RV32M1-VEGA Development Board

1. Introduction

This guide describes the hardware for the RV32M1-VEGA Development Board. The RV32M1-VEGA development board is a small, low-power, and cost-effective evaluation and development board for application prototyping and demonstration of the RV32M1 device. These evaluation boards offer easy-to-use mass-storage-device mode flash programmer, a virtual serial port, and standard programming and run-control capabilities.

The RV32M1 is an ultra-low power, highly integrated single-chip device that enables Bluetooth Low Energy (BLE), Generic FSK (at 250, 500, 1000 and 2000 kbps) or IEEE Standard 802.15.4 with Thread support for portable, extremely low-power embedded systems.

The RV32M1 integrates a radio transceiver operating in the 2.36 GHz to 2.48 GHz range supporting a range of FSK/GFSK and O-QPSK modulations, an ARM Cortex-M4 CPU, ARM Cortex-M0+ CPU, RISC-V RI5CY CPU, RISC-V ZERO_RISCY CPU, 1.25 MB Flash and 384 KB SRAM, BLE Link Layer hardware, 802.15.4 packet processor hardware and peripherals optimized to meet the requirements of the target applications.

Contents

1. Introduction 1
2 Overview and description 2
  2.1 Overview 2
  2.2 Feature description 3
  2.3 OpenSDA serial and debug 5
3 Functional description 6
  3.1 RF circuit 6
  3.2 Clocks 7
  3.3 Power management 7
  3.4 Universal Serial Bus (USB) 8
  3.5 Secure Digital Host Controller (SDHC) 9
  3.6 Serial flash memory 10
  3.7 Accelerometer + Magnetometer Combo Sensor 10
  3.8 Visible light sensor 11
  3.9 User application LEDs 12
  3.10 User buttons 12
4 Headers and jumpers 14
  4.1 Arduino compatible I/O headers 14
  4.2 Jumper table 21
5 References 22
6 Revision history 23

2. Overview and description

The RV32M1-VEGA development board is an evaluation environment supporting RISC-V cores based RV32M1 Wireless Microcontrollers (MCU). The RV32M1 integrates a radio transceiver operating in the 2.36 GHz to 2.48 GHz range (supporting a range of FSK/GFSK and O-QPSK modulations) an ARM Cortex-M4 CPU, an ARM Cortex-M0+ CPU, a RSIC-V RI5CY MCU and a RSIC-V ZERO_RISCY MCU into a single package. www.open-isa.org supports the RV32M1 with GNU toolchain and software that include hardware evaluation and development boards, Eclipse-based software development IDE, applications, drivers, custom PHY usable with IEEE Std. 802.15.4 compatible MAC, and BLE Link Layer. The RV32M1-VEGA development board consists of the RV32M1 device with a 32 MHz reference oscillator crystal, RF circuitry (including antenna), 32-Mbit external serial flash, and supporting circuitry in the popular Freedom board form-factor. The board is a standalone PCB and supports application development with Bluetooth Low Energy, Geeric FSK and IEEE Std. 802.15.4 protocol stacks including Thread.

2.1 Overview

Figure 1 is a high-level block diagram of the RV32M1-VEGA board features:

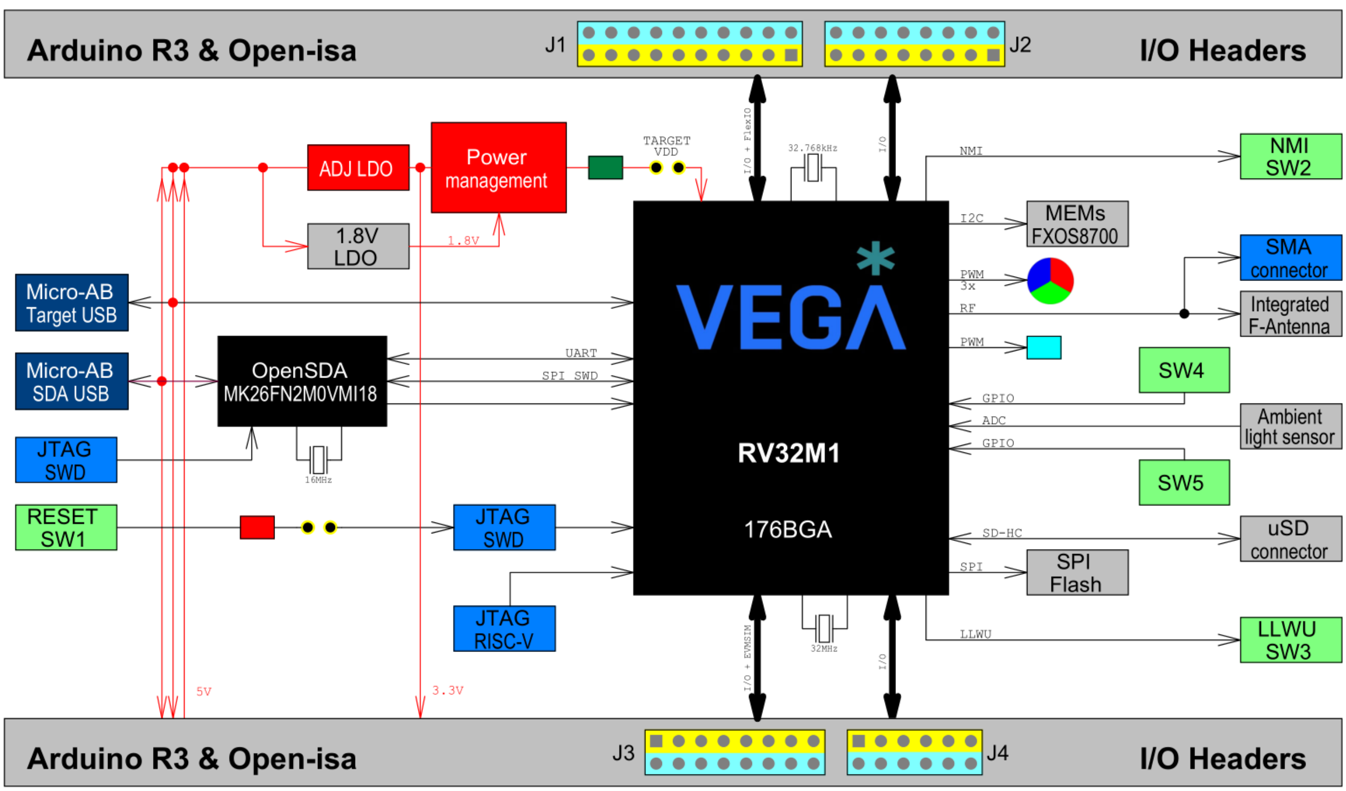

Figure 1. RV32M1-VEGA block diagram

2.2 Feature description

The RV32M1-VEGA development board is the most diverse reference design containing the RV32M1 device and all necessary I/O connections for use as a stand-alone board, or connected to an application. Figure 2 shows the RV32M1-VEGA development board.

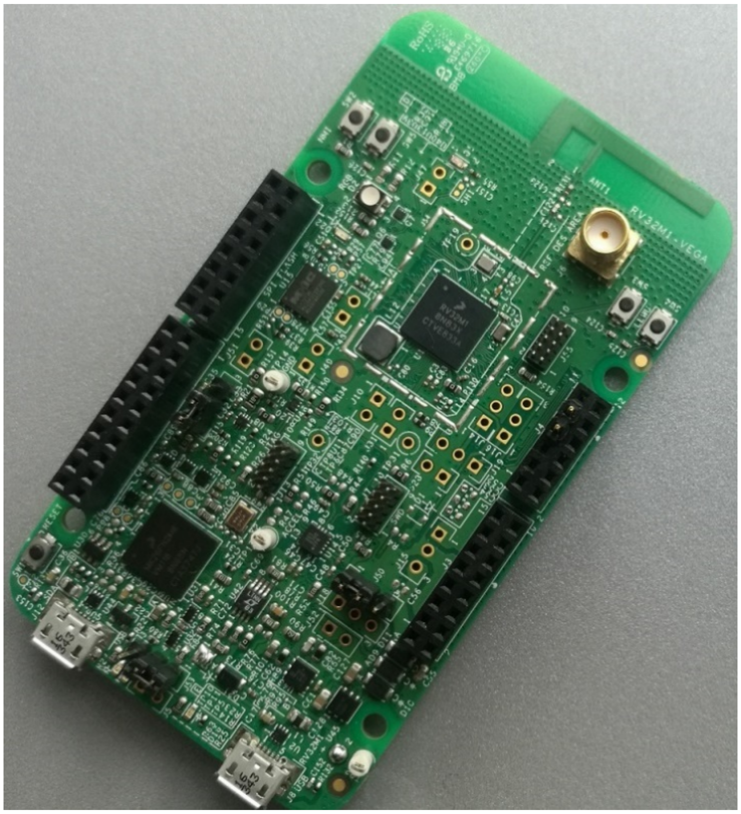

Figure 2. RV32M1-VEGA development board

The RV32M1-VEGA development board has these features:

- Ultra-low-power RV32M1 Wireless MCU supporting BLE, Generic FSK, and IEEE Std. 802.15.4 (Thread) platforms
- IEEE Std. 802.15.4-2006 compliant transceiver supporting 250 kbps O-QPSK data in 5.0 MHz channels, and full spread-spectrum encoding and decoding
- Fully compliant Bluetooth v4.2 Low Energy (BLE)
- Reference design area with small-footprint, low-cost RF node:
    - Single-ended input/output port
    - Low count of external components
    - Programmable output power from -30 dBm to +3.5 dBm at the SMA connector
    - Receiver sensitivity is -100 dBm, typical (@1 % PER for 20-byte payload packet) for 802.15.4 applications, at the SMA connector
    - Receiver sensitivity is -95 dBm (for BLE applications) at the SMA connector
- Integrated PCB inverted F-type antenna and SMA RF port (requires moving C122 to C121)
- Selectable power sources
- DC-DC converter with Buck and Bypass operation modes

- 32 MHz reference oscillator
- 32.768 kHz reference oscillator
- 2.4 GHz frequency operation (ISM and MBAN)
- USB device mode interface with micro USB connector
- 32-Mbit (4 MB) external serial flash memory for Over-the-Air Programming (OTAP) support
- FXOS8700CQ Digital Sensor, 3D Accelerometer (±2g/±4g/±8g) + 3D Magnetometer
- Integrated Open-Standard Serial and Debug Adapter (OpenSDA)
- One RGB LED indicator
- One red LED status indicator
- One green LED power indicator
- One red LED reset indicator
- One amber LED OpenSDA activity indicator
- Four push-button switches

Figure 3 shows the main board features and Input/output headers for the RV32M1-VEGA board:

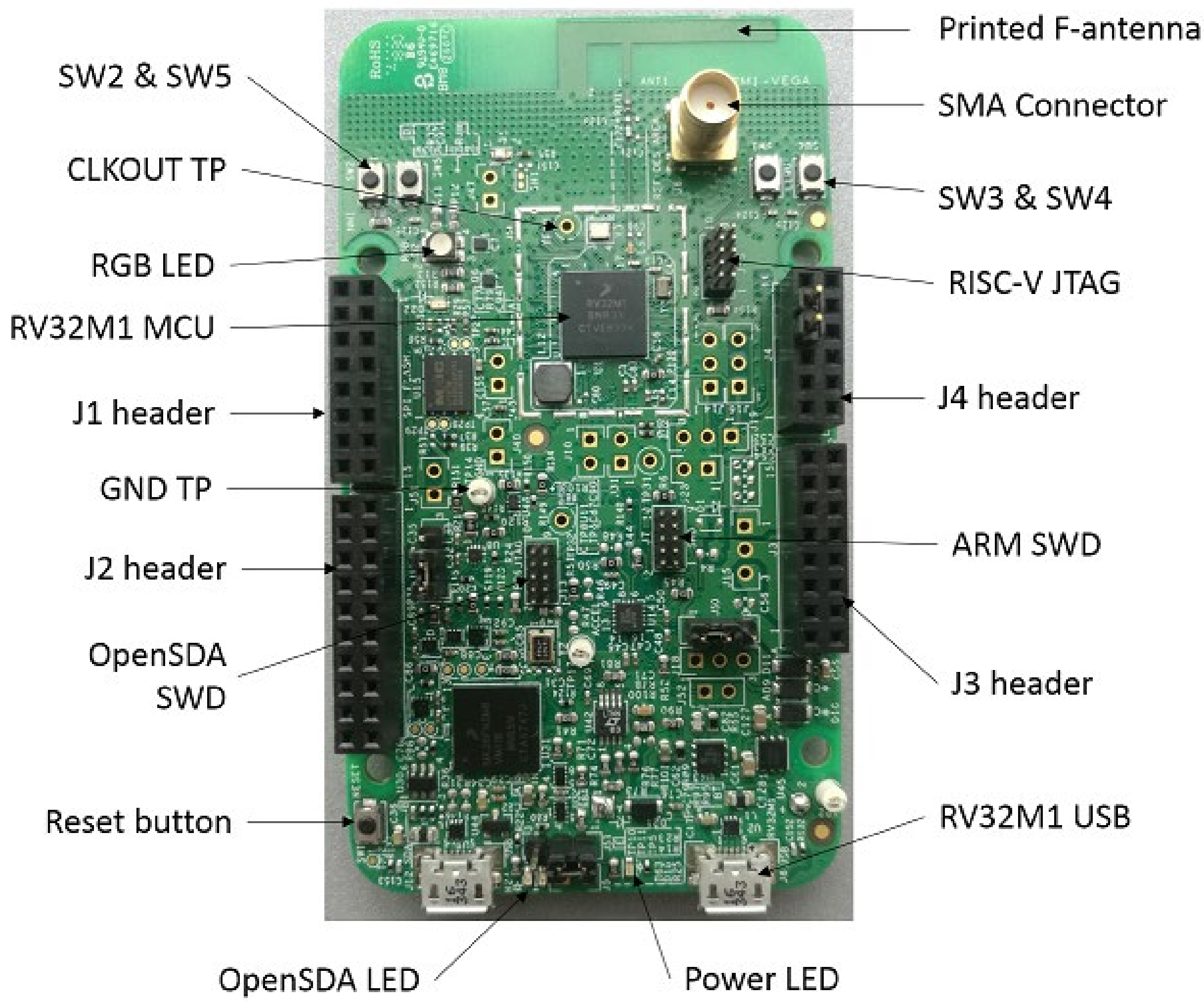

Figure 3. RV32M1-VEGA component placement

2.3 OpenSDA serial and debug

The RV32M1-VEGA development board includes OpenSDA v2.4 - a serial and debug adapter circuit that includes an open-source hardware design, an open-source bootloader, and debug interface software. It bridges serial and debug communications between a USB host and an embedded target processor as shown in Figure 4. The hardware circuit is based on a Kinetis K26 family MCU (MK26FN2M0VMI18) with 2 MB of embedded flash and an integrated USB controller. OpenSDA v2.4 comes preloaded with the DAPLink bootloader - an open-source mass storage device (MSD) bootloader and the Interface firmware, which provides an MSD flash programming interface, a virtual serial port interface, and a CMSIS-DAP debug protocol interface. For more information on the OpenSDA v2.4 software, see mbed.org, https://github.com/mbedmicro/DAPLink.

