

UM11158

LPCXpresso55S69/55S28 Development Boards

Rev. 1.6 — 17 March 2022

User manual

Document information

Info	Content
Keywords	LPC55S69, LPC55S6x, LPCXpresso55S69, LPC55S69-EVK, LPC55S28, LPCXpresso55S28, LPC55S28-EVK
Abstract	LPCXpresso55S69/55S28 user manual



Revision history

Rev	Date	Description
1.6	20220317	<ul style="list-style-type: none"> In Section 10 “Errata” added title Section 10.1 “Brown-out detect (BOD) may be triggered during boot (code execution from ROM)”. Added Section 10.2 “SPI ISP mode” and Section 10.3 “Power supply rise time”.
1.5	20200505	Added information about crystals. Updated information about 1.8V operation. Added clarifications about labeling of external serial connector. Renamed section 10 to Errata, added information about regulator errata (Revision A2 boards.)
1.4	20191114	Added information about tool version requirements.
1.3	20191029	Manual now also covers LPCXpresso55S28. Updated P4 information (after Revision A1 boards, 1.8V option may be used). Corrected pin out diagrams for alternative I2C connection on P19.
1.2	20190425	Added note about SD1_POWER_EN control in expansion connectors section. Corrected Red/Blue port pin assignments. Updated Figures 1 & 2.
1.1	20190314	Additional information about expansion connectors and use of J-link. Updated board images.
1.0	20190308	Added references in jumper/connector Indicators/buttons/connector table. Added notes about 3.3V only operation of Revision A/A1 boards. Added new figures for expansion connectors.
0.2	20181015	Added official document number. Section 4 updated/corrected.
0.1	20180816	Initial internal release

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, Keil uVISION and IAR Embedded Workbench. The LPCXpresso55S69 board (order code LPC55S69-EVK) and LPCXpresso55S28 board (order code LPC55S28-EVK) are the evaluation and development platforms for the LPC55S6x and LPC55S2x/LPC552x families of MCUs respectively.

See <https://www.nxp.com/demoboard/LPC55S69-EVK> or <https://www.nxp.com/demoboard/LPC55S69-EVK> for more information on these boards, including tutorial videos, development software and board hardware design files. Both board designs are identical except for the LPC target MCU fitted; the information in this manual applies to both board types unless otherwise noted. The abbreviation LPC55Sxx is used to collectively refer to the LPC55S69 or LPC55S28 device on these boards. **Note: the schematics for this board indicate components that are not installed by default with a red cross.**

