

UM11192

LPCXpresso54S018M Board User Manual

Rev. 1.0 — 13th February 2019

User manual

Document information

Info	Content
Keywords	LPC54S018M-EVK, LPC54S018J2M, LPC54S018J4M, LPC54018J2M, LPC54018J4M
Abstract	LPCXpresso54S018M User Manual



Revision history

Rev	Date	Description
1.0	20180204	Initial revision

Contact information

For more information, please visit: <http://www.nxp.com>

For sales office addresses, please send an email to: salesaddresses@nxp.com

1. Introduction

The LPCXpresso™ family of boards provides a powerful and flexible development system for NXP's LPC Cortex®-M family of MCUs. They can be used with a wide range of development tools, including NXP's MCUXpresso IDE. The LPCXpresso54S018M development board has been developed by NXP to enable development with the LPC54S018JxM and LPC54018JxM family devices. It is very similar to the LPCXpresso54S018 design, with only a few minor modifications to accommodate the differences between these two series of devices. The board uses an LPC54S018J4M in a BGA180 package.

The LPCXpresso54S018M can provide CAN-FD support when combined with the OM13099 CAN-FD shield or using external transceivers connected via the available expansion site connectors.

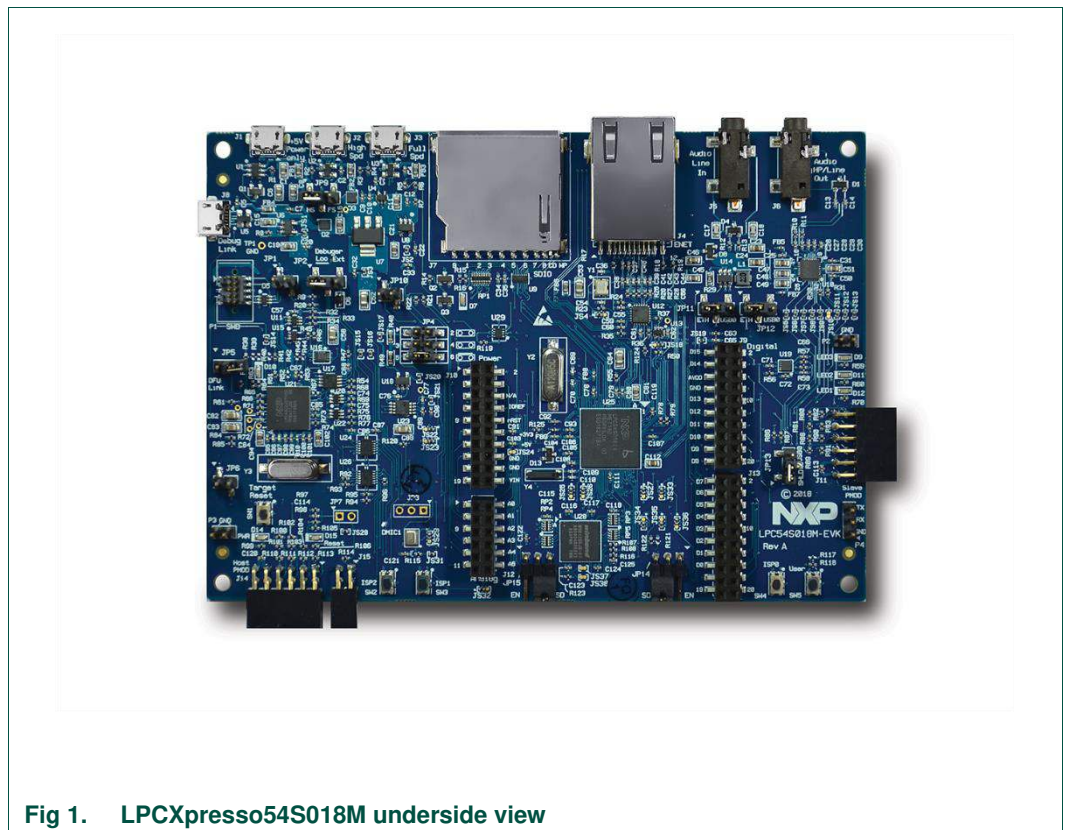


Fig 1. LPCXpresso54S018M underside view

The following aspects of interfacing to the board are covered by this guide:

- Main board features.
- Setup for use with development tools.
- Board interface connectors.
- Jumper settings.

2. Feature summary

The LPCXpresso54S018M board includes the following features:

- On-board, high-speed USB based, Link2 Debug Probe with CMSIS-DAP and SEGGER J-Link protocol options:
 - Link2 probe can be used with on-board LPC54S018J4M or external target.
 - UART and SPI port bridging from LPC54S018J4M target to USB via the on-board Debug Probe.
 - Support for external Debug Probe.
- 3 x user LEDs
- Target Reset, ISP (3) and user buttons
- Expansion options based on popular standards:
 - Arduino UNO compatible expansion site with additional LPCXpresso V3 standard connections
 - PMod™ compatible expansion port
 - Host connection / general purpose expansion port
- On-board 3.3V regulator with external power supply options.
- Built-in power consumption measurement for target LPC54S018J4M MCU.
- Winbond 128Mb W9812G6JB-6I SDRAM
- Knowles SPH0641LM4H digital microphone.
- Full size SD/MMC card slot.
- NXP MMA8652FCR1 accelerometer.
- Stereo audio codec with line in/out.
- High and full speed USB ports with micro A/B connector for host or device functionality.
- 10/100Mbps Ethernet (RJ45 connector).
- Rocktech RK043FN02H 272x480 color LCD with capacitive touch screen. Available separately.

2.1 Board layout and settings

This section provides a quick reference guide to the main board components, configurable items, visual indicators, and expansion connectors. [Fig 2](#) shows the layout of the components on the LPCXpresso54S018M board.

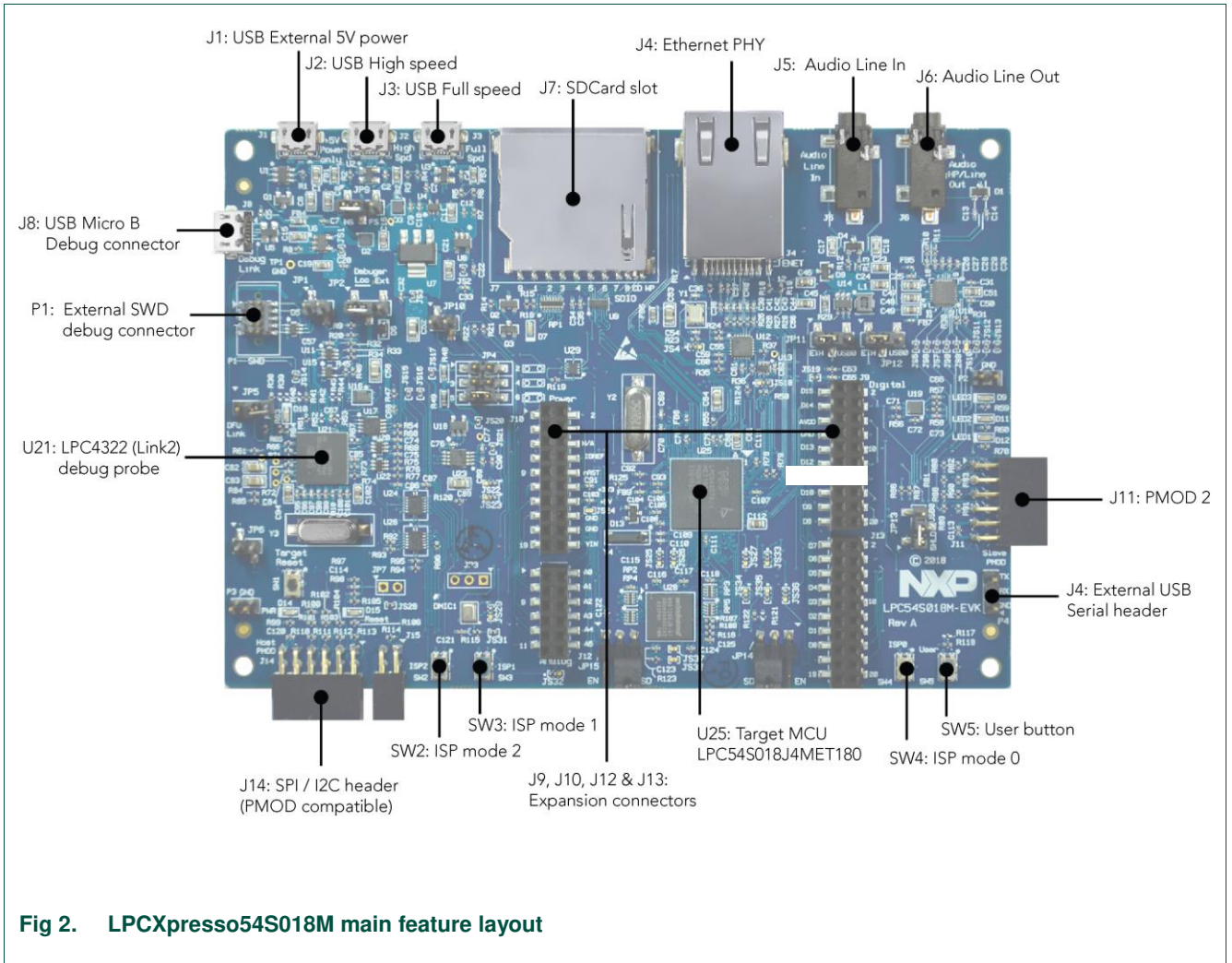


Fig 2. LPCXpresso54S018M main feature layout

The optional LCD panel is mounted on the reverse side of the board, connected to the circuitry via two flex cable connectors. 4 double-sided adhesive pads are provided on the board to attach the LCD. The LCD panel is available for purchase separately - visit <https://www.nxp.com> and search for part number RK043FN02H-CT or contact an NXP distributor to order this panel.

Fig 3 shows the location of indicators and jumpers.

