Firmware Integration Manual

MARS & BlueShift Development Kits

Introduction

The VX modular augmented reality system (MARS) is available as a complete AR reference design. The MARS is a tested system that was designed to reduce lead time and development risk for new headset designs.

MARS offers everything needed for USB Type-C PD negotiation, sensor integration, and onboard data processing for plug-n-play accessory modules. The reference supports two FHD CNED displays, 5 Gbit/s data bus, high-speed multiplexing, and full device control via the serial, MIPI, and I2C interfaces. For more information, refer to the Module Communication Section.

This document describes the process of connecting to and programming the dedicated application microcontroller in the system. Additional resources can be found here:

www.vx-inc.com/technical-resources.

To improve ease of customization and reduce development effort, the MARS has a dedicated application microcontroller (ATMEGA32U4) designed to allow the user to program system functionality specific to their needs. The microcontroller is programmable over USB through the main device USB-C port.

The microcontroller can interface with:

- Backlight Brightness Control of the CNED
- All of the user modules through
  - I2C
  - Digital IO / PWM
- Ambient Light Sensor
- Virtual Serial Comm Port, for active communication over the USB 2.0 Bus on the USB-C connection
- User Buttons (optional on specific user modules)
- External Illumination (optional on specific user modules)

Hardware Overview

Module Communication

The MARS has 3 module locations: Right, Center, and Left, that allow for a variety of sensors and components to be added with little development effort. All of the accessory modules have the following communication interfaces available:

USB

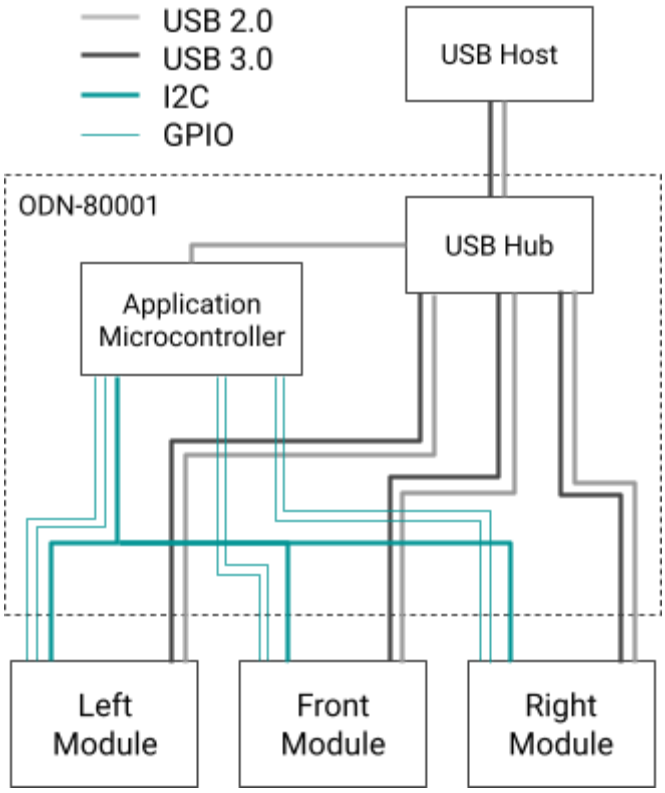
Both USB 2.0 and USB 3.0 fed from an onboard USB Hub. All USB devices accessible from the USB Host through the USB-C Cable. The USB 3.0 lines running to each module contain a single full duplex high-speed lane.