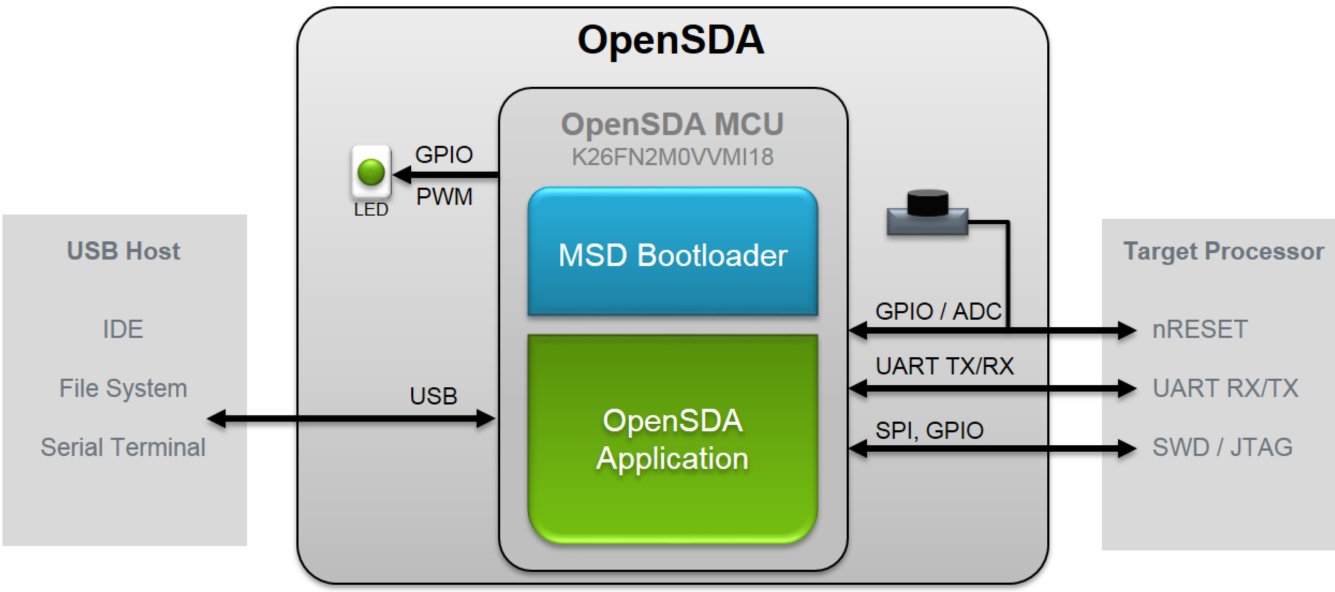

Figure 4. OpenSDA v2.4 high-level block diagram

OpenSDA v2.4 is managed by a Kinetis K26 MCU built on the ARM Cortex-M4 core. The OpenSDA v2.4 circuit includes a status LED (D5) and a pushbutton (SW1). The pushbutton asserts the Reset signal to the RV32M1 target MCU. It can also be used to place the OpenSDA v2.4 circuit into bootloader mode. UART and GPIO signals provide an interface to either the SWD debug port or the K26. The OpenSDA v2.4 circuit receives power when the USB connector J12 is plugged into a USB host.

2.3.1 Virtual serial port

A serial port connection is available between the OpenSDA v2.4 MCU and pins PTC7 and PTC8 of the RV32M1.

NOTE

To enable the Virtual COM features, a driver must be installed. Download the driver at https://developer.mbed.org/handbook/Windows-serial-configuration

3. Functional description

The six-layer board provides the RV32M1 with its required RF circuitry, 32 MHz reference oscillator crystal, and power supply with a DC-DC Buck converter, and Bypass modes. The layout for this base-level functionality can be used as a reference layout for your target board.

3.1 RF circuit

The RV32M1-VEGA RF circuit provides an RF interface for users to begin application development. A minimum matching network to the MCU antenna pin is provided through C13 and L5.

An optional SMA is located at J6. This is enabled by rotating the 10pF capacitor in C122 to the location of C121. Figure 5 shows the RF circuit in detail.

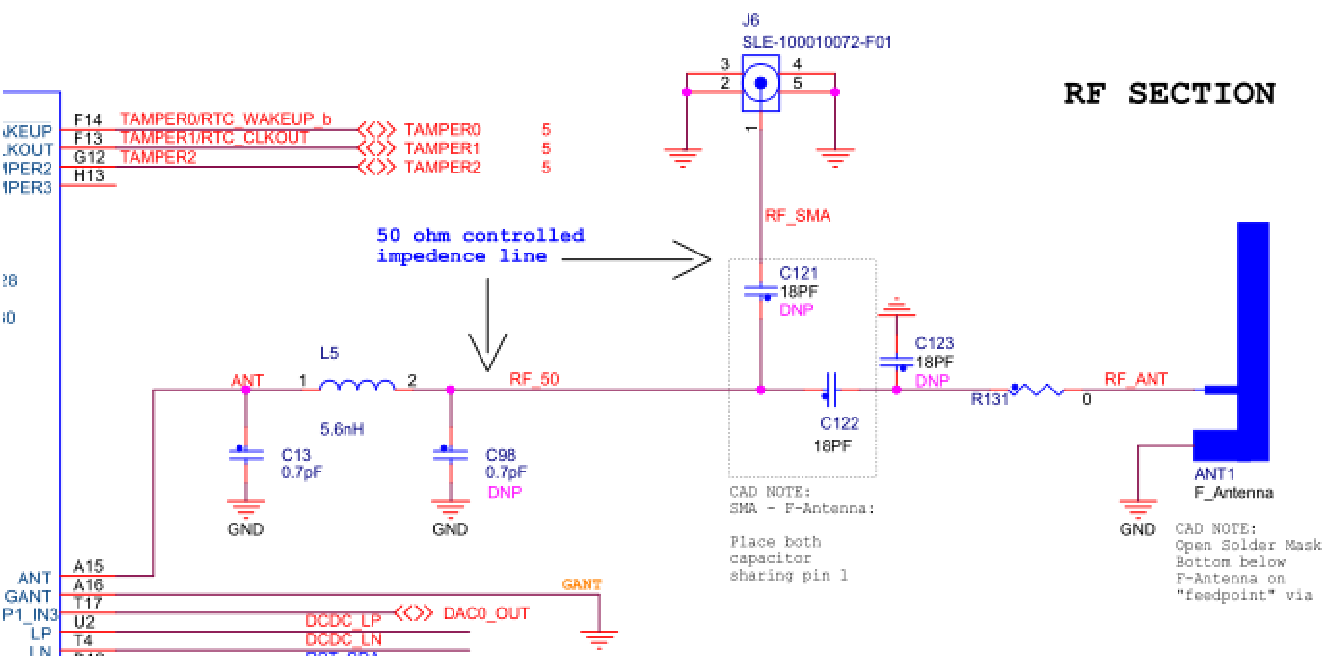

Figure 5. RV32M1-VEGA RF circuit

3.2 Clocks

The RV32M1-VEGA board provides two clocks. A 32 MHz for clocking the MCU and Radio, and a 32.768 kHz to provide an accurate low power time base:

- 32 MHz Reference Oscillator
    - The IEEE Std. 802.15.4 requires the frequency to be accurate to less than ±40 ppm
    - Internal load capacitors provide the crystal load capacitance
    - To measure the 32 MHz oscillator frequency, enable the RF_CLKOUT signal to provide buffered output clock signal to TP19

- 32.768 kHz Crystal Oscillator (for accurate low-power time base)
    - A 32.768 kHz crystal Y1 is provided
    - Internal load capacitors provide the entire crystal load capacitance
    - To measure the 32.768 kHz oscillator frequency, enable the RTC_CLKOUT signal to be available on the TAMPER1 pin. This can be observed at J4-3

3.3 Power management

There are several different ways to power and measure current on the RV32M1-VEGA board. The RV32M1-VEGA power distribution scheme is shown in Figure 6:

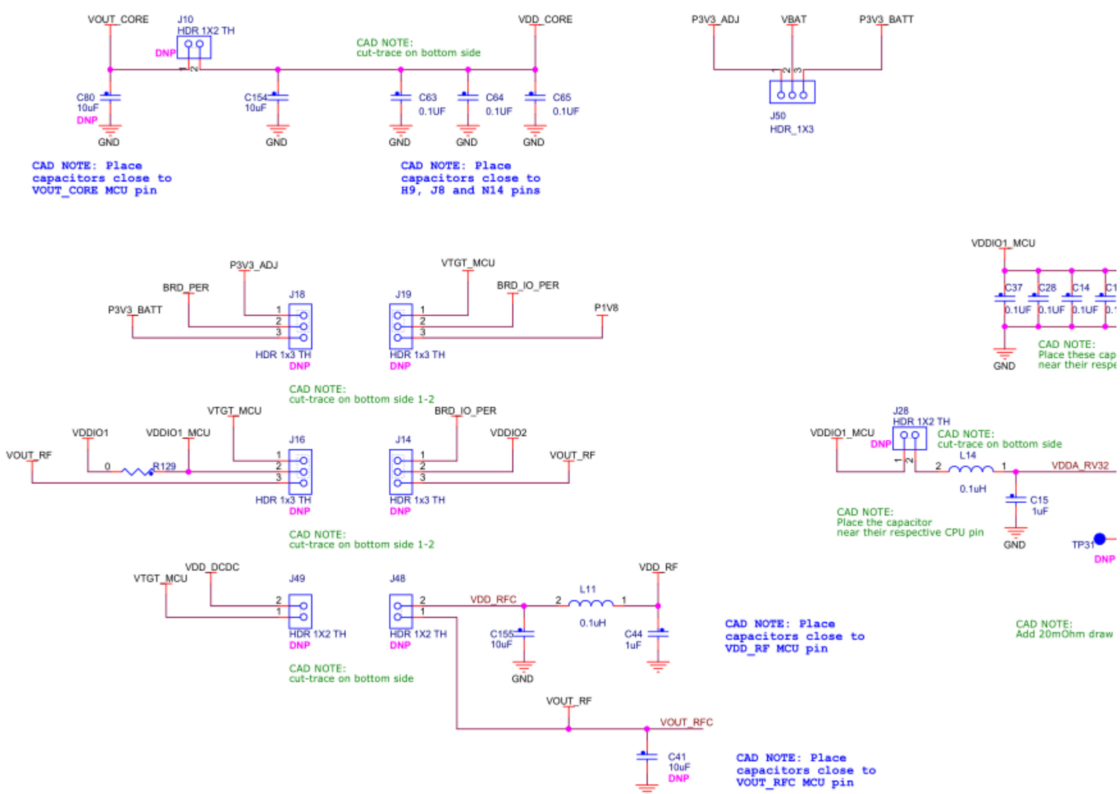

Figure 6. RV32M1-VEGA power management circuit

The RV32M1-VEGA board will typically be powered by a 5V source by one of the following means:

- OpenSDA micro USB type B connector (J12)
- RV32M1 micro USB type B connector (J8)
- Through the header J3 pin-10
- Optional 5V regulator populated at J15

The 5V supply then powers an adjustable regulator, U43, and a 1.8V regulator, U45. The adjustable regulator is preset to provide a nominal 3.3V output. The adjustable regulator output can be controlled by connecting an external supply in series with a 3.9k resistor to J52-1. The external supply range of 0.9V to 2.7V will adjust the regulator output from 3.6V to 1.8V.
The RV32M1 and supporting circuitry can then be powered using the adjustable regulator or a CR2032 coin cell selected by means of J18. The 1.8V regulator provides the ability to run the device with split supplies.
Typical power supply configurations are shown in Table 1

Table 1. **RV32M1-VEGA power supply configurations**

Description	J10	J48	J14	J16	J19
Single Supply Operation, IO @ 3.3V	1-2	1-2	1-2	1-2	1-2
Single Supply Operation, RF & IO @ 1.8V	1-2	1-2	2-3	2-3	open
Dual IO, 3.3V and 1.8V	1-2	1-2	1-2	1-2	2-3
Full bypass	open	open	open	open	open

These jumpers provide access to insert ammeters in all the supplies connecting to the RV32M1 device. They also provide a means of connecting external supplies to any of the RV32M1 power pins.
In the case of using a single supply, an ammeter can be placed across J18 to measure the entire system current. Alternatively, an ammeter can be placed across J53 pins 2 and 3 to measure current with the LEDs and sensor core taken out of the reading. To minimize the current drawn by the other board components and measure the current drawn by just the RV32M1 device, the following steps are recommended

- Cut the trace under J5 to isolate the power indicator LED (if using J18)
- Cut the trace under J47 to isolate the photo transistor
- Place the SPI flash in ultra-low power mode by writing the command value 0xB9.

3.4 Universal Serial Bus (USB)

The RV32M1 MCU features a full-speed USB module with device capability and built-in transceiver. The RV32M1-VEGA board routes the USB D+ and D- signals from the RV32M1 MCU directly to the onboard micro USB connector (J8) via the required 33ohm resistors. Figure 7 shows the complete USB circuit.

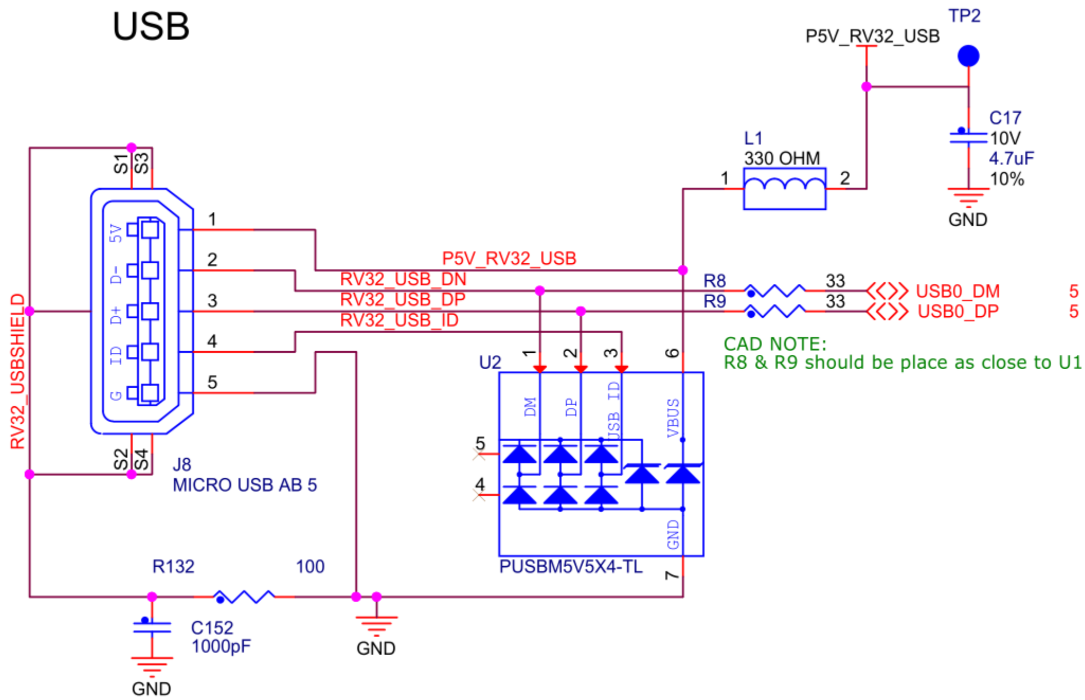

Figure 7. USB connector circuit

3.5 Secure Digital Host Controller (SDHC)

A micro secure digital (SD) card slot is supported on the RV32M1-VEGA. The SD card detect pin is an open switch that shorts with VDD when the card is inserted. The SD card VDD is supplied by VDDI01 and it must be configured to be at least 2.7V. The SD card connections are shown in Figure 8.

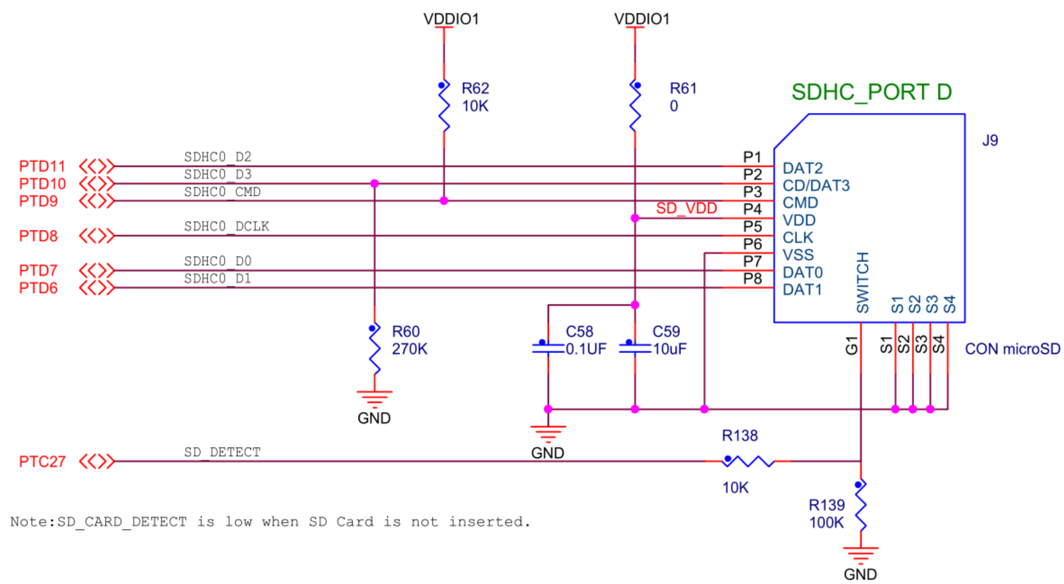

Figure 8. Micro SD card connector circuit

3.6 Serial flash memory

Component U15 is the MX25R3235FZNIL0 32-Mbit (4 MB) serial flash memory with SPI interface. It is intended for Over-the-Air Programming (OTAP) or for storing the non-volatile system data, or parameters.

Figure 9 below shows the memory circuit:

- Memory power supply is VDDIO1_SDA_SPI
- Discrete pull-up resistors pads are provided for the SPI port
- The memory uses a dedicated SPI port
- The SPI Write Protect and Reset has a discrete pull-up resistor
- Series zero ohm resistors are provided if it is desired to isolate the memory from the RV32M1 device.