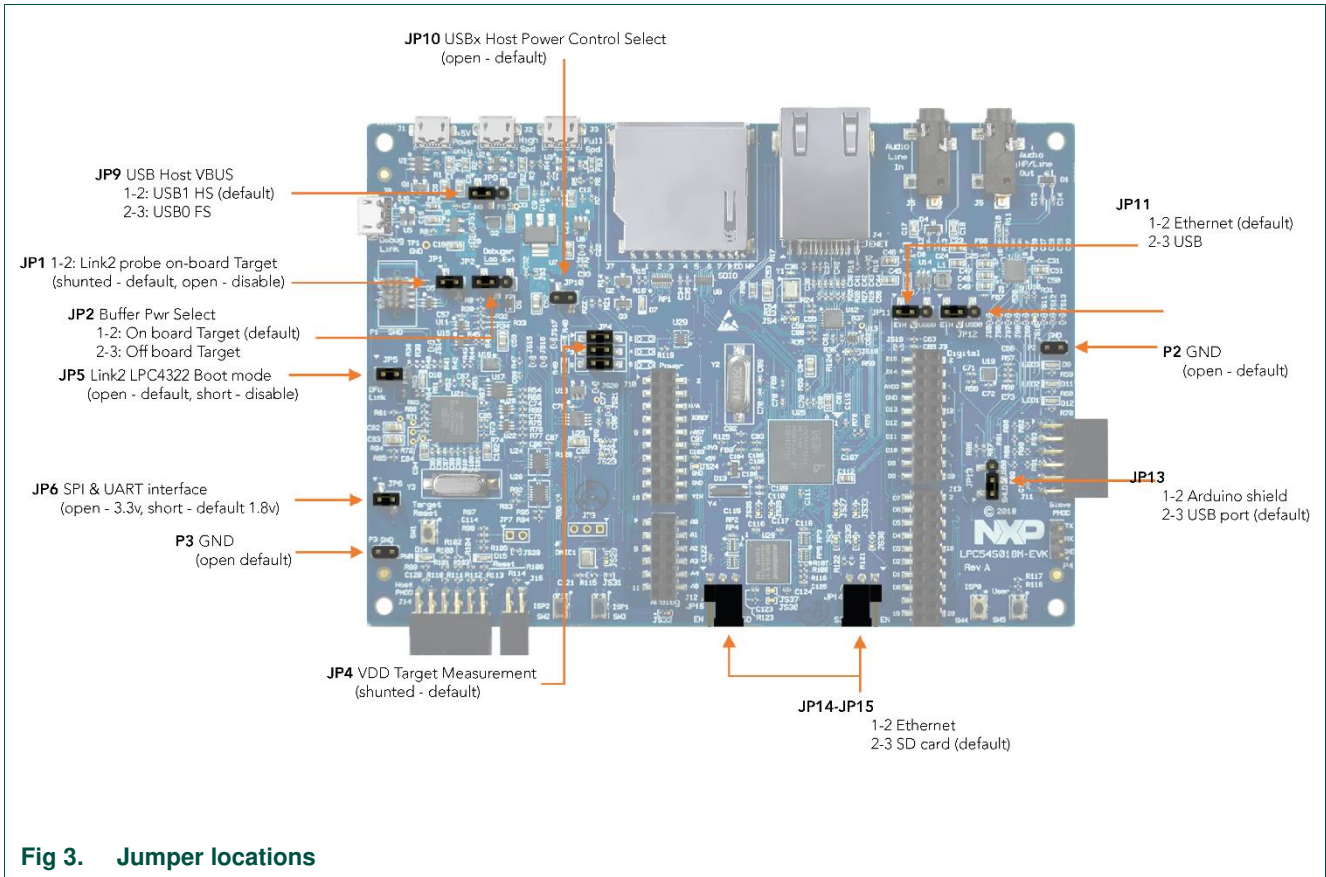


Fig 3. Jumper locations

Table 1 lists the function of each jumper.

Table 1. Jumpers

Circuit ref	Description	Section
JP1	<p>Target processor selection for the on-board Debug Probe.</p> <p>Jumper open (default) the LPC54S018J4M Target SWD interface enabled. Normal operating mode where the Target SWD is connected to either the on-board Link2 Debug Probe or an external Debug Probe.</p> <p>Jumper shunted, the LPC54S018J4M Target SWD interface is disabled. Use this setting only when the on-board Link2 Debug Probe is used to debug an off-board target MCU.</p>	3, 4
JP2	<p>Buffer Power Selection</p> <p>For On-board Target place in position 1-2 (default)</p> <p>For Off-board Target place in position 2-3</p>	3, 4
JP3	<p>This header (not installed by default) provides a convenient connection point to provide external ADC positive and negative voltages. To inject these voltages at this header SJ22 (for VREFN) and/or SJ23 (for VREFP) need to be moved from the default 1-2 position to the 2-3 position.</p>	See Schematic

Circuit ref	Description	Section
JP4	<p>This set of 3 jumpers control various selections for power measurement:</p> <p>Position 1-2 and 3-4 are in parallel with 1 ohm resistors. Current can be measured across these jumper headers to determine current flow into the LPC54S018J4M target.</p> <p>Position 5-6 (installed by default) can be left open and a current meter connected between these pins to directly measure current flow into the LPC54S018J4M target.</p>	6
JP5	<p>Link2 (LPC43xx) force DFU boot – 2 position jumper pins.</p> <p>1) Jumper open (default) for Link2 to follow the normal boot sequence. The Link2 will boot from internal flash if image is found there. With the internal flash erased the Link2 normal boot sequence will fall through to DFU boot.</p> <p>Jumper shunted to force the Link2 to DFU boot mode. Use this setting to reprogram the Link2 internal flash with a new image (using the LPCScript utility) or to use the MCUXpresso IDE with CMSIS-DAP protocol.</p> <p>Note that the LPCXpresso54S018M Link2 flash is pre-programmed with a version of CMSIS-DAP firmware by default.</p>	4
JP6	<p>When open (default), the “Bridge” UART and SPI connections from the Link2 probe are driven to the LPC54S018J4M target.</p> <p>Install JP6 when using the SPI interface at connector J14 and/or FC0 UART at P4 (FTDI). Note that this disables the Link2 SPI and UART (bridge) probe connections.</p>	8.3
JP7 (not installed by default)	<p>JP7 may be fitted to provide a convenient way to enable/disable the reset signal to/from the expansion connectors. Solder jumper JS28 should be removed if JP7 is to be used.</p>	See schematic
JP9	<p>USB host Vbus selection</p> <p>Note that only one of USB0 or USB1 can be configured as a USB host port at any given time (this is a board restriction, not a limitation of the LPC54S018J4M.)</p> <p>Install jumper in position 1-2 for USB1 (High Speed) to provide Vbus (i.e. enable USB host capability) (Default)</p> <p>Install jumper in position 2-3 for USB0 (Full Speed) to provide Vbus (i.e. enable USB host capability)</p>	8.2
JP10	<p>USB host power control selection</p> <p>This jumper selects routing of USB port power and overcurrent detect from either the USB0 or USB1 ports of the LPC546x8/540xx. Note that only one of USB0 or USB1 can be configured as a USB host port at any given time (this is a board restriction, not a limitation of the LPC546x8/540xx.)</p> <p>Leave open when using USB1 (High Speed) as a USB host (Default)</p> <p>Install jumper for USB0 (Full Speed) to provide Vbus (i.e. enable USB host capability)</p>	8.2

Circuit ref	Description	Section
JP11 & JP12	<p>USB0 host / Ethernet TXD/RXD selection</p> <p>Due to sharing of pin functionality on this development board, it is not possible to support the Ethernet port and USB0 overcurrent feature simultaneously. For both JP11 and JP12:</p> <p>Install jumper in position 1-2 to enable Ethernet (Default)</p> <p>Install jumper in position 2-3 for USB0 (Full Speed) (i.e. enable USB host capability)</p>	8.2
JP13	<p>USB0 host / Ethernet selection</p> <p>Due to sharing of pin functionality on this development board, the P4-7 port pin is used for either the USB0 port or as a general purpose signal on the expansion connector.</p> <p>Install jumper in position 1-2 to route P4-7 to the expansion connector.</p> <p>Install jumper in position 2-3 for USB0 (Full Speed) (i.e. enable USB host capability)</p>	8.2, 8.6
JP14	<p>P1_16 Ethernet / EMC selection</p> <p>Due to sharing of pin functionality on the 180 pin LPC54S018J4M package, on this board port pin P1_16 must routed via JP14 to either be used as the Ethernet MDC signal or EMC address line A10. Place in location 1-2 (EN) when using Ethernet or position 2-3 (SD) when using EMC.</p>	
JP15	<p>P1_23 Ethernet / EMC selection</p> <p>Due to sharing of pin functionality on the 180 pin LPC54S018J4M package, on this board port pin P1_23 must routed via JP15 to either be used as the Ethernet MDIO signal or EMC address line A11. Place in location 1-2 (EN) when using Ethernet or position 2-3 (EN) when using EMC.</p>	

Table 2 describes the board LED and button functions, and connectors.

Table 2. LEDs, buttons and connectors

Circuit Ref	Description	Section
D7	<p>SD card slot power enable</p> <p>This LED illuminates when power is enabled to the SD card slot (controlled by LPC54S018J4M port P2-5.)</p>	n/a
D9, D11, D12	<p>User LEDs</p> <p>These LEDs are for application use. They are illuminated when the driving signal from the LPC54S018J4M is low. The LEDs are driven by ports P2-2 (D9), P3-3 (D11) and P3-14 (D12).</p>	n/a
D10	<p>Link2 boot mode</p> <p>Link2 LPC43xx BOOT0_LED indicator. Reflects the state of LPC43xx Link2 MCU P1_1. When the boot process fails, D1 will toggle at a 1 Hz rate for 60 seconds. After 60 seconds, the LPC43xx is reset.</p>	n/a

Circuit Ref	Description	Section
D14	Target power This LED illuminates when the 3.3V supply to the LPC54S018J4M is present.	n/a
D15	Reset LED This LED illuminates when reset is asserted either via the expansion connector or when reset button SW1 is pressed.	n/a
SW1	Reset button Press and release this button to reset the LPC546x8/540xx. Note that this does not reset the Link2 Debug Probe.	n/a

