ZKit-ARM-1769, ARM Dev. Kit

User Manual

0.1, May 2013

Table of Contents

Chapter 1. Introduction

ZKit-ARM-1769 is a ARM Cortex M3 based micro-controller development kit from Zilogic Systems. ZKit-ARM-1769 is designed for easy usage, rapid prototyping and product design.

1. Features

The ZKit-ARM-1769 has the following features

- Graphics display and on-board keys
- Well defined IO connector interface for I²C, SPI, GPIO and SIO
- Powered through USB port
- Ethernet and CAN network support
- Programmable through USB
- Free and open source compiler and programmer
- Ready to go with Zilogic's Relay, Motor, Display boards etc., add-on boards.

2. Applications

- Motherboard for embedded products
- Embedded application prototyping
- Teaching and learning embedded systems

3. Board Details

The ZKit-ARM-1769 offers the following features:

- NXP LPC1769 micro-controller with 512KB Flash and 64KB RAM
- 120MHz ARM CPU
- Networking Interfaces
  - 10/100Mbps Ethernet Interface
  - Two Channel CAN Bus Interface (2.0 B)
  - RS-485 Bus Interface
  - UART Interfaces (TTL and USB-Serial)
- Device Interface
  - USB 2.0 Device Interface
  - SPI Interface
  - I²C Interface
- Storage
  - microSD Connector
  - 512KB on-chip Flash
  - 2Kb I²C EEPROM
- Analog Interface
  - ADC, 12-bit, 4 channels
  - DAC, 10-bit, 1 channel
  - PWM, 5 channels
- User Interface
  - 128x64 graphics LCD, with backlight
  - Five button keypad

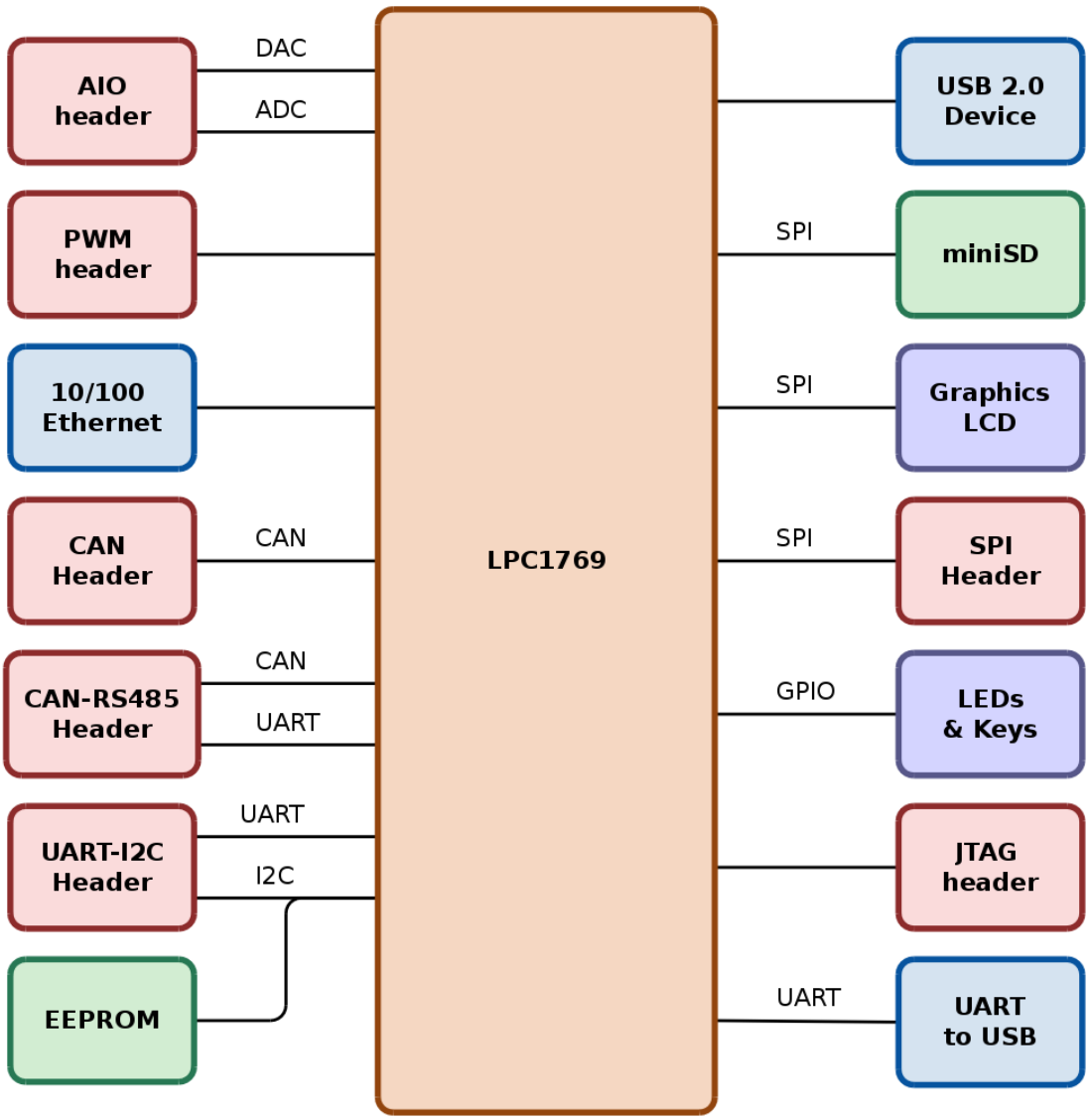
  - 2 debug LEDs
- Flashing/Debugging Interface
  - UART serial console
  - JTAG