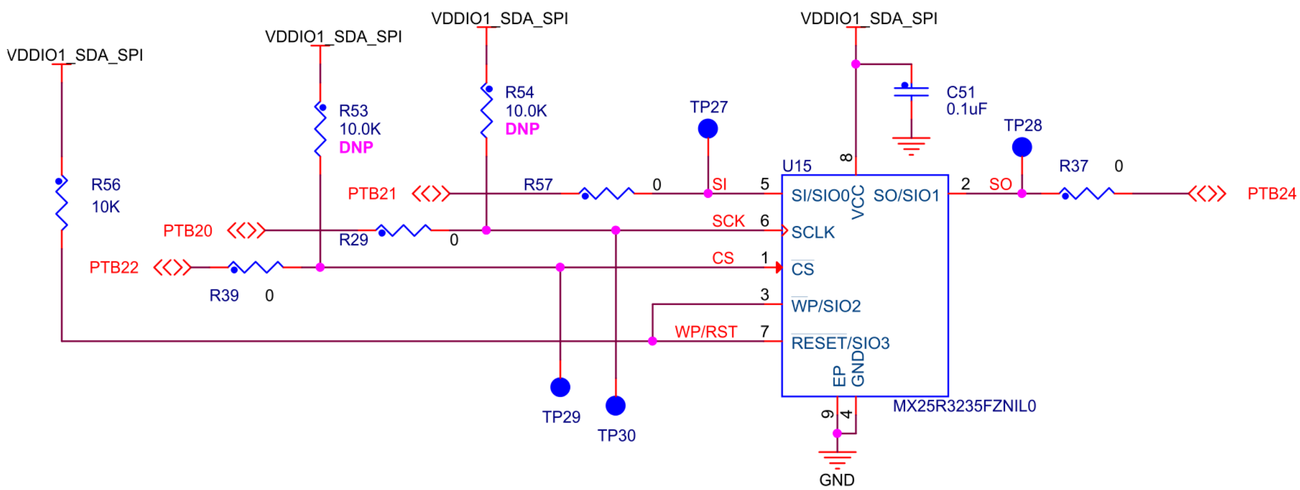

Figure 9. MX25R3235FZNIL0 32-Mbit (4 MB) serial flash memory circuit

3.7 Accelerometer + Magnetometer Combo Sensor

Component U14 is a FXOS8700CQ sensor, a six-axis sensor with integrated linear accelerometer and magnetometer with very low power consumption, and selectable I^2C. Figure 10 shows the sensor circuit.

- The sensor core is powered by the BRD_PER rail and the sensor IO is powered by the BRD_IO_PER rail
  - This allows the sensor IO to be operated at a lower voltage than the sensor core supply
- Discrete pull-up resistors for the I^2C bus lines are provided
- Default address is configured as 0x1E:
  - Address can be changed by pull-up/pull-down resistors on SA0 and SA1 lines
- There are two interrupt signals routed
- The I^2C uses dedicated lines for the I2C interface and GPIO connections
- Series zero ohm resistors and shorting links are provided if it is desired to isolate the sensor from the RV32M1 device.

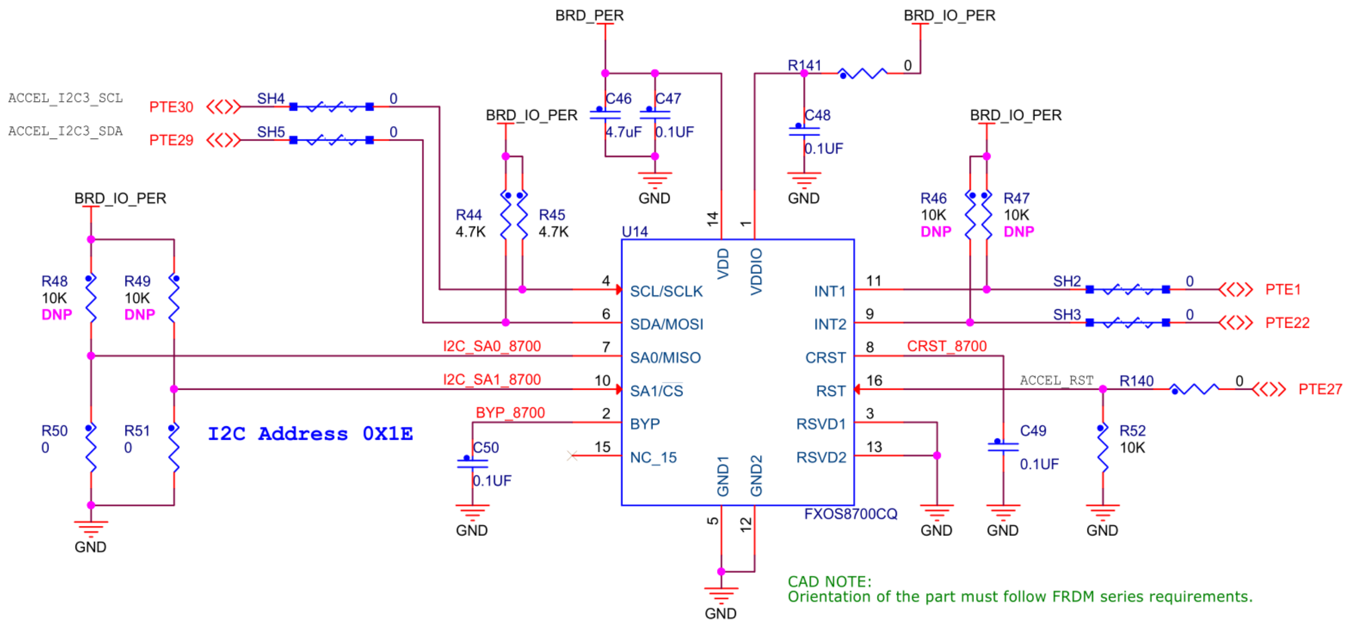

Figure 10. FXOS8700CQ combo sensor circuit

3.8 Visible light sensor

One phototransistor (Q1) is connected to ADC input channel SE3 of the RV32M1 for evaluating the ADC module as shown in Figure 11.

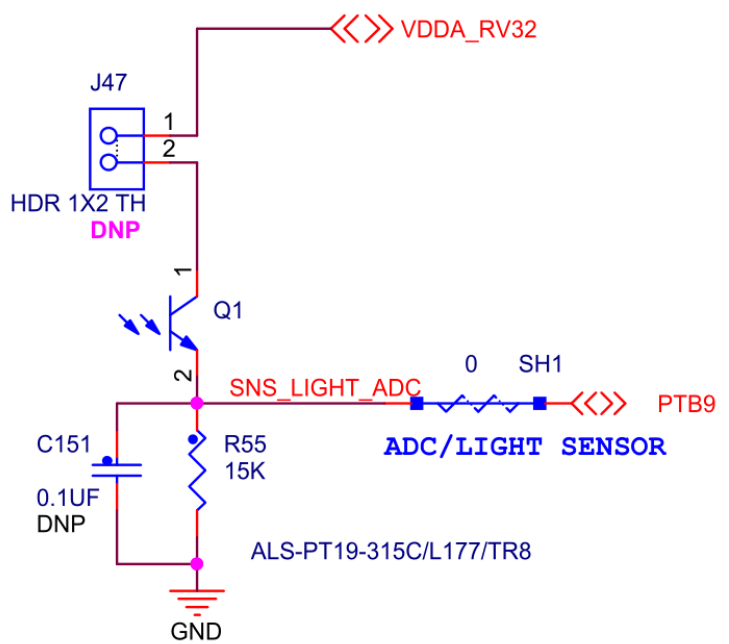

Figure 11. Visible Light Sensor circuit

The light sensor Output is shared with header J4 pin-6. The light sensor maybe isolated from the RV32M1 device, and header J4 pin-6, by cutting the shorting link SH1. The light sensor is powered by

VDDA_RV32 so if VREFH is configured to be less than VDDA_RV32, the maximum voltage the ADC can convert will be that of VREFH.
With no light reaching the light sensor, there will be a small current drawn from VDDA_RV32. If it is desired to measure the lowest MCU current, then the trace under J47 will need to be cut.

3.9 User application LEDs

The RV32M1-VEGA provides an RGB LED for user applications. A single red LED, D22, is provided as a general status indicator. Figure 12 shows the circuitry for the LEDs.

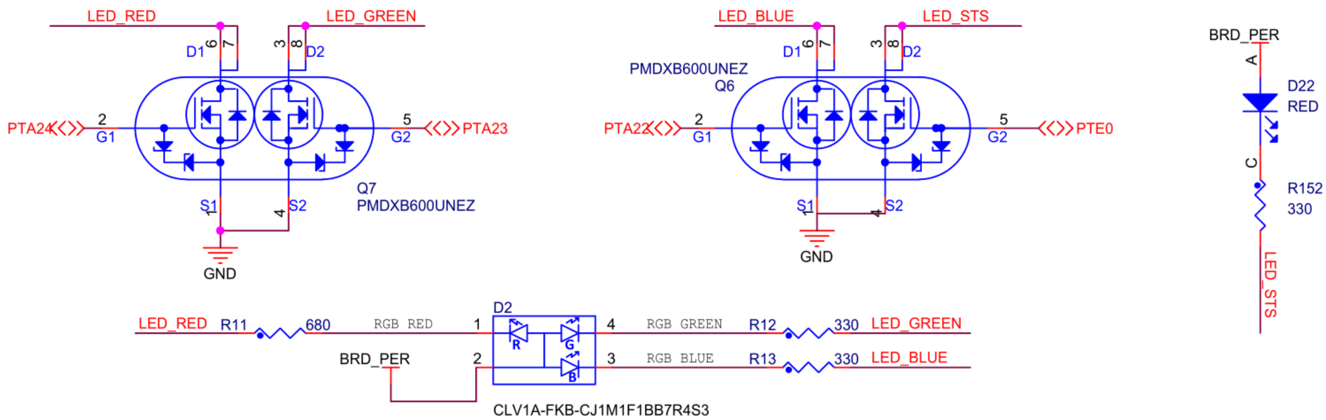

Figure 12. RV32M1-VEGA RGB LED circuit

The LEDs are powered by the BRD_PER rail and controlled by Q6 and Q7. This allows the LEDs to operate while being controlled by GPIO that are powered at a voltage less than BRD_PER. The Blue and Green LED in the RGB LED will not illuminate when BRD_PER is at lower voltages.

3.10 User buttons

Four tactile buttons are populated on the RV32M1-VEGA for Human Machine Interaction (HMI). Figure 13 shows the circuit for the tactile buttons.

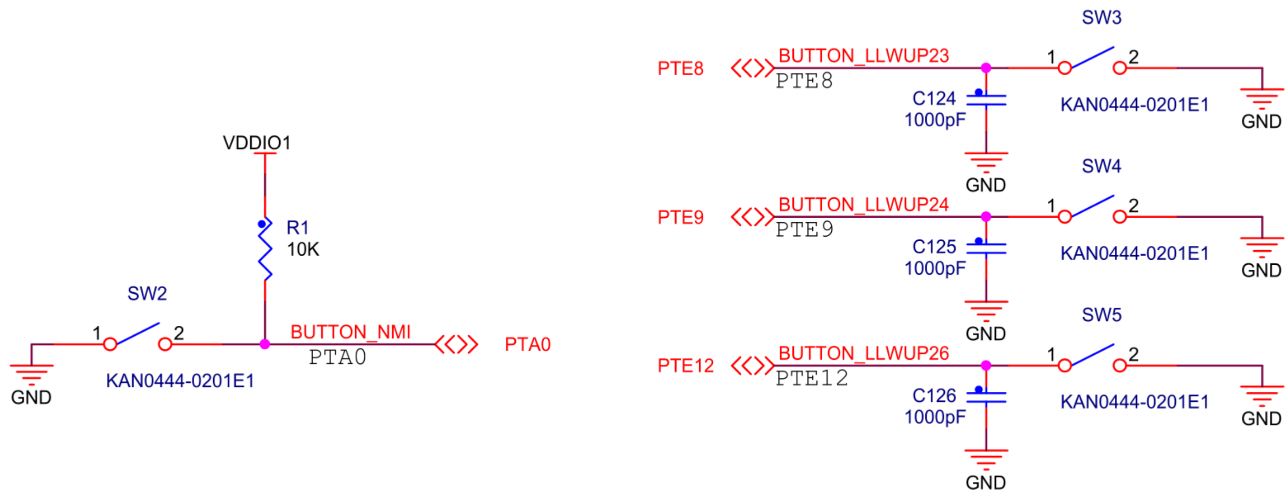

Figure 13. RV32M1-VEGA HMI circuit.

SW2 provides an external pull up device. It is connected to the RV32M1 NMI pin. This provides the option of using this switch as an NMI/wake up source, ROM bootloader boot option source or as a general-purpose input with interrupt capability.

SW3, SW4 and SW5 all provide general purpose inputs with interrupt and wake up capability. The internal pull up device for each pin must be enabled when these are being used.

