISM, etc.

J2 pin	(Typical) RF-module usage	LPC4088 MCU connection
1	+3.3V supply, decoupled.	
2	DOUT from RF-module.	P4_23, typically a UART-RXD signal.
3	DIN from RF-module.	P4_22, typically a UART-TXD signal.
4	CD from RF-module.	P4_18 on the LPC4088.
5	RESET to RF-module (active low).	P4_17 on the LPC4088.
6	Connected to LED6.	
9	DTR from RF-module.	P4_19, typically a UART-DTR signal.

10	Ground	
12	CTS from RF-module.	P4_20, typically a UART-CTS signal.
13	Connected to LED5.	
16	RTS from RF-module.	P4_21, typically a UART-RTS signal.