Chapter 2. Board Design

1. Overview

A bird's eye view of the devices available on the board, is shown in the following block diagram. Each device connectivity is described in detail in the following sections.

Figure 2.1. Block Diagram



2. Locating Components

The location of the components on the board is indicated in the following diagram.

Figure 2.2. Front View

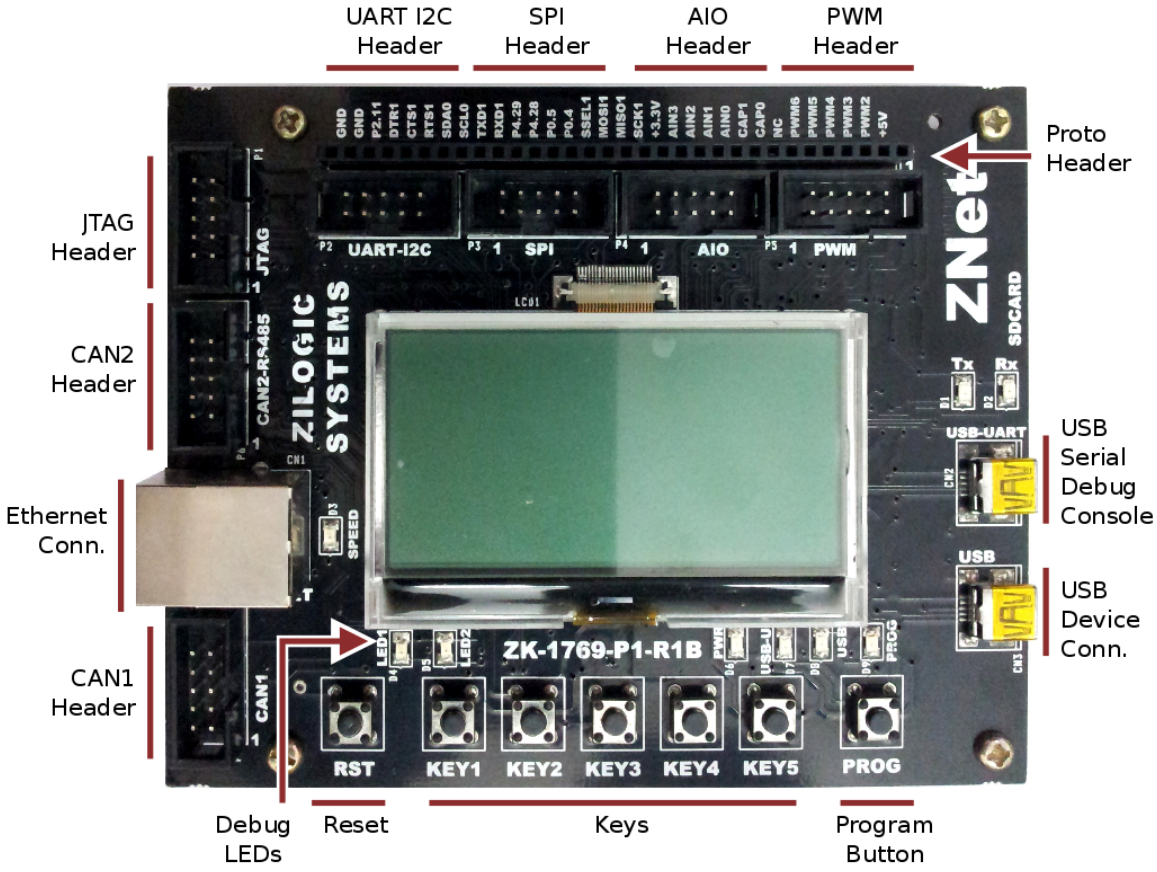
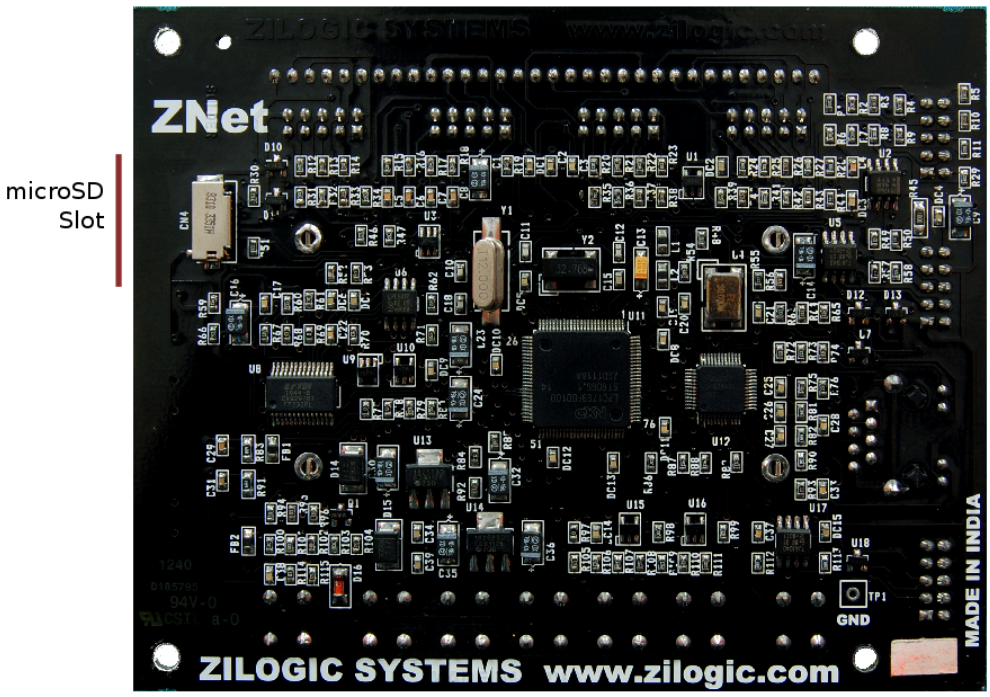