4. Headers and jumpers

4.1 Arduino compatible I/O headers

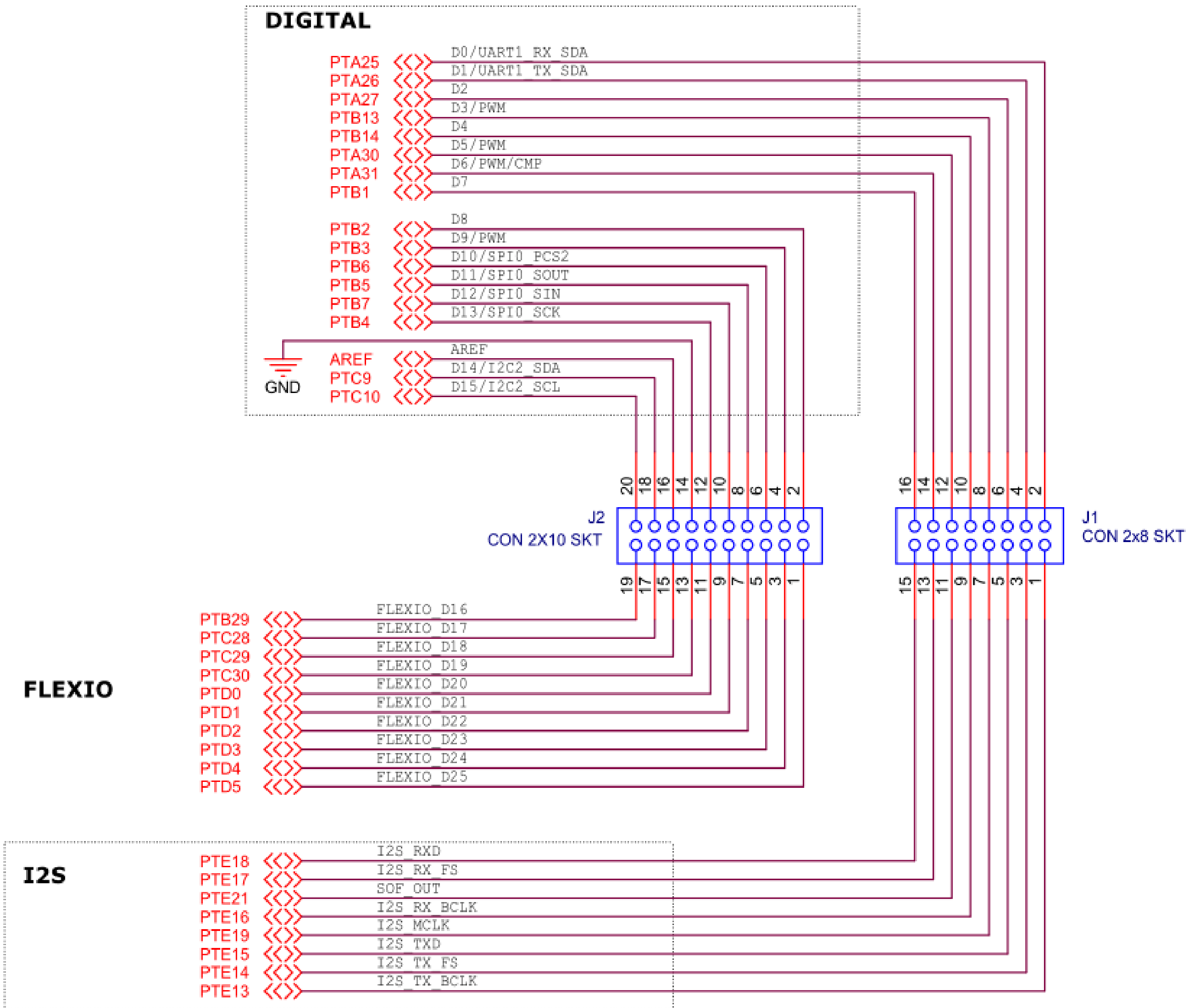

Figure 14. RV32M1-VEGA I/O header pinout

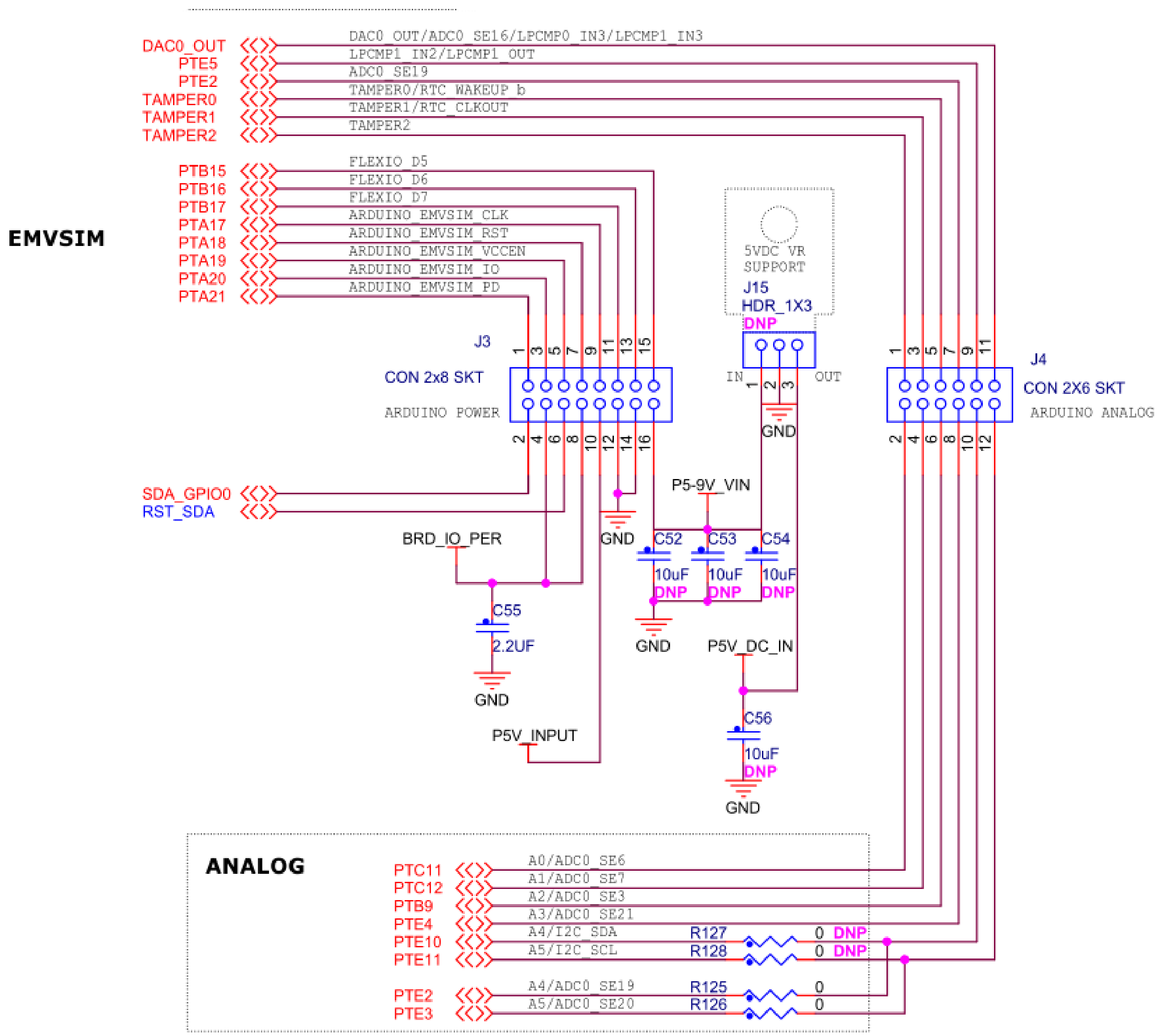

Figure 15. RV32M1-VEGA I/O header pinout

Table 2 shows the signals that can be multiplexed to each pin.

Table 2. **Arduino compatible header/connector pinout (J1 and J2)**

Header	Pin No	Name	Type / RV32M1 Pin	GPIO	Functions
J1	1		Freedom Proprietary	PTE18	TPM2_CH2 / I2S0_RXD / FXIO0_D8
			RV32M1 (Pin K13)		

	2	D0	Arduino Uno R3	PTA25	LPUART1_RS / LPI2C2_SCLS / LPSPI3_SOUT
			RV32M1 (Pin B5)		
	3		Freedom Proprietary	PTE17	TPM2_CH1 / I2S0_RX_FS / FXIO_D7
			RV32M1 (Pin L15)		
	4	D1	Arduino Uno R3	PTA26	LPUART1_TX / LPI2C2_SCLS / LPSPI3_PCS2
			RV32M1 (Pin A5)		
	5		Freedom Proprietary	PTE21	TPM2_CH4 / I2S0_TXD1 / USB0_SOF_OUT / FXIO0_D10
			RV32M1 (Pin J17)		
	6	D2	Arduino Uno R3	PTA27	LPUART1_CTS / LPSPI3_SIN
			RV32M1 (Pin A3)		
	7		Freedom Proprietary	PTE16	TPM2_CH0 / I2S0_RX_BCLK / FXIO0_D6
			RV32M1 (Pin L14)		
	8	D3	Arduino Uno R3	PTB13	TPM3_CH0 / LPUART2_CTS / LPI2C1_SDA / LPI2C0_SDAS / FXIO0_D3
			RV32M1 (Pin G3)		
	9		Freedom Proprietary	PTE19	TPM2_CH3 / I2S0_MCLK / FXIO_D9
			RV32M1 (Pin K16)		
	10	D4	Arduino Uno R3	PTB14	LPUART2_RTS / LPI2C1_SCL / LPI2C0_SCLS / TPM3_CH1 / FXIO0_D4
			RV32M1 (Pin G2)		
	11		Freedom Proprietary	PTE15	TPM3_CLKIN / I2S0_TXD / FXIO0_D5
			RV32M1 (Pin L17)		
	12	D5	Arduino Uno R3	PTA30	LLWU_P3 / LPUART2_CTS / LPSPI1_SOUT / TPM1_CH0
			RV32M1 (Pin A1)		

	13		Freedom Proprietary	PTE14	TPM3_CH1 / LPI2C3_HREQ / I2S0_TX_FS / FXIO0_D4
			RV32M1 (Pin L16)		
	14	D6	Arduino Uno R3	PTA31	TPM1_CH1 / LPUART2_RTS / LPSPI1_PCS2
			RV32M1 (Pin AC4)		
	15		Freedom Proprietary	PTE13	TPM3_CH0 / LPI2C3_SCLS / I2S0_BCLK / FXIO0_D3
			RV32M1 (Pin N17)		
	16	D7	Arduino Uno R3	PTB1	LPUART2_RX / LPSPI1_PCS0 / I2S0_TXD1
			RV32M1 (Pin J12)		
J2	1		Freedom Proprietary	PTD5	ADC0_S38 / SDHC0_D4 / EMVSIM0_VCCEN / FXIO0_D25
			RV32M1 (Pin N10)		
	2	D8	Arduino Uno R3	PTB2	TPM0_CH0 / LPUART2_RX / LPSPI0_PCS1 / I2S0_TXD0
			RV32M1 (Pin D12)		
	3		Freedom Proprietary	PTD4	LPSPI2_PCS1 / SDHC0_D5 / EMVSIM0_RST / FXIO0_D24
			RV32M1 (Pin N8)		
	4	D9	Arduino Uno R3	PTB3	TPM0_CH1 / LPUART1_TX LPSPI0_PCS3 / I2S0_TX_FS
			RV32M1 (Pin C1)		
	5		Freedom Proprietary	PTD3	TPM2_CLKIN / LPSPI0_PCS0 / SDHC0_D6 / EMVSIM0_CLK / FXIO0_D23
			RV32M1 (Pin T8)		
	6	D10	Arduino Uno R3	PTB6	TPM0_CH4 / LPI2C1_SDA / LPSPI0_PCS2 / I2S0_RX_BCLK
			RV32M1 (Pin E1)		
	7		Freedom Proprietary	PTD2	LPSPI0_SIN / SDHC0_D7 / FXIO0_D22
			RV32M1 (Pin U7)		