Circuit Ref	Description	Section																																		
SW2, SW3, SW4	<p>ISP / User buttons</p> <p>These switches can be used to force the LPC54S018J4M in to ISP boot modes, as shown below. Note that ISP boot is also affected by OTP bit settings, and behavior will also be modified based on port activity as the boot ROM executes. Refer to the device User Manual for more information. Signal is low when the button is pressed.</p> <table border="1"> <thead> <tr> <th>Mode / Boot source</th> <th>ISP2 (P0-6)</th> <th>ISP1 (P0-5)</th> <th>ISP0 (P0-4)</th> </tr> </thead> <tbody> <tr> <td>Auto boot: If OTP BOOT_SRC not set, LPC54S018J4M will look for a valid image in internal flash, then external SPI flash, then external parallel flash.</td> <td>High</td> <td>High</td> <td>High</td> </tr> <tr> <td>USART/I2C/SPI boot</td> <td>High</td> <td>High</td> <td>Low</td> </tr> <tr> <td>SPI Boot: Boot from SPI NOR flash connected to Flexcomm 9. If no valid image is found, ISP boot will commence. This board does not have on-board SPI flash, but the SPI port signals to support this are available on connectors J9 and J13.</td> <td>High</td> <td>Low</td> <td>High</td> </tr> <tr> <td>Reserved</td> <td>High</td> <td>Low</td> <td>Low</td> </tr> <tr> <td>SPIFI boot: Boot from internal flash memory. If no valid image is found, ISP boot will commence, depending on OTP bit settings.</td> <td>Low</td> <td>High</td> <td>High</td> </tr> <tr> <td>USB0 (full speed) ISP DFU</td> <td>Low</td> <td>High</td> <td>Low</td> </tr> <tr> <td>USB1 (high speed) ISP DFU</td> <td>Low</td> <td>Low</td> <td>High</td> </tr> </tbody> </table>	Mode / Boot source	ISP2 (P0-6)	ISP1 (P0-5)	ISP0 (P0-4)	Auto boot: If OTP BOOT_SRC not set, LPC54S018J4M will look for a valid image in internal flash, then external SPI flash, then external parallel flash.	High	High	High	USART/I2C/SPI boot	High	High	Low	SPI Boot: Boot from SPI NOR flash connected to Flexcomm 9. If no valid image is found, ISP boot will commence. This board does not have on-board SPI flash, but the SPI port signals to support this are available on connectors J9 and J13.	High	Low	High	Reserved	High	Low	Low	SPIFI boot: Boot from internal flash memory. If no valid image is found, ISP boot will commence, depending on OTP bit settings.	Low	High	High	USB0 (full speed) ISP DFU	Low	High	Low	USB1 (high speed) ISP DFU	Low	Low	High			
Mode / Boot source	ISP2 (P0-6)	ISP1 (P0-5)	ISP0 (P0-4)																																	
Auto boot: If OTP BOOT_SRC not set, LPC54S018J4M will look for a valid image in internal flash, then external SPI flash, then external parallel flash.	High	High	High																																	
USART/I2C/SPI boot	High	High	Low																																	
SPI Boot: Boot from SPI NOR flash connected to Flexcomm 9. If no valid image is found, ISP boot will commence. This board does not have on-board SPI flash, but the SPI port signals to support this are available on connectors J9 and J13.	High	Low	High																																	
Reserved	High	Low	Low																																	
SPIFI boot: Boot from internal flash memory. If no valid image is found, ISP boot will commence, depending on OTP bit settings.	Low	High	High																																	
USB0 (full speed) ISP DFU	Low	High	Low																																	
USB1 (high speed) ISP DFU	Low	Low	High																																	
	<p>The ISP pins are sampled by the LPC54S018J4M boot ROM code immediately following reset, so to initiate an ISP boot press and hold the required ISP buttons while pressing and releasing the reset button (SW1.)</p> <p>Following reset, these buttons may also be used by a user application.</p> <p>Note: A board power cycle (holding down ISP required buttons while power is applied) may be required to ensure correct driving of the ISP pins when the EMC bus has been initialized, since these pins are shared with the EMC D2-D3 pins, and the SDRAM chip select may float at reset.</p>																																			
SW5	<p>User button</p> <p>This button is connected to LPC54S018J4M port pin P1-1 and is provided for user applications. Port P1-1 is pulled to ground when the button is pressed.</p>																																			
J1	<p>External +5V power</p> <p>Micro USB connection for power to the LPC54S018J4M target and peripheral circuitry (excluding Link2 Debug Probe).</p>																																			

Circuit Ref	Description	Section
J2	<p>LPC54S018J4M High Speed USB connector (USB1)</p> <p>This micro AB connector enables connection from the LPC54S018J4M USB1 port to host or slave devices. An adaptor (not supplied) is typically required to connect USB slave devices (mouse, keyboard, etc.)</p> <p>Note that when using this USB port as a host, power must be supplied via the J1 connector in order to power the USB device being connected to the board.</p>	
J3	<p>LPC54S018J4M Full Speed USB connector (USB0)</p> <p>This micro AB connector enables connection from the LPC54S018J4M USB1 port to host or slave devices. An adaptor (not supplied) is typically required to connect USB slave devices (mouse, keyboard, etc.)</p> <p>Note that when using this USB port as a host, power must be supplied via the J1 connector in order to power the USB device being connected to the board. Also note that jumpers JP9 through JP13 must be changed from their default position to use this port in host mode.</p>	
J4	<p>LPC54S018J4M Ethernet connector</p> <p>This RJ45 connector provides a 10/100Mbps connection to the Ethernet PHY being driven by the LPC546x8/540xx.</p>	
J5	<p>Audio line input jack</p> <p>3.5mm audio input jack for the audio codec</p>	
J6	<p>Audio line output jack</p> <p>3.5mm audio input jack for the audio code</p>	
J7	<p>SD/MMC card slot</p> <p>Full size SD/MMC card slot connected to the SDIO interface of the LPC546x8/540xx.</p>	
J8	<p>Link2 Debug Probe connector</p> <p>Micro USB type B connection for the on-board Link2 Debug Probe. Note: do not use this connection when using an external Debug Probe.</p>	
J9, J10, J12, J13	<p>Expansion connectors</p> <p>0.1" pitch connectors for addition of Arduino R3 shield or other expansion daughter boards / circuitry.</p>	
J11	<p>Peripheral expansion PMod connector</p> <p>0.1" pitch 2x6 connector following the PMod standards. This connector is primarily intended for adding external peripherals using I²C and/or SPI bus, but is also suitable for general purpose I/O connections.</p>	
J14	<p>Host/peripheral expansion connector</p> <p>0.1" pitch 2x6 connector for host connection / expansion. This connector is primarily intended for connection and external host using I²C and/or SPI bus, but is also suitable for general purpose I/O connections or peripherals.</p>	8.3
J15	<p>Host reset control</p> <p>This connector provides a reset input to the LPC54S018J4M along with ground signals.</p>	8.3
J16, J17	<p>LCD and touch screen display connectors</p> <p>These connectors are dedicated for the LCD and touch screen. Use these connectors to attach the optional LCD panel flex connectors.</p>	n/a

3. Getting Started

The LPCXpresso54S018M board is pre-programmed with a spirit level demonstration and self-test diagnostic program. The spirit level demonstration uses the User LEDs and on-board accelerometer. Connect a micro USB cable from connector J1 to a power source (computer or power supply), place the board on a flat surface, and the 3 User LEDs will all light; tilting the board (rotating about the axis of the horizontal center of the board) will cause the LED pattern to change to illustrate the angle of tilt.

The pre-programmed application also includes a suite of self-test functions. Please see Section 4 for full details of that application.

3.1 Starting a debug session using the on-board (Link2) Debug Probe

By default, the LPCXpresso54S018M is configured to use the on-board Debug Probe (Link2) to debug the on-board target (LPC54S018J4M), using the CMSIS-DAP debug protocol pre-programmed into the Link2 Flash memory. The MCUXpresso IDE or other development tools that support the CMSIS-DAP protocol can be used in the default configuration.

Note that when using the MCUXpresso IDE, the on-board Link2 can also be booted in DFU mode by installing a jumper on JP5; if this is done then the IDE will download CMSIS-DAP to the probe as needed. Using DFU boot mode will ensure that the most up-to-date / compatible firmware image is used with the MCUXpresso IDE. Note that spare jumpers are provided in the board packaging.

NOTE: if the Debug Probe is set up to boot in DFU mode, the USB bridge functions (virtual comm port) and Debug Probe features will not be available if the board is not first initialized by the MCUXpresso IDE.

For further information and tutorial videos please visit the Getting Started tab on the <https://www.nxp.com/demoboard/LPC54S018M-EVK> board page.

3.1.1 Installation steps for use with MCUXpresso IDE

1. Download and install the MCUXpresso IDE.
2. Configure and download an SDK package (with the MCUXpresso IDE tool chain option selected) from the MCUXpresso SDK Builder utility (<http://mcuxpresso.nxp.com>), or use the pre-built example available in the online Getting Started Guide tutorial.
3. Start the IDE, using a new or already created workspace. Drag and drop the SDK package zip file into the Installed SDK panel at the bottom right of the IDE.
4. In the Quickstart panel at the bottom right of the IDE, select "Import SDK example(s)...". Select the board you are using in the dialog that appears, click Next then select example(s) from the demo_apps or driver_examples, then click Finish.
5. Click on the project you imported in the Project Explorer at the left panel in the IDE, then click "Build" in the Quickstart panel. Monitor the Console window at the bottom right panel of the IDE to see the build process complete.
6. Recommended: Install JP5 to force the Link2 Debug Probe to boot in DFU mode before powering the board (see notes above).

7. Ensure jumper JP2 is fitted in position 1-2 (local target powered), and JP1 is not installed (target SWD enabled). These are the default positions set during board manufacture.
8. Connect the LPCXpresso54S018M board to the USB port of your host computer by connecting a micro USB cable to connector J8 (“USB Debug-Link”). The board will boot and run the pre-installed demo.
9. Allow about 10 seconds for the LPCXpresso54S018M devices to enumerate for the first time; the device will appear as “LPC Device” in the host computer system (if booted in DFU mode).
10. In the Quickstart panel of the IDE, click “Debug”. The IDE will search for the board; once a dialog appears showing the board has been found, click OK to connect and start the debug session.
11. If the first attempt to debug a project fails in the IDE, cancel the debug session and reprogram the board. On some machines the drivers take longer to enumerate for the first time, so these steps should correct this issue.

The board is now ready to be used with the MCUXpresso SDK examples in the package you have loaded. For more information on the MCUXpresso tool suite, including video tutorials, please visit <https://www.nxp.com/mcuxpresso>.

When the board has been running applications that use the EMC port, it is recommended to force the LPC54S018J4M target into a known state by performing an ISP boot before attempting to run your first example code. This can be achieved by pressing and holding down the ISP0 button while applying power to the board.

3.1.2 Installation steps to use Keil and IAR tools

1. Download and install LPCScript or the Windows drivers for LPCXpresso boards (<http://www.nxp.com/lpcutilities>). This will install required drivers for the board.
2. Ensure JP5 is open to force the Link2 Debug Probe to boot from internal flash.
3. Ensure jumper JP2 is fitted in position 1-2 (local target powered), and JP1 is not installed (target SWD enabled). These are the default positions set during board manufacture.
4. Connect the LPCXpresso54S018M board to the USB port of your host computer, connecting a micro USB cable to connector J8 (“USB Debug-Link”). The board will boot and run the pre-installed demo.
5. Allow about 10 seconds for the LPCXpresso54S018M devices to enumerate for the first time. It is not necessary to check the Hardware Manager, however if this is done there will be five devices; four under Human Interface Devices (CMSIS-DAP, LPC-SIO, two HID Compliant Devices, and a USB Input Device) and one under Ports (LPC-LinkII Ucom.)
6. Your board is now ready to use with your 3rd party tool. Follow the instructions for those tools for using a CMSIS-DAP probe.

When the board has been running applications that use the EMC port, it is recommended to force the LPC54S018J4M target into a known state by performing an ISP boot before attempting to run your first example code. This can be achieved by pressing and holding down the ISP0 button while applying power to the board.

3.2 Starting a debug session using an external Debug Probe

Code running on the LPC54S018J4M target can be debugged using an external Debug Probe that conforms to the standard ARM debug connector. To use an external Debug Probe connect the probe to the SWD connector (P1) and connect power via the micro USB connector J1.

Note: The Debug link connector J8 must be left unconnected so that the Link2 Debug Probe is left unpowered and does not contend with the SWD interface signals from the external Debug Probe.

4. Self-test demo application

The LPCXpresso54S018M board is pre-programmed with a spirit level demonstration and self-test diagnostic program. The spirit level demonstration uses the User LEDs and on-board accelerometer (as described earlier in this manual). The self-test application is available for download from the LPCXpresso54S018M board page under the software and tools tab.

4.1 Self-test application and VCOM driver installation

The self-test application uses the VCOM port for control and to report self-test results. For Windows 7 or 8 host platforms, the LPC-Link2 serial port driver must be installed before attempting to use this feature (unless previously installed). If MCUXpresso IDE or LPCScript has been used on the Host system being employed then this driver will already have been installed; if not then please download and install LPCScript from <https://www.nxp.com/LPCScript>. Note that it is not necessary to install these drivers when using Window 10, MacOS or Linux.

4.1.1 Serial terminal setup

The self-test application is controlled via a serial link to a host computer. You may either use the VCOM port or attach a serial to USB cable to header P4 and installing a jumper on JP6.

When using the VCOM option, after completing any necessary VCOM driver installation, connect the board to a host computer over USB using connector J8 on the Board. The Board should enumerate on the host computer as several devices, including a VCOM device called LPC-LinkII VCOM. Note the port number of this device using your host computer device manager (see Section 5.3 for more details).

Use a serial terminal emulation program (such as PuTTY or TeraTerm), open a connection to the VCOM port or USB serial cable and set the baud rate to 115200, 8 bits, not parity and no flow control.

4.1.2 Running self-tests

When the board powers up or reset, the spirit level demo will run. To enter self test mode, press the User button (SW5). The spirit level demo will stop running, the test menu will be displayed on the serial terminal and a prompt to enter the test name will be displayed, as shown in Fig 4. Type the name of the test you wish to run, then press enter. Tests will return to the prompt after running in most cases, but some tests require the User button to be pressed to be terminated. Reset can also be pressed to exit test mode.

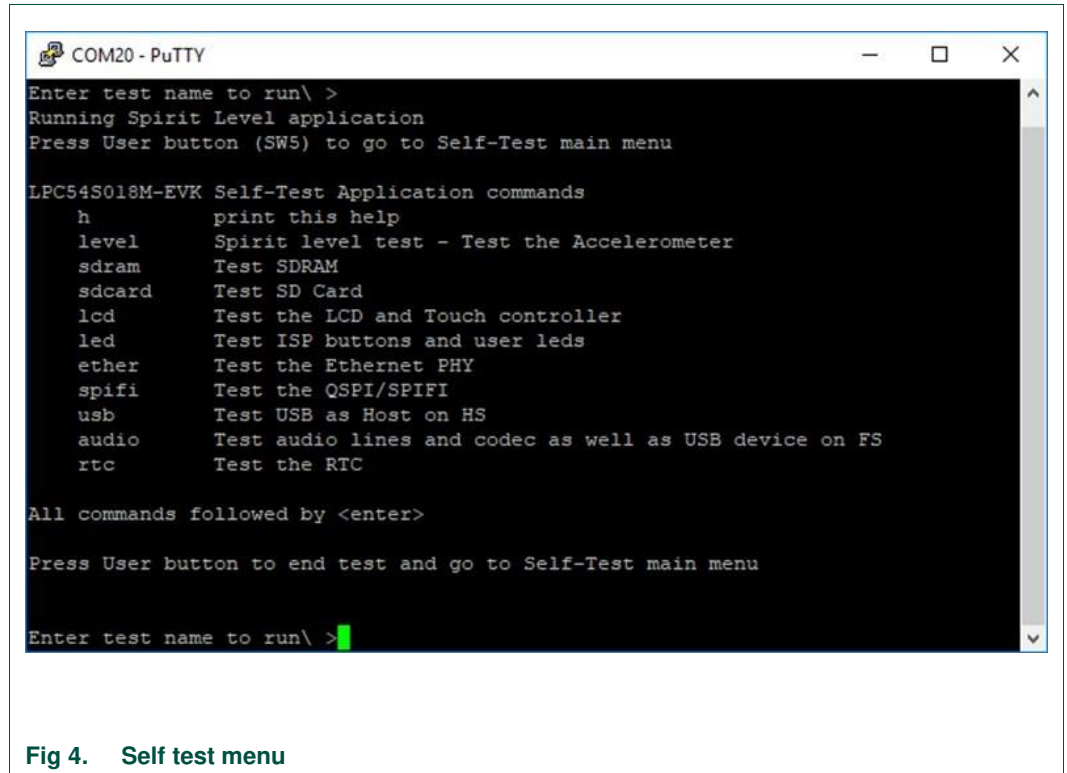


Fig 4. Self test menu

The tests available are as follows:

- level: run the spirit level demo. Press the User button to end this and return to the self-test menu.
- sdram: test on-board sdram. JP14 and JP15 must be set to SDRAM (SD) position for this test to run correctly.
- sdcard: test SD card. Waits for an SD card to be inserted, reads basic card information then performs a read/write test
- lcd: Test the LCD and touch screen (if fitted), but running a simple touch/draw application). Press the User button to exit this test.
- led: Tests the ISP buttons and LEDs. Holding down ISP0 will cause light LED3 to blink, ISP1 for LED2 and ISP0 for LED0. Press the User button to exit this test.
- ether: Checks for presence of an Ethernet host connection then transmits 20 packets. Press User button to exit this test. JP14 and JP15 must be set to Ethernet (EN) position for this test to run correctly. It may be necessary to reset the board to exit this test if a failure occurs.
- spifi: Tests the flash of the LPC54S018J4M.
- usb: USB host test for use with a mouse. JP9 must be set to the 1-2 position and power must be connected to J1. After reporting some diagnostic messages, connect a mouse to J2. The PID and VID of the mouse will be reported, followed by mouse location and button status changes. Press the User button to exit this test.

- audio: Audio codec test. In this test the board will operate as an USB audio device. Connect the host computer to J3 and the will enumerate as a generic audio device. Audio playback from the host computer can be heard by plugging a powered speaker or headphones into J6. Press the User button to exit this test.
- rtc: Performs a test of the RTC crystal. Enter a time for the test to run (in seconds. After the timeout period the test will display a message to confirm the RTC worked correctly.

5. Link2 debug features

This section describes the features provided by the on-board Link2 Debug Probe. See the [Getting Started](#) section for more information on how to program the device.

The Link2 Debug Probe is implemented using an LPC432x MCU (circuit reference U21), which provides a high speed USB port interface to the host computer that runs the development tools. This device is not intended for developer use and should only be used with approved firmware images from NXP. The Link2 on-chip flash memory is factory programmed with a firmware image that supports CMSIS-DAP debug protocol, but also includes several other USB end point functions:

- Virtual COM (VCOM) port: a serial device that can be used with any host computer application design for serial port communication (e.g. Teraterm, puTTY, etc.) Set the terminal program for baud rate to 115200, no parity, 8 bit data, 1 stop bit, no flow control.
- SWO trace end point: this virtual device is used by MCUXpresso to retrieve SWO trace data. See the MCUXpresso IDE documentation for more information.
- Power measurement probe: this virtual device is used by MCUXpresso to retrieve power measurement data from boards that support this feature (e.g. LPCXpresso V3 boards, including the LPCXpresso546x8/540xx.) See Section 9 for more details on this feature.
- I2S/SPI bridges: bridge device from I2C and SPI ports of the LPC54S018J4M target.

All of these devices are independent of each other and of the CMSIS-DAP debug device that is enumerated when the board is connected to a host computer; for example, the VCOM port can be used if the board is running an application when no debugger is running.

In order to correctly install and use the Link2 device on the LPCXpresso54S018M (required for any debugging purpose) for Windows host computers, install the drivers first. These drivers will automatically be installed when MCUXpresso IDE has already been installed. If these IDEs are not being used, it is recommended LPCScript be installed as this also includes the required drivers. All these tools and utilities are available for free download at <https://www.nxp.com>.

The CMSIS-DAP firmware image installed at the factory (and by LPCScript) will uniquely identify itself to the host computer so that more than one board can be connected to that host computer at any time. Some toolchains cannot discern between multiple debug devices; refer to your toolchain documentation for more information (note the MCUXpresso does support multiple LPCXpresso board targets.) It is recommended that LPCScript be used to update the Debug Probe firmware in order to ensure the latest version is being used (see Section 5.2.)

Note: The Link2 only boots when the board is power cycled; the reset button on the board does not reset the Link2.

When using MCUXpresso IDE, the Link2 can be automatically booted with the latest / most appropriate firmware for that IDE version by installing JP5 before powering up the board. This is the recommended approach for the MCUXpresso IDE. Note that if JP5 is installed the VCOM port (and other devices mentioned above) device will not appear until the IDE boots the Debug Probe. The Debug Probe is booted once a debug session is started (that is, the IDE attempts to download code to the target).

5.1 What the Link2 boot LED indicates

LED D10 is the Link2 MCU BOOT0_LED indicator. This LED reflects the state of Link2 MCU pin P1_1. When the boot process fails, D1 will toggle at a 1 Hz rate for 60 seconds. After 60 seconds, the Link2 MCU is reset.

5.2 Programming the Link2 firmware

As mentioned earlier in this section, it is not normally necessary to program the Link2 firmware. However, this can easily be accomplished using the supporting utility, LPCScript.

To program the Link2 Flash the Link2 device (LPC432x) must be in DFU mode. If the Link2 already has a valid image in the flash, it will need to be forced into DFU mode by placing a jumper on JP5, and power cycling (disconnecting then reconnecting power via J8.) Link2 MCU programming is performed using the LPCScript utility (see <http://www.nxp.com/lpcutilities>). After installing the utility, run either of the Program LPC-Link2 with CMSIS-DAP or Program LPC-Link2 with Segger J-link utilities to program the firmware, and follow the on-screen instructions. Further instructions for using the tool are located on the utility web page.

5.3 VCOM port

The identifier of the VCOM port will vary between boards and hosts. To determine the COM port, open the Windows operating system Device Manager. This can be achieved by going to the Windows operating system Start menu and typing "Device Manager" in the search bar. In the device manager look under "Ports"; the LPC-LinkII UCom Port device and its name should be visible. NOTE: this VCOM port will only appear if (a) the Debug Probe has been programmed with the CMSIS-DAP firmware and the Debug Probe DFU link is removed at power up, or (b) if the Debug Probe has been configured for DFU boot (JP5 installed) at power up and MCUXpresso IDE has booted it (by starting a debug session.)

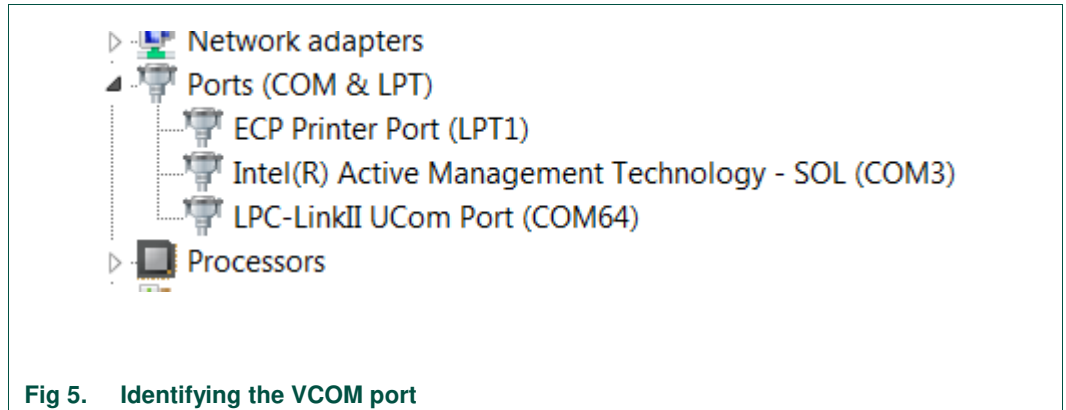


Fig 5. Identifying the VCOM port

If the J-Link firmware image has been programmed into the Debug Probe and DFU boot mode is not being used, then a VCOM device called Jlink CDC UART port will appear instead of the LPC-LinkII UCom port.

5.4 Configuring the LPCXpresso54S018M to debug an external target

The LPCXpresso54S018M board's Link2 Debug Probe may be used to debug an off-board target MCU. The on-board Link2 Debug Probe is capable of debugging target MCU's with a VDDIO range of 1.6V to 3.6V. To keep the on-board target LPC54S018J4M MCU from interfering with the SWD interface, JP1 must be fitted. The Link2 Debug Probe SWD should be connected by a ribbon cable between the P1 connector to the off-board target MCU SWD interface. Power the LPCXpresso54S018M board from the Link USB connector J8, and fit jumper JP2 across pins 2 - 3 (External Target).

6. Troubleshooting debug sessions

In some cases, the ability to make a debug connection can be lost due to the LPC54xxx target running code that does not allow the host debugger system to gain control of it. Some reasons for this can be:

- Application code causing a hard fault
- Application code disabling the SWD debug port pins
- The boot image has an invalid header

In many cases, the board can be put back in a state where the debugger can gain control by using the ISP buttons and pressing and releasing the reset button (for example, holding down ISP0 to force the device into the serial ISP boot mode). If problems persist with ISP mode entry being required, try mass erasing the flash; this can be done using the MCUXpresso IDE device programming utility.

LPC54S018JxM devices do not have any pull-ups enabled at reset, and in cases where an application is using the SDRAM this could cause the EMC_D2 through EMC_D4 signals to be driven by that device (since the P1_12 port of the target could be floating). Since these signals are shared with the ISP0-2 pins, an undesired ISP boot mode could

be inadvertently selected. In these situations, try removing power for several second, then hold down ISP0 reconnecting power to the board.

7. Board power connections & measurement

The LPCXpresso54S018M board requires +5V input to power the on-board voltage regulators which in turn power the Link2 Debug Probe and other +3.3V circuits, the LPC54S018J4M target and other +1.8V circuits, and the Arduino +5V and +3.3V power rails. When the main external power source is from the Link2 side USB micro B-type connector (J8), both the Link side and LPC54S018J4M Target sections of the board are powered. When the main external power is from the power only USB micro B-type connector (J1), or one of more of the USB device ports (USB0 at J3 or USB1 at J2), only the LPC54S018J4M target and supporting devices and peripherals are powered.

7.1 LPCXpresso54S018M current measurement

The LPC54S018J4M current can be measured by measuring the voltage across a sense resistor in series with the supply, a current meter or using the on board current measurement circuit. Each of these methods will be described in subsections below. There is no current monitoring of the Link2 section circuits on the board. The Target side power going to LEDs and support ICs is not monitored by the current measurement circuit. The LPC54S018J4M BGA package has the core and IO power both sourced from the same VDD pins.

When a shield board is attached, attempting to measure the lowest possible power the LPC54S018J4M IO pins must be configured according to how the software has configured the shield board to ensure there is no extra current from the LPC54S018J4M IO ports that have external pull-up or pull-down resistors enabled.

There are several leakage paths through the various devices on the LPCXpresso54S018M board, so this board may not be suitable for accurately measuring the lowest possible leakage current achievable in a target application.

7.1.1 LPC54S018J4M Vsense resistor current measurement

The voltage across a pair of 1Ω resistors in series with the target LPC54S018J4M VDD can be manually measured at JP4 across pins 1 and 4 on the PCB. For a higher range of current a jumper should be installed between pins 3 and 4 of JP4 to short out one of these resistors. The voltmeter positive probe is applied to JP4 pin 1 (see silkscreen labelling) and pin 4. Use Ohm's law to calculate the current (LPC54S018J4M current = measured voltage / 2 Ω). As an example, if the measured voltage is 20 mV, then $20\text{e-}3 / 2 \Omega = 10 \text{ mA}$.

Note: The input current to the MAX9634 used in the on-board current measurement will be included in the voltage measured across this resistor.

7.1.2 LPC54S018J4M VDD current measurement using a current meter

A current meter may be inserted at JP4 between pins 5 and 6 to measure the LPC54S018J4M VDD input current. Note that a jumper needs to be installed in the 5-6 position when an ammeter is not present so power can reach the LPC54S018J4M target.

7.1.3 LPC54S018J4M VDD current measurement

The LPCXpresso54S018M board has an on-board current measurement circuit consisting of a MAX9634T (U18) current monitor chip and a 12-bit ADC (ADC122S021, U23) with a 12-bit sample at 50k to 200ksps. The on-board MAX9634T current monitor measures the voltage across the LPC54S018J4M VDD Vsense resistors; either 2 Ω or 1 Ω if JP4 jumper 3-4 is installed. The MAX9634 multiplies the sense voltage by 25 to provide a voltage range suitable for the ADC to measure. A 2-input analog mux selects between the LPC54S018J4M current monitor and the output off a MAX9634T current monitor chip on an expansion board (with compatible current measurement circuit on-board). The current measurement circuit is controlled by the Link2 processor and is not user programmable. Power measurement utilities to use this feature are available in the MCUXpresso IDE.

Due to input offset voltage variations in the MAX9634, the current measurement circuit is not recommended for measuring current below 150 μA. See Fig 6 as a guideline for measurement error versus measured current.

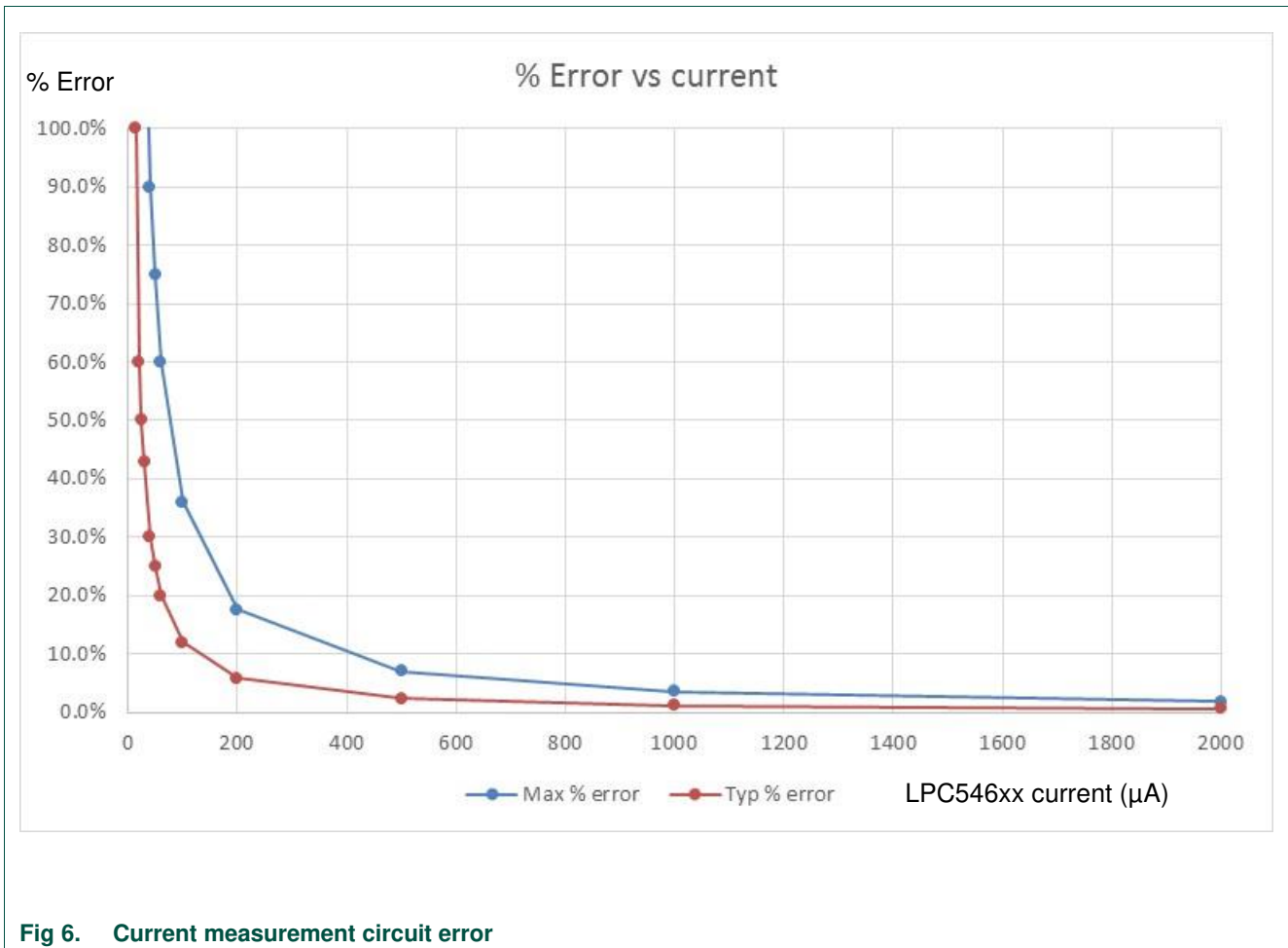


Fig 6. Current measurement circuit error

7.1.4 Shield board current measurement

To use the on-board current measurement circuitry, any expansion board must match the functionality of the LPCXpresso54S018M. Refer to the board schematics for more information.

8. Board peripheral connections

8.1 LPC54S018J4M USART ports and VCOM support

The LPCXpresso54S018M board provides access to USART connections from Flexcomm4 at the expansion connector J13.

On LPCXpresso54S018M boards the USART from Flexcomm0 supports ISP UART mode booting and is used by default for semi-hosting debug messages. This USART is connected to a serial port on the Link2 Debug Probe, which provides a bridging function to USB to a virtual com port (or “VCOM” port) on a host computer connected to the debug link connector, J8. On revision C boards TXD and RXD from this USART are also available at header P4, for an external serial connection to be made. When using the P4 connector, JP6 must be installed; note that when JP6 is installed the VCOM port functionality is disabled.

The factory default CMSIS-DAP Link2 image includes UART bridge functionality (VCOM support), and this firmware is also available with the LPCScript utility, available at <http://www.nxp.com/lpcutilities>. The J-Link firmware image available from LPCScript (version 1.9 onwards) or directly from SEGGER also has this feature (also see Section 5.3.)

8.1.1 P3 USART header

The P4 header is designed to be used with an external serial to USB or other 3.3V serial device. P4 is only present on Revision C (and later) boards. [Table 3](#) shows the pin out information. below. Note that the connections at P4 are made directly to the LPC54S018J4M device, so care should be taken to avoid ESD discharge and voltages outside the ratings of the device I/O pins.

Table 3. P4 connections

Pin	Function
1	Ground
2	RXD (from LPC54S018J4M)
3	TXD (from LPC54S018J4M)

8.2 USB ports

The LPCXpresso54S018M incorporates connectors for the two USB ports of the LPC54S018J4M device – a full speed (USB0) and a high speed port (USB1). Both of these USB ports support device and host functionality, but only one can be used as a host port at any given time. Jumper settings must be set appropriately for host mode selection, as described in this section.

The MCUXpresso SDK for LPC54S018J4M includes USB stacks for both host and device, with several supporting application examples.

8.2.1 USB full-speed port (USB0)

The LPC54S018J4M Full Speed (FS) USB port is connected to micro AB USB connector J3. This section describes functionality support by this board and associated jumper settings for this board.

8.2.1.1 Device mode

The FS port supports USB device mode operation; no jumper settings are needed to configure this mode. When using the port in device mode, a standard micro USB cable can be used.

8.2.1.2 Host mode

The FS port supports USB device mode operation but, due to sharing of some signals and power supply circuitry, requires the following settings:

- Connect external +5V power via USB connector J1. This is required for the FS USB port to be able to power the devices connected to it.
- Install JP10
- For each of JP9, JP11, JP12, JP13 install a jumper in position 2-3.
- Use of a micro AB adaptor to connect an external USB device may be required, as most USB device have either a type A or micro B connector.

Note: The Ethernet and HS USB Host functions are disabled when the board is configured for FS USB Host operation. Also, note that Expansion connector port PIO4-7 is no longer available when using this mode.

8.2.2 USB High Speed port (USB1)

The LPC54S018J4M High Speed (HS) USB port is connected to micro AB USB connector J2. This section describes functionality support by this board and associated jumper settings for this board.

8.2.2.1 Host mode

The HS port supports USB device mode operation but, due to sharing of some signals and power supply circuitry with the FS port, requires the following settings:

- Connect external +5V power via USB connector J1. This is required for the HS USB port to be able to power the devices connected to it.
- Ensure JP10 is not installed (default).
- Install a jumper at JP9 in position 1-2 (default).
- Use of a micro AB adaptor to connect an external USB device may be required, as most USB device have either a type A or micro B connector.

(Note: Although not required for the HS USB port operation, it is recommended to check JP11, JP12 and JP13 have a jumper in position 1-2 to enable Ethernet operation and availability of the I/O at the Expansion connector D8.)

8.3 Host Expansion Header (J14)

This header provides connectivity from the LPC54S018J4M target to a remote Host, peripherals or other devices. [Table 4](#) shows the connections.

Table 4. Host Expansion Header signals

Pin	J14 signal	Flexcomm # / Port
1	GPIO/SPI-SSEL in/out	3 / PIO0-1
2	GPIO/SPI-MOSI in/out	3 / PIO0-3
3	GPIO/SPI-MISO out/in	3 / PIO0-2
4	GPIO/SPI-SCK in/out	3 / PIO0-0
5	GND	
6	VDD (3.3V)	
7	GPIO/INT (out/in)	PIO0-5 (default) or PIO0-4, set by JS32
8	GPIO/RESET (out)	PIO3-13
9	SCL	1 / PIO0-14 Rev C and earlier 1 / PIO0-13 Rev D and later
10	SDA	1 / PIO0-13 Rev C and earlier 1 / PIO0-14 Rev D and later
11	GND	
12	VDD (3.3V)	

This connector shares SPI interface signal connections with the SPI bridge function (i.e. connection from the LPC54S018J4M to the Link2 Debug Probe); only one of these functions may be used at once. By default jumper JP6 is open, connection the Flexcomm 3 SPI port to the Link2 device; install JP6 if using the SPI port on J14.

By default, solder jumpers JS15 and JS16 connect the LPC54S018J4M Flexcomm 1 I²C port to the Link2 device; these may be removed if they will interfere with other devices interfaced to J14. Note there are 2.2K Ω pull-up resistors to 3.3V for these SDA and SCL connections.

8.4 PMod Slave Expansion Header

This header provides connectivity from the LPC54S018J4M target to PMod peripherals or other devices. [Table 5](#) shows the connections.

Table 5. PMod Slave Expansion Header signals

Pin	PMod standard connection	J11 signal	Flexcomm # / Port
1	GPIO/SPI-SSEL out / UART CTS in	GPIO/SPI-SSEL in/out	9 / PIO4-6 (default) or PIO3-30, set by JS27
2	GPIO/SPI-MOSI out / UART TXD out	GPIO/SPI-MOSI in/out	9 / PIO3-21
3	GPIO/SPI-MISO in / UART RXD in	GPIO/SPI-MISO out/in	9 / PIO3-22
4	GPIO/SPI-SCK out / UART RTS out	GPIO/SPI-SCK in/out	9 / PIO3-20
5	GND	GND	

Pin	PMod standard connection	J11 signal	Flexcomm # / Port
6	VCC (3.3V)	VDD (3.3V)	
7	GPIO/INT in	GPIO/INT (out/in)	PIO2-0
8	GPIO/RESET (out)	GPIO/RESET (out)	PIO3-11
9	GPIO/SCL	SCL	2 / PIO3-24
10	GPIO/SDA	SDA	2 / PIO3-23
11	GND	GND	
12	VCC (3.3V)	VDD (3.3V)	

Note that the I²C and SPI ports on J11 are also connected to the Expansion connector J9, and the INT and RESET connections to J12.

8.5 Expansion connectors (including Arduino connectivity)

The LPCXpresso54S018M board includes four expansion connectors (J9, J10, J12 and J13) incorporate an Arduino Uno revision 3 footprint in their inner rows. These connectors provide access to the CAN interfaces, additional digital microphone support signals, I²S, I²C, USART, SPI and GPIO/INT/PWM connections. Note that several of the signals available at these connectors are shared with other connectors or board functions, so may not be usable if those other functions are being used.

[Table 6](#) shows the connector circuit references and purposes.

Table 6. Expansion connectors

Reference	Description
J9	The odd number pins are compatible with Arduino Uno rev3 Digital 15:8, AREF, SDA & SCL connector. The even numbered pins are used for external access and expansion of LPC54S018J4M signals not used by the Arduino Uno rev3 compatible interface.
J13	The odd numbered pins 1 – 15 are compatible with Arduino Uno rev3 Digital 7:0 connector. The even numbered pins, and odd numbered pins 17 and 19, are used for external access and expansion of LPC54S018J4M signals not used by the Arduino Uno rev3 compatible interface.
J10	The even numbered pins 6 – 20 are compatible with the Arduino Uno rev3 Power connector standard. On Revision E and later boards, pins 13 and 15 are used to provide access to Smart Card 0 port (board modifications required to use this feature – see Section 8.7).
J12	The even numbered pins 2 – 12 are compatible with Arduino Uno rev3 Analog connector. The odd numbered pins are used for external access and expansion of LPC54S018J4M signals not used by the Arduino Uno rev3 compatible interface.

Below shows the sharing of signals between the expansion connectors and other connectors or circuit functions. Refer to the board schematics for more details.

Table 7. Expansion Connector signal sharing

Function	Connector	Shared with
I ² C	J9 pin 14 & 15	Accelerometer, PMod Slave Connector, Audio Codec, LCD touch panel controller
SPI	J9 pins 9, 11,13,15	PMod Slave Connector
GPIO D8	J9 pin 19	Jumper (JP13) selection for USB0 FS host port. Only one of USB0 Host mode or GPIO function on this port may be used.
GPIO	J9 pin 6	User LED 3
GPIO	J9 pin 8	User LED 2
I ² S	J9 pins 12, 14, 16, 18, 20	Audio codec. To use an I ² S device connected to these pins solder jumpers JS6 through JS9 and JS11 through JS13 may need to be removed.

8.6 Ethernet port

The LPC54S018J4M incorporates an on-board 10/100 Mbps Ethernet MAC. The LPCXpresso54S018M board includes an on-board LAN8720A-CP PHY and RJ45 jack with integrated link status LEDs.

The MCUXpresso SDK for LPCXpresso54S018M includes drivers, LwIP stack and several examples to show how the Ethernet function can be used.

Due to sharing of pin functionality on the 180 pin LPC54S018J4M package, on this board port pins P1_16 and P1_23 must routed via jumpers JP14 and JP15 respectively to either be used for Ethernet (MDC and MDIO signals) or to EMC address lines A10 and A11. Place in location 1-2 (EN) when using Ethernet or position 2-3 (SD) when using EMC.

8.7 Smart Card support

Smart Card 0 interface connections are routed to J10 pins 13 (P0_20) and 15 (P0_21) via solder jumpers JS38 and JS37. Since these signals are also used for the EMC (SDRAM) address lines A2 and A3, these solder jumpers are left open by default. If these solder jumpers are closed, then care should be taken not to access the SDRAM and thus cause bus contention.

9. On-board memory

9.1 SDRAM

The board includes a Winbond 128Mb SDRAM connected to the 16-bit external memory interface available on the LPC54S018J4M BGA180 part. The MCUXpresso SDK includes drivers that have optimized external memory settings for use of this memory.

Due to sharing of pin functionality on the 180 pin LPC54S018J4M package, on this board port pins P1_16 and P1_23 must routed via jumpers JP14 and JP15 respectively to either be used for Ethernet (MDC and MDIO signals) or to EMC address lines A10 and A11. Place in location 1-2 (EN) when using Ethernet or position 2-3 (SD) when using EMC.

10. Other board features

This section describes board functions not mentioned elsewhere in this document.

10.1 LCD with capacitive touch

The LPCXpresso54S018M has connectors and adhesive pads for a Rocktech LCD to be added by the user. This LCD panel may be ordered from <https://www.nxp.com> (use part number RK043FN02H-CT). When installing the panel it is recommended that an example program that uses the LCD panel (such as the emWIN examples provided in the SDK) is used to test the connections before securing the panel in place with the adhesive pads.

Note: The LCD data lines are shared with the PDM0 interface of the LPC54S018J4M device (the LCD is enabled by default), and solder jumpers may need to be modified to change between these configurations (see section [10.5](#) for more details.)

On-board voltage pump circuitry is provided to drive the LCD backlight using an AP5724WG device. This is enabled using PIO3-31.

The MCUXpresso SDK includes drivers and example applications for the LCD and capacitive touch controller.

10.2 Audio codec

The LPCXpresso54S018M board incorporates a Cirrus Logic (Wolfson) WM8904 audio codec, powered via a dedicated regulator. This codec has both I²C (for control) and I²S (for data) interfaces.

The I²C interface of the codec is routed to Flexcomm 2 of the LPC54S018J4M, the same connection as used for the other I²C devices on the board; the Codec has an address of 0b0011010.

The Flexcomm 6 and Flexcomm 7 interfaces of the LPC54S018J4M are connected respectively to the Codec for data sent to / received from the Codec. These I²S connections from the LPC54S018J4M are also connected to the Expansion connectors (see Section 8.5 for more information); solder jumpers may be removed to disconnect the on-board codec if an external I²S device is to be used instead.

The line input to the codec is routed to a 3.5mm stereo jack socket (J5) through the circuit shown in Fig 7.

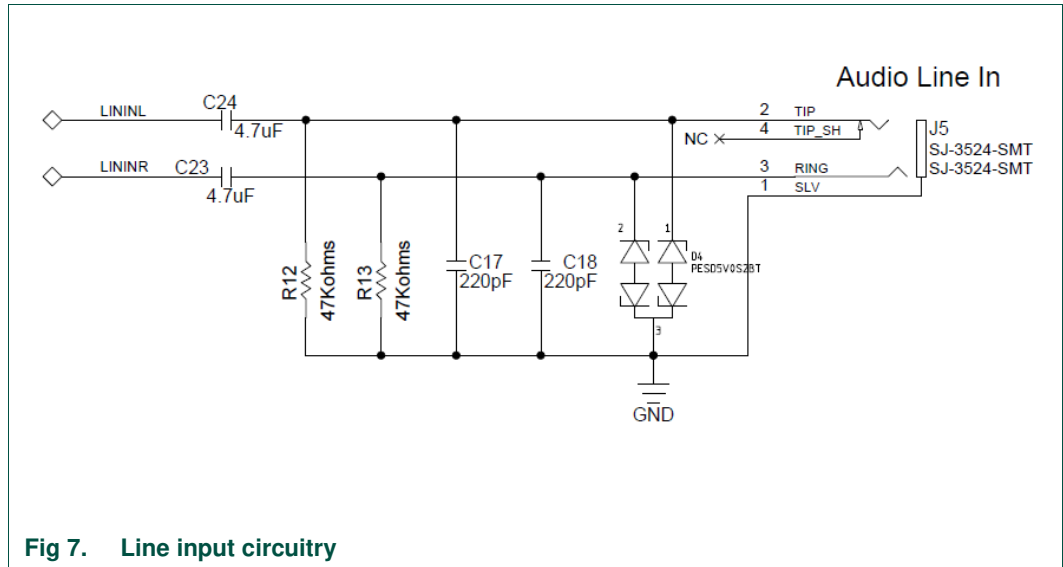


Fig 7. Line input circuitry

A second 3.5 mm stereo jack socket (J6) provides a headphone / line out from the codec, via the circuit shown in Fig 8.

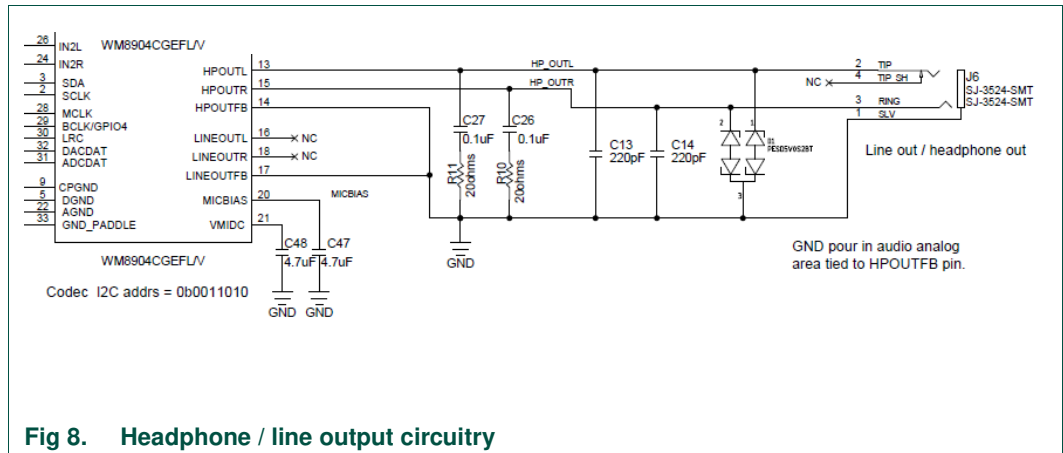


Fig 8. Headphone / line output circuitry

The MCUXpresso SDK includes drivers and example code for the audio codec.

10.3 SD card

The full size SD card (J7) included in the LPCXpresso54S018M board provides a 4-bit SDIO interface to support memory cards, plug-in WiFi modules, etc. Power enable to the socket is provided via PIO2-5, with LED D7 providing a visual indication when power is applied.

The MCUXpresso SDK includes drivers and examples of using this interface with SD memory cards.

10.4 Accelerometer

The board includes an NXP MMA8652FCR1 accelerometer, interfaced to Flexcomm 2 (ports PIO3-23/PIO3-24) of the LPC54S018J4M. The accelerometer has an I²C address of 0b0011101.

10.5 Digital Microphone

The LPC54S018J4M device incorporates dual digital microphone interfaces. A low power Knowles SPH0641LM4H digital microphone is incorporated on the board, interfaced to the PDM1 microphone interface. Note that the Knowles digital microphone is designed to be mounted on the underside of a board, with audio passing through a hole in the board.

Note that the PDM0 interface of the LPC54S018J4M is also available for interfacing to an external microphone via the Expansion connectors, but due to signal sharing with this function and LCD data lines 14 and 15, solder jumpers JS33 and JS36 need to be moved to the 2-3 position to enable this function (thus disabling correct LCD operation).

The MCUXpresso SDK includes drivers and demonstration code for capturing data from the on-board digital microphone and an external microphone of the same type (the latter requires board modification).

10.6 USER button

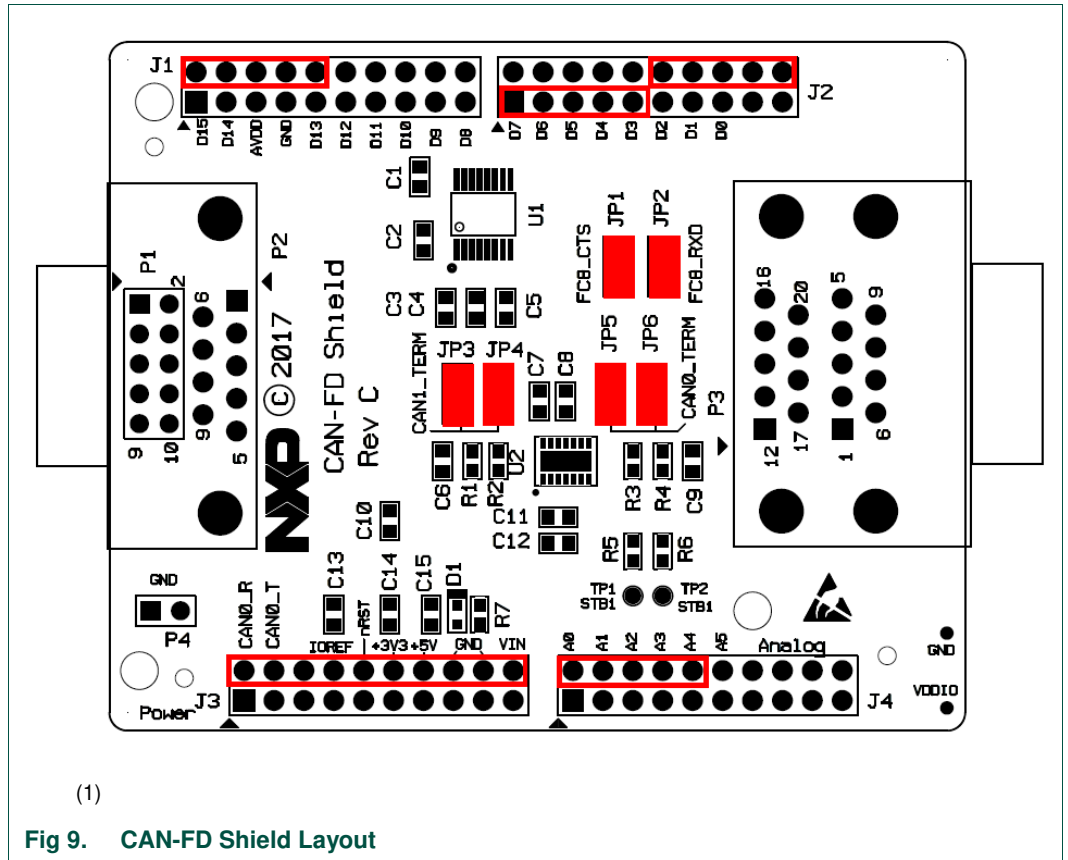
This button can be used to generate an interrupt by pulling down the PIO1-1 of the LPC54S018J4M to ground.

11. Dual CAN transceiver Shield (OM13099)

The OM13099 shield board is a dual CAN/CAN-FD transceiver, available as a standalone item from NXP and distributors, which has been designed to work with LPC54xxx series evaluation boards. This board features the NXP TJA1059 dual transceiver, plus an RS232C transceiver. This section provides a brief overview of the board and how to use it with the LPC54S018M-EVK.

11.1 Board layout and jumpers

[Fig 9 shows the](#) Shield board layout. The diagram highlights the connections used by the Shield board on the expansion connectors, and the location of the jumpers.



The CAN-FD Shield has a double-height DB9 connector installed by default (P3) for CAN/CAN-FD connections.

The Shield board layout includes provision for a DB9 connector to be installed (P2) but a standard 0.1" header (P1) is installed by default; P1 would need to be unsoldered in order for P2 to be fitted.

Note that header P4 is provided to allow convenient access to ground.

11.2 Jumper settings

Jumper settings are described in [Table 8](#).

Table 8. CAN-FD Shield Jumper settings

Circuit ref	Description	Default
JP1	RS232C transceiver CTS connection. When fitted, the RS232C transceiver CTSn signal is connected to expansion connector J4 pin 8 (Arduino A4). On the LPCXpresso54S018M boards A4 is connected to pin PIO3_4; this pin is shared with the accelerometer interrupt output.	Installed
JP2	RS232C transceiver RXD connection. When fitted, the RS232C transceiver R1OUT signal is connected to expansion connector J2 pin 20. On the LPCXpresso54S018M boards A4 is connected to pin PIO2_17.	Installed

Circuit ref	Description	Default
JP3 & JP4	Install these jumpers to provide termination on CAN/CAN-FD connection 2.	Installed
JP5 & JP6	Install these jumpers to provide termination on CAN/CAN-FD connection 2.	Installed

11.3 Installation

To install the Shield board, carefully align the J1 pin 1 of the Shield board with J9 pin 1 of the LPCXpresso board, and J3 pin 1 of the Shield board with J10 pin 2 of the LPCXpresso board, ensuring that all leads from the Shield are lined up with the corresponding sockets of the LPCXpresso board. Apply even pressure to the Shield board edges until its leads are fully seat in the sockets. It should not require much pressure to install the Shield board; if the required force seems high, stop and re-check the pins are aligned properly.

11.4 Drivers and example code

The MCUXpresso SDK package for the board includes drivers (called mcan drivers) and examples using polling and interrupt methods.

12. Legal information

12.1 Disclaimers

Limited warranty and liability — Information in this document is believed to be accurate and reliable. However, NXP Semiconductors does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

In no event shall NXP Semiconductors be liable for any indirect, incidental, punitive, special or consequential damages (including - without limitation - lost profits, lost savings, business interruption, costs related to the removal or replacement of any products or rework charges) whether or not such damages are based on tort (including negligence), warranty, breach of contract or any other legal theory.

Notwithstanding any damages that customer might incur for any reason whatsoever, NXP Semiconductors' aggregate and cumulative liability towards customer for the products described herein shall be limited in accordance with the Terms and conditions of commercial sale of NXP Semiconductors.

Right to make changes — NXP Semiconductors reserves the right to make changes to information published in this document, including without limitation specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

Suitability for use — NXP Semiconductors products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an NXP Semiconductors product can reasonably be expected to result in personal injury, death or severe property or environmental damage. NXP Semiconductors accepts no liability for inclusion and/or use of NXP Semiconductors products in such equipment or applications and therefore such inclusion and/or use is at the customer's own risk.

Applications — Applications that are described herein for any of these products are for illustrative purposes only. NXP Semiconductors makes no representation or warranty that such applications will be suitable for the specified use without further testing or modification.

Customers are responsible for the design and operation of their applications and products using NXP Semiconductors products, and NXP Semiconductors accepts no liability for any assistance with applications or customer product design. It is customer's sole responsibility to determine whether the NXP Semiconductors product is suitable and fit for the customer's applications and products planned, as well as for the planned application and use of customer's third party customer(s). Customers should

provide appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP Semiconductors does not accept any liability related to any default, damage, costs or problem which is based on any weakness or default in the customer's applications or products, or the application or use by customer's third party customer(s). Customer is responsible for doing all necessary testing for the customer's applications and products using NXP Semiconductors products in order to avoid a default of the applications and the products or of the application or use by customer's third party customer(s). NXP does not accept any liability in this respect.

Evaluation products — This product is provided on an "as is" and "with all faults" basis for evaluation purposes only. NXP Semiconductors, its affiliates and their suppliers expressly disclaim all warranties, whether express, implied or statutory, including but not limited to the implied warranties of non-infringement, merchantability and fitness for a particular purpose. The entire risk as to the quality, or arising out of the use or performance, of this product remains with customer.

In no event shall NXP Semiconductors, its affiliates or their suppliers be liable to customer for any special, indirect, consequential, punitive or incidental damages (including without limitation damages for loss of business, business interruption, loss of use, loss of data or information, and the like) arising out of the use of or inability to use the product, whether or not based on tort (including negligence), strict liability, breach of contract, breach of warranty or any other theory, even if advised of the possibility of such damages.

Notwithstanding any damages that customer might incur for any reason whatsoever (including without limitation, all damages referenced above and all direct or general damages), the entire liability of NXP Semiconductors, its affiliates and their suppliers and customer's exclusive remedy for all of the foregoing shall be limited to actual damages incurred by customer based on reasonable reliance up to the greater of the amount actually paid by customer for the product or five dollars (US\$5.00). The foregoing limitations, exclusions and disclaimers shall apply to the maximum extent permitted by applicable law, even if any remedy fails of its essential purpose.

12.2 Trademarks

Notice: All referenced brands, product names, service names and trademarks are property of their respective owners.

13. List of figures

Fig 1.	LPCXpresso54S018M underside view.....	3
Fig 2.	LPCXpresso54S018M main feature layout	5
Fig 3.	Jumper locations.....	6
Fig 4.	Self test menu	15
Fig 5.	Identifying the VCOM port.....	18
Fig 6.	Current measurement circuit error	20
Fig 7.	Line input circuitry	27
Fig 8.	Headphone / line output circuitry.....	27
Fig 9.	CAN-FD Shield Layout.....	29

14. List of tables

Table 1.	Jumpers	6
Table 2.	LEDs, buttons and connectors	8
Table 3.	P4 connections	21
Table 4.	Host Expansion Header signals	23
Table 5.	PMod Slave Expansion Header signals	23
Table 6.	Expansion connectors.....	24
Table 7.	Expansion Connector signal sharing.....	25
Table 8.	CAN-FD Shield Jumper settings	29

15. Contents

1.	Introduction	3	8.2	USB ports	21
2.	Feature summary	4	8.2.1	USB full-speed port (USB0).....	22
2.1	Board layout and settings.....	4	8.2.1.1	Device mode	22
3.	Getting Started	12	8.2.1.2	Host mode	22
3.1	Starting a debug session using the on-board (Link2) Debug Probe	12	8.2.2	USB High Speed port (USB1)	22
3.1.1	Installation steps for use with MCUXpresso IDE	12	8.2.2.1	Host mode	22
3.1.2	Installation steps to use Keil and IAR tools	13	8.3	Host Expansion Header (J14)	23
3.2	Starting a debug session using an external Debug Probe	14	8.4	PMod Slave Expansion Header.....	23
4.	Self-test demo application	14	8.5	Expansion connectors (including Arduino connectivity)	24
4.1	Self-test application and VCOM driver installation	14	8.6	Ethernet port.....	25
4.1.1	Serial terminal setup	14	8.7	Smart Card support	25
4.1.2	Running self-tests	14	9.	On-board memory	25
5.	Link2 debug features	16	9.1	SDRAM	25
5.1	What the Link2 boot LED indicates	17	10.	Other board features	26
5.2	Programming the Link2 firmware	17	10.1	LCD with capacitive touch	26
5.3	VCOM port	17	10.2	Audio codec.....	26
5.4	Configuring the LPCXpresso54S018M to debug an external target	18	10.3	SD card	27
6.	Troubleshooting debug sessions	18	10.4	Accelerometer	28
7.	Board power connections & measurement	19	10.5	Digital Microphone.....	28
7.1	LPCXpresso54S018M current measurement... ..	19	10.6	USER button	28
7.1.1	LPC54S018J4M Vsense resistor current measurement	19	11.	Dual CAN transceiver Shield (OM13099)	28
7.1.2	LPC54S018J4M VDD current measurement using a current meter	19	11.1	Board layout and jumpers.....	28
7.1.3	LPC54S018J4M VDD current measurement	20	11.2	Jumper settings	29
7.1.4	Shield board current measurement	21	11.3	Installation	30
8.	Board peripheral connections	21	11.4	Drivers and example code	30
8.1	LPC54S018J4M USART ports and VCOM support	21	12.	Legal information	31
8.1.1	P3 USART header	21	12.1	Disclaimers.....	31
			12.2	Trademarks	31
			13.	List of figures	32
			14.	List of tables	33
			15.	Contents	34

Please be aware that important notices concerning this document and the product(s) described herein, have been included in the section 'Legal information'.