Figure 2.3. Back View



3. Power Supply

The ZKit-ARM-1769 can be powered from a PC USB port, or an external power supply with Mini-B USB connector.

The external power supply, if used, should be a regulated power supply. The regulated power supply should have the following characteristics.

Output Voltage	5V
Output Current	500mA
Connector	Mini-B USB

4. CPU

The heart of the ZKit-ARM-1769 is a NXP LPC1769 micro-controller. The LPC1769 is an 32-bit ARM 3.3V low power micro-controller with 512KB Flash, 64KB of data RAM and supports In-System Programming (ISP).

The main features of the micro-controller are listed below.

- ARM Cortex-M3 processor, running at frequencies of up to 120 MHz
- Up to 512 KB on-chip flash programming memory
- Up to 64 KB On-chip SRAM
- In-System Programming (ISP) and In-Application Programming (IAP)
- Ethernet MAC with RMII interface
- USB 2.0 full-speed device/Host/OTG controller
- Four UARTs with fractional baud rate generation
- CAN 2.0B controller with two channels
- SPI controller with synchronous, serial, full duplex communication
- Two SSP controllers
- Three enhanced I²C bus interfaces
- I2S (Inter-IC Sound) interface
- 70 General Purpose I/O (GPIO) pins
- 12-bit/8-ch Analog/Digital Converter (ADC)
- 10-bit Digital/Analog Converter (DAC)
- Four general purpose timers/counters
- One motor control PWM
- WatchDog Timer (WDT)
- ARM Cortex-M3 system tick timer
- Standard JTAG test/debug interface
- Integrated PMU (Power Management Unit)
- Four low power modes
- Single 3.3 V power supply

5. USB Serial

The ZKit-ARM-1769 has a FT232R USB to UART converter, connected to UART0 of the MCU. The ZKit-ARM-1769 uses the USB UART to provide a serial console interface. It is also used for In-System Programming (ISP), to download the firmware.

5.1. In-System Programming

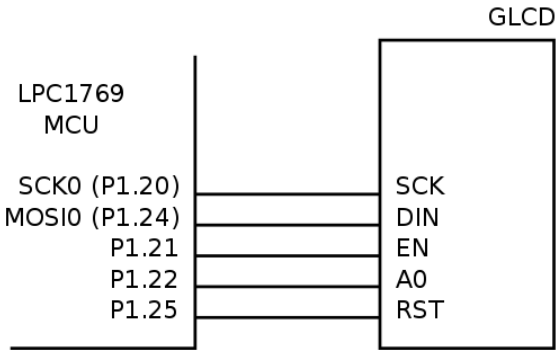
The ZKit-ARM-1769 has a PROG push button, which can be used to select between Programming mode and Serial Communication mode. When the board is powered on, it is in Serial Communication mode. The PROG button, can be used to switch to Programming mode. The current mode is indicated by the PROG LED. ON indicates Programming mode, OFF indicates Serial Communication mode.