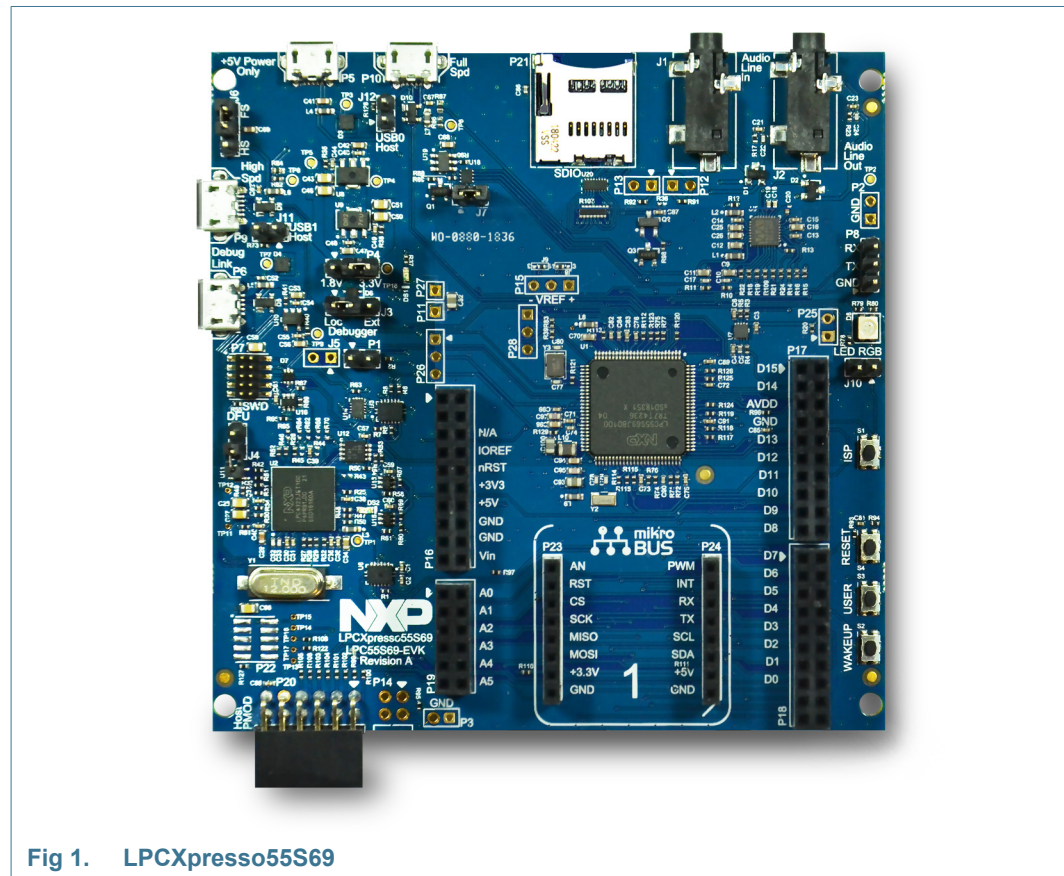


Fig 1. LPCXpresso55S69

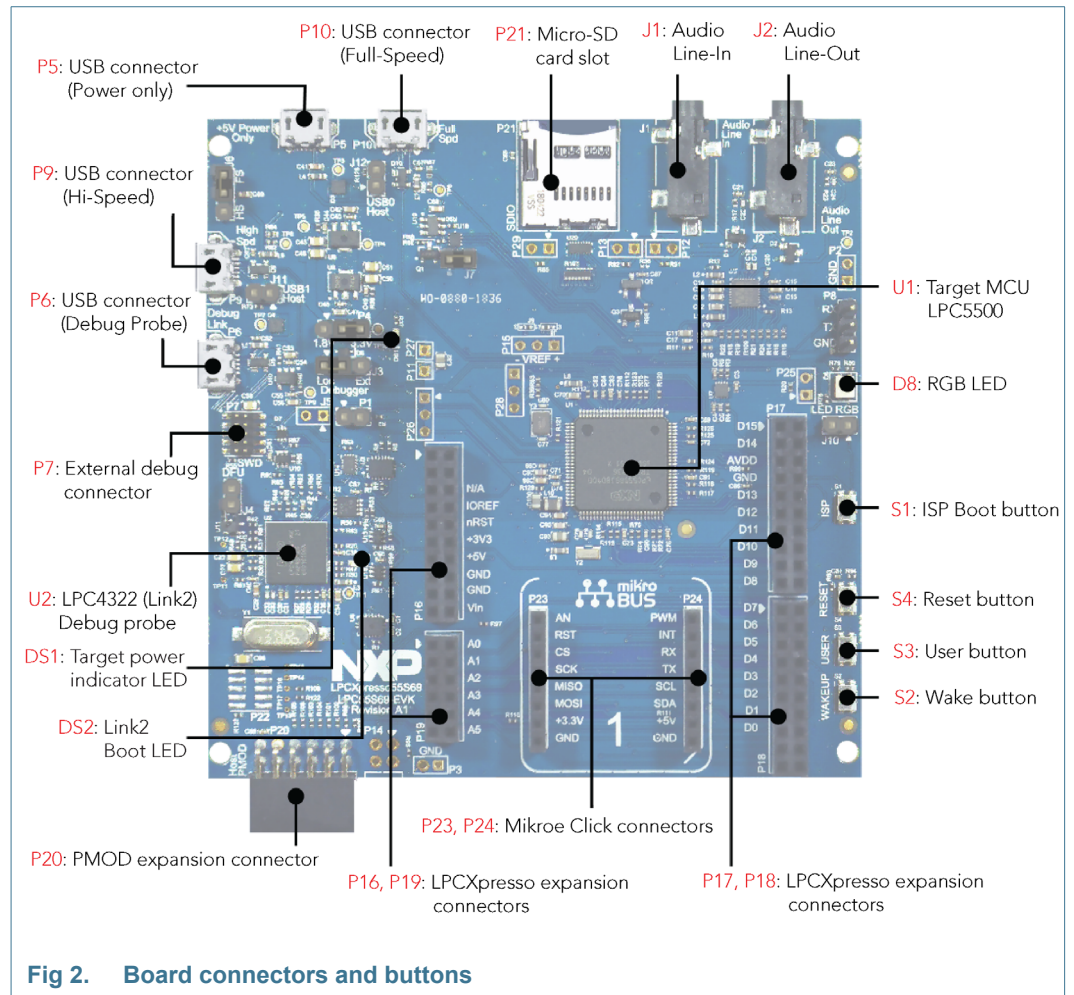
The LPCXpresso55S69 and LPCXpresso55S28 boards include the following features:

- LPC55S69 Dual Cortex-M33 core processor or LPC55S28 Cortex-M33 processor

- Onboard, high-speed USB, Link2 debug probe with CMSIS-DAP and SEGGER J-Link protocol options
- UART and SPI port bridging from LPC55Sxx target to USB via the onboard debug probe
- Optional external debug probes with trace option (10 or 20 pin Cortex-M connectors)
- RGB user LED
- Reset, ISP, User/Wakeup and user buttons
- Multiple Expansion options, including Arduino UNO, Mikroe Click and PMod
- Micro SD card slot
- NXP MMA8652FCR1 accelerometer
- Stereo audio codec with line in/out
- High / full speed USB port with micro A/B connector for the host or device functionality
Reset button

2. Board layout and Settings

Figure 2 shows the layout of the board (top side), indicating location of the connectors and buttons, while Figure 3 shows locations of jumpers and headers. Table 1 provides a description of connectors, jumpers, LEDs and buttons.



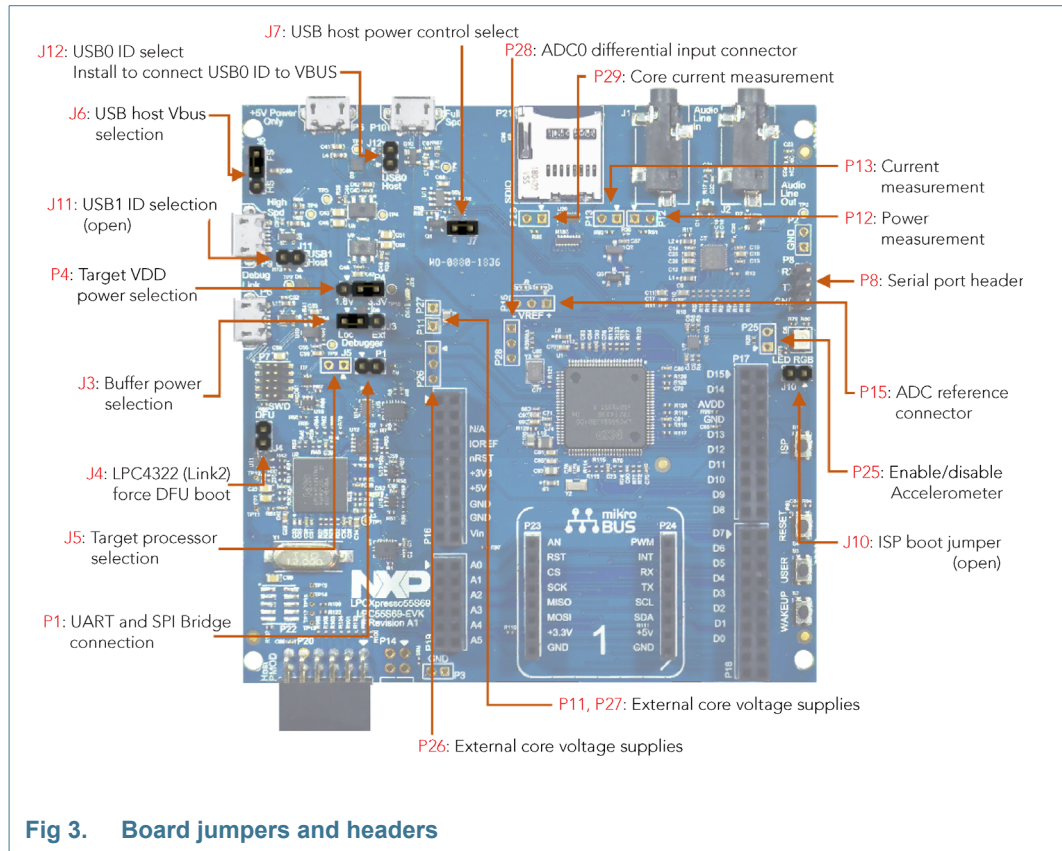


Fig 3. Board jumpers and headers

Table 1. Indicators, buttons, connectors and LEDs

Circuit reference	Description	Default	Reference
DS1	Target power indicator LED	n/a	n/a
DS2	Link2 boot LED	n/a	Section 4.1
D8	RGB User LED	n/a	Section 7.4
J1	Audio codec line input jack	n/a	Section 7.1
J2	Audio codec line output jack	n/a	Section 7.1
J3	Buffer Power Selection For On-board Target place in position 1-2 (default) For Off-board Target place in position 2-3	1-2	Section 3.2, Section 4.4
J4	Link2 (LPC43xx) force DFU boot. Leave this 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. Install this jumper 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. Note that the Link2 flash is pre-programmed with a version of CMSIS-DAP firmware by default.	Open	Section 3., Section 4.

Table 1. Indicators, buttons, connectors and LEDs

Circuit reference	Description	Default	Reference
J5	<p>Target processor selection for the on-board Debug Probe.</p> <p>Jumper open (default) the LPC55Sxx 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 LPC55Sxx 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>	Not installed	Section 4.
J6	<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 LPC55Sxx.)</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>	1-2 (USB1)	Section 6.1
J7	<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 LPC55Sxx. 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 LPC55Sxx.)</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>	Installed (USB1)	Section 6.1
J10	ISP boot jumper for LPC55Sxx. Installing this jumper ties port P0_5 to ground, forcing the LPC55Sxx into ISP mode whenever it is reset.	Open	Section 3.2, Section 7.5.1
J11	USB1 ID selection: USB1 ID is normally pulled to ground through a 100Kohm resistor. Installing this jumper connects USB1 ID to VBUS.	Open	Not required. May be removed in future versions
J12	USB0 ID selection: USB1 ID is normally pulled to ground through a 100Kohm resistor. Installing this jumper connects USB0 ID to VBUS.	Open	
P1	<p>When open (default), the "Bridge" UART and SPI connections from the Link2 probe are driven to the LPC55Sxx target.</p> <p>Install P1 when using the SPI interface at connector P20 and/or FC0 UART at P8. Note that this disables the Link2 SPI and UART (bridge) connections.</p>	Open	Section 4.
P4	Target VDD power selection. For Revision A1 boards this jumper must be placed in the 3.3V position (2-3). An external supply voltage to the LPC55Sxx can also be applied through pin 2 of this header	2-3 (3.3V)	Section 5.
P5	<p>External +5V power</p> <p>Micro USB connection for power to the LPC55Sxx target and peripheral circuitry (excluding Link2 Debug Probe).</p>	n/a	Section 5.
P6	<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>	n/a	Section 4.
P7	<p>10 pin external debug probe / off board target connector</p> <p>This standard Cortex-M debug connector is used either to (1) connect and off-board debug probe or (2) to connect an external debug target.</p>	n/a	Section 3.2.3

Table 1. Indicators, buttons, connectors and LEDs

Circuit reference	Description	Default	Reference
P8	Serial port header 0.1" header providing convenient access to Flexcom 0 USART (the USART used for ISP boot). When using this port install jumper P1 to disable the Link2 connection to this port.	n/a	Section 6.2
P9	LPC55Sxx USB1 (high speed) micro AB port connector	n/a	Section 6.1
P10	LPC55Sxx USB0 (full speed) micro AB port connector	n/a	Section 6.1
P11, P26, P27	Headers for external core voltage supplies (not installed). Do not use these headers unless instructed to do so by NXP.	n/a	n/a
P12	This header can be used to measure the voltage drop across a 2.43 ohm resistor connected in line with the LPC55Sxx VDD supply, and hence measure current consumption.	Not installed	Section 5.2
P13	Current measurement header for LPC55Sxx. This header is in line with the VDD supply of the LPC55Sxx, with a zero ohm resistor (R92) installed in parallel with it. To measure supply current to the LPC55Sxx, remove R92 and insert an ammeter between the pins of P13.	Not installed	Section 5.2
P14	Host header for LPC55Sxx reset control. This header is provided for convenient connection of a reset control input to the LPC55Sxx from an off-board host.	Not installed	See schematic
P15	ADC reference connector This header provides an access point to inject positive and negative voltage references for the LPC55Sxx ADC. An external positive reference may be connected to pin 1. Note that a solder jumper / zero ohm resistor must be in position 2-3 on J8. An external negative reference may be connected to pin 3. Note that a solder jumper / zero ohm resistor must be in position 2-3 on J9.	VREFP=VDD VREFN=GND	Section 9.1
P16-P19	LPCXpresso expansion connectors, including Arduino R3 compatible site. Provides connectivity to SPI (high speed SPI), USART (Flexcom 2), I2C (Flexcom 4 and Flexcom 1), I2S (Flexcom 7), ADC0, GPIO and PLU pins.	n/a	Section 8.
P20	PMod expansion connector / Host interface connector This connector provides access to the SPI (Flexcom 3) and I2C (Flexcom 1) ports that support LPC55Sxx ISP mode, along with 2 GPIO lines. When using the SPI port at this connector ensure jumper P1 is installed to disable the connection to the Link2 debug probe.	Not installed	Section 8.
P21	SD card slot 8-bit, full size card slot connected to the LPC55Sxx SD0 interface. Supports 3.3V operation only.	n/a	Section 7.2
P23, P24	Mikroe Click site Provides connectivity to standard Mikroe Click connectors. Shares SPI, ADC, I2C and USART connections with the P16-P19 expansion connectors.	n/a	Section 8.
P25	A header may be added at P25 to provide a convenient way to enable/disable the on-board accelerometer. If this header is used, remove the zero ohm resistor R20, which is in parallel with this header.	Not installed	Section 7.3
P28	ADC0 differential input connector. This header provides access to the ADC0 negative and positive inputs, plus a ground pin.	n/a	See schematic

Table 1. Indicators, buttons, connectors and LEDs

Circuit reference	Description	Default	Reference
S1	ISP/User button When pressed, this button connect sport pin P0_5 to ground. Hold this button down while pressing and releasing Reset (S4) to force the LPC55Sxx into ISP mode. Port P0_5 may also be reconfigured as a GPIO in user application code to enable this button to be used for other, user-defined purposes. A 2.2k ohm pull up is used to pull P0_5 high when the button is not pressed.	n/a	Section 7.5.1
S2	Wake/User button This button, when pressed, pulls P1_18 to ground. A 100K ohm pull up to VDD is connected to P1_18.	n/a	Section 7.5.2
S3	User button This button, when pressed, pulls P1_9 to ground. A 100K ohm pull up to VDD is connected to P1_9, and this port is also connected to pin 1 of expansion connector P18.	n/a	Section 7.5.2
S4	Reset button When pressed, reset is applied to the LPC55Sxx, expansion connector (Arduino reset) P16, the Mikroe Click site reset, and U14 (debug connector P7 reset when debugging an off-board target).	n/a	Section 7.5.3

3. Getting started

This section describes how to first power up the board and then how to start a first debug session using the MCUXpresso SDK. The board is pre-programmed with a simple program indicating that the target MCU is running. Connect a micro USB cable from connector P5 to a host computer or power supply to power-up the board and run this program.

The following debug probes can be used with those development and the board (check compatibility between debug probe and IDE used, including support for Arm Cortex M33 cores):

- On-board debug probe (LPC4322 “Link2”)
- LPC-Link2 (OM13054) debug probe from NXP or Embedded Artists
- SEGGER J-link probes (version 9 and newer). Ensure driver version 6.44a or later is installed.
- P&E Micro probes
- Keil ULINK2 probes
- IAR I-Jet probes

Other debug probes may also be supported by IAR and Keil tools and by other IDEs/debug tools; refer the websites of these companies for further information.

3.1 Tool version requirements

To ensure correct debugger operation, the following tool versions are recommended:

- IAR Embedded Workbench (EWARM) version 8.40.2 or later.
- Keil uVISION Device Family Pack (DFP) version 12.0.1 or later.

- MCUXpresso IDE version 11.1.1 or later
- SEGGER J-Link drivers version 6.72 or later. When using J-Link probes with non-SEGGER IDEs care should be taken to ensure the IDE being used is referencing the correct version of the driver.

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

By default, the Board is configured to use the on-board Debug Probe (Link2) to debug the on-board target (LPC55Sxx), 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. Check with your toolchain vendor for availability of specific device support packs for the LPC55Sxx series devices.

Note that when using the MCUXpresso IDE, the on-board Link2 can also be booted in DFU mode by installing a jumper on J4; 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: if the Debug Probe is set up to boot in DFU mode, the USB bridge functions (virtual COM port) and Debug Probe features will not be available if the board is not first initialized by the MCUXpresso IDE.

3.2.1 Installation steps for use with MCUXpresso IDE

Download and install the MCUXpresso IDE.

Configure and download an SDK package (with the MCUXpresso IDE tool chain option selected) from the MCUXpresso SDK Builder utility (<http://mcuxpresso.nxp.com>).

Recommended: Install J4 to force the Link2 Debug Probe to boot in DFU mode (see notes above).

Ensure jumper -- is fitted in position -- (local target powered), and -- is not installed (target SWD enabled). These are the default positions set during board manufacture.

Connect the board to the USB port of your host computer, connecting a micro USB cable to connector P6. The board will boot and run the pre-installed demo.

Allow about 10 seconds for the LPCXpresso55S69/S28 Board to enumerate for the first time; the device will appear as "LPC Device".

If the first attempt to debug a project fails in the IDE, cancel the debug session and repower 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 for the LPCXpresso55S69/55S28.

When the board is used for the first time, it is recommended to force the LPC55Sxx 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 ISP button while pressing and releasing the reset button or installing the LPC55Sxx ISP jumper.

3.2.2 Installation steps to use Keil and IAR tools with on-board debug probe

Download and install LPCScript (<https://www.nxp.com/lpcscript>). This will install required drivers for the board. Note that the Link2 (LPC4322 device) is pre-programmed with CMSIS-DAP firmware. If you have previously installed LPCScript please ensure version 2.1 or later is installed.

Ensure jumper J3 is fitted in position 1-2 (local target powered), and J5 is not installed (target SWD enabled). These are the default positions set during board manufacture.

The Link2 Debug Probe on the board will have been programmed with CMSIS-DSP firmware during manufacture, but can also be updated with J-link OB for LPC-Link2. To start the board with this firmware running, ensure J4 is not installed as power is connected, so the Link2 Debug Probe boots from internal flash. If you wish to update/change the Link2 firmware please follow the steps described in [Section 4.2](#).

Connect the Board to the USB port of your host computer, connecting a micro USB cable to connector P6 ("Debug-Link"). The board will boot and run the pre-installed demo. Allow about 30 seconds for the Link2 devices to enumerate for the first time. It is not necessary to check the Hardware Manager on your host computer, however if this is done there will be five devices (if using CMSIS-DAP protocol); 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.)

Your board is now ready to use with your 3rd party tool. Follow the instructions for those tools for using a CMSIS-DAP probe. MCUXpresso IDE can also be used with the board after setting up the board this way.

When the board is used for the first time, it is recommended to force the target into 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 ISP button while pressing and releasing the reset button.

3.2.3 Starting a debug session using an external Debug Probe

Code running on the LPC55Sxx target can be debugged using an external Debug Probe that conforms to the standard ARM Cortex-M debug connector. To use an external Debug Probe connect the probe to the SWD connector (P7) and connect power via the micro USB connector P5.

Note: The Debug link USB connector P6 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. Also note that the VCOM function provided by the on-board Debug Probe is not available when using an external debug probe.

4. On-board (Link2) Debug probe

This section describes the features provided by the on-board Link2 Debug Probe, including how to use this to debug an external target.

The Link2 Debug Probe is implemented using an LPC432x MCU (circuit reference U2), 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 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.
- I2S/SPI bridges: bridge device from I2C and SPI ports of the LPC 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 LPCXpresso55S69 (required for any debugging purpose) for Windows 7 or 8 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/lpcscript>.

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 IDE does support multiple LPCXpresso board targets.)

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 J4 (DFU jumper) before powering up the board. This is the recommended approach for the MCUXpresso IDE. Note that if J4 is installed when powering the board then the VCOM port (and other devices mentioned above) device will not appear until the MCUXpresso 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).

4.1 Link2 boot LED

LED DS2 is the Link2 MCU BOOT0_LED indicator. This LED reflects the state of Link2 MCU pin P1_1. When the boot process fails, DS2 will toggle at a 1 Hz rate for 60 seconds. After 60 seconds, the Link2 MCU is reset.

4.2 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 shunt on J4 (DFU), and power cycling (disconnecting then reconnecting power via P6.) Link2 MCU programming is performed using the LPCScript utility (see <http://www.nxp.com/lpcscript>). Instructions for using the tool are located at the same web page

After installing LPCScript, use either the “Program LPC-Link2 with CMSIS-DAP” or “Program LPC-Link2 with JLink” utilities provided in that installation to update the firmware, following on-screen instructions.

4.3 VCOM port

The identifier of the VCOM port will vary between boards and hosts as each board will enumerate with a unique identifier. On Windows, 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 (J4) is removed at power up, or (b) if the Debug Probe has been configured for DFU boot (J4 installed) at power up and MCUXpresso IDE has booted it (by starting a debug session.)

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.

Note that the VCOM port is not available when the Link2 Debug Probe is not powered.

4.4 Configuring the LPCXpresso55S69 to debug an off-board target

The LPCXpresso55S69 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 LPC55Sxx MCU from interfering with the SWD interface, J5 must be fitted (note: the header for this jumper is not installed during board production). 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 LPCXpresso55S69 board from the Link USB connector P6, and fit jumper J3 across pins 2 - 3 (External Target).

5. Board Power

The LPCXpresso55S69 board requires +5V input to power the on-board voltage low dropout linear regulators, of which there are 3, all available from Torex Semiconductor.

The Link2 Debug probe has a 2.5V regulator (U10) which draws power from USB connector P6 (“Debug Link”) only. The Debug Probe is unpowered if P6 is unconnected.

There are two other regulators, providing the option of 1.8V (U8) or 3.3V (U9) to the other devices on the board; these regulators can be powered by any of the following (+5V) sources:

- USB +5V Power only connector (P5)
- USB Full Speed connector for LPC55Sxx (P10)
- USB High Speed connector for LPC55Sxx (P9)
- USB Debug Link connector (P6)

The +5V sources above are connected via protection diodes to prevent reverse powering of any of them by another source. The +5V output of these combined sources is also connected to the Arduino and Mikroe expansion sites. Note that if P9 or P10 is configured as a USB host port then those connectors will not supply power to the board and another source must be provided.

5.1 1.8V operation

LPCXpresso55S69 Revision A1 boards (with LPC55S69 revision 0A silicon) require a minimum supply voltage of 1.85V, so jumper P4 must be set in the 3.3V position, or an external supply provided to the center pin of P4.

For Revision A2 boards, P4 may be set in the 1.8V position, however note that this is the minimum voltage supply for the LPC55Sxx devices, so variations in the 1.8V regulator tolerance and/or loading on the regulator may cause the brown-out detect (BOD) to trigger, especially when using the device at higher operating frequencies (when current draw is higher). The default BOD settings may be changed to avoid this problem, before higher clock speeds and/or higher current draw I/Os are enabled (e.g. at the start of main()) with a function call to the power library, as shown below:

```
POWER_SetBodVbatLevel(kPOWER_BodVbatLevel1650mv, kPOWER_BodHystLevel50mv, false);
```

It should be noted that this workaround may not work in all cases due to the regulator used on Revision A2 boards having a tolerance that does not guarantee the minimum required voltage output; see [Section 10 “Errata”](#).

Note that the MMA8652 Accelerometer and use of most SD cards require the board to be configured for 3.3V operation. The power connection to the Accelerometer may be disconnected by removing R20; installing header P25 provides a convenient way to connect/disconnect this.

Note that the forward drop voltage of the green and blue user LEDs (which are driven directly from GPIOs of the LED device) is too high for these to operate when the board is configured for 1.8V operation

For further details, refer to the board schematics.

5.2 Measuring LPC55Sxx device supply current

For further details, refer to the board schematics. Current supply to the LPC55Sxx Core can be measured via P29 (supply into VDDA and the on-board DC-DC converter) by removing R85. Supply current into the entire device (VDDA, DC-DC and I/O) can be measured via P13 (removing R92), or by measuring the voltage drop across the 2.43 ohm resistor (R91) which is accessible at P12.

6. Board serial connections

This section describes connections between LPC55Sxx on-board serial peripherals and connectors for use with off-board devices.

6.1 LPC55Sxx USB Ports

The Board incorporates micro AB connectors for both of USB0 (Full Speed, connector P10) and USB1 (High Speed, connector P9) ports of the LPCXpresso55S69. Both of these ports are capable of operating as a device or a host, and this is why micro AB connectors are used.

The Board can only be configured to support one of these ports as a host at a time (this is a limitation of the board, not the LPC55Sxx). When either port is used as a host, power must be supplied to the board via the +5V power only micro B USB port. Jumper J6 is used to connect power from a load switch to the VBUS signal of the port selected; J6 should be placed in position 1-2 for USB1 or position 2-3 for USB0. If neither USB port is being used as a USB host this jumper is not required.

Note that USB_VBUS1 pin is a 5V tolerant pin, so a resistor divider on the USB1_VBUS pin is not required, but is installed on Revision A1 and A2 boards. These resistors may be removed if VBUS detection issue is observed.

6.2 USART header

Header P8 is provided as a convenient way to use an LPC55Sxx USART with a serial to USB cable. Flexcom 0 ports (P0_29 and P0_30) are used for this feature, since these ports are assigned for USART ISP mode. Note that these ports are shared with the Link2 debug probe (LPC4322) and PMod connector; when using this header ensure that there is no conflicting device connected to the PMod connector and that jumper P1 is installed to disable the connection to the Link2. Always ensure the serial cable is only connected when the board is powered to prevent the LPC55S69 being powered through this TXD/RXD pins. Note that there is no level shifter between the LPC55Sxx target and this header, so a UART to serial cable with voltage levels matching the setting of the P4 power supply voltage selection jumper must be used.

NOTE: the silkscreen on the board is labeled from the perspective of a serial cable connected to the board, so P8 pin 3 (labeled "RXD" on the board silkscreen) is connected to PIO0_30 (Flexcomm0 TXD), and P8 pin 2 (labeled "TXD" on the board silkscreen) is connected to PIO0_29 (Flexcomm0 RXD).

7. On-board peripherals

This section describes how the on-board peripheral devices of the Board are connected to the LPC55Sxx and relevant configuration options. For full details of these devices please refer to the individual device datasheets. For circuit further details, refer to the board schematics.

7.1 Audio Codec

The LPCXpresso55S69 board incorporates a Cirrus Logic (Wolfson) WM8904 audio codec. This codec has both I2C (for control) and I2S (for data) interfaces. The I2C interface of the codec is routed to Flexcomm 4 of the LPC55Sxx, 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 LPC55Sxx are connected respectively to the Codec for data sent to / receive from the Codec. The board design assumes pad sharing between Flexcom 6 and Flexcom 7 for I2S connections.

Table 2. Audio Codec port connections

Circuit reference	Port	Flexcoms
I2S SCK	P0_21	Flexcom 6 & 7, pad shared
I2S TX/RX Data	P0_20	Flexcom 6 & 7, pad shared
I2S WS	P0_19	Flexcom 6 & 7, pad shared
I2C SDA	P1_21	Flexcom 4
I2C SCL	P1_20	Flexcom 4

I2S connections from the LPC55Sxx are also connected to the Expansion connectors; solder jumpers may be removed to disconnect the on-board codec if an external I2S device is to be used instead.

Line input (J1) and line output (J2) 1/4" stereo jack sockets provide analog I/O connections to the Codec. See schematic for further information.

7.2 Micro SD card slot

The micro SD card (P21) included in the LPCXpresso55S69 board provides a 4-bit SDIO interface to support memory cards, plug-in WiFi modules, etc. Power enable to the socket is provided via transistor enabled by P1_0.

Software drivers and related examples for the SD card are provided as part of the MCUXpresso SDK.

Table 3. Micro SD card connections

SDIO interface signals	Port	Notes
SDIO Clock	P0_7	IOCON function 2
SDIO Command	P0_8	IOCON function 2
D0	P0_9	IOCON function 2
D1	P0_17	IOCON function 2
D2	P0_24	IOCON function 2

Table 3. Micro SD card connections

SDIO interface signals	Port	Notes
D3	P0_25	IOCON function 2
Card detect	P0_21	IOCON function 2
Power enable	P1_0	GPIO. Active low.

7.3 Accelerometer

The board includes an NXP MMA8652FCR1 accelerometer, interfaced to Flexcom 4 (ports P1_21 and P1_20) with its interrupt output connected to P1_19 (this signal is also shared with the Expansion connector P18 via a 100 ohm resistor (R5).) The accelerometer has an I2C address of 0b0011101.

Note that the accelerometer can only be used when the board is configured for 3.3V operation. Resistor R20 may be removed to disconnect the MMA8652 from the supply rail, and P25 may be installed to provide a convenient way to connect/disconnect it to/from power.

Resistors R3 and R4 are zero ohm links, provided to enable the MMA8652 to be disconnected from P1_20 and P1_21 to avoid leakage effects when the board is configured for 1.8V operation.

I2C software drivers are provided as part of the MCUXpresso SDK, and example code is provided to illustrate how to read values from the accelerometer.

7.4 RGB User LED

An User controlled RGB LED is provided on the Board, located on the right hand edge. The LEDs in this device are controlled by LPC55Sxx ports P1_4 (Blue), P1_6 (Red), P1_7 (Green) with the LEDs being illuminated when the respective LED is pulled low. Note that these port lines are also connected to expansion connector P18.

7.5 Buttons

Four buttons are provided on the LPCXpresso55S69, as described in this section.

7.5.1 ISP (S1)

P0_5 is pulled up to VDD via a 2.2 kohm resistor; pressing this button will pull P0_5 low. This port is read by the LPC55Sxx boot code to determine if the MCU should enter In System Programming (ISP) mode. The main purpose of this button is to force the LPC55Sxx into ISP mode at boot time; to do this hold down this button while pressing and releasing the Reset button, or while connecting power to the board. While the main purpose of ISP mode is to program the flash of the LPC55Sxx, forcing the MCU into this mode places it into a known state in which development tools can gain debug control. If an application in flash is crashing or disabling the SWD port unintentionally, then ISP mode can be useful to recover control of the board. Refer to the LPC55S6x/S2x/2x User Manual for more information on ISP mode.

The ISP button can also be used in user application code. Care should be taken if P0_5 is configured as an output driving high, since pressing this button will short it to ground.

Note that jumper J10 may also be used as a convenient way to always assert ISP when the LPC55Sxx is reset or powered up. This jumper is in parallel with S1.

7.5.2 User (S3) and Wake (S2) buttons

The User (S3) and Wake (S2) buttons are intended for user application use. These buttons pull ports P1_9 (User) and P1_18 (Wake) low when the button is pressed. 100kohm resistors are used to pull these two ports to VDD then the buttons are not pressed.

7.5.3 Reset

Pressing this button will assert reset to the LPC55Sxx and to the Mikroe and Arduino connector sites. Note that the Debug Probe (LPC4322) is not reset when this button is pressed.

7.6 Crystals

For applications requiring high-speed communication such as high-speed USB and accurate clock such as real-time clock (RTC), two external crystals are provided on the board, 16MHz crystal for XTAL32M and 32.768KHz for XTAL32K.

External capacitors are not required on the crystal pins, but are installed on Revision A1 and A2 boards. These may be removed for better performance if issues are observed. See the “Chapter 12: LPC55S6x/LPC55S2x/LPC552x Cap Bank API” of the LPC55S6x/LPC55S2x/LPC552x User manual for further information.

8. Expansion connectors

The LPCXpresso55S69 includes 3 expansion connector sets, incorporating support for Arduino UNO R3, Mikroe Click and PMod standards. The Arduino UNO connector footprint is surrounded by additional connectors that are compatible with other LPCXpresso V3 boards. Pin out diagrams are provided below in [Figure 4](#) and [Figure 5](#). For further details, refer to the board schematics.

8.1 Mikroe Click Site

The connectors P23 and P24 provide a Mikroe Click module site for the wide range of add-on modules available from MikroElektronika. Note that this site shares the same SPI, I2C and UART connections as the Arduino/LPCXpresso V3 expansion connectors.

8.2 Host / PMod Connector

Connector P20 is a optional connector (not installed by default) which can provide convenient access for a remote host to the SPI and I2C ports of the LPC55Sxx that support ISP mode. When setting up the Board to be used with a host, jumper P1 must be installed to prevent conflict with the Link2 (which can also emulate an SPI or I2C host). Jumper J4 can be used to force the LPC55Sxx to boot in ISP mode, and connector P14

can be installed and used to provide an external reset control input from the host, The connector footprint and connection can also be used to connect to external PMod peripherals that follow the PMod I2C, or SPI (Type 2 or 2A) standards. Refer to the PMod specifications from Digilent Inc. for more information.

Table 4. Micro SD card connections

SDIO interface signals	Port	Notes
SDIO Clock	P0_7	IOCON function 2
SDIO Command	P0_8	IOCON function 2
D0	P0_9	IOCON function 2
D1	P0_17	IOCON function 2
D2	P0_24	IOCON function 2
D3	P0_25	IOCON function 2
Card detect	P0_21	IOCON function 2
Power enable	P1_0	GPIO. Active low.

8.3 Arduino and LPCXpresso V3 expansion connectors

The connectors P16-P19 provide Arduino UNO R3 compatibility and access to several other signals for use in prototyping. The overall footprint is compatible with the LPCXpresso V3 expansion connector footprint used on other LPCXpresso boards of this type. Note that some ports used on these connectors are shared with other devices/connectors on the board, including the Mikroe Click site, audio codec and RGB LED. Refer to the schematic for further information.

When using an Arduino shield with the Board, align P17 of the Arduino shield (SCL) with pin 1 of connector P17 on the LPCXpresso55S69 board.

Table 5. Micro SD card connections

Expansion connector	Port
P17	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 LPC55Sxx signals not used by the Arduino Uno rev3 compatible interface.
P18	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 LPC55Sxx signals not used by the Arduino Uno rev3 compatible interface.
P16	The even numbered pins 6 – 20 are compatible with the Arduino Uno rev3 Power connector standard.
P19	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 LPC55Sxx signals not used by the Arduino Uno rev3 compatible interface.

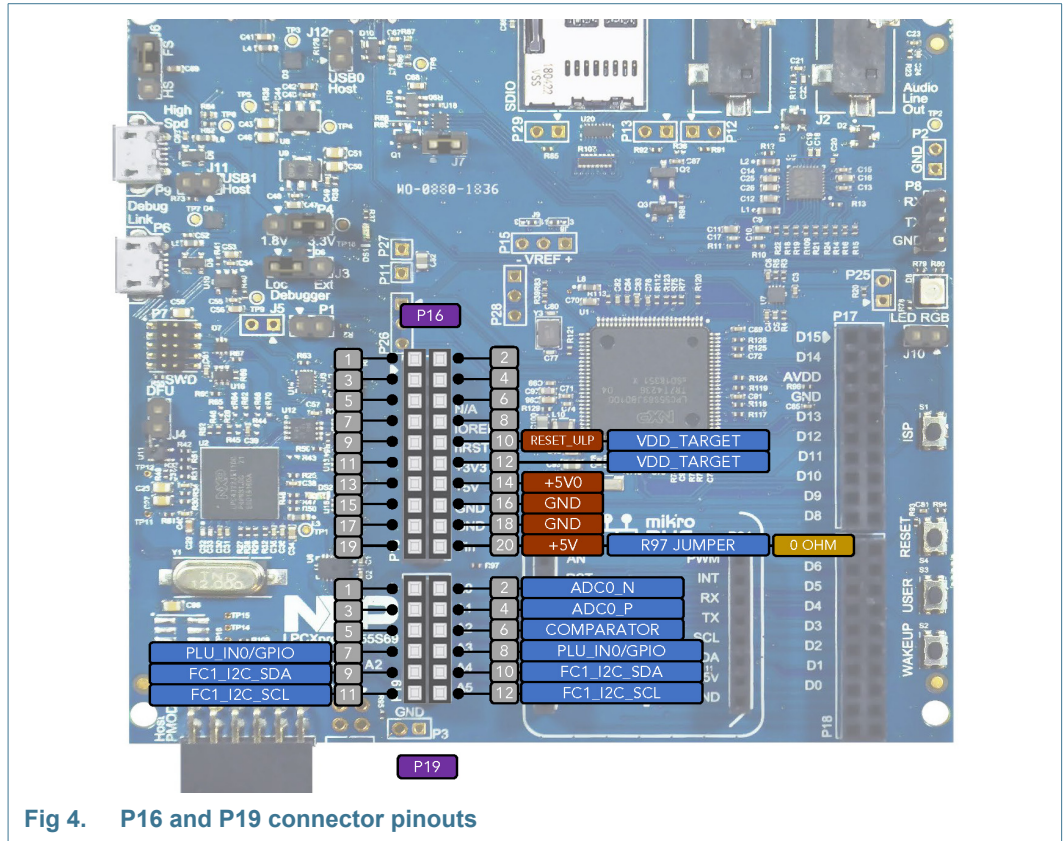


Fig 4. P16 and P19 connector pinouts

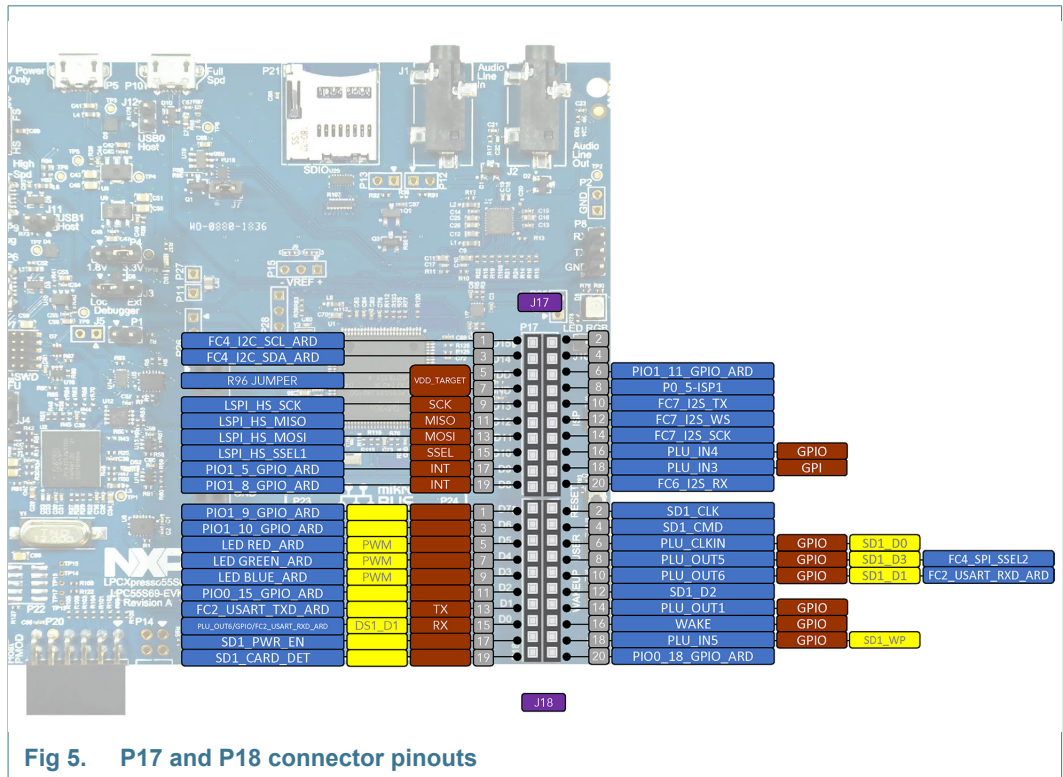


Fig 5. P17 and P18 connector pinouts

SDIO port 1 of the LPC55Sxx is routed to the expansion connector to provide access to this interface. Note that the power enable signal used for this SDIO interface is not one of the port pins that supports in SD1_POWER_EN function; this because all port pins that support this function are required for other features of the board. PIO1_16 is used instead for the power enable function, configured as a GPIO.