	8	D11	Arduino Uno R3	PTB5	TPM0_CH3 / LPUART1_RTS / LPSPI0_SOUT / I2S0_MCLK
			RV32M1 (Pin D2)		
	9		Freedom Proprietary	PTD1	LPUART1_RTS / LPSPI0_PCS2 / FXIO0_D21
			RV32M1 (Pin P7)		
	10	D12	Arduino Uno R3	PTB7	TPM0_CH5 / LPI2C1_SDAS / LPSPI0_SIN / I2S0_RX_FS
			RV32M1 (Pin E2)		
	11		Freedom Proprietary	PTD0	TPM0_CH0 / LPUART1_CTS / LPSPI0_SOUT / FXIO0_D20
			RV32M1 (Pin T7)		
	12	D13	Arduino Uno R3	PTB4	TPM0_CH2 / LPUART1_CTS / LPSPI0_SCK / I2S0_TX_BCLK
			RV32M1 (Pin C2)		
	13		Freedom Proprietary	PTC30	TPM0_CH1 / LPUART1_TX / LPSPI0_SCK / FXIO0_D19
			RV32M1 (Pin R7)		
	14	GND	Arduino Uno R3		
	15		Freedom Proprietary	PTC29	TPM0_CH2 / LPUART1_RX / LPSPI0_PCS3 / FXIO0_D18
			RV32M1 (Pin N6)		
	16	ARE F	Arduino Uno R3		
	17		Freedom Proprietary	PTC28	TPM0_CH3 / LPSPI0_PCS1 / FXIO0_D17
			RV32M1 (Pin U5)		
	18	D14	Arduino Uno R3	PTC9	LLWU_P16 / TPM0_CH2 / LPUART0_CTS / LPI2C0_SDA / LPSPI0_SOUT
			RV32M1 (Pin R1)		
	19		Freedom Proprietary	PTB29	LPUART3_TX / I2S0_TX_FS / FXIO0_D16

			RV32M1 (Pin L3)		
	20	D15	Arduino Uno R3	PTC10	TPM0_CH3 / LPUART0_RTS / LPI2C0_SCL / LPSPI0_PCS2
			RV32M1 (Pin R2)		

Table 3. **Arduino compatible header/connector pinout (J3 and J4)**

Header	Pin No	Name	Type / RV32M1 Pin	GPIO	Functions
	1		Freedom Proprietary		
			RV32M1 (Pin B7)	PTA21	TPM2_CH3 / LPSPI2_SOUT / EMVSIM0_PD
	2	N.C.			
	3		Freedom Proprietary		
			RV32M1 (Pin C7)	PTA20	TPM2_CH4 / LPSPI2_SCK / LPSPI1_PCS1 / EMVSIM0_IO
	4	3V3	Arduino Uno R3		
	5		Freedom Proprietary		
			RV32M1 (Pin D7)	PTA19	TPM2_CH5 / LPSPI2_PCS3 / LPSPI3_SCK / EMVSIM0_VCCEN
	6	RESET	Arduino Uno R3		
	7		Freedom Proprietary		
			RV32M1 (Pin D8)	PTA18	LPSPI2_PCS1 / LPSPI3_PCS3 / EMVSIM0_RST
	8	3V3	Arduino Uno R3		
	9		Freedom Proprietary		
			RV32M1 (Pin F7)	PTA17	LPI2C2_HREQ / LPSPI3_PCS1 / EMVSIM0_CLK
	10	5V	Arduino Uno R3		
	11		Freedom Proprietary		
			RV32M1 (Pin K5)	PTB17	LPUART3_RTS / LPI2C3_SCLS / FXIO0_D7
J3	12	GND	Arduino Uno R3		

	13		Freedom Proprietary		
			RV32M1 (Pin H5)	PTB16	LPUART3_CTS / LPI2C3_SDA / FXIO0_D6
	14	GND	Arduino Uno R3		
	15		Freedom Proprietary		
			RV32M1 (Pin G1)	PTB15	TPM0_CLKIN / LPI2C1_HREQ / LPI2C3_SCL / FXIO0_D5
	16	Vin	Arduino Uno R3		
J4	1		Freedom Proprietary		
			RV32M1 (Pin G12)	TAMPER2	
	2	A0	Arduino Uno R3		
			RV32M1 (Pin T1)	PTC11	LLWU_P17 / TPM0_CH4 / LPI2C1_SDA / LPI2C0_SDAS / LPSPI0_SIN
	3		Freedom Proprietary		
			RV32M1 (Pin F13)	TAMPER1	
	4	A1	Arduino Uno R3		
			RV32M1 (Pin R3)	PTC12	LLWU_P18 / TPM0_CH5 / LPI2C1_SCL / LPI2C0_SCLS / LPSPI0_PCS0
	5		Freedom Proprietary		
			RV32M1 (Pin F14)	TAMPER0	
	6	A2	Arduino Uno R3		
			RV32M1 (Pin F4)	PTB9	ADC0_SE3 / LPI2C1_SCL / LPSPI0_PCS1 / I2S0_RXD1 / FXIO0_D0
	7		Freedom Proprietary		
			RV32M1 (Pin P12)	PTE2	ADC0_SE19 / LPI2C0_SCLS / LPSPI3_PCS3 / SDHC0_D0
	8	A3	Arduino Uno R3		
			RV32M1 (Pin M11)	PTE4	ADC0_SE21 / TPM1_CLKIN / LPI2C0_SCL / LPSPI3_SOUT / SDHC0_D6
	9		Freedom Proprietary		
			RV32M1 (Pin R17)	PTE5	LPI2C0_HREQ / LPSPI3_PCS2 / SDHC_DCLK

	10	A4	Arduino Uno R3 (Pin M13)	PTE10	LLWU_P25 / TPM3_CH0 / LPUART3_CTS / LPI2C3_SCLK / SDHC0_D4
			RV32M1 (Pin P12)	PTE2	ADC0_SE19 / LPI2C0_SCLS / LPSPI3_PCS3 / SDHC0_D0
	11		Freedom Proprietary		
			RV32M1 (Pin T17)	DAC0_OUT	
	12	A5	Arduino Uno R3 (Pin M14)	PTE11	TPM3_CH1 / LPUART3_RTS / LPI2C3_SCL / SDHC0_D3 / FXIO0_D2
			RV32M1 (Pin N12)	PTE3	LLWU_P22 / ADC0_SE20 / TPM0_CLKIN / LPI2C0_SDA / SDHC0_D7

4.2 Jumper table

Table 4 describes the jumper settings on the RV32M1-VEGA. * denote jumper selection is shorted on board by default. Bold text indicates default selection.

Table 4. **RV32M1-VEGA jumper table**

Signal	Jumper designator	Option	Setting
VDD_CORE	J10	**1-2**	**VDD_CORE to VOUT_CORE**
		Open	Core power bypassed / off
VBAT	J50	**1-2**	**VBAT powered by 3V3 LDO**
		2-3	VBAT powered by coin cell battery
BRD_PER	J18	**1-2**	**Non-MCU circuitry powered by 3V3 LDO**
		2-3	Non-MCU circuitry powered by external coin cell battery
BRD_IO_PER	J19	**1-2**	**IO power for peripheral board components powered by VTGT_MCU**
		2-3	IO power for peripheral board components powered by 1V8 LDO
VDDIO1_MCU	J16	**1-2**	**VDDIO1 powered by VTGT_MCU**
		2-3	VDDIO1 powered by VOUT_RF

VDDIO2	J14	**1-2**	**VDDIO2 powered from BRD_IO_PER**
		2-3	VDDIO2 powered from VOUT_RF

VDD_DCDC	J49	**1-2**	**VDD_DCDC powered by VTGT_MCU**
		Open	VDD_DCDC unpowered.

VDDA_RV32	J28	**1-2**	**VDDA_RV32 powered by VDDIO1_MCU**
		Open	VDDA_RV32 unpowered.

VREGIN	J31	***1-2**	**VREGIN powered by P5V_INPUT**
		Open	VREGIN unpowered.

OpenSDA Level Shifter Power	J51	**1-2**	**VDDIO1_SDA and VDDIO1_SDA_SPI powered by VDDIO1.**
		Open	VDDIO1_SDA and VDDIO1_SDA_SPI unpowered and isolated.

Visible Light Sensor	J47	***1-2**	**Visible light sensor powered by VDDA_RV32.**
		Open	Visible light sensor unpowered.

Power LED	J5	***1-2**	**Power LED powered by BRD_PER**
		Open	Power LED unpowered.

SDA Voltage Sense	J52	**1-2**	**OpenSDA VDD sense circuit is connected to P3V3 ADJ supply**
		Open	OpenSDA VDD sense circuit disconnected.

5 References

Following references are available on www.open-isa.org:

1. *RV32M1-VEGA-SCH: Schematics*
2. *RV32M1-VEGA-LAYOUT: Layout*
3. *RV32M1RM: Reference Manual*
4. *RV32M1DS: Datasheet*

6 Revision history

Rev.	Date	Substantive change(s)
0	11/2018	Initial release

VEGA