In Programming mode, the RTS is connected to the RESET of the MCU, and DTR is connected to the pin P2.10 of the MCU. Flash programming applications like Smash and Flash Magic can utilize this feature to switch the device into ISP mode automatically, without user intervention.

6. LCD Display

The ZKit-ARM-1769 has a 128x64 monochrome GLCD display, TM12864. The TM12864 GLCD has a Sitronix ST7567 controller. The LCD is connected to the SSP0 (SPI controller) of the MCU. The following diagram shows the LCD pin connection details.

Figure 2.4. LCD Connection Diagram

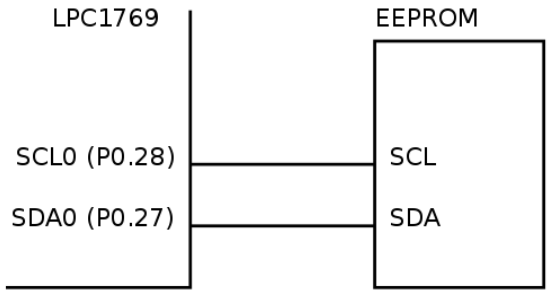


7. I²C EEPROM

The ZKit-ARM-1769 has a CAT24AA02 EEPROM for data storage. The CAT24AA02 is a 2-Kbit Serial EEPROM. The memory is accessed via I²C bus. The maximum bus speed supported by the device is 400 kbit/s

The I²C EEPROM is connected to I²C0 (on-chip I²C controller) of the MCU. The following diagram shows the EEPROM pin connection details.

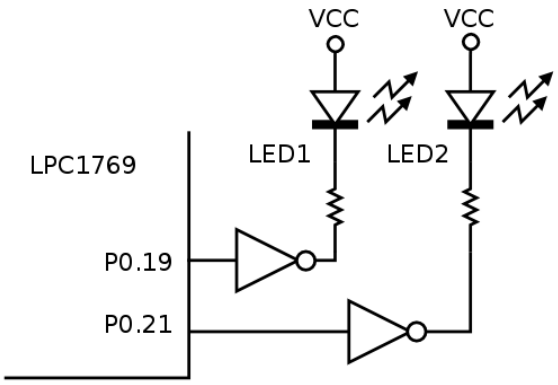
Figure 2.5. I²C EEPROM Connection Diagram



8. Debug LEDs

The ZKit-ARM-1769 has two debug LEDs connected to P0.19 and P0.21, through an buffer. By driving P0.19 and P0.21 high, the LEDs can be switched On.

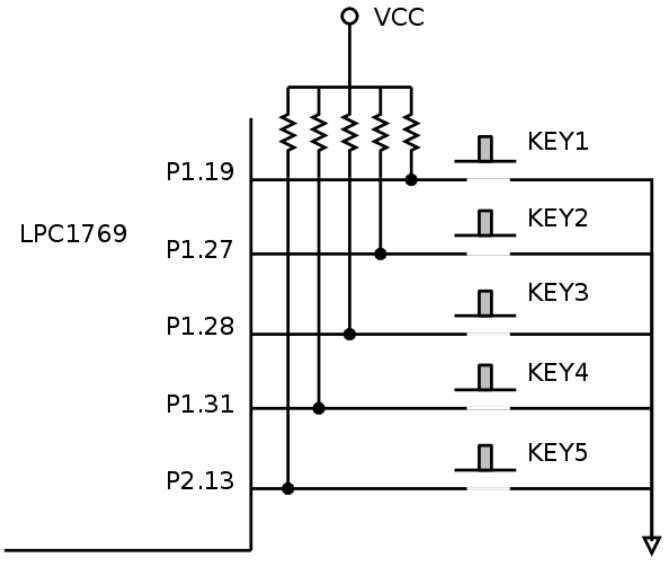
Figure 2.6. LEDs Connection Diagram



9. Keypad

The ZKit-ARM-1769 has 5 tactile push button switches connected to `P1.19`, `P1.27`, `P1.28`, `P1.31` and `P2.13`. The keypad connection details are shown in the following diagram.