I2C

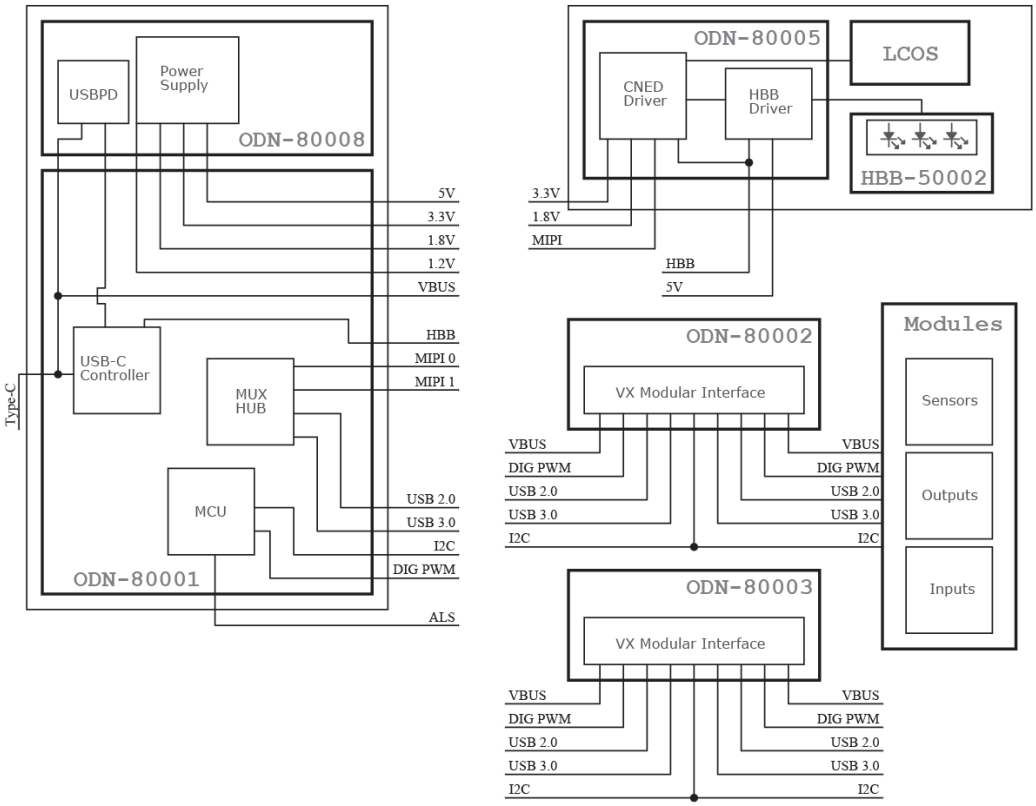
Connected to the Dedicated Application Microcontroller, each module can support multiple I2C devices with the Microcontroller acting as the host. The parameters of the bus and configuration of all devices connected to the bus are controlled by the application microcontroller.

GPIO

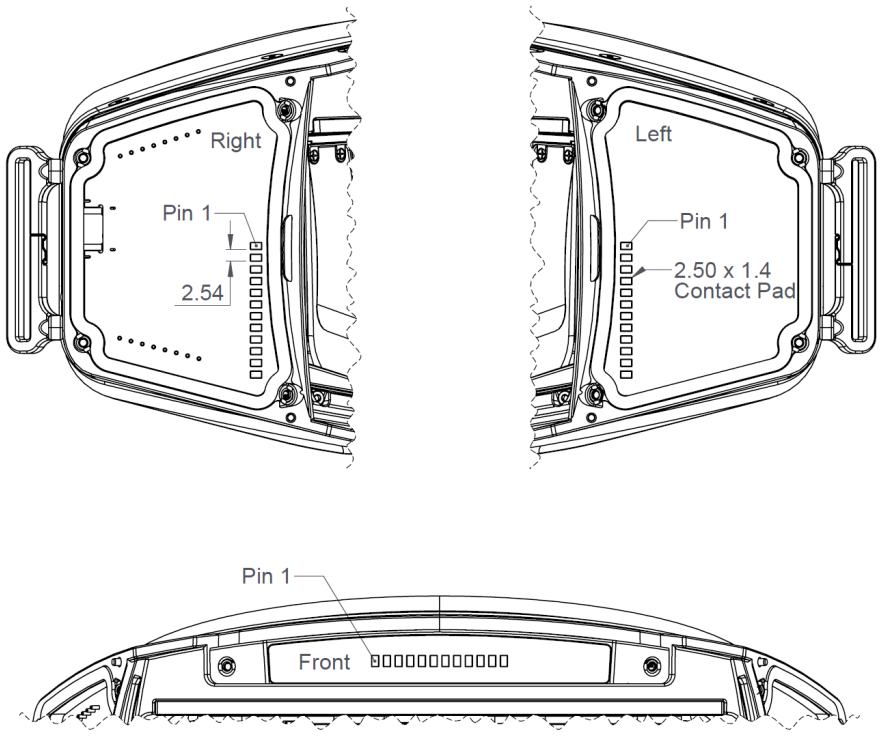
Each one of the accessory bays has 2 dedicated GPIO pins directly connected to the Dedicated Application Microcontroller. This can be used as Digital IO or PWM.



Functional Block Diagram



Module Pinout Diagram



Module Pinouts

ATMEGA32U4 I/O and I2C Pinouts highlighted in the table below

Right Module

Connector - Pin	Name	Type	Description
J2 - 1	DGND	Ground	Device Ground
J2 - 2	USB_RMOD_N	Input / Output	Right Module USB 2.0, Negative
J2 - 3	USB_RMOD_P	Input / Output	Right Module USB 2.0, Positive
J2 - 4	USB_RMOD_TX_P	Input / Output	Right Module USB 3.0 Transmit, Positive
J2 - 5	USB_RMOD_TX_N	Input / Output	Right Module USB 3.0 Transmit, Negative
J2 - 6	USB_RMOD_RX_P	Input / Output	Right Module USB 3.0 Receive, Positive
J2 - 7	USB_RMOD_RX_N	Input / Output	Right Module USB 3.0 Receive, Negative
J2 - 8	**RMOD_PWM_D5**	**Input / Output**	**Sensor Input/Output for Right Module**
J2 - 9	**RMOD_PWM_D6**	**Input / Output**	**Sensor Input/Output for Right Module**
J2 - 10	**SDA**	**Input / Output**	**I^2C Data**
J2 - 11	**SCL**	**Input**	**I^2C Clock**
J2 - 12	VBUS_15V	Power	USBPD VBUS Power

Front Module

Connector - Pin	Name	Type	Description
J16 - 1	DGND	Ground	Device Ground
J16 - 2	USB_FMOD_N	Input / Output	Front Module USB 2.0, Negative
J16 - 3	USB_FMOD_P	Input / Output	Front Module USB 2.0, Positive
J16 - 4	USB_FMOD_TX_P	Input / Output	Front Module USB 3.0 Transmit, Positive
J16 - 5	USB_FMOD_TX_N	Input / Output	Front Module USB 3.0 Transmit, Negative
J16 - 6	USB_FMOD_RX_P	Input / Output	Front Module USB 3.0 Receive, Positive
J16 - 7	USB_FMOD_RX_N	Input / Output	Front Module USB 3.0 Receive, Negative
J16 - 8	**FMOD_DIG_D7**	**Input / Output**	**Sensor Input Or Digital Output for Front Module**
J16 - 9	**FMOD_DIG_D8**	**Input / Output**	**Sensor Input Or Digital Output for Front Module**
J16 - 10	**SDA**	**Input / Output**	**I^2C Data**
J16 - 11	**SCL**	**Input**	**I^2C Clock**
J16 - 12	VBUS_15V	Power	USBPD VBUS Power

Left Module

Connector - Pin	Name	Type	Description
J21 - 1	DGND	Ground	Device Ground
J21 - 2	USB_LMOD_N	Input / Output	Left Module USB 2.0, Negative
J21 - 3	USB_LMOD_P	Input / Output	Left Module USB 2.0, Positive
J21 - 4	USB_LMOD_TX_P	Input / Output	Left Module USB 3.0 Transmit, Positive
J21 - 5	USB_LMOD_TX_N	Input / Output	Left Module USB 3.0 Transmit, Negative
J21 - 6	USB_LMOD_RX_P	Input / Output	Left Module USB 3.0 Receive, Positive
J21 - 7	USB_LMOD_RX_N	Input / Output	Left Module USB 3.0 Receive, Negative

J21 - 8	**LMOD_PWM_D9**	**Input / Output**	**Sensor Input/Output for Left Module**
J21 - 9	**LMOD_PWM_D10**	**Input / Output**	**Sensor Input/Output for Left Module**
J21 - 10	**SDA**	**Input / Output**	**I^2C Data**
J21 - 11	**SCL**	**Input**	**I^2C Clock**
J21 - 12	VBUS_15V	Power	USBPD VBUS Power

Dedicated Application Microcontroller

The onboard user-programmable dedicated application microcontroller is an ATMEGA32U4-MU. The microcontroller has a 16MHz crystal attached and no hardware reset button.

This is further described in the programming section of this document and the sample application code in the "MARS Firmware Application Examples" document.

USB Device and Bootloader

This device comes preloaded with an Arduino bootloader. When the bootloader is included on the microcontroller the device automatically enumerates as a Virtual Comm Port over USB 2.0, assuming the Arduino drivers have been installed.

Application Microcontroller Pinouts

The following are direct pinouts from the Dedicated Application Microcontroller (ATMEGA32U4) leading to the accessory module pins. Note that the function of each module pin can vary depending on the Module installed. There is example code written for the Arduino development environment outlined in the "MARS Firmware Application Examples" document, found here: www.vx-inc.com/technical-resources.

Microcontroller Pin	Arduino Pin	Name	Pin Mode	Description
PB5	D9	LMOD_PWM_D9	GPIO	GPIO For the Left Module
PB6	D10	LMOD_PWM_D10	GPIO	GPIO For the Left Module
PE6	D7	FMOD_DIG_D7	GPIO	GPIO For the Front Module
PB4	D8	FMOD_DIG_D8	GPIO	GPIO For the Front Module
PC6	D5	RMOD_PWM_D5	GPIO	GPIO For the Right Module
PD7	D6	RMOD_PWM_D6	GPIO	GPIO For the Right Module
PD1	SDA	SDA	I2C	Common I2C Data for all accessory modules
PD0	SCL	SCL	I2C	Common I2C Clock for all accessory modules
PB7	D11	BACKLIGHT	PWM	Backlight Control for the CNEDs (PWM)
PF7	A0	ALS	Analog	Analog Input for the ambient brightness level

Setting CNED Backlight

The backlight brightness is based on the duty cycle of the BACKLIGHT pin. Where there are 10 settings:

0 - Off
1 - 9 Intermediate Brightness Levels
10 - Maximum Brightness

The duty cycle is interpreted and not directly controlled by a hardware element. To set a specific brightness level, use the following equation to determine the duty cycle that must be set on the backlight pin.

% Duty Cycle = ((Brightness 0 - 10) * 10 + 15) / 255;

Image Sensor Module Module Application

The ODN-50005 Imaging Sensor Module is an example module created by VX in order to demonstrate cameras and user buttons on the BlueShift headset. The module connects in the Right Module Bay and utilizes:

- USB 2.0:
  - RGB Image Sensor
  - IR Image Sensor
- I2C: User button inputs
- PWM:
  - Adjustable brightness White LED
  - Adjustable brightness IR LED

There is an example of this application linked in the “MARS Firmware Application Examples” document.

The pinouts for the microcontroller to the ODN-50005 are as follows:

Microcontroller Pin	Arduino Pin	Name	Pin Mode	Description
PC6	D5	RMOD_PWM_D5	PWM	PWM for the White LED
PD7	D6	RMOD_PWM_D6	PWM	PWM for the IR LED
PD1	SDA	SDA	I2C	Common I2C Data for all accessory modules
PD0	SCL	SCL	I2C	Common I2C Clock for all accessory modules
PB7	D11	BACKLIGHT	PWM	Backlight Control for the CNEDs (PWM)
PF7	A0	ALS	Analog	Analog Input for the ambient brightness level

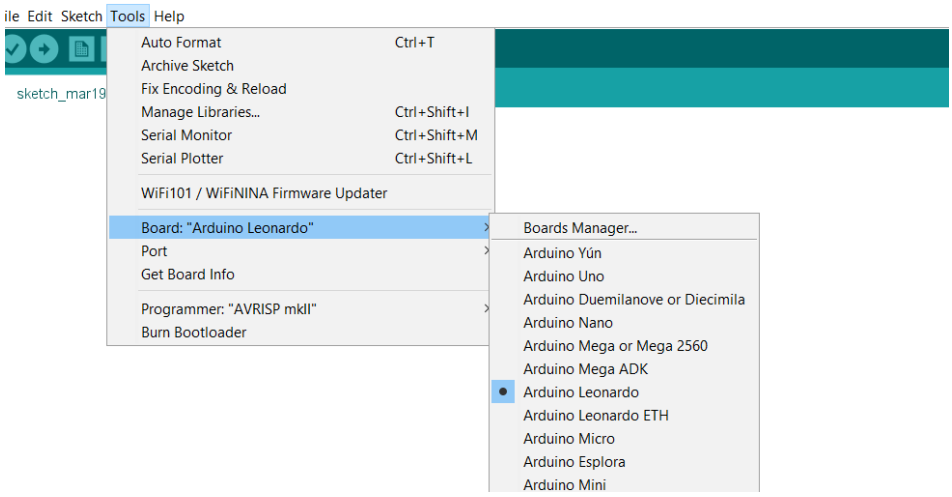
The I2C expander containing the buttons is at I2C address 0x20. When the device is read, an 8-bit packet should be returned, where the up button press is connected to LSB 0 and the down button to LSB 4. Both buttons are active low.

The PWM brightness for the White and IR LEDs are simply set on their corresponding pins, where 0% duty cycle is off and 100% duty cycle is full brightness.

Programming

The application microcontroller comes shipped with a USB Bootloader to allow programming VIA the Arduino IDE for the user's convenience. To get started with the Arduino IDE, visit https://www.arduino.cc/en/software and download the latest version. Install the software and connect the headset using a USB-C cable.

In the Arduino IDE select the Arduino Leonardo:

