LPCXpresso55S36UM

LPCXpresso55S36 Development Board User Manual

Rev. 3 — 17 August 2023 User manual

Document information

Information	Content
Keywords	LPCXpresso55S36UM, LPCXpresso55S36, LPC55S36, LPC5536, MCU-Link, MCUXpresso IDE
Abstract	The LPCXpresso55S36 board is a powerful and flexible, evaluation and development platform for NXP LPC55S36/LPC5536 microcontrollers (MCUs).

1 LPCXpresso55S36 overview

The LPCXpresso55S36 board is a powerful and flexible, evaluation and development platform for NXP LPC5536 microcontroller (MCU). It belongs to the LPCXpresso family of boards — boards for NXP LPC MCUs based on Arm Cortex-M cores.

The board is compatible with the Arduino UNO R3 and Mikroe click boards. It can be used with a wide range of development tools, including NXP MCUXpresso IDE, Keil µVision, and IAR Embedded Workbench. The board is lead-free and RoHS-compliant.

For debugging the LPC5536 MCU, the LPCXpresso55S36 board uses an onboard debug probe, known as MCU-Link OB (OB stands for "onboard"), which is based on another MCU, LPC55S69. For simplicity, MCU-Link OB is referred to as "MCU-Link debug probe" or just "MCU-Link" and the LPC5536 MCU is referred to as "target MCU" in this document.

This document provides details about the LPCXpresso55S36 board interfaces, power supplies, clocks, push buttons, jumpers, and LEDs.

1.1 Acronyms

Table 1 lists the acronyms used in this document.

Table 1. Acronyms

Acronym	Description
4PDT	Four pole double throw
ADC	Analog-to-digital converter
BGA	Ball grid array
CAN	Controller area network
Codec	Coder/decoder
DAC	Digital-to-analog converter
DMIC	Digital PDM microphone
DNP	Do not populate
FD	Flexible data-rate
GPIO	General-purpose input/output
HID	Human interface device
HS	High-speed
I2C	Inter-integrated circuit
I3C	Improved inter-integrated circuit
I2S	Inter-IC sound
IDE	Integrated development environment
ISP	In-System Programming
LED	Light-emitting diode
MCU	Microcontroller unit
NFC	Near field communication
Op-amp	Operational amplifier

Table 1. Acronyms*...continued*

Acronym	Description
PDM	Pulse-density modulation
POR	Power-on reset
PWM	Pulse width modulation
RC	Resistor-capacitor
SDK	Software development kit
SMT	Surface-mount technology
SPI	Serial peripheral interface
SWD	Serial wire debug
SWO	Serial wire debug trace output
UART	Universal asynchronous receiver/transmitter
USART	Universal synchronous/asynchronous receiver/transmitter
USB	Universal serial bus
VCOM	Virtual communication

1.2 Related documentation

Table 2 lists and explains the additional documents and resources that you can refer to for more information on the LPCXpresso55S36 board. Some of the documents listed below may be available only under a non-disclosure agreement (NDA). To request access to these documents, contact your local field applications engineer (FAE) or sales representative.

Table 2. Related documentation

Document	Description	Link
LPC553x Reference Manual	Intended for system software and hardware developers and application programmers who want to develop products with LPC553x/LPC55S3x MCU	LPC553xRM.pdf
LPC553x Data Sheet	Provides information about electrical characteristics, hardware design considerations, and ordering information	LPC553x.pdf
LPC55S6x/LPC55S2x/LPC552x User manual (UM11126)	Intended for system software and hardware developers and application programmers who want to develop products with LPC55S6x/LPC55S2x/LPC552x MCU	UM11126.pdf
MCU-Link Energy Measurement Capabilities Application Note (AN13660)	This document shows the results of the MCU-Link in different power settings and compares it to a third-party tool	AN13660.pdf

1.3 Board kit contents

Table 3 lists the items included in the LPCXpresso55S36 board kit.

Table 3. Board kit contents

Item description	Quantity
LPCXpresso55S36 board	1

Table 3. Board kit contents*...continued*

Item description	Quantity
USB cable	1
M3 x 40 mm female-female brass hex standoff	4
M3 x 5 mm stainless steel Phillips pan head screw	4
LPCXpresso55S36 Quick Start Guide	1

1.4 Block diagram

Figure 1 shows the LPCXpresso55S36 block diagram.

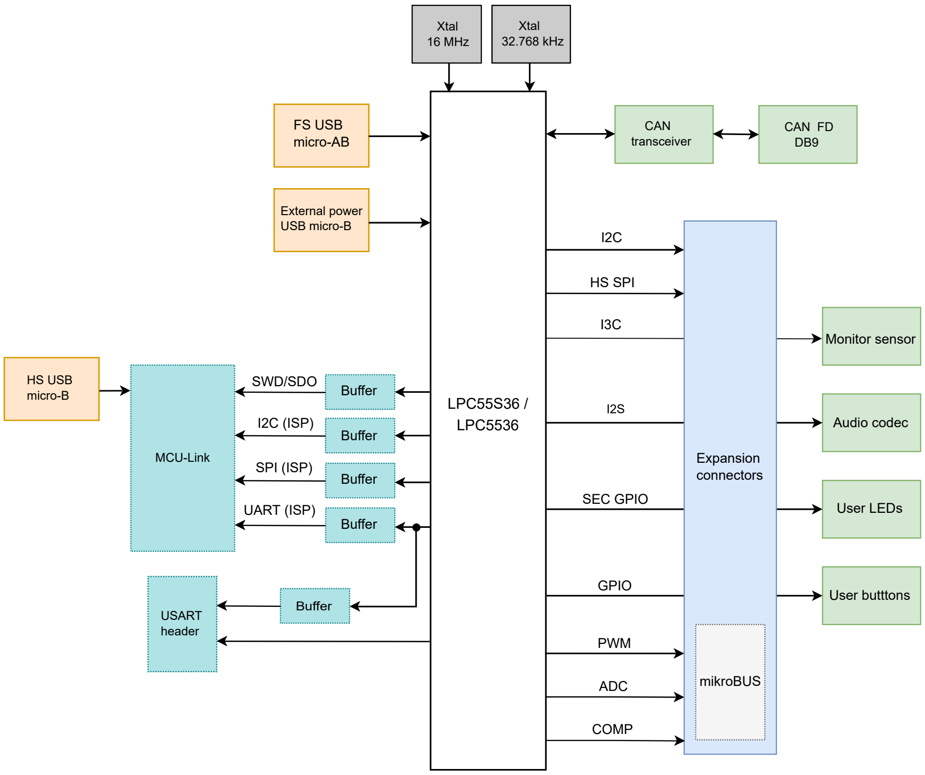


Figure 1. LPCXpresso55S36 block diagram

1.5 Board features

Table 4 describes the features of the LPCXpresso55S36 board.

Table 4. LPCXpresso55S36 features

Board feature	Target MCU feature used	Description
MCU (target MCU)		LPC5536 MCU with 32-bit Arm Cortex-M33 core running at a frequency of up to 150 MHz ***Note:*** *For details on the LPC5536 MCU, see LPC553x Data Sheet and LPC553x Reference Manual.*
USB	USB 2.0 full-speed host and device controller (USB0)	A USB micro-AB connector for the target MCU USB0 connection. The board also provides a USB micro-B

Table 4. LPCXpresso55S36 features*...continued*

Board feature	Target MCU feature used	Description
		connector as a power-only connector and another USB micro-B connector for MCU-Link.
USART	Flexcomm modules 0 and 2	• FC0 USART connection is supported as: – An external UART connection from a 1x3 pin header or a USB-to-UART bridge for connecting MCU-Link to the target MCU – An external UART connection from one of the connectors of Motor 2 + Arduino socket • FC2 USART connection is supported as an external UART connection from one of the connectors of mikroBUS socket
I2C	Flexcomm modules 1, 3, and 7	• FC1 I2C: Supports a USB-to-I2C bridge between MCU-Link and the target MCU • FC3 I2C: Connects to one of the connectors of Motor 2 + Arduino socket • FC7 I2C: Connects to another connector of Motor 2 + Arduino socket, one of the connectors of mikroBUS socket, audio codec, audio expansion connector, and external DMIC connector
I2S	Flexcomm modules 0, 2, 3, 4, and 6	I2S connections are supported on an audio codec and audio expansion connector
HS SPI	Flexcomm module 8	Three HS SPI slave selects are used: • SS0: Connects to one of the connectors of mikroBUS socket • SS1: Supports a USB-to-SPI bridge between MCU-Link and the target MCU • SS3: Connects to one of the connectors of Motor 2 + Arduino socket
CAN	Controller area network flexible data (MCAN) controller	A high-speed CAN transceiver for driving CAN signals between the target MCU and a DB9 connector
I3C	I3C module (I3C0)	I3C slave devices: • 6-axis motion tracking device with integrated 3-axis gyroscope and 3-axis accelerometer • One of the connectors of the Motor 1 + unused MCU pin socket
Octal flash	FlexSPI host controller	512 Mbit octal serial NOR flash memory
DMIC	Digital PDM microphone (DMIC) module	Onboard DMIC and an external DMIC connector
mikroBUS		mikroBUS socket with two 1x8 position receptacles
Motor control and Arduino	Motor control subsystem (Flex PWM0, FlexPWM1, ADC0, ADC1, DAC0, and DAC1)	Two Arduino UNO compatible sockets, each with four receptacles (2x6, 2x8, 2x10, and 2x8 positions)
Power supply		The board accepts the following external power supplies: • External 5 V power through one of the three USB connectors • VIN1 power from one of the connectors of the Motor 1 + unused MCU pin socket

Table 4. LPCXpresso55S36 features*...continued*

Board feature	Target MCU feature used	Description
		• VIN2 power from one of the connectors of Motor 2 + Arduino socket
Clock		Two crystals (16 MHz and 32.768 kHz) provide clocks to the target MCU
Debug		• MCU-Link debug probe (onboard debug probe) can be used to debug the target MCU using a USB micro-B connector provided on the board. It supports CMSIS-DAP and SEGGER J-Link protocol options. • An external debug probe can also be attached for debugging the target MCU • MCU-Link debug probe can also be used for debugging an external target MCU

1.6 Board pictures

Figure 2 shows the top-side view of the LPCXpresso55S36 board, with connectors, push buttons, and LEDs highlighted.

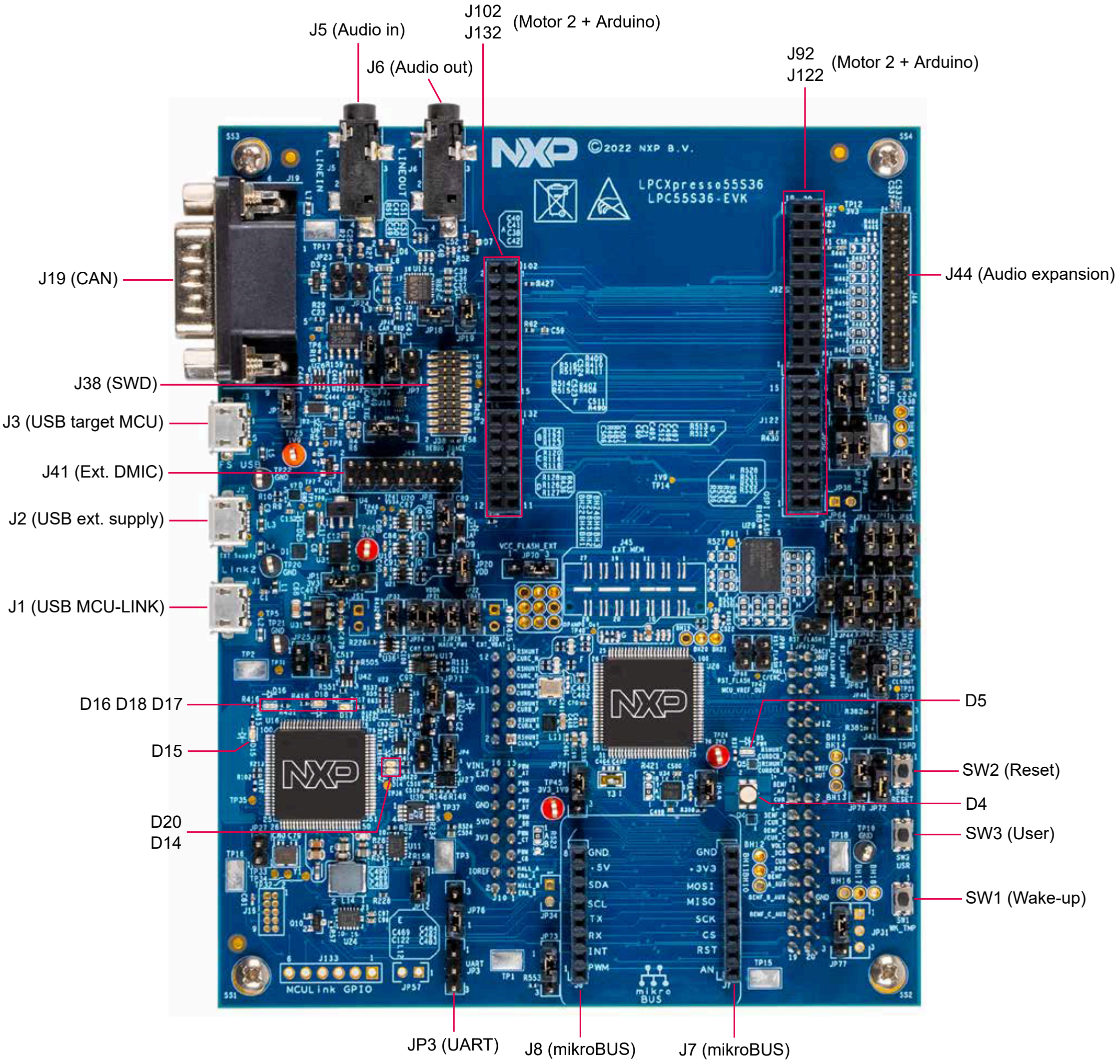


Figure 2. LPCXpresso55S36 connectors, push buttons, and LEDs (top-side view)

Figure 3 shows the bottom-side view of the LPCXpresso55S36 board, with four connectors and one LED highlighted.

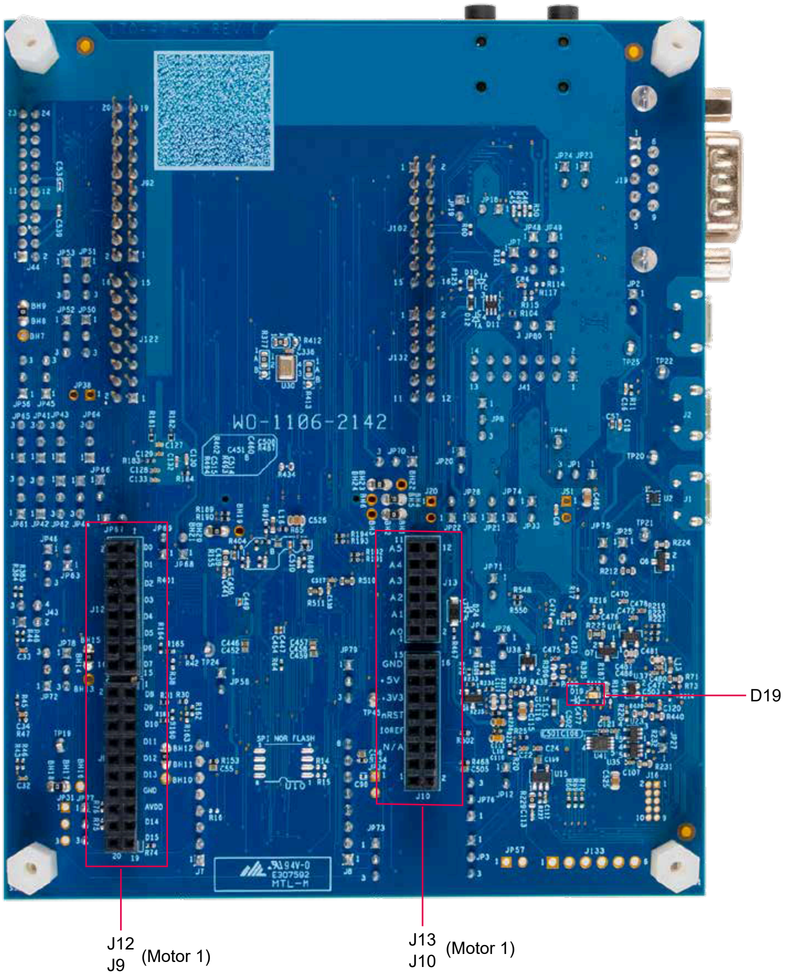


Figure 3. LPCXpresso55S36 connectors and LEDs (bottom-side view)

Figure 4 shows jumpers of the LPCXpresso55S36 board.

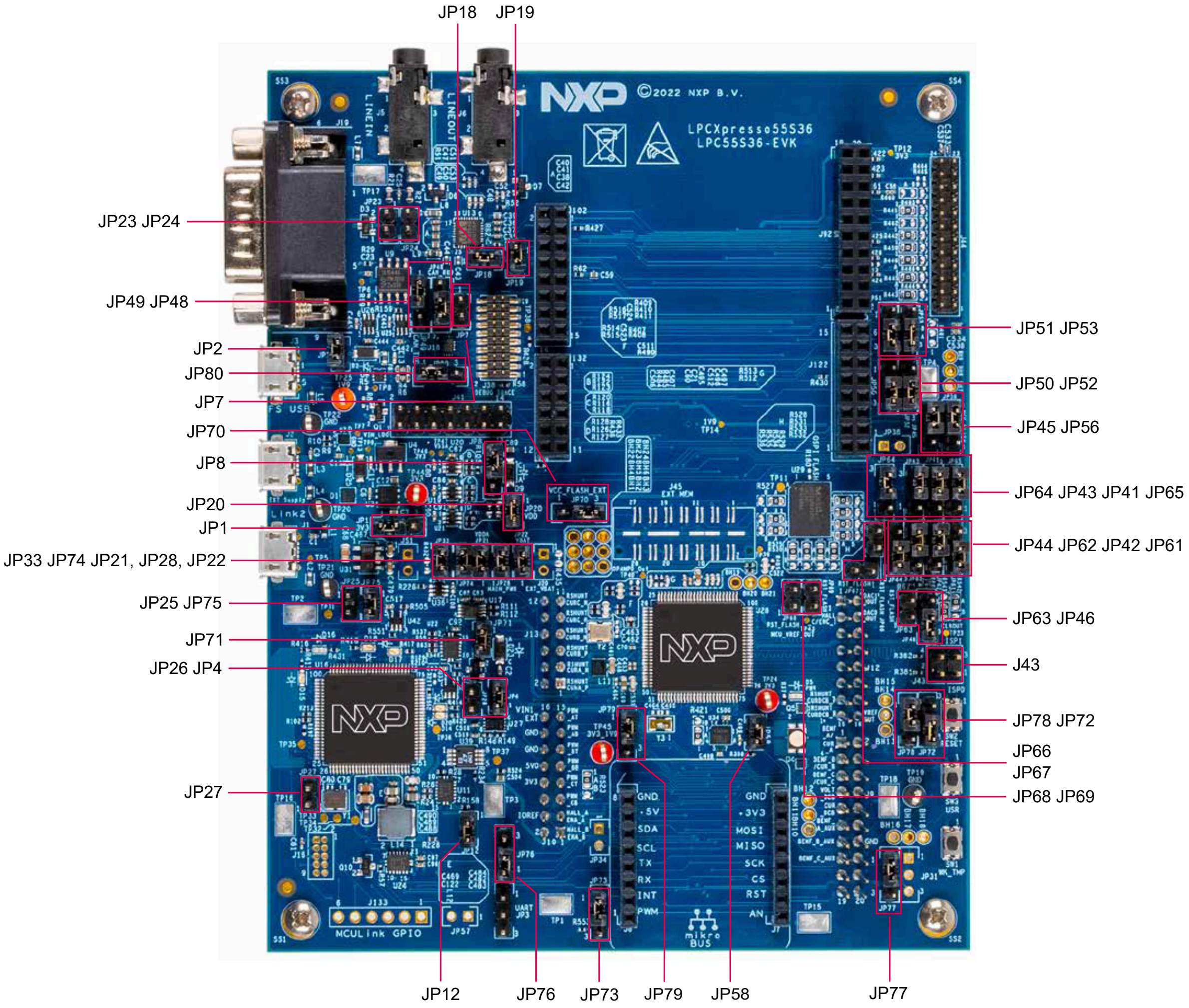


Figure 4. LPCXpresso55S36 jumpers

1.7 Connectors

The LPCXpresso55S36 connectors are shown in Figure 2 and Figure 3. The connectors are described in Table 5.

Table 5. LPCXpresso55S36 connectors

Part identifier	Connector type	Description	Reference section
J1	USB 2.0 micro-B receptacle	MCU-Link debug probe USB connector	Section 2.3
J2	USB 2.0 micro-B receptacle	External 5 V power connector	
J3	USB 2.0 micro-AB receptacle	Target MCU USB connector	
JP3	1x3 pin header	Connector for accessing the Flexcomm 0 (USART) port of the target MCU for an external UART connection	Section 2.4
J5	4-pin audio jack	Audio input connector for onboard audio codec	Section 2.6

Table 5. LPCXpresso55S36 connectors*...continued*

Part identifier	Connector type	Description	Reference section
J6		Audio output connector for onboard audio codec	
J44	2x12 pin header	Audio expansion connector	
J19	DB9 male connector	High-speed CAN connector	Section 2.8
J45 (DNP)	2x14 pin header	External flash connector	Section 2.10
J41	2x7 pin header	External digital PDM microphone (DMIC) connector	Section 2.11
J7	1x8 position receptacle	mikroBUS socket connectors	Section 2.12
J8			
J13	2x6 position receptacle	Motor 1 control + unused MCU pin socket connectors (placed on bottom side of PCB)	Section 2.13.1
J10	2x8 position receptacle		
J9	2x10 position receptacle		
J12	2x8 position receptacle		
J132	2x6 position receptacle	Motor 2 control + Arduino socket connectors	Section 2.13.2
J102	2x8 position receptacle		
J92	2x10 position receptacle		
J122	2x8 position receptacle		
J38	2x10 pin header	Debug (SWD) connector to connect an external debug probe or external debug target	Section 3.4
JP31 (DNP)	1x3 pin header	ADC reference connector. JP31 is not populated on the board. If populated, it can be used as an access point to inject positive and negative voltage references at its pins 1 and 3, respectively, for the target MCU ADC.	For more information on these connectors, see LPCXpresso55S36 schematics
J16 (DNP)	2x5 pin header	MCU-Link SWD connector	
J133 (DNP)	1x6 pin header	MCU-Link GPIO header	

1.8 Jumpers

Table 6 describes the LPCXpresso55S36 jumpers.

Table 6. LPCXpresso55S36 jumpers

Part identifier	Jumper type	Description	Reference section
JP1	1x3 pin header	3.3 V / 1.9 V voltage selection jumper: • Pins 1-2 shorted (default setting): 3.3 V supply is in use • Pins 2-3 shorted: 1.9 V supply is in use • Pins 1-2-3 open: Using pin 2 of JP1, an external voltage can be supplied to the target MCU (LPC5536)	Section 2.1

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
JP2	1x2 pin header	USB host VBUS enable jumper: • Open: The target MCU USB0 port is set to Device mode • Shorted (default setting): The target MCU USB0 port is set to Host mode (USB0 provides VBUS power). Ensure to supply external power to USB connector J2.	
JP71	1x2 pin header	VIN_LDO power source selection jumper: • Open: VIN_LDO supply is powered from SYS_5V0 supply • Shorted (default setting): VIN_LDO supply is powered from VIN1/VIN2 supply that comes from Motor 1 + unused MCU pin socket / Motor 2 + Arduino socket	
JP12	1x2 pin header	USART connector disable jumper: • Open: UART connection from USART connector JP3 is driven to the target MCU • Shorted (default setting): UART connection from USART connector JP3 is disabled	Section 2.4
JP26	1x2 pin header	USB-to-UART bridge disable jumper: • Open (default setting): MCU-Link is connected to the target MCU through a USB-to-UART bridge • Shorted: The USB-to-UART bridge between MCU-Link and the target MCU is disabled. Use this setting when using USART connection from USART connector JP3.	
JP4	1x2 pin header	USB-to-I2C bridge disable jumper: • Open: MCU-Link is connected to the target MCU through a USB-to-I2C bridge • Shorted (default setting): The USB-to-I2C bridge between MCU-Link and the target MCU is disabled	Section 2.5
JP19	1x2 pin header	Audio codec digital audio input enable jumper: • Open: Audio codec cannot receive digital audio input • Shorted (default setting): Audio codec can receive digital audio input from the target MCU	Section 2.6.1
JP18	1x2 pin header	Audio codec digital audio output enable jumper: • Open: Digital audio output from audio codec is disabled • Shorted (default setting): Audio codec can send digital audio output to the target MCU	
JP34 (DNP)	1x2 pin header	SPI NOR flash memory power enable jumper. SPI memory and jumper JP34 are not populated on the board.	Section 2.7
JP57	1x2 pin header	USB-to-SPI bridge disable jumper: • Open: MCU-Link is connected to the target MCU through a USB-to-SPI bridge	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Shorted (default setting): The USB-to-SPI bridge between MCU-Link and the target MCU is disabled, and the target MCU HS SPI port can be used for other purposes. For example, the HS SPI port can be used to establish a SPI connection with the SPI NOR flash memory.	
JP23	1x2 pin header	HS CAN transceiver high-level CAN bus line (CANH) disable jumper: • Open (default setting): Signal transmission is enabled over CANH bus line • Shorted: Signal transmission is disabled over CANH bus line	Section 2.8
JP24	1x2 pin header	HS CAN transceiver low-level CAN bus line (CANL) disable jumper: • Open (default setting): Signal transmission is enabled over CANL bus line • Shorted: Signal transmission is disabled over CANL bus line	
JP48	1x3 pin header	Target MCU MCAN controller input pin selection jumper: • Pins 1-2 shorted (default setting): Target MCU MCAN controller receives input through the MCU pin PIO1_3 • Pins 2-3 shorted: Target MCU MCAN controller receives input through the MCU pin PIO1_22 pin	
JP49	1x3 pin header	Target MCU MCAN controller output pin selection jumper: • Pins 1-2 shorted (default setting): Target MCU MCAN controller transmits output through the MCU pin PIO1_2 • Pins 2-3 shorted: Target MCU MCAN controller transmits output through the MCU pin PIO0_30	
JP56	1x3 pin header	Octal flash memory voltage selection jumper: • Pins 1-2 shorted: Octal flash memory is powered by VCC_3V3_1V9 supply • Pins 2-3 shorted (default setting): Octal flash memory is powered by VCC_1V9 supply	Section 2.10
JP63	1x2 pin header	Octal flash memory reset jumper. By default, jumper pins are open. To reset the flash memory, short the jumper.	
JP41	1x3 pin header	Target MCU pin PIO0_2 I/O selection jumper: • Pins 1-2 shorted: Target MCU pin PIO0_2 sends/receives input/output to/from jumper JP61 • Pins 2-3 shorted (default setting): Target MCU pin PIO0_2 sends/receives input/output to/from Motor 2 + Arduino socket connector J102	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
JP61	1x3 pin header	Jumper JP41 pin 1 I/O selection jumper: • Pins 1-2 shorted (default setting): Jumper JP41 pin 1 sends/receives serial data to/from octal flash memory / external flash memory • Pins 2-3 shorted: Jumper JP41 pin 1 provides I2C/I2S clock or sends/receives analog signal	
JP42	1x3 pin header	Target MCU pin PIO0_6 input selection jumper: • Pins 1-2 shorted: Target MCU pin PIO0_6 sends/receives input/output to/from octal flash memory / external flash memory • Pins 2-3 shorted (default setting): Target MCU pin PIO0_6 sends/receives input/output to/from Motor 2 + Arduino socket connector J102	
JP43	1x3 pin header	Target MCU pin PIO0_25 I/O selection jumper: • Pins 1-2 shorted: Target MCU pin PIO0_25 sends/receives I/O to/from octal flash memory / external flash memory • Pins 2-3 shorted (default setting): Target MCU pin PIO0_25 sends/receives I/O to/from Motor 2 + Arduino socket connector J122	
JP44	1x3 pin header	Target MCU pin PIO0_3 I/O selection jumper: • Pins 1-2 shorted: Target MCU pin PIO0_3 sends/receives input/output to/from jumper JP62 • Pins 2-3 shorted (default setting): Target MCU pin PIO0_3 sends/receives input/output to/from Motor 2 + Arduino socket connector J102	
JP62	1x3 pin header	Jumper JP44 pin 1 I/O selection jumper: • Pins 1-2 shorted (default setting): Jumper JP44 pin 1 sends/receives serial data to/from octal flash memory / external flash memory • Pins 2-3 shorted: Jumper JP44 pin 1 sends/receives I2C/I2S data / analog signal	
JP45	1x3 pin header	Target MCU pin PIO0_21 I/O selection jumper: • Pins 1-2 shorted: Target MCU pin PIO0_21 sends/receives input/output to/from octal flash memory / external flash memory • Pins 2-3 shorted (default setting): Target MCU pin PIO0_21 sends/receives input/output to/from Motor 2 + Arduino socket connector J102	
JP64	1x3 pin header	Target MCU pin PIO1_29 function control jumper: • Pins 1-2 shorted: Target MCU pin PIO1_29 sends/receives serial data to/from octal flash memory / external flash memory • Pins 2-3 shorted (default setting): Target MCU pin PIO1_29 acts as a GPIO pin	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
JP65	1x3 pin header	Target MCU pin PIO1_27 function control jumper: • Pins 1-2 shorted: Target MCU pin PIO1_27 sends/receives serial data to/from octal flash memory / external flash memory • Pins 2-3 shorted (default setting): Target MCU pin PIO1_27 acts as a GPIO pin	
JP66	1x2 pin header	Target MCU pin PIO1_16 I/O enable jumper: • Open (default setting): Target MCU pin PIO1_16 input/output is disabled • Shorted: Target MCU pin PIO1_16 sends/ receives serial data to/from octal flash memory / external flash memory	
JP67	1x2 pin header	Target MCU pin PIO1_15 I/O enable jumper: • Open (default setting): Target MCU pin PIO1_15 input/output is disabled • Shorted: Target MCU pin PIO1_15 sends/ receives serial data to/from octal flash memory / external flash memory	
JP68	1x2 pin header	Target MCU pin PIO0_19 output enable jumper: • Open (default setting): Target MCU pin PIO0_19 output is disabled • Shorted: Target MCU pin PIO0_19 provides serial clock input to octal flash memory / external flash memory	
JP69	1x2 pin header	Target MCU pin PIO0_4 I/O enable jumper: • Open (default setting): Target MCU pin PIO0_4 input/output is disabled • Shorted: Target MCU pin PIO0_4 sends/ receives serial data to/from octal flash memory / external flash memory	
JP70	1x3 pin header	External flash connector voltage selection jumper: • Pins 1-2 shorted: External flash connector is powered by VCC_3V3_1V9 supply • Pins 2-3 shorted (default setting): External flash connector is powered by VCC_1V9 supply	
JP38 (DNP)	1x2 pin header	External flash memory reset jumper. External flash connector and jumper JP38 are not populated on the board.	
JP58	1x2 pin header	I3C master pullup resistor control enable jumper: • Open: Pullup resistor control is disabled for I3C master (target MCU) • Shorted (default setting): Pullup resistor control is enabled for I3C master	Section 2.13.1
JP25	1x2 pin header	MCU-Link target (LPC55S69) force ISP mode jumper:	Section 3.2

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Open (default setting): MCU-Link follows normal boot sequence (MCU-Link boots from internal flash if a boot image is found). With the internal flash erased, the MCU-Link normal boot sequence falls through to ISP boot mode. • Shorted: MCU-Link is forced to ISP mode (USB1). Use this setting to reprogram the MCU-Link internal flash with a new image or use the MCUXpresso IDE with CMSIS-DAP protocol. ***Note:*** *By default, MCU-Link flash is preprogrammed with a version of CMSIS-DAP firmware.*	
JP7	1x2 pin header	MCU-Link debug probe target selection jumper: • Open (default setting): The MCU-Link debug probe or an external debug probe can be connected to the onboard target MCU. Use this setting when the target MCU operates in Normal mode (its debug interface is enabled). • Shorted: The MCU-Link debug probe can be connected to an external target MCU. Use this setting when using the MCU-Link debug probe for debugging an external target MCU.	Section 3.4
JP8	1x3 pin header	Target power selection jumper: • Pins 1-2 shorted (default setting): Onboard target MCU is used as debug target or an external target MCU is used as debug target but it uses board power • Pins 2-3 shorted: An external target MCU is used as debug target and it uses its own power	
JP33	1x2 pin header	MCU_VCC_3V3_1V9 supply enable jumper: • Open: MCU_VCC_3V3_1V9 supply is OFF • Shorted (default setting): MCU_VCC_3V3_1V9 supply is available	Section 3.8
JP20	1x2 pin header	MCU_VDD supply enable jumper: • Open: MCU_VDD supply is OFF • Shorted (default setting): MCU_VDD supply is available	
JP21	1x2 pin header	Target MCU VDDA power enable jumper: • Open: MCU_VDDA power supply is OFF • Shorted (default setting): MCU_VDDA power supply is available	
JP22	1x2 pin header	MCU_VBAT supply enable jumper: • Open: MCU_VBAT supply is OFF • Shorted (default setting): MCU_VBAT supply is available	
JP28	1x2 pin header	MCU_MAIN supply enable jumper: • Open: MCU_MAIN supply is OFF	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Shorted (default setting): MCU_MAIN supply is available	
JP74	1x2 pin header	These two jumpers are used together to connect a power supply to the current measurement circuit for measuring its current. By default, both are shorted.	
JP75	1x2 pin header		
J20 (DNP)	1x2 pin header	Target MCU external VBAT power enable jumper. J20 is not populated on the board. If populated, it can be used to supply VBAT power to the target MCU from an external power source.	For more information on these jumpers, see LPCXpresso55S36 schematics
J43	2x2 pin header	Target MCU boot configuration jumper. Pins 1-2 of J43 connect to PIO0_5 pin (ISP0) of the target MCU. Pins 3-4 of J43 connect to PIO0_7 pin (ISP1) of the target MCU. Pins 1-2 and pins 3-4 of J43 together determine the boot source for the target MCU. • Pins 1-2 and pins 3-4 open (default settings): Both ISP0 and ISP1 are High. Target MCU (LPC5536) follows normal boot sequence. • Pins 1-2 open and pins 3-4 shorted: ISP0 is High and ISP1 is Low. Target MCU boots in In-System Programming (ISP) mode.	
JP27	1x2 pin header	MCU-Link serial wire debug (SWD) disable jumper: • Open (default setting): MCU-Link SWD feature is enabled. MCU-Link drives SWD of target MCU or external target MCU. • Shorted: MCU-Link SWD feature is disabled. This setting of JP27 can be used when connecting an external debug probe for debugging the target MCU. This setting of JP27 is also recommended when using shared I3C signals.	
JP46	1x3 pin header	Target MCU pin PIO1_31 input selection jumper: • Pins 1-2 shorted: Target MCU monitors USB0 VBUS power • Pins 2-3 shorted (default setting): Target MCU receives input from jumper JP53	
JP53	1x3 pin header	Jumper JP46 pin 3 I/O selection jumper: • Pins 1-2 shorted: Jumper JP46 pin 3 sends/receives clock I/O to/from audio codec / audio expansion connector • Pins 2-3 shorted (default setting): Jumper JP46 pin 3 sends/receives PWM data to/from Motor 2 + Arduino socket connector J102	
JP50	1x3 pin header	Target MCU pin PIO1_21 function control jumper: • Pins 1-2 shorted: Target MCU pin PIO1_21 sends/receives I2C data	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Pins 2-3 shorted (default setting): Target MCU pin PIO1_21 sends/receives PWM data	
JP51	1x3 pin header	Target MCU pin PIO1_23 function control jumper: • Pins 1-2 shorted: Target MCU pin PIO1_23 provides I2C clock input • Pins 2-3 shorted (default setting): Target MCU pin PIO1_23 sends/receives PWM data	
JP52	1x3 pin header	Target MCU pin PIO1_25 function control jumper: • Pins 1-2 shorted: Target MCU pin PIO1_25 acts as UART/GPIO pin • Pins 2-3 shorted (default setting): Target MCU pin PIO1_25 acts as PWM pin	
JP72	1x3 pin header	Target MCU pin PIO0_22 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_22 connects to LED_GREEN signal • Pins 2-3 shorted: Target MCU pin PIO0_22 connects to USB0_VBUS signal	
JP73	1x3 pin header	Target MCU pin PIO0_17 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_17 receives interrupt signal from mikroBUS socket (connector J8) / SW3 button • Pins 2-3 shorted: Target MCU pin PIO0_17 communicates to audio codec	
JP76	1x3 pin header	Target MCU pin PIO1_1 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO1_1 communicates to MCU-Link / SPI memory (DNP) through HS SPI slave select 1 signal • Pins 2-3 shorted: Target MCU pin PIO1_1 communicates to debug trace connector (J38) through TRACE_CLKOUT signal	
JP77	1x3 pin header	Target MCU pin PIO0_24 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_24 communicates to 6-axis motion tracking device (U34) / Motor 1 connector J9 through I3C SDA signal • Pins 2-3 shorted: Target MCU pin PIO0_24 communicates to debug trace connector (J38) through TRACE_D0 signal	
JP78	1x3 pin header	Target MCU pin PIO0_29 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_29 communicates to MCU-Link / USART connector JP3 / audio expansion connector	

Table 6. LPCXpresso55S36 jumpers*...continued*

Part identifier	Jumper type	Description	Reference section
		• Pins 2-3 shorted: Target MCU pin PIO0_29 communicates to debug trace connector (J38) through TRACE_D2 signal	
JP79	1x3 pin header	Target MCU pin PIO0_28 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_28 communicates through FC0 SCK / I3C0 pull-up resistor control / USB0 overcurrent signal • Pins 2-3 shorted: Target MCU pin PIO0_28 communicates to debug trace connector (J38) through TRACE_D3 signal	
JP80	1x3 pin header	Target MCU pin PIO0_30 signal selection jumper: • Pins 1-2 shorted (default setting): Target MCU pin PIO0_30 communicates to MCU-Link / USART connector JP3 through FC0 RXD signal • Pins 2-3 shorted: Target MCU pin PIO0_30 communicates to debug trace connector (J38) through TRACE_D1 signal	

1.9 Push buttons

The LPCXpresso55S36 push buttons are shown in Figure 2 and are described in Table 7.

Table 7. LPCXpresso55S36 push buttons

Part identifier	Switch name	Description
SW1	Wake-up button (WK_TMP)	Wakes up the target MCU (LPC5536) from Deep Power-Down mode. It connects to the target MCU pin PIO1_18.
SW2	Reset button (RESET)	Causes the target MCU and board peripherals to reset to their default states and executes the boot code. If the target MCU is in Deep Power-Down mode, it comes out of this mode. SW2 connects to the target MCU pin RESETN.
SW3	User button (USR)	The purpose of this button can be defined in user application code. It connects to the target MCU pin PIO0_17.

1.10 LEDs

Table 8 describes the LPCXpresso55S36 light-emitting diodes (LEDs) that correspond to the target MCU. The board also has some MCU-Link specific LEDs, which are described in Section 3.9.

Table 8. LPCXpresso55S36 LEDs

Part identifier	LED color	LED name	Description
D4	Red/green/blue	RGB	User application LED. It is connected to the VCC_3V3 supply. This LED can be controlled through a user application.
D5	Red	PWR	Indicates system power-on status. It is connected to the VCC_3V3 supply. When the board is powered up, this LED turns ON.

2 LPCXpresso55S36 functional description

This section describes the features and functions of the LPCXpresso55S36 board. You can use the functionality described in this section as a reference while designing your own target board.

Note: *For details on the LPC5536 MCU, see LPC553x Data Sheet and LPC553x Reference Manual.*

2.1 Power supplies

The LPCXpresso55S36 board is powered up using the following primary power supplies:

- External 5 V power through USB micro-B connector J1
- External 5 V power through USB micro-B connector J2
- External 5 V power through USB micro-AB connector J3
- VIN1 (5 V max.) power from Motor 1 + unused MCU pin socket connector J10
- VIN2 (5 V max.) power from Motor 2 + Arduino socket connector J102

The primary power supplies are used to produce secondary power supplies for the board to power up other board components, including the target MCU (LPC5536), MCU-Link, mikroBUS socket, motor control and Arduino sockets, audio codec, DMIC, octal flash memory, high-speed CAN transceiver, push buttons, and LEDs.

Figure 5 shows the LPCXpresso55S36 power supply diagram.

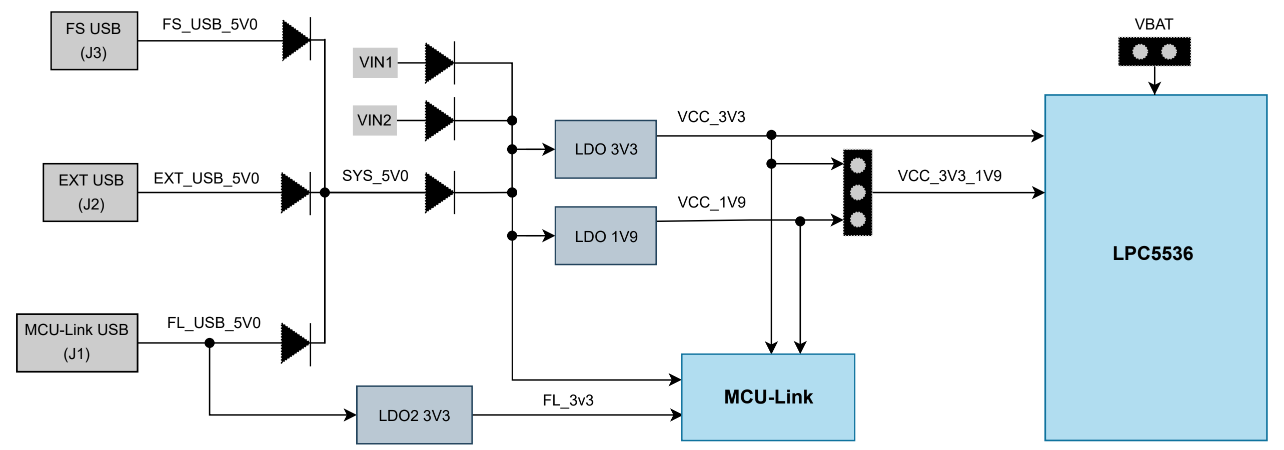


Figure 5. Power supply diagram

Table 9 describes the LPCXpresso55S36 power supplies.

Table 9. LPCXpresso55S36 power supplies

Power source	Manufacturing part number	Power supply rail	Description
USB connector J1	ZX62D-B-5PA8(30) (Hirose Electric)	FL_USB_5V0 (5 V)	• One of the three sources for SYS_5V0 supply • Supplies power to MCU-Link 3.3 V regulator (U31) • Provides USB1_VBUS power to MCU-Link • Should always be powered when using the board
USB connector J2	ZX62D-B-5PA8(30) (Hirose Electric)	EXT_USB_5V0 (5 V)	• Another source for SYS_5V0 supply • Produces HOST_USB_VBUS supply through load switch U7 (NX5P3090UK)

Table 9. LPCXpresso55S36 power supplies*...continued*

Power source	Manufacturing part number	Power supply rail	Description
			• Should be powered when using the target MCU USB0 in Host mode
USB connector J3	ZX62D-AB-5P8(30) (Hirose Electric)	FS_USB_5V0 (5 V)	Third source for SYS_5V0 supply
From FL_USB_5V0 / EXT_USB_5V0 / FS_USB_5V0		SYS_5V0 (5 V)	• One of the three sources for VIN_LDO supply • One of the three power sources for audio expansion connector • Supplies power to mikroBUS socket connector J8, Motor 1 + unused MCU pin socket connector J10, Motor 2 + Arduino socket connector J102, high-speed CAN transceiver, and voltage translators (U25 and U26) • Produces P5VA supply, which is used in current measurement circuit
Motor 1 + unused MCU pin socket connector J10		VIN1 (5 V max.)	Another source for VIN_LDO supply
Motor 2 + Arduino socket connector J102		VIN2 (5 V max.)	Third source for VIN_LDO supply
From VIN1 / VIN2 / SYS_5V0 supply through jumper JP71		VIN_LDO (5 V max.)	Supplies power to the target MCU 3.3 V regulator (U3) and 1.9 V regulator (U4)
U31	XC6227C331 PR-G (Torex Semiconductor)	3V3 (3.3 V)	Supplies power to MCU-Link, MCU-Link serial wire debug (SWD) connector (J16) (DNP), MCU-Link LEDs, I2C repeater / voltage translator (U27), voltage translators (U17, U19, U20, U21, U22, U24, and U41), and buffer (U42)
U3	NCP692MN33T2G (ON Semiconductor)	VCC_3V3 (3.3 V at 1 A max.)	• One of the two sources for VCC_3V3_1V9 supply • Another power source for audio expansion connector • Provides USB0_3V3 power to the target MCU • Supplies power to the target MCU LEDs (D4 and D5) and voltage translator (U11)
U4	XC6203E192 PR-G (Torex Semiconductor)	VCC_1V9 (1.9 V at 40 mA max.)	• Another source for VCC_3V3_1V9 supply • Third power source for audio expansion connector • One of the two power sources for audio codec, octal flash memory, and external flash connector (DNP)
From VCC_3V3 / VCC_1V9 supply through jumper JP1		VCC_3V3_1V9 (3.3 V or 1.9 V)	• Another power source for audio codec, octal flash memory, and external flash connector • Produces MCU_VCC_3V3_1V9 supply • Supplies power to onboard DMIC, external DMIC connector, mikroBUS socket connector J7, Motor 1 + unused MCU pin socket connector J10, Motor 2 + Arduino socket connectors J92 and J102, SPI NOR flash memory (DNP), SW1, SW2, and SW3 buttons, the target MCU D5 LED, debug interface four pole double throw

Table 9. LPCXpresso55S36 power supplies*...continued*

Power source	Manufacturing part number	Power supply rail	Description
			(4PDT) switch (U18), debug trace connector (J38), 6-axis motion tracking device (U34), I2C repeater / voltage translator (U27), voltage translators (U11, U22, U24, U25, and U26), and current measurement circuit • Produces VCC_IF_LINK supply, which provides power to MCU-Link, debug trace connector (J38), voltage translators (U17, U19, U20, U21), buffer (U33), and current measurement circuit
From VCC_3V3_1V9 supply through jumper JP33		MCU_VCC_3V3_1V9 (3.3 V or 1.9 V)	• Produces MCU_VDD, MCU_VDDA, MCU_VBAT, and MCU_MAIN supplies • Supplies power to current measurement circuit
From MCU_VCC_3V3_1V9 supply through jumper JP20		MCU_VDD (3.3 V or 1.9 V)	• Provides VDDIO power to the target MCU • Supplies power to boot configuration jumper (J43)
From MCU_VCC_3V3_1V9 supply through jumper JP21		MCU_VDDA (3.3 V or 1.9 V)	Produces MCU_VDDA_F supply, which supplies power to the target MCU and operational amplifiers (op-amps)
From MCU_VCC_3V3_1V9 supply through jumper JP22		MCU_VBAT (3.3 V or 1.9 V)	Provides VBAT power to the target MCU
From MCU_VCC_3V3_1V9 supply through jumper JP28		MCU_MAIN (3.3 V or 1.9 V)	Provides VDD_MAIN power to the target MCU

The LPCXpresso55S36 board supports current measurement for board power supplies through the onboard MCU-Link debug probe. For more details, see Section 3.8.

2.2 Clocks

Table 10 describes the clocks available on the LPCXpresso55S36 board.

Table 10. LPCXpresso55S36 clocks

Clock generator	Clock	Frequency	Destination
Y2: Crystal (CX3225SA16000D0PTW)	XTAL32M_[P,N]	16 MHz	Target MCU (LPC5536)
Y3: Crystal (FC-135 32.7680KA-AG3)	XTAL32K_[P,N]	32.768 kHz	Target MCU
Y1: Crystal (CX3225GA16000D0PTVCC)	MCU_LINK_[P,N]_16MHz	16 MHz	MCU-Link (LPC55S69)

2.3 USB interface

The LPCXpresso55S36 board has three USB connectors, each can accept 5 V external power. Two out of the three USB connectors can work as both USB ports and power connectors. The third one only works as a power connector.

Table 11 explains the LPCXpresso55S36 USB connectors.

Table 11. LPCXpresso55S36 USB connectors

Part identifier	Connector type	Device	Description
J3	USB 2.0 micro-AB receptacle	ZX62D-AB-5P8(30) (Hirose Electric)	Connects to full-speed USB host and device controller (USB0) of the target MCU (LPC5536). The USB0 port can work in either Host or Device mode. In USB0 Host mode, the J2 connector must be used as the power supply source (5 V) for the board. In USB0 Device mode, the J3 connector can serve as a power supply source (5 V) for the board.
J2	USB 2.0 micro-B receptacle	ZX62D-B-5PA8(30) (Hirose Electric)	Supplies 5 V power to the target MCU and peripheral circuitry, excluding MCU-Link debug probe. When the target MCU USB0 port is in Host mode, the J2 connector provides USB host power for the J3 connector.
J1	USB 2.0 micro-B receptacle	ZX62D-B-5PA8(30) (Hirose Electric)	Used to create MCU-Link high-speed USB connection to the host computer. J1 is also used to provide a 5 V supply to the board.

2.4 USART interface

The target MCU has a USART module, which can be accessed using its Flexcomm modules 0-7. In the LPCXpresso55S36 board, the USART module is accessed using Flexcomm modules 0 and 2 of the target MCU.

Figure 6 shows the USART connection diagram of the LPCXpresso55S36 board.

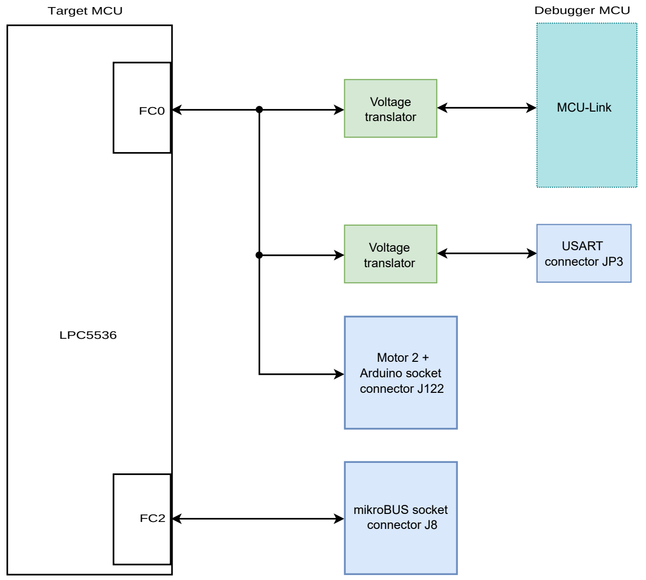


Figure 6. USART diagram

Table 12 explains the LPCXpresso55S36 USART connections.

Table 12. LPCXpresso55S36 USART connections

Flexcomm module	Voltage translator	Connector / MCU-Link target
FC0	Bidirectional voltage translator U22 (74AVC4 TD245BQ). It supports a USB-to-UART bridge between MCU-Link and the target MCU, by	MCU-Link target (LPC55S69). The USB-to-UART bridge between MCU-Link and the target

Table 12. LPCXpresso55S36 USART connections*...continued*

Flexcomm module	Voltage translator	Connector / MCU-Link target
	shifting voltage levels of signals between the two devices from 3.3 V to 3.3/1.9 V and vice versa. The USB-to-UART bridge can be disabled (so that the target MCU USART port can be used for other purposes) by shorting jumper JP26.	MCU can be used for debugging the target MCU from MCU-Link.
	Bidirectional voltage translator U11 (74AVC4 TD245BQ). It creates an external UART connection from USART connector JP3, by shifting voltage levels of signals between JP3 and the target MCU from 3.3 V to 3.3/1.9 V and vice versa. Before using this connection, short jumper JP26 to disable the USB-to-UART bridge between MCU-Link and the target MCU. The UART connection is disabled by default. It can be enabled by opening jumper JP12.	USART connecter JP3. It is a 1x3 pin header with the following pinout: • Pin 1: FC0_RXD • Pin 2: FC0_TXD • Pin 3: Ground (GND) JP3 supports an external UART connection using a USB-to-serial cable. It operates at 3.3 V. To prevent the board from powering up using the USB-to-serial cable, power on the board before connecting the USB-to-serial cable.
		Motor 2 control + Arduino socket connector J122. It allows a UART connection between the target MCU and board/module attached to the socket, using its pin 2 (RXD) and pin 4 (TXD).
FC2		mikroBUS socket connector J8. It allows a UART connection between the target MCU and click board plugged into the mikroBUS socket, using its pin 3 (RX) and pin 4 (TX).

2.5 I2C interface

The target MCU has an I2C module, which can be accessed using its Flexcomm modules 0-7. In the LPCXpresso55S36 board, the I2C module is accessed using Flexcomm modules 1, 3, and 7 of the target MCU.

Figure 7 shows the I2C diagram of the LPCXpresso55S36 board.

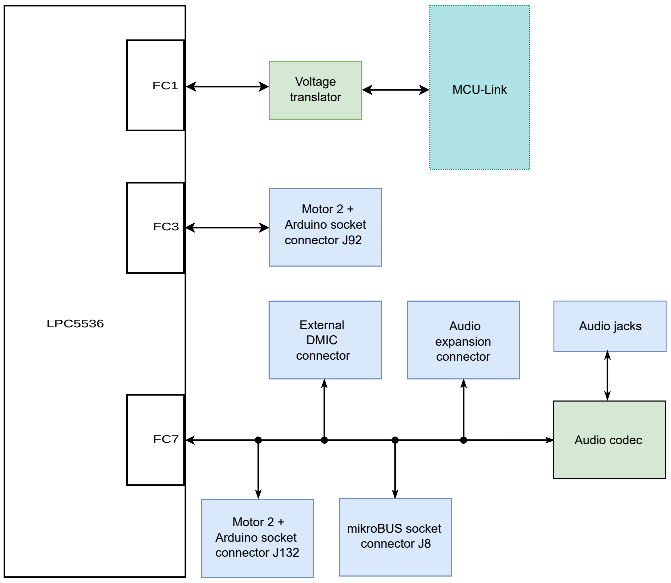

LPC5536
FC1
FC3
FC7
Voltage translator
MCU-Link
Motor 2 + Arduino socket connector J92
External DMIC connector
Audio expansion connector
Audio jacks
Audio codec
Motor 2 + Arduino socket connector J132
mikroBUS socket connector J8

Figure 7. I2C diagram

Table 13 shows the LPCXpresso55S36 I2C bus device map.

Table 13. I2C bus device map

Flexcomm module	I2C address	Device	Description
		LPC5536JBD100 (U28)	Target MCU. It acts as I2C master on the board.
FC1		LPC55S69JBD100 (U16)	MCU-Link. A USB-to-I2C bridge is supported between MCU-Link and the target MCU using an I2C repeater / a voltage translator (U27). Jumper JP4 can be used to enable/disable the USB-to-I2C bridge.
FC3	I2C address is defined by the connected board/ module	Motor 2 control + Arduino socket connector J92	Provides I2C connectivity to the board/module connected to Motor 2 control + Arduino socket
FC7	I2C address is defined by the connected DMIC	TSW-107-07-F-D (J41)	External digital PDM microphone (DMIC) connector. It provides I2C connectivity to the external DMIC connected to J41.
	I2C address is defined by the connected audio device	TMM-112-01-G-D (J44)	Audio expansion connector. It provides I2C connectivity to the external audio device connected to J44.
	0x34/0x35	WM8904 (U13)	Audio codec. It provides I2C connectivity to audio line in / line out connectors (J5 and J6).
	I2C address is defined by the connected board/ module	Motor 2 control + Arduino socket connector J132	Provides I2C connectivity to the board/module connected to Motor 2 control + Arduino socket
	I2C address is defined by the	mikroBUS socket connector J8	Provides I2C connectivity to the click board plugged-in into the mikroBUS socket

Table 13. I2C bus device map*...continued*

Flexcomm module	I2C address	Device	Description
	plugged-in click board		

2.6 I2S interface

Inter-IC sound (I2S) is a three-wire serial bus with one data, one clock, and one word select (frame trigger) signals. Each of the Flexcomm modules 0-5 of the target MCU provides one channel pair of I2S. One of these modules can be set as I2S master and others as I2S slaves. Each of the Flexcomm modules 6-7 of the target MCU provides four channel pairs of I2S. These modules can only be set as I2S slaves.

In the LPCXpresso55S36 board, I2S connections are supported on Flexcomm modules 0, 2, 3, 4, and 6. Figure 8 shows the I2S diagram of the LPCXpresso55S36 board.

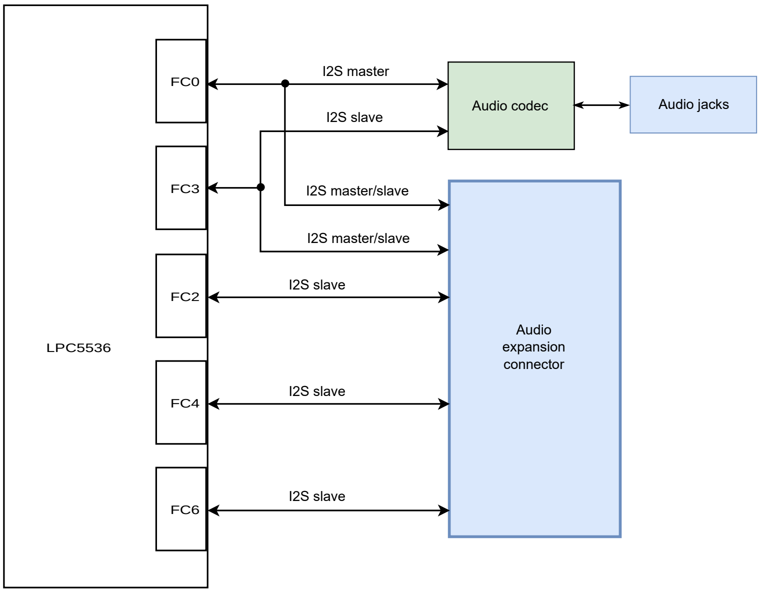


Figure 8. I2S diagram

Table 14 shows the LPCXpresso55S36 I2S bus device map.

Table 14. I2S bus device map

Device	I2S master	I2S slaves
Audio codec (U13)	FC0	FC3
Audio expansion connector (J44)	FC0/FC3	• When FC0 is used as I2S master; FC2, FC3, FC4, and FC6 are I2S slaves • When FC3 is used as I2S master; FC0, FC2, FC4, and FC6 are I2S slaves

2.6.1 Audio codec and audio expansion connector

The LPCXpresso55S36 board has an audio coder/decoder (codec) for encoding/decoding audio data. The audio codec is connected to two audio jacks, one for receiving input for audio codec and other for sending output from audio codec.

The board also provides an audio expansion connector, which can be used to connect an external audio device. The routing of signals between the target MCU and audio codec (or audio expansion connector) is controlled through various configuration resistors. The configuration resistors of the audio codec are populated, allowing it to communicate to the target MCU. The configuration resistors of the audio expansion connector are not populated on the board.

Communication between audio codec / audio expansion connector and the target MCU is enabled using I2C bus (for control signals) and I2S bus (for data signals).

The audio codec, audio jacks, and audio expansion connector are described in Table 15.

Table 15. Audio codec, audio jacks, and audio expansion connector

Part identifier	Manufacturing part number	Description
U13	WM8904	Audio codec. It receives analog audio from J5 jack, encodes it as digital signals, decodes digital signals back to analog audio, and sends it to J6 jack.
J5	SJ-3524-SMT	Audio input jack. It receives audio data from an external audio source and sends it to audio codec.
J6		Audio output (headphone output) jack. It sends out audio data received from audio codec.
J44	TMM-112-01-G-D	2x12 pin header for connecting an external audio device

Two jumpers (JP19 and JP18) are provided for testing digital audio input/output to/from audio codec. JP19 is used to enable/disable digital audio input to audio codec. JP18 is used to enable/disable digital audio output from audio codec.

2.7 HS SPI interface

The target MCU has a high-speed SPI (HS SPI) controller, which is accessible through its Flexcomm 8 module. The HS SPI controller can act as a SPI master or slave. It supports up to four SPI slave select inputs/outputs (SS0-SS3). In the LPCXpresso55S36 board, the HS SPI connections are supported through slave selects 0, 1, and 3.

Figure 9 shows the HS SPI diagram of the LPCXpresso55S36 board.

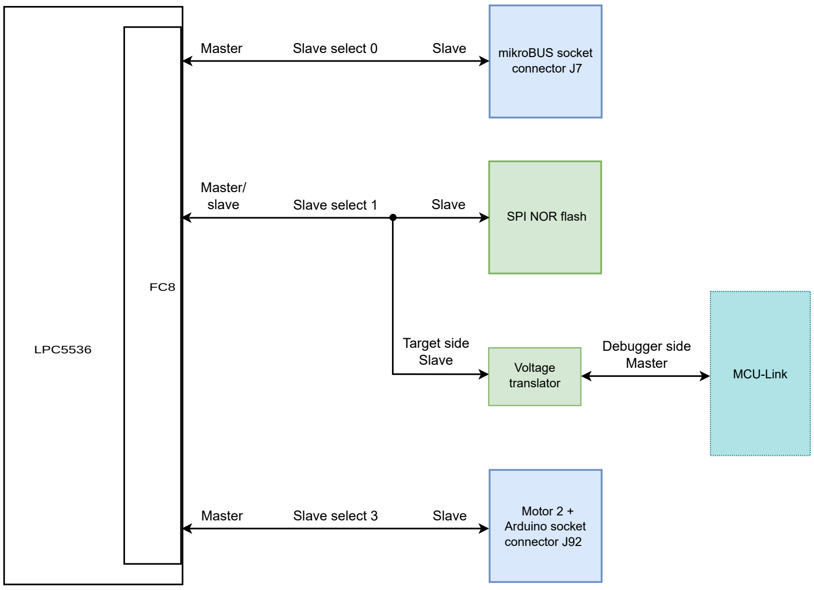


Figure 9. HS SPI diagram

Table 16 describes the HS SPI connections on the LPCXpresso55S36 board.

Table 16. HS SPI connections

Slave select	Voltage translator	Slave device / MCU-Link target
Slave select 0		mikroBUS socket connector J7. It allows a SPI connection between the target MCU and click board plugged into the mikroBUS socket.
Slave select 1		8 Mbit SPI NOR flash memory (U10) (not populated on the board). Supported part number is MX25R8035FM2IL0. If the SPI NOR flash memory is attached, it is powered up by shorting jumper JP34 (not populated on the board). Using this connection also requires disabling the USB-to-SPI bridge between MCU-Link and the target MCU by shorting jumper JP57 (shorted by default).
	Bidirectional voltage translator U24 (74AVC4TD245BQ). It supports a USB-to-SPI bridge between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to 3.3/1.9 V and vice versa. Jumper JP57 is used to enable/disable the USB-to-SPI bridge. By default, jumper JP57 is shorted, and the USB-to-SPI bridge is disabled. To enable the USB-to-SPI bridge, open jumper JP57.	MCU-Link target (LPC55S69). The USB-to-SPI bridge between MCU-Link and the target MCU can be used for debugging the target MCU (slave) from MCU-Link (master).
Slave select 3		Motor 2 control + Arduino socket connector J92. It allows a SPI connection between the target MCU and board/module attached to the socket.

2.8 CAN interface

The target MCU has a controller area network flexible data (MCAN) controller, which supports CAN flexible data-rate (FD). The LPCXpresso55S36 board allows communication with MCAN controller through a high-speed (HS) CAN transceiver, which drives CAN signals between MCAN controller and a physical two-wire CAN bus that terminates at a DB9 connector.

Figure 10 shows the CAN diagram of the LPCXpresso55S36 board.

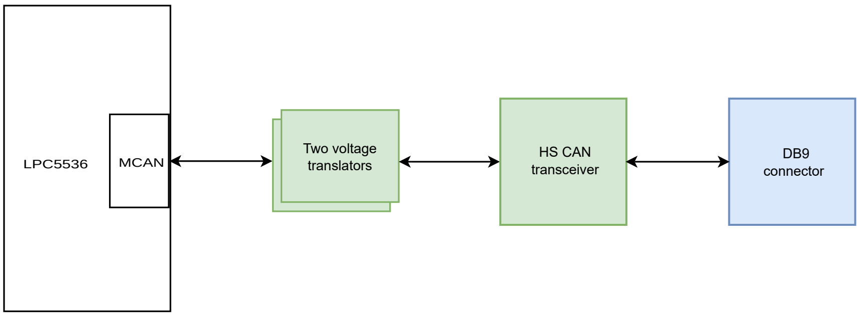


Figure 10. CAN diagram

Table 17 explains the LPCXpresso55S36 CAN connections.

Table 17. LPCXpresso55S36 CAN connections

CAN controller	Voltage translators	CAN transceiver	Connector
MCAN	Two voltage translators U25 and U26 (74LVC1T45GW,125). U25 and U26 shift voltage levels of signals between the target MCU and HS CAN transceiver from 3.3/1.9 V to 5 V and vice versa.	High-speed CAN transceiver U9 (NXP TJA1044GT), which is designed for high-speed CAN applications in the automotive industry, providing the differential transmit and receive capabilities to a CAN protocol controller. It converts and sends analog data from CAN bus lines to the target MCU as digital data, and converts and sends digital data from the target MCU to the CAN bus lines as analog data.	DB9 male connector J19. It allows external CAN connection with the CAN bus.

Jumpers JP23 and JP24 are used to enable/disable high-level CAN bus line (CANH) and low-level CAN bus line (CANL), respectively.

Jumper JP48 is used to select the target MCU input pin for the MCAN controller. Jumper JP49 is used to select the target MCU output pin for the MCAN controller.

2.9 I3C interface

The MIPI Alliance improved inter-integrated circuit (MIPI I3C) brings major improvements in use and power over I2C, and provides an alternative to SPI for mid-speed applications. The target MCU has an I3C module, which is used as an I3C master for devices (I3C slaves) connected to I3C bus on the LPCXpresso55S36 board.

The I3C and SWD signals are shared on the LPCXpresso55S36 board. While using I3C, it is recommended to short jumper JP27 to isolate MCU-Link from driving the shared signals.

Figure 11 shows the I3C diagram of the LPCXpresso55S36 board.

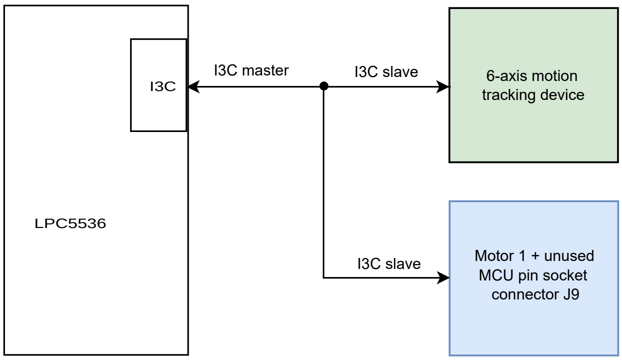

I3C
LPC5536
I3C master
I3C slave
6-axis motion tracking device
I3C slave
Motor 1 + unused MCU pin socket connector J9

Figure 11. I3C diagram

Table 18 describes the I3C slave devices on the LPCXpresso55S36 board.

Table 18. I3C slave devices

Part identifier	Manufacturing part number	Description
U34	ICM-42688-P	6-axis motion tracking device with integrated 3-axis gyroscope and 3-axis accelerometer
J9		One of the four connectors of the Motor 1 control + unused MCU pin socket

2.10 Octal flash memory

The target MCU has a flexible SPI (FlexSPI) host controller, which supports Single/Dual/Quad/Octal mode data transfer (1/2/4/8 bidirectional data lines). The LPCXpresso55S36 board supports communication with the FlexSPI host controller through an octal flash memory. The board also supports populating a flash connector, which can be used to connect an external flash memory to be used as an alternative to the onboard octal flash memory.

The routing of signals between the target MCU and octal flash memory (or external flash connector) is controlled through various configuration resistors. The configuration resistors of the octal flash memory are populated, allowing it to communicate to the target MCU. The external flash connector and its configuration resistors are not populated on the board.

The octal flash memory and the supported external flash connector are described in Table 19.

Table 19. Octal flash memory and external flash connector

Part identifier	Manufacturing part number	Description
U29	MX25UM51345GXDI00	24-pin ball grid array (BGA), 512 Mbit octal serial NOR flash memory. The footprint also supports the following part numbers: • S26KS256SDPBHV02 • S27KS0641DPBHI023 • APS6408L-OBM-BA (same pinout as S27KS0641DPBHI023)
J45	QSE-014-01-F-D-DP-A	2x14 pin header for connecting an external flash memory

The default configuration for the LPCXpresso55S36 board uses a 3.3 V supply rail for the target MCU. However, the octal flash memory connected on the board is not a 3.3 V device. Therefore, by default, the octal flash memory is disconnected from the target MCU signals to protect it from any damage.

To use the flash memory, change the settings of the jumpers listed in Table 20 from default settings to octal flash settings. This selects the 1.9 V supply rail for the target MCU and the flash memory, and connects the FlexSPI signals to the flash memory. If you need to change the supply rail back to 3.3 V, then ensure to change the octal flash jumpers back to the default settings to avoid damaging the flash.

Table 20. Octal flash specific jumpers

Jumper	Octal flash signal at jumper	Default jumper setting	Jumper setting for octal flash signal
JP1	VCC_3V3_1V9	1-2 shorted (VCC_3V3)	2-3 shorted (VCC_1V9)
JP56	VCC_FLASH	2-3 shorted (VCC_1V9)	2-3 shorted (VCC_1V9)
JP63	RST_FLASH	Open	Shorted
JP41	FLEXSPI0_D3/P0_2	2-3 shorted	1-2 shorted
JP42	FLEXSPI0_D0/P0_6	2-3 shorted	1-2 shorted
JP43	FLEXSPI0_DQS/P0_25	2-3 shorted	1-2 shorted
JP44	FLEXSPI0_D2/P0_3	2-3 shorted	1-2 shorted
JP45	FLEXSPI0_SS0_N/P0_21	2-3 shorted	1-2 shorted
JP64	FLEXSPI0_D7/P1_29	2-3 shorted	1-2 shorted
JP65	FLEXSPI0_D6/P1_27	2-3 shorted	1-2 shorted
JP66	FLEXSPI0_D4/P1_16	Open	Shorted
JP67	FLEXSPI0_D5/P1_15	Open	Shorted
JP68	FLEXSPI0_SCLK/P0_19	Open	Shorted
JP69	FLEXSPI0_D1/P0_4	Open	Shorted

2.11 DMIC and external DMIC connector

The target MCU has a digital microphone interface subsystem, which includes a dual-channel (channels 0 and 1) digital PDM microphone (DMIC) module. The LPCXpresso55S36 board supports communication with the DMIC module of the target MCU through a DMIC device. The board also provides a DMIC connector, which can be used to connect an external DMIC device to be used as an alternative to the onboard DMIC.

The routing of signals between the target MCU and onboard DMIC (or external DMIC connector) is controlled through various configuration resistors. The configuration resistors of the onboard DMIC are populated, allowing it to communicate to the target MCU. The configuration resistors of the external DMIC connector are not populated on the board.

The onboard DMIC and external DMIC connector are described in Table 21.

Table 21. Onboard DMIC and external DMIC connector

Part identifier	Manufacturing part number	Description
U30	SPH0641LM4H-1	Onboard DMIC, placed on the bottom side of PCB. The voice comes from the top side of PCB via a through-hole.
J41	TSW-107-07-F-D	2x7 pin header for connecting an external DMIC

2.12 mikroBUS socket

A mikroBUS socket is a pair of 1x8 position receptacles (connectors) with a proprietary pin configuration and silkscreen markings. It allows maximum hardware expandability with smallest number of pins.

The LPCXpresso55S36 board has a mikroBUS socket with two 1x8 position receptacles, J7 and J8. Figure 12 shows the silkscreen markings and pinouts of the mikroBUS socket connectors with explanation.

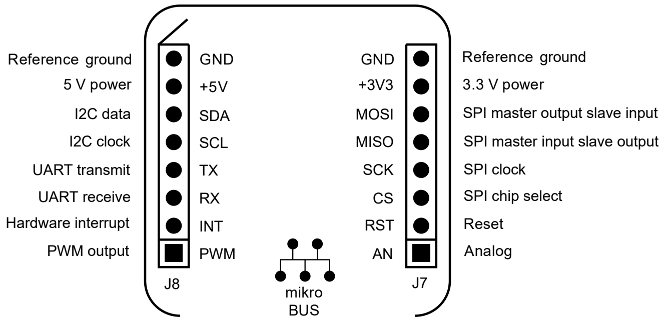


Figure 12. mikroBUS socket connector pinouts

The LPCXpresso55S36 mikroBUS socket supports different types of add-on boards, called *click boards*, which are plug and play solutions to add new functionality to the board design. A click board has a pair of 1x8 pin headers that connect to the two receptacles of a mikroBUS socket.

MikroElektronika (MIKROE) is one of the manufactures of click boards. You can find details of some example click boards for LPCXpresso55S36 mikroBUS socket at MIKROE website.

2.13 Motor control and Arduino sockets

The LPCXpresso55S36 board has two Arduino sockets, each with four connectors (2x6, 2x8, 2x10, and 2x8 position receptacles). In each socket, the two 2x8 position receptacles are placed diagonally opposite to each other. Each socket is pin-compatible with an Arduino Uno revision 3 (R3) board. Each socket allows communication with one FlexPWM module of motor control subsystem of the target MCU (LPC5536).

2.13.1 Motor 1 control + unused MCU pin socket

The Motor 1 control + unused MCU pin socket allows communication with the following modules/pins of the target MCU:

- FlexPWM0 module of motor control subsystem
- Analog-to-digital converter (ADC) modules, ADC0 and ADC1
- Digital-to-analog converter (DAC) modules, DAC0 and DAC1
- Some of the unused MCU pins

The socket includes the following four connectors:

- J13 (2x6 position receptacle)
- J10 (2x8 position receptacle)
- J9 (2x10 position receptacle)
- J12 (2x8 position receptacle)

Figure 13 shows the pinouts of Motor 1 control + unused MCU pin socket connectors. Odd pins (inner side pins) of the socket are used for motor control. Even pins (outer side pins) of the socket are either connected to unused MCU pins or used for other purposes.

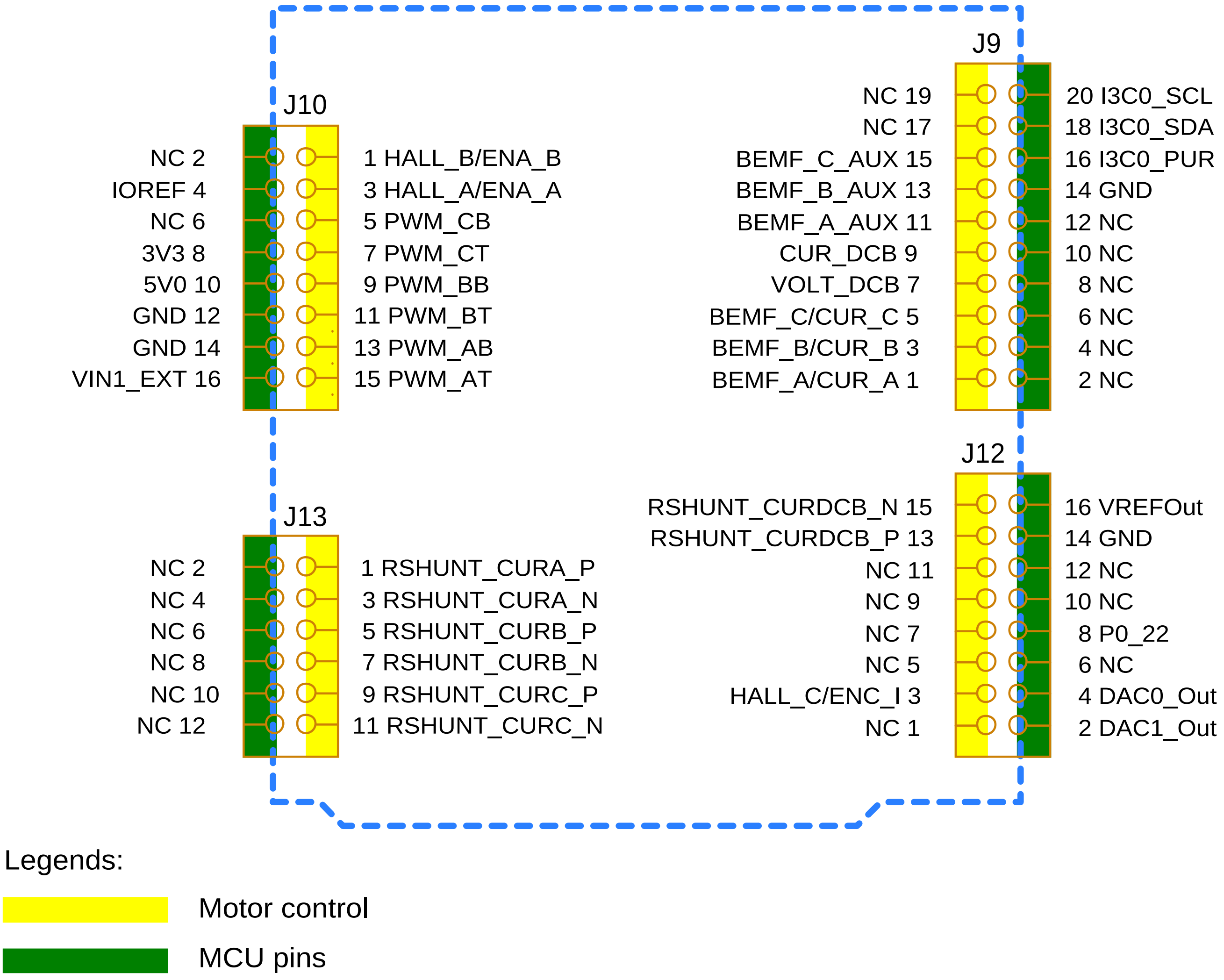


Figure 13. Motor 1 control + unused MCU pin socket connector pinouts

Jumper JP58, which is placed between signals on pins 16 and 18 of connector J9, is used to enable/disable pullup resistor control for I3C master.

2.13.2 Motor 2 control + Arduino socket

The Motor 2 control + Arduino socket allows communication with the following modules of the target MCU:

- FlexPWM1 module of motor control subsystem
- Analog-to-digital converter (ADC) modules, ADC0 and ADC1
- HS SPI controller

The Motor 2 control + Arduino socket is pin-compatible with an Arduino UNO board. The socket includes the following four connectors:

- J132 (2x6 position receptacle)
- J102 (2x8 position receptacle)
- J92 (2x10 position receptacle)
- J122 (2x8 position receptacle)

Figure 14 shows the pinouts of Motor 2 control + Arduino socket connectors. Odd pins (inner side pins) of the socket are used for motor control. Even pins (outer side pins) of the socket provide connection with an external Arduino board.

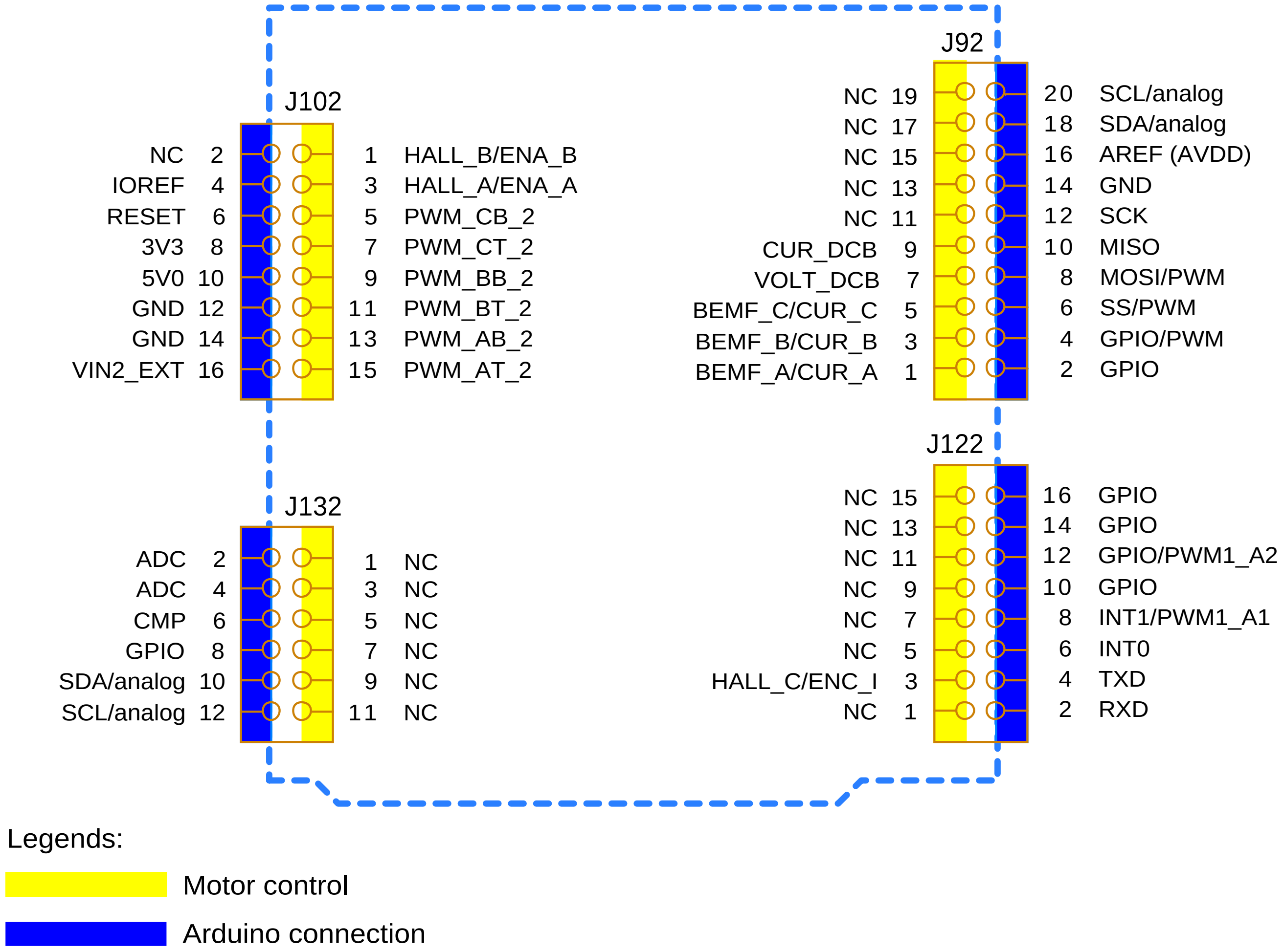


Figure 14. Motor 2 control + Arduino socket connector pinouts

2.14 GPIOs and interrupts

Table 22 lists the target MCU pins that are used as general-purpose inputs/outputs (GPIOs) on the LPCXpresso55S36 board. Some of these pins are also used for other purposes on the board.

Table 22. LPCXpresso55S36 GPIOs

Target MCU pin	Board devices where GPIO function is used	GPIO mode control
PIO1_14	Audio expansion connector (pin 21)	Only for GPIO function
PIO1_13	Motor 2 control + Arduino socket (pin 8 of connector J132)	Only for GPIO function
PIO1_27	Motor 2 control + Arduino socket (pin 2 of connector J92)	To enable GPIO mode, short pins 2-3 of jumper JP65
PIO1_29	Motor 2 control + Arduino socket (pin 4 of connector J92)	To enable GPIO mode, short pins 2-3 of jumper JP64
	Audio expansion connector (pin 22)	

Table 22. LPCXpresso55S36 GPIOs*...continued*

Target MCU pin	Board devices where GPIO function is used	GPIO mode control
PIO1_8	Motor 2 control + Arduino socket (pin 10 of connector J122)	To enable GPIO mode, set configuration resistor R523 in position B (2-3 shorted)
PIO1_25	Motor 2 control + Arduino socket (pin 12 of connector J122)	GPIO mode is enabled by default (pins 1-2 of jumper JP52 are shorted)
PIO1_0	Motor 2 control + Arduino socket (pin 14 of connector J122)	Only for GPIO function
PIO1_28	Motor 2 control + Arduino socket (pin 16 of connector J122)	Only for GPIO function

The onboard audio codec is the only interrupt source on the LPCXpresso55S36 board. Interrupt triggered by audio codec is received at the target MCU GPIO pin PIO1_13.

2.15 Board errata

Table 23 summarizes all known errata for the LPCXpresso55S36 board.

Table 23. LPCXpresso55S36 board errata summary

Erratum	Workaround	Applicable board revision	Applicable schematics revision
Erratum 36: D19 LED is placed on bottom side instead of top side	None	C	SCH-55007_A / SPF-47745_C
Erratum 46: JP57 jumper is DNP	Populate and short JP57 jumper	C	SCH-55007_A / SPF-47745_C
Erratum 47: Incorrect silkscreen labels for some J9 and J10 pins	See Section 2.13.1	C	SCH-55007_A / SPF-47745_C
Erratum 48: DMIC PDM clock signals indicate incorrect clock	CLK0 in the DMIC PDM signal names should be read as CLK1	C	SCH-55007_A / SPF-47745_C
Erratum 51: To use PDM DATA0, PDM CLK1 must be enabled	Set the EN_CH1 bit of DMIC register CHANEN to 1	C, D	SCH-55007_A / SPF-47745_C, SPF-47745_D

2.15.1 Erratum 36: D19 LED is placed on bottom side instead of top side

2.15.1.1 Description

MCU-Link LED D19, which indicates if USBSIO bridge traffic is present, is placed on the bottom side of the board. Verifying the LED status while working on the board in inconvenient as the LED is placed on the bottom side.

2.15.1.2 Impact

Difficultly in verifying the LED status.

2.15.1.3 Workaround

None.

2.15.1.4 Fix plan

D19 LED will be moved to the top side of the board in next board revision.

2.15.2 Erratum 46: JP57 jumper is DNP

2.15.2.1 Description

Jumper JP57, which disables the USB-to-SPI bridge between MCU-Link and the target MCU, is not populated on the board. By default, the USB-to-SPI bridge is enabled and the target MCU SPI connection is used with MCU-Link. To use the target MCU SPI connection with other peripherals, such as SPI NOR flash memory, JP57 jumper must be populated and shorted.

2.15.2.2 Impact

The target MCU cannot connect to the SPI NOR flash memory.

2.15.2.3 Workaround

Populate and short JP57 jumper.

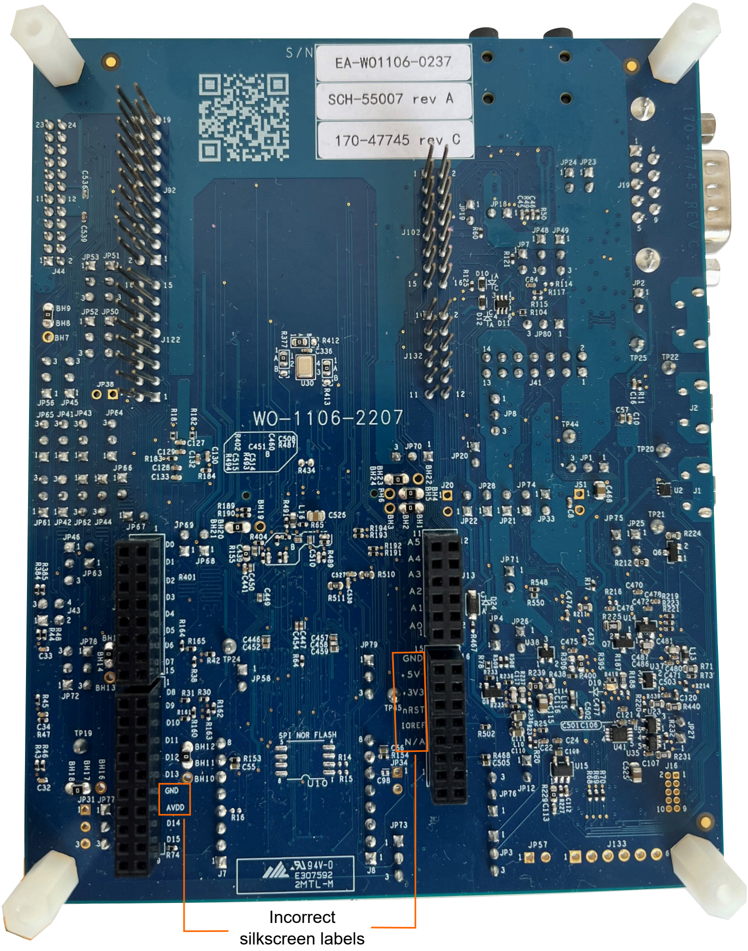
2.15.2.4 Fix plan

JP57 jumper will be populated and shorted in next board revision.

2.15.3 Erratum 47: Incorrect silkscreen labels for some J9 and J10 pins

2.15.3.1 Description

Some pins of Motor 1 control socket connectors J9 and J10, which are placed on the bottom side of the board, have incorrect silkscreen labels. The J9 and J10 pin silkscreen labels highlighted in Figure 15 are incorrect.



Incorrect silkscreen labels

Figure 15. Incorrect J9 and J10 pin silkscreen labels

2.15.3.2 Impact

Difficulty in determining correct J9 and J10 pins.

2.15.3.3 Workaround

See Section 2.13.1 to determine correct J9 and J10 pins.

2.15.3.4 Fix plan

Silkscreen labels will be corrected for J9 and J10 pins in next board revision.

2.15.4 Erratum 48: DMIC PDM clock signals indicate incorrect clock

2.15.4.1 Description

In the LPC5536 MCU, the DMIC PDM clock (at the MCU pin PIO_7) uses DMIC PDM channel 1. However, in the LPCXpresso55S36 board schematics, the DMIC PDM clock signals are named with CLK0, instead of CLK1. For example, two DMIC PDM clock signals are named as P0_7-PDM_CLK0_ISP1 and P0_7-PDM_CLK0_OnBoard.

2.15.4.2 Impact

DMIC PDM clock signal names are confusing.

2.15.4.3 Workaround

CLK0 in DMIC PDM clock signals in the board schematics should be read as CLK1. For example, P0_7-PDM_CLK0_ISP1 and P0_7-PDM_CLK0_OnBoard should be read as P0_7-PDM_CLK1_ISP1 and P0_7-PDM_CLK1_OnBoard, respectively.

2.15.4.4 Fix plan

The signal names will be corrected in the next revision of the board schematics.

2.15.5 Erratum 51: To use PDM DATA0, PDM CLK1 must be enabled

2.15.5.1 Description

On the LPCXpresso55S36 board, PDM DATA0 at the target MCU pin PIO0_27 is paired with the PDM clock CLK1 (instead of CLK0) at the target MCU pin PIO0_7 and the PDM data and clock signals are connected to the DMIC interface. To use PDM DATA0 at PIO0_27, PDM CLK1 at PIO0_7 must be enabled.

2.15.5.2 Impact

The DMIC interface of the board cannot be used.

2.15.5.3 Workaround

Enable PDM CLK1 at PIO0_7 by setting the EN_CH1 bit (bit 1) of DMIC register CHANEN to 1. Additionally, enable DMIC internal clock by configuring the PDMDIV bits of DMIC register DIVHFCLK1.

2.15.5.4 Fix plan

No fix planned.

2.16 Board operating conditions

The LPCXpresso55S36 board should be used in laboratory conditions (0 ℃ to 50 ℃). It is not intended for extended temperature testing. See *LPC5536 Data Sheet* for complete details on device operating conditions.

3 MCU-Link OB debug probe

This section describes the MCU-Link OB debug probe and explains how to connect it to the target MCU (LPC5536).

3.1 MCU-Link overview

MCU-Link is a debug probe architecture jointly developed by NXP and Embedded Artists. The MCU-Link architecture is based on the LPC55S69 MCU, which is based on the Arm Cortex-M33 core.

NXP uses MCU-Link OB on its evaluation boards, for example, LPCXpresso55S36. MCU-Link is also available as a standalone debug probe in the following two types:

- MCU-Link (base model): For more details, see https://www.nxp.com/design/microcontrollers-developer-resources/mcu-link-debug-probe:MCU-LINK
- MCU-Link Pro (fully featured): For more details, see https://www.nxp.com/design/microcontrollers-developer-resources/mcu-link-pro-debug-probe:MCU-LINK-PRO

All these MCU-Link probes use the same firmware.

NXP evaluation boards have some configuration strap pins that determine the type of MCU-Link used. The MCU-Link firmware reads these pins during boot process to determine the MCU-Link type.

In the LPCXpresso55S36 board, some of the strap pins can be configured via jumpers; however, most of these configuration changes are made before powering up the board.

3.1.1 Supported firmware options

Supported firmware options for the LPCXpresso55S36 MCU-Link debug probe are described below:

- By default, the LPCXpresso55S36 MCU-Link debug probe is programmed with firmware based on CMSIS-DAP protocol from NXP. This firmware also supports all other features supported in hardware. For information on how to update MCU-Link firmware, see Section 3.2.
- A custom version of J-Link Lite is also available but this firmware is limited to debug and VCOM features
- Other firmware options available for MCU-Link OB may not provide the buffer enable / direction control support that is required to correctly configure the LPCXpresso55S36 hardware. Therefore, care should be taken while using such firmware.

3.1.2 Using MCU-Link with development tools

The MCU-Link debug probe can be used with IDEs supported within the MCUXpresso ecosystem (MCUXpresso IDE, IAR Embedded Workbench, and Keil MDK). To get started on one of these IDEs:

1. Go to the LPCXpresso55S36 Development Board page on the NXP website.
2. Click the **GET STARTED** button next to the **BUY OPTIONS** button.

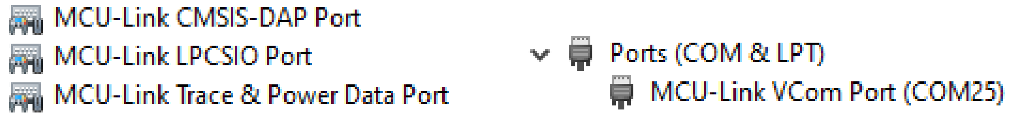
Note: *Other IDEs that support CMSIS-DAP or J-Link protocol can also use the MCU-Link debug probe; refer to the documentation for these IDEs for more information.*

3.2 Installing device drivers and updating MCU-Link firmware

MCU-Link is supported on host computers running on Windows 10, MacOS X, and Ubuntu Linux operating systems (OSs). For each of these OSs, an MCU-Link firmware package is available that includes the host device drivers, MCU-Link firmware, and scripts to program CMSIS-DAP and J-Link firmware. The host device drivers are included in the firmware package to configure the host so that it displays user-friendly device names.

To download and install the host device drivers and update the MCU-Link firmware, follow these steps:

1. Go to the MCU-Link Debug Probe page on the NXP website.
2. Click **Design Resources**, and then click **SOFTWARE**. Installation packages for all three OSs are displayed.
3. Download the firmware package applicable to your host OS.
4. Depending on the host OS, install the firmware package (Linux/MacOS) or execute the installer program (Windows). The firmware package is installed/unzipped to the MCU-LINK_installer_Vx_xxx directory (where Vx_xxx indicates the version number, for example, V2_220). The directory has a "Readme.txt" file and few subdirectories that contain other package components, including host device drivers, MCU-Link firmware, and scripts to program CMSIS-DAP and J-Link firmware.
   Host drivers are usually installed automatically during firmware package installation. In case you need to install the drivers manually, refer to the "Host Configuration" section of the "Readme.txt" file.
   Before powering up the board for updating MCU-Link firmware, you must switch MCU-Link to ISP mode (USB1).
5. To force MCU-Link to boot in ISP mode (USB1), short jumper JP25.
6. Connect the J1 connector on the board to the USB port of the host computer through a USB micro-B cable. MCU-Link gets powered up in ISP mode (USB1). The red MCU-Link status LED (D16) lights up (for more details on MCU-Link LEDs, see Section 3.9). The board gets enumerated as a human interface device (HID) class device.
7. Program the MCU-Link firmware into the MCU-Link internal flash using the instructions provided in the "Firmware Installation Guide" section of the "Readme.txt" file. Use the scripts provided to program the CMSIS-DAP or J-Link firmware.
8. Disconnect the board from the host computer, remove jumper JP25, and reconnect the board.
   Now, the system behaves differently based on whether the CMSIS-DAP or J-Link firmware was programmed:
   - If the CMSIS-DAP firmware was programmed, then the red status LED (D16) fades in/out repeatedly (“breathing”), while the green USB status LED (D14) stays ON constantly. In this case, four devices (three HID devices and one VCOM port) get enumerated, as shown in Figure 16.



MCU-Link CMSIS-DAP Port
MCU-Link LPCSIO Port
MCU-Link Trace & Power Data Port

Ports (COM & LPT)
MCU-Link VCom Port (COM25)

Figure 16. MCU-Link enumerated devices for CMSIS-DAP firmware

   - If the J-Link firmware was programmed, then only the green VCOM LED (D17) is used to indicate status. The VCOM LED lights up when MCU-Link boots. Then, when you start a new debug session, the LED blinks to indicate that debug activity is in progress. In this case, two devices (one J-Link driver USB device and one J-Link CDC UART port VCOM port) get enumerated.

Now, your LPCXpresso55S36 board is ready for use. If you use the board with MCUXpresso IDE version 11.3 or higher, you are notified in case a more recent firmware version is available for MCU-Link. If you use the board with a different IDE, ensure that latest MCU-Link firmware version is installed on the board.

3.3 Supported MCU-Link features

MCU-Link includes several mandatory and optional features. Table 24 summarizes the MCU-Link features supported on the LPCXpresso55S36 board.

Table 24. Supported MCU-Link features

Feature	Description
Serial wire debug (SWD) / serial wire debug trace output (SWO)	Allows SWD-based debugging with SWO for profiling and/or low overhead debug standard I/O communication
Virtual communication (VCOM) serial port	Adds a serial COM port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-UART bridge

Table 24. Supported MCU-Link features*...continued*

Feature	Description
USB serial input/output (USBSIO[1]) port	Adds a USB serial I/O port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-SPI bridge or USB-to-I2C bridge. It also allows MCU-Link to be used as a USB-to-GPIO bridge to connect the MCU-Link LPC55S69 target to the host computer.
External debug probe support	Allows debugging the target MCU (LPC5536) using an external debug probe, instead of MCU-Link. Support for external debug probe is enabled by disabling SWD feature. While using external debug probe, VCOM and USBSIO features can be used.
External target support[1]	Allows debugging an external target MCU using MCU-Link
Energy/power/current/voltage consumption measurement[1] (MCUXpresso IDE only)	Allows onboard measurement of current drawn by the target MCU

[1] J-Link firmware does not support this feature.

3.4 Supported debug scenarios

In the LPCXpresso55S36 board, the MCU-Link debug probe target can be either the LPC5536 MCU or an external target compliant with MCU-Link. The board also allows you to use an external debugger for debugging the LPC5536 MCU, in place of the MCU-Link debug probe.

Table 25 describes the debug scenarios supported on the LPCXpresso55S36 board.

Table 25. Supported debug scenarios

Debug scenario	Feature support
Use MCU-Link as debugger for the target MCU (LPC5536)	• Serial wire debug (SWD): – MCU-Link SWD feature is enabled (MCU-Link SWD disable jumper JP27 is open) – The target MCU SWD interface is connected to MCU-Link – The target MCU SWD connector J38 is not used for external connection • Virtual communication (VCOM): MCU-Link VCOM feature is enabled • USB serial input/output (USBSIO): MCU-Link USBSIO feature is enabled • Target selection: Target selection jumper JP7 is open • Target power selection: Target power selection jumper JP8 is set to 1-2 shorted
Use an external debugger to debug the target MCU (LPC5536)	• SWD: – Disable MCU-Link SWD feature by shorting MCU-Link SWD disable jumper JP27 – Connect the external debugger to the target MCU SWD connector J38. The target MCU SWD interface is connected to the external debugger. • VCOM: MCU-Link VCOM feature can be used • USBSIO: MCU-Link USBSIO feature can be used • Target selection: Target selection jumper JP7 is open • Target power selection: Target power selection jumper JP8 is set to 1-2 shorted
Use MCU-Link as debugger for an external target MCU	• SWD: – MCU-Link SWD feature is enabled (MCU-Link SWD disable jumper JP27 is open) – Connect the external target MCU to the target MCU SWD connector J38 • VCOM: MCU-Link VCOM feature is not supported • USBSIO: MCU-Link USBSIO feature is not supported

Table 25. Supported debug scenarios*...continued*

Debug scenario	Feature support
	• Target selection: Short target selection jumper JP7 • Target power selection: – If the external target MCU has to get power from the board, then set target power selection jumper JP8 to 1-2 shorted – If the external target MCU has to use its own power, then set jumper JP8 to 2-3 shorted

3.5 Connecting to a target through a USB-to-UART bridge

MCU-Link supports the VCOM serial port feature, which adds a serial COM port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-UART bridge.

In the LPCXpresso55S36 board, MCU-Link is connected to the Flexcomm 0 (USART) port of the target MCU through a voltage translator U22, which enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to 3.3/1.9 V and vice versa.

To use MCU-Link as a USB-to-UART bridge, verify the following jumper settings and connect the J1 connector on the board to the USB port of the host computer:

- Jumper JP25 is open (MCU-Link is in Normal mode)
- Jumper JP26 is open (USB-to-UART bridge is enabled)

When you boot the LPCXpresso55S36 board, a VCOM port with the name MCU-Link Vcom Port (COMxx) is enumerated on the host computer, where “xx” may vary from one computer to another. Each MCU-Link based board has a unique VCOM number associated with it.

The USB-to-UART bridge (VCOM feature) can be disabled so that the target MCU USART port can be used for other purposes. For example, to create an external UART connection from USART connecter JP3 using a USB-to-serial cable.

The USB-to-UART bridge can be disabled by disabling voltage translator U22 (setting it to high impedance). To disable U22, unpower the LPCXpresso55S36 board and short jumper JP26. Shorting/opening JP26 after powering up the board has no impact on the functions/features of the MCU-Link firmware. When VCOM feature is disabled, then the target MCU UART port can be used for other purposes.

3.6 Connecting to a target through a USB-to-SPI or USB-to-I2C bridge

MCU-Link supports the USB serial input/output (USBSIO) port feature, which adds a USB serial I/O port on the host computer, and connects it to the target MCU by using MCU-Link as a USB-to-SPI bridge or USB-to-I2C bridge. Support for the USBSIO feature can be enabled on the host computer using the libusbsio library, which is a free host library from NXP for Windows/Linux/MacOS systems. For more details on the libusbsio library, see https://www.nxp.com/libusbsio.

In the LPCXpresso55S36 board, MCU-Link is connected to the Flexcomm 8 (HS SPI) port of the target MCU using slave select 1 connection, through a voltage translator U24, which enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to 3.3/1.9 V and vice versa.

MCU-Link is also connected to the Flexcomm 1 (I2C) port of the target MCU through an I2C repeater / a voltage translator U27, which enables communication between MCU-Link and the target MCU, by shifting voltage levels of signals between the two devices from 3.3 V to 3.3/1.9 V and vice versa.

To use MCU-Link as a USB-to-SPI or USB-to-I2C bridge, the board must be connected to the host computer through a USB cable from its J1 connector. A USB-to-SPI bridge can be used to program the target MCU using the ISP mode and/or to emulate the host system. A USB-to-I2C bridge can be used to program the target

MCU using the ISP mode and/or to emulate the host system / board peripherals. NXP MCUXpresso Secure Provisioning Tool (SEC) provides an option to program the target via a USB-to-SPI or USB-to-I2C bridge (see MCUXpresso SEC documentation for more details).

The USBSIO feature can be disabled for SPI or I2C so that the target MCU SPI/I2C port can be used for other purposes. Disabling this feature instructs the firmware not to enumerate the USB endpoint for USBSIO (which is called “MCU-Link LPCSIO” for backward compatibility reasons). Disabling the USBSIO feature also frees more USB bandwidth for the SWO profiling and energy measurement features of MCU-Link.

The USBSIO feature can be disabled for SPI by disabling voltage translator U24 (setting it to high impedance). By default, U24 is disabled (jumper JP57 is shorted); therefore, the USBSIO feature is disabled for SPI. The feature can be enabled by opening jumper JP57. Shorting/opening JP57 after powering up the board has no impact on the functions/features of the MCU-Link firmware. When USBSIO feature is disabled for SPI, then the target MCU SPI port can be used for other purposes.

The USBSIO feature can be disabled for I2C by opening jumper JP4.

3.7 Connecting to a target through a USB-to-GPIO bridge

The MCU-Link USBSIO feature also allows MCU-Link to be used as a USB-to-GPIO bridge to connect the MCU-Link LPC55S69 target to the host computer. Support for the USBSIO feature can be enabled on the host computer using the libusbsio library, which is a free host library from NXP for Windows/Linux/MacOS systems. For more details on the libusbsio library, see https://www.nxp.com/libusbsio.

The USB-to-GPIO bridge can be used to control a limited number of GPIO pins of LPC55S69. The LPCXpresso55S36 board supports a 1x6 pin GPIO header J133 (not populated on the board) for accessing these GPIO pins on the board.

Table 26 lists the LPC55S69 GPIO pins that can be controlled through USB-to-GPIO bridge and shows how these pins are mapped to GPIO header pins.

Table 26. USB GPIO pin assignments

LPC55S69 pin	GPIO header J133 pin	Signal direction
PIO1_1	Pin 1	Output from MCU-Link
PIO1_9	Pin 2	Output from MCU-Link
PIO1_20	Pin 3	Input to MCU-Link
PIO1_21	Pin 4	Output from MCU-Link
PIO1_31	Pin 3	Output from MCU-Link
PIO1_7	Pin 6	Input to MCU-Link

3.8 Measuring target MCU power consumption

The LPCXpresso55S36 board includes circuitry to measure voltage, current, and power/energy consumption for the target MCU (LPC5536) by the onboard MCU-Link debug probe. This measurement data can be analyzed and displayed using MCUXpresso IDE version 11.5.1 or later. For more details on the MCUXpresso IDE, see https://www.nxp.com/mcuxpresso/ide.

It is recommended to use the latest MCU-Link firmware. If the latest firmware version is not installed, then the MCUXpresso IDE shows a message indicating that the firmware version is not the latest one. For instructions on updating MCU-Link firmware, see Section 3.2.

The power measurement feature is intended for low power measurements, with the target MCU running at up to 3.6 V. The design uses the built-in 16-bit ADC of the MCU-Link, with power data sampled at up to 100 kS/s. At high sample rates, MCUXpresso IDE fails to capture all data; therefore, you must adjust the sample rate using the configuration options available in the energy measurement configuration settings of the tool.

When MCU-Link is powered on, the measurement circuit initiates self-calibration. No user intervention is needed for setting up calibration or adjusting measurement ranges.

Table 27 shows current measurement configuration settings. The maximum measurable current is 50 mA. Accuracy may vary with temperature and it is only mentioned for reference purposes.

Table 27. Current measurement configuration settings

Measurement range	Resolution	Accuracy (typical)
200 nA to 400 μA	200 nA	1%
> 400 μA to 50 mA	5 μA	1%

In the LPCXpresso55S36 board, the onboard MCU-Link is used to measure the current drawn by the target MCU. For current measurement, the supported voltage range is 1.7 V – 3.6 V. By default, current measurement is disabled on the board.

Table 28 shows the power supplies and jumpers corresponding to various currents of the target MCU (LPC5536).

Table 28. Target MCU currents

Target MCU current	Power supply	Jumper that drives power supply
IDD current (total current)	MCU_VCC_3V3_1V9	JP33 (shorted by default)
VDD current	MCU_VDD	JP20 (shorted by default)
VDDA current	MCU_VDDA	JP21 (shorted by default)
VBAT current	MCU_VBAT	JP22 (shorted by default)
VDD_MAIN current	MCU_MAIN	JP28 (shorted by default)

For measuring the current drawn from a board power supply, the power supply is connected in line with the current measurement circuit through jumpers JP74 and JP75. By default, these two jumpers are shorted and they connect the MCU_VCC_3V3_1V9 supply (power supply corresponding to the total target MCU current (IDD)) to the current measurement circuit.

To measure the total current drawn by the target MCU, configure the MCU_VCC_3V3_1V9 supply as follows:

1. Open jumper JP33, which drives the MCU_VCC_3V3_1V9 supply. When JP33 is shorted, the MCUXpresso energy measurement tool shows zero current because the measurement circuit is bypassed (short circuited by JP33). When JP33 is open, the energy measurement tool can read the MCU_VCC_3V3_1V9 supply current correctly.
2. Ensure that jumpers JP74 and JP75 are shorted.

To measure an individual target MCU current (VDD, VDDA, VBAT, or VDD_MAIN), configure the corresponding power supply as follows:

1. Ensure that jumper JP33 is shorted.
2. Open the jumper corresponding to the power supply.
3. Open jumpers JP74 and JP75.
4. Connect pin 1 of the power supply jumper to pin 2 of JP74.
5. Connect pin 2 of the power supply jumper to pin 2 of JP75.

3.9 MCU-Link status LEDs

The LPCXpresso55S36 board has seven status indicator LEDs for MCU-Link. Table 29 lists these LEDs and describes how each LED behaves in different MCU-Link modes.

Table 29. MCU-Link LEDs

Part identifier	LED name / color	MCU-Link mode		
		Normal mode (with CMSIS-DAP firmware)	**Normal mode (with J-Link firmware)**	**Firmware Update (ISP) mode**
D14	USB / green	Lights up after successful USB enumeration at startup. Afterward, the LED stays ON.	Remains OFF	Remains OFF
D15	SWO / green	Indicates if serial wire debug trace output (SWO) data is being received from the target MCU	Remains OFF	Remains OFF
D16	Status / red	Indicates heartbeat (fades in/out repeatedly), with SWD activity overlaid. The LED blinks rapidly at startup, if an error occurs.	Remains OFF	Lights up when MCU-Link target (LPC55S69) boots in ISP mode
D17	VCOM / green	Indicates if VCOM port is transmitting/receiving data	Lights up when MCU-Link boots, and then blinks when debug activity happens	Remains OFF
D18	TPWR / green	Reserved for future use	Remains OFF	Remains OFF
D19	SIO / green	Indicates if USBSIO bridge traffic is present	Remains OFF	Remains OFF
D20	NRG / green	Indicates if energy measurement communication is happening between the target MCU and MCU-Link	Remains OFF	Remains OFF

4 Revision history

Table 30 summarizes the revisions to this document.

Table 30. Revision history

Revision number	Release date	Description
3	17 August 2023	Updated board pictures in Section 1.6
		Updated jumper J57 details in Table 6
		Updated slave select 1 connections in Figure 9 and Table 16
		Updated Table 23 and text above it
		Updated jumper J57 details in Section 3.6
		Updated D18 LED name in Table 29
2	5 September 2022	Added new board pictures in Section 1.6
		Updated details related to jumpers JP74 and JP75 in Table 6
		In Section 2.1: • Added a link to the "Measuring target MCU power consumption" section • Removed subsection "Current measurement"
		Updated Section 2.15 and added five subsections to it
		Updated Section 3.8
1	24 January 2022	In Section 1.8: • Removed jumper JP30 details • Changed default settings of jumpers JP41, JP42, JP43, JP44, JP45, JP46, JP50, JP51, JP52, JP53, JP63, JP64, JP65, JP66, JP67, JP68, and JP69 • Added new jumpers JP72, JP73, JP74, JP75, JP76, JP77, JP78, JP79, and JP80
		In Table 9, changed power supply for onboard DMIC and external DMIC connector from VCC_1V9 to VCC_3V3_1V9
		Updated Table 20 and text above it
		Added a new paragraph at the end of Section 2.11
		In Section 2.12: • Updated Figure 12 • Removed the "Example click boards" table
		Updated signals at pins 13 and 15 of connector J12 in Figure 13
		Updated Section 2.15
		Restructured Section 3
		Updated Section 3.1
		Updated Section 3.1.2 and removed its subsections
		Updated Table 25
0	15 September 2021	Initial release

5 Legal information

5.1 Definitions

Draft — A draft status on a document indicates that the content is still under internal review and subject to formal approval, which may result in modifications or additions. NXP Semiconductors does not give any representations or warranties as to the accuracy or completeness of information included in a draft version of a document and shall have no liability for the consequences of use of such information.

5.2 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. NXP Semiconductors takes no responsibility for the content in this document if provided by an information source outside of NXP Semiconductors.

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 and its suppliers accept 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.

Terms and conditions of commercial sale — NXP Semiconductors products are sold subject to the general terms and conditions of commercial sale, as published at http://www.nxp.com/profile/terms, unless otherwise agreed in a valid written individual agreement. In case an individual agreement is concluded only the terms and conditions of the respective agreement shall apply. NXP Semiconductors hereby expressly objects to applying the customer's general terms and conditions with regard to the purchase of NXP Semiconductors products by customer.

Export control — This document as well as the item(s) described herein may be subject to export control regulations. Export might require a prior authorization from competent authorities.

Suitability for use in non-automotive qualified products — Unless this document expressly states that this specific NXP Semiconductors product is automotive qualified, the product is not suitable for automotive use. It is neither qualified nor tested in accordance with automotive testing or application requirements. NXP Semiconductors accepts no liability for inclusion and/or use of non-automotive qualified products in automotive equipment or applications.

In the event that customer uses the product for design-in and use in automotive applications to automotive specifications and standards, customer (a) shall use the product without NXP Semiconductors' warranty of the product for such automotive applications, use and specifications, and (b) whenever customer uses the product for automotive applications beyond NXP Semiconductors' specifications such use shall be solely at customer's own risk, and (c) customer fully indemnifies NXP Semiconductors for any liability, damages or failed product claims resulting from customer design and use of the product for automotive applications beyond NXP Semiconductors' standard warranty and NXP Semiconductors' product specifications.

Translations — A non-English (translated) version of a document, including the legal information in that document, is for reference only. The English version shall prevail in case of any discrepancy between the translated and English versions.

Security — Customer understands that all NXP products may be subject to unidentified vulnerabilities or may support established security standards or specifications with known limitations. 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 B.V. - NXP B.V. is not an operating company and it does not distribute or sell products.

5.3 Trademarks

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

NXP — wordmark and logo are trademarks of NXP B.V.

Contents

Date of release: 17 August 2023
Document identifier: LPCXpresso55S36UM