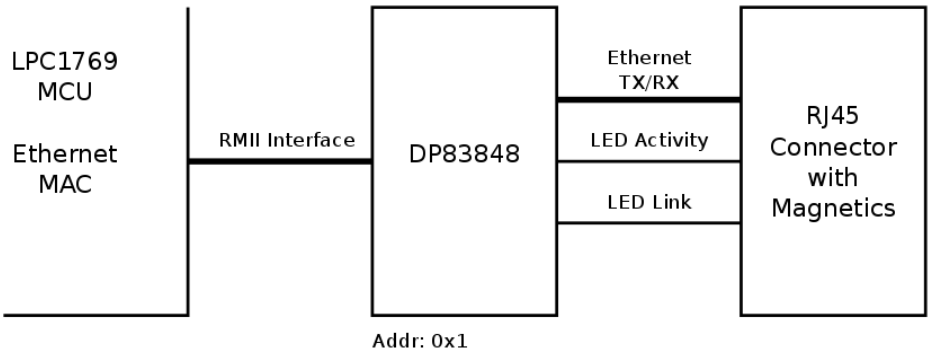
Figure 2.7. Keypad Connection Diagram



10. Ethernet

The ZKit-ARM-1769 has a 10/100 Fast Ethernet interface. The on-chip MAC of the LPC1769 is connected to a TI DP83848 PHY. The PHY address `0x1`. The PHY and RJ45 jack connection details are shown in the following diagram.

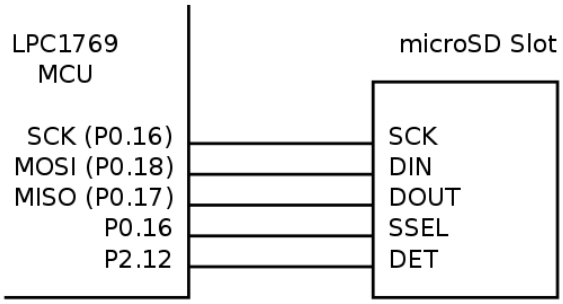
Figure 2.8. Ethernet Connection Diagram



11. microSD Card Slot

The ZKit-ARM-1769 has a microSD slot, connected to the SPI controller of the MCU. The card select pin of the microSD slot is connected to `P0.16` of the MCU, and the card detect pin is connected to `P2.12` of the MCU.

Figure 2.9. microSD Slot Connection Diagram



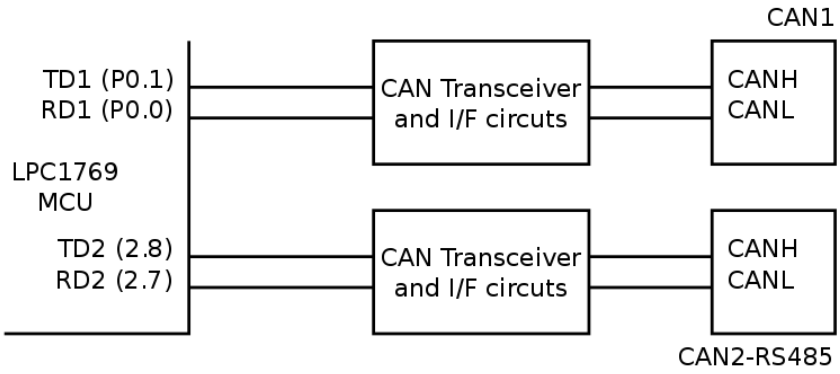
12. USB Device Interface

The ZKit-ARM-1769 has a mini-USB device connector, connected to the USB controller of the MCU. A USB device stack is to be used on the MCU to implement USB HID, Mass Storage, Serial, and other USB device classes.

13. CAN Interface

The ZKit-ARM-1769 has two CAN interfaces terminated on FRC headers, `CAN1` and `CAN2-RS485`. The LPC1769 MCU has two CAN MACs, `CAN0` and `CAN1`. Each CAN MAC is connected to TJA1040, a high speed CAN transceiver IC. The `CANH` and `CANL` signals from the transceiver are terminated on a 10-pin FRC header.

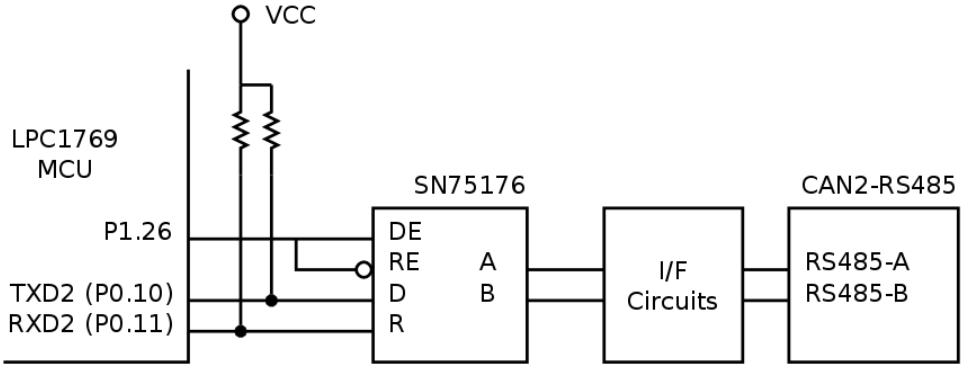
Figure 2.10. CAN Connection Diagram



14. RS485 Interface

The ZKit-ARM-1769 has one RS485 interface, connected to UART2 of the MCU. The UART signals are connected to SN75176A, a differential bus transceiver IC. The differential signals from the transceiver are terminated on a 10-pin FRC header. The GPIO P1.26, is used to disable/enable transmission.

Figure 2.11. RS485 Connection Diagram



15. JTAG Debug Interface

The ZKit-ARM-1769 has a 10-pin ARM mini-JTAG connector for debug purposes. This can be used with any standard JTAG debugger.

Chapter 3. Connectors

This chapter describes the connectors in the ZKit-ARM-1769.

1. SPI Pinmap

The SPI header is terminated with serial peripheral interface (SPI) bus, 4 general purpose IO and power supply. Add-on boards with SPI interface and general purpose IOs like MMC/SD card,EEPROM etc., can be connected through this header.

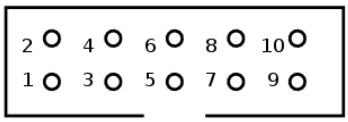


Table 3.1. SPI Header

Pin #	Header Signal	MCU Signal	Signal Type
1	VCC	-	+5V
2	SCK	P0.7/SCK1	TTL Out
3	MISO	P0.8/MISO1	TTL In [1]
4	MOSI	P0.9/MOSI1	TTL Out
5	SS	P0.6/SSEL1	TTL Out
6	DIO0	P0.4	TTL In/Out [1]
7	DIO1	P0.5	TTL In/Out [1]
8	DIO2	P4.28	TTL In/Out [1]
9	DIO3	P4.29	TTL In/Out [1]
10	GND	-	Ground

[1] 5V tolerant Input

VCC (Pin 1)	This is the +5V power supply for the external devices. The supply has a total current limit of 200mA when powered through USB.
SCK (Pin 2)	This is Serial Clock signal.
MISO (Pin 3)	This is the Master Input, Slave Output signal.
MOSI (Pin 4)	This is the Master Output, Slave Input signal.
SS (Pin 5)	This is the SPI chip select signal.
DIO (Pin 6-9)	These are digital input/output signals. These lines can be used to interface any extra signals required for a SPI devices like SD Card, etc., or can be used as chip selects for four other devices.
GND (Pin 10)	This is the ground signal. All other signals are referenced to the this signal.

2. UART-I2C Pinmap

The UART-I2C header is terminated with serial communication signals, I²C signals and power supply. Add-on boards, with different functionality, can be connected through this header.

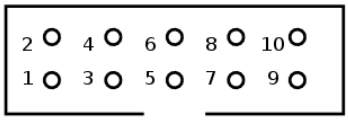


Table 3.2. UART-I2C Header

Pin #	Header Signal	MCU Signal	Signal Type
1	VCC	-	+5V
2	RXD	P2.1/RXD1	TTL In [1]
3	TXD	P2.0/TXD1	TTL Out
4	SCL	P0.28/SCL0	OC [2]
5	SDA	P0.27/SDA0	OC [2]
6	DIO0	P0.22/RTS1	TTL In/Out [1]
7	DIO1	P2.2/CTS1	TTL In/Out [1]
8	DIO2	P0.20/DTR1	TTL In/Out [1]
9	DIO3	P2.11/nEINT1	TTL In/Out [1]
10	GND	-	Ground

[1] 5V tolerant input

[2] Open collector, with 3.3V pull-up

VCC (Pin 1)	This is the +5V power supply for the external devices. The supply has a total current limit of 200mA when powered through USB.
RXD (Pin 2)	This is receive line of serial IO.
TXD (Pin 3)	This is transmit line of serial IO.
SCL, SDA (Pin 4, 5)	These are I²C bus signals(clock, data), and can be used to connect I²C devices. The signals are pulled up to 3.3V, through a 4.7K resistor.
DIO (Pin 6-9)	These are digital input/output signals. These pins can be used for hand-shake and flow control signals like DTR, RTS, CTS, etc.
GND (Pin 10)	This is the ground signal. All other signals are referenced to this signal.

3. PWM Pinmap

The PWM header is terminated with 5 pulse width modulation signals and power supply. Add-on boards like LED control, motor control can be connected through this header.

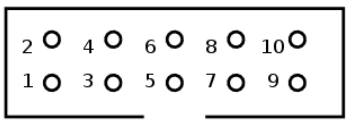


Table 3.3. PWM Header

Pin #	Signal Name	MCU Signal	Signal Type
1	VCC	-	+5V
2	PWM 0	P3.25/PWM1.2	TTL Out
3	PWM 1	P3.26/PWM1.3	TTL Out
4	PWM 2	P2.3/PWM1.4	TTL Out
5	PWM 3	P2.4/PWM1.5	TTL Out
6	PWM 4	P2.5/PWM1.6	TTL Out
7	PWM 5	-	TTL Out

Pin #	Signal Name	MCU Signal	Signal Type
8	Freq-In 0	P2.6/PCAP1.0	TTL In [1]
9	Freq-In 1	P1.29/ PCAP1.1	TTL In [1]
10	GND	GND	Ground

[1] 5V tolerant input

VCC (Pin 1) — This is the +5V power supply for the external add-on boards. The supply has a total current limit of 200mA when powered through USB.

PWM (Pin 2 - 7) — These are PWM output signals. The PWM signal when active produces a stream of pulses whose width can be controlled through software. An important parameter of a PWM signal is the **duty cycle**. The duty cycle is defined as the ratio between the pulse duration and pulse period of a rectangular waveform.

The PWM signal can be used to control the power delivered to a load, by controlling the duty cycle of the PWM signal. PWM signals are generally used for Motor speed control, LED brightness control, power supplies and wave form generation.

The PWM signal is a 5V CMOS/TTL output.

Freq-In (Pin 8, 9) — These are input signals, used for event counting and frequency measurement. These signals are 5V tolerant CMOS/TTL inputs.

4. AIO Pinmap

The AIO header is terminated with 4 ADC channels, 1 DAC and power supply. Sensors can be connected to this header.

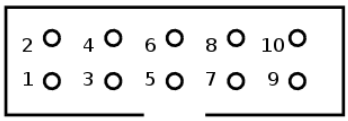


Table 3.4. AIO Header

Pin #	Signal Name	MCU Signal	Signal Type
1	VCC	-	+5V
2	ADC 0	P0.23/AD0.0	Analog Input
3	ADC 1	P0.24/AD0.1	Analog Input
4	ADC 2	P0.25/AD0.2	Analog Input
5	ADC 3	P0.26/AD0.3	Analog Input
6	ADC 4	-	Analog Input
7	ADC 5	-	Analog Input
8	DAC 0	P0.26/AOUT	Analog Output
9	VREF-OUT	3.3V	+3.3V
10	GND	GND	Ground

VCC (Pin 1) — This is the +5V power supply for the external add-on boards. The supply has a total current limit of 200mA when powered through USB.

ADC (Pin 2-5) — These are analog input signals connected to the ADC. The max input voltage is 3.3V.

DAC (Pin 8)	This is analog output signal connected to the DAC. Output voltage range is 0 - 3.3V.
VREF-OUT (Pin 9)	This is the ADC's reference voltage.
GND (Pin 10)	This is the ground signal. All other signals are referenced to this signal.

5. CAN1 Pinmap

The CAN1 header is terminated with 1 CAN interface. CAN devices can be connected to this header.

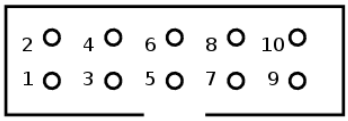


Table 3.5. CAN1 Header

Pin #	Signal Name	Signal Type
1	-	-
2	-	-
3	CANL	Can Bus
4	CANH	Can Bus
5	GND	Ground
6	-	-
7	-	-
8	VCC	+5V
9	-	-
10	-	-

VCC (Pin 8)	This is the +5V power supply for the external add-on boards. The supply has a total current limit of 200mA when powered through USB.
CANL, CANH (Pin 3, 4)	These are CAN bus signals.
GND (Pin 10)	This is the ground signal. All other signals are referenced to this signal.

6. CAN2-RS485 Pinmap

The CAN2-RS485 header is terminated with 1 CAN interface and 1 RS-485 interface. CAN devices and RS485 nodes can be connected to this header.

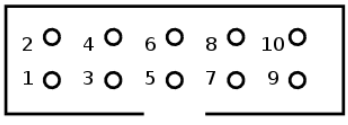


Table 3.6. CAN2-RS485 Header

Pin #	Signal Name	Signal Type
1	-	-
2	-	-

Pin #	Signal Name	Signal Type
3	CANL	Can Bus
4	CANH	Can Bus
5	GND	Ground
6	RS485-A	RS485 Bus
7	RS485-B	RS485 Bus
8	VCC	+5V
9	RS485-GND	Ground
10	-	-

VCC (Pin 8) — This is the +5V power supply for the external add-on boards. The supply has a total current limit of 200mA when powered through USB.

CANL, CANH (Pin 3, 4) — These are CAN bus signals.

RS485-A, RS485-B (Pin 6, 7) — These are RS485 bus signals.

RS485-GND (Pin 9) — This is used to ground the shield of the RS485 twisted pair cable.

GND (Pin 10) — This is the ground signal. All other signals are referenced to this signal.

7. J1 Proto Header Pinmap

The J1 Proto Header provides the signals available on the FRC-ports, through a socket header, for quick prototyping, using bread-boards and single strand wires. For the signal descriptions, refer to the corresponding FRC header.

Table 3.7. J1 Proto Header

Pin #	Header Signal	MCU Signal	Signal Type
1	+5V	-	+5V
2	PWM2	P3.25/PWM1.2	TTL In/Out [1]
3	PWM3	P3.26/PWM1.3	TTL In/Out [1]
4	PWM4	P2.3/PWM1.4	TTL In/Out [1]
5	PWM5	P2.4/PWM1.5	TTL In/Out [1]
6	PWM6	P2.5/PWM1.6	TTL In/Out [1]
7	NC	-	TTL In/Out [1]
8	CAP0	P2.6/PCAP1.0	TTL In/Out [1]
9	CAP1	P1.29/PCAP1.1	TTL In/Out [1]
10	AIN0	P0.23/AD0.0	Analog Input
11	AIN1	P0.24/AD0.1	Analog Input
12	AIN2	P0.25/AD0.2	Analog Input
13	AIN3	P0.26/AD0.3	Analog Input
14	+3.3V	-	+3.3V
15	SCK1	P0.7/SCK1	TTL In/Out [1]
16	MISO1	P0.8/MISO1	TTL In/Out [1]
17	MOSI1	P0.9/MOSI1	TTL In/Out [1]
18	SSEL1	P0.6/SSEL1	TTL In/Out [1]
19	P0.4	P0.4	TTL In/Out [1]

Pin #	Header Signal	MCU Signal	Signal Type
20	P0.5	P0.5	TTL In/Out [1]
21	P4.28	P4.28	TTL In/Out [1]
22	P4.29	P4.29	TTL In/Out [1]
23	RXD1	P2.1/RXD1	TTL In/Out [1]
24	TXD1	P2.0/TXD1	TTL In/Out [1]
25	SCL0	P0.28/SCL0	TTL In/Out [1]
26	SDA0	P0.27/SDA0	TTL In/Out [1]
27	RTS1	P0.22/RTS1	TTL In/Out [1]
28	CTS1	P2.2/CTS1	TTL In/Out [1]
29	DTR1	P0.20/DTR1	TTL In/Out [1]
30	P2.11	P2.11	TTL In/Out [1]
31	GND	-	Ground
32	GND	-	Ground

[1] 5V tolerant input

[2] Open collector, with 3.3V pull-up

Appendix A. Legal Information

1. Copying

2. Disclaimers

NO WARRANTY. ZILOGIC SYSTEMS' DEVELOPMENT KITS (AND TECHNICAL SUPPORT, IF ANY) ARE PROVIDED "AS IS" AND WITHOUT ANY WARRANTY OF ANY KIND, EXPRESS OR IMPLIED. TO THE MAXIMUM EXTENT PERMITTED UNDER APPLICABLE LAWS, ZILOGIC SYSTEMS EXPRESSLY DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NONINFRINGEMENT. ZILOGIC SYSTEMS DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN ZILOGIC SYSTEMS' DEVELOPMENT KITS WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION WILL BE UNINTERRUPTED OR ERROR-FREE, OR THAT DEFECTS IN ZILOGIC SYSTEMS' DEVELOPMENT KITS WILL BE CORRECTED. FURTHERMORE, ZILOGIC SYSTEMS DOES NOT WARRANT OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE ZILOGIC SYSTEMS' DEVELOPMENT KITS IN TERMS OF THEIR CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES, SO THE ABOVE EXCLUSION MAY NOT APPLY OR MAY BE LIMITED.

Limitation of Liability. Zilogic Systems' development kits are not designed, authorized or warranted to be suitable for use in medical, military, aircraft, space or life support equipment, not in applications where failure or malfunction of a Zilogic Systems product can resonably be expected to result in personal injury, death or severe property or environmental damage. Zilogic Systems accepts no liability for inclusion and/or use of Zilogic Systems' development kits in such equipment or applications and therefore such inclusion and/or use is at the customer’s own risk.