9. Other features

This section describes miscellaneous board features that we not covered elsewhere in this manual.

9.1 LPC55Sxx ADC references and inputs

By default, the LPC55Sxx ADC positive reference is connected to VDDA and the negative reference to ground. Different reference voltages may be used by moving the zero ohm link J8 or J9 (for positive or negative reference respectively) from the 1-2 to the 2-3 position, and connecting an external reference to P15 (pin 1 for positive, pin 3 for negative).

An external differential ADC input signal can be applied at P28 or at the P19 expansion connector. This connection is routed to PIO0_16 and PIO0_23.

10. Errata

10.1 Brown-out detect (BOD) may be triggered during boot (code execution from ROM)

Initial production LPCXpresso55S69 boards, marked as Revision A1, use LPC55S69 revision 0A silicon. Please refer to errata documentation from NXP for full details of device errata (https://www.nxp.com/docs/en/errata/ES_LPC55S6x.pdf).

Revision A2 boards have a 1.8V regulator installed that has a 2% tolerance. This can result in the supply voltage being below the specification of the LPC55Sxx device and thus in some cases the brown-out detect (BOD) of the device may trigger during boot (code execution from ROM). If this occurs it will prevent an application programmed in SRAM or flash from executing. In this situation an external 1.8V supply may be connected to the center pin of P4 to power the LPC55Sxx device. Note that board designs should use a regulator that guarantees a minimum 1.8V supply; alternative regulators with 1.9V or 2V outputs are available from Torex in their XC6203 family in the same footprint, and also in the XC6503 range, and other alternatives are available from other suppliers.

Revision A1 and A2 boards have some passive components installed that are not necessary and should be omitted in application designs; see [Section 6.1 “LPC55Sxx USB Ports”](#) and [Section 7.6 “Crystals”](#).

10.2 SPI ISP mode

SPI ISP mode is not supported since the pins assigned use Flexcomm 3. SPI ISP mode is supported only on the High-Speed SPI port pins.

10.3 Power supply rise time

The minimum rise time of the power supply on the LPC55S69/S28 does not meet the power-up ramp conditions specified in the datasheet. Please see the data sheet and VBAT_DCDC.1 errata in the errata sheet.

11. Board operating conditions

These boards have been designed for use in laboratory conditions (0 to 50 degrees Celsius) and should not be used for extended temperature testing. Refer to the data sheet of the LPC55xx MCU being used for full operating conditions for that device.

12. Contents

1	Introduction	3
2	Board layout and Settings	5
3	Getting started	9
3.1	Tool version requirements.	9
3.2	Starting a debug session using the on-board (Link2) Debug Probe.	10
3.2.1	Installation steps for use with MCUXpresso IDE . 10	
3.2.2	Installation steps to use Keil and IAR tools with on-board debug probe	11
3.2.3	Starting a debug session using an external Debug Probe	11
4	On-board (Link2) Debug probe	12
4.1	Link2 boot LED	13
4.2	Programming the Link2 firmware	13
4.3	VCOM port	13
4.4	Configuring the LPCXpresso55S69 to debug an off-board target	13
5	Board Power	14
5.1	1.8V operation	14
5.2	Measuring LPC55Sxx device supply current .	15
6	Board serial connections	15
6.1	LPC55Sxx USB Ports	15
6.2	USART header	15
7	On-board peripherals	16
7.1	Audio Codec	16
7.2	Micro SD card slot	16
7.3	Accelerometer	17
7.4	RGB User LED	17
7.5	Buttons	17
7.5.1	ISP (S1)	17
7.5.2	User (S3) and Wake (S2) buttons	18
7.5.3	Reset	18
7.6	Crystals	18
8	Expansion connectors	18
8.1	Mikroe Click Site	18
8.2	Host / PMod Connector	18
8.3	Arduino and LPCXpresso V3 expansion connectors	19
9	Other features	21
9.1	LPC55Sxx ADC references and inputs.	21
10	Errata	21
10.1	Brown-out detect (BOD) may be triggered during boot (code execution from ROM)	21
10.2	SPI ISP mode	21
10.3	Power supply rise time	22
11	Board operating conditions	22
12	Contents	23

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Limited warranty and liability — Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer’s technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

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.

Security — Customer understands that all NXP products may be subject to unidentified or documented vulnerabilities. Customer is responsible for the design and operation of its applications and products throughout their lifecycles to reduce the effect of these vulnerabilities on customer’s applications and products. Customer’s responsibility also extends to other open and/or proprietary technologies supported by NXP products for use in customer’s applications. NXP accepts no liability for any vulnerability. Customer should regularly check security updates from NXP and follow up appropriately. Customer shall select products with security features that best meet rules, regulations, and standards of the intended application and make the ultimate design decisions regarding its products and is solely responsible for compliance with all legal, regulatory, and security related requirements concerning its products, regardless of any information or support that may be provided by NXP. NXP has a Product Security Incident Response Team (PSIRT) (reachable at PSIRT@nxp.com) that manages the investigation, reporting, and solution release to security vulnerabilities of NXP products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, ICODE, JCOP, LIFE, VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, Altivec, CodeWarrior, ColdFire, ColdFire+, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, Tower, TurboLink, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, µVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks

Table continues on the next page...

and service marks licensed by Power.org. M, M Mobileye and other Mobileye trademarks or logos appearing herein are trademarks of Mobileye Vision Technologies Ltd. in the United States, the EU and/or other jurisdictions.

© NXP B.V. 2018-2022.

All rights reserved.

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

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

Date of release: 17 March 2022

Document identifier: UM11158

The logo for ARM, consisting of the lowercase letters 'arm' in a blue, sans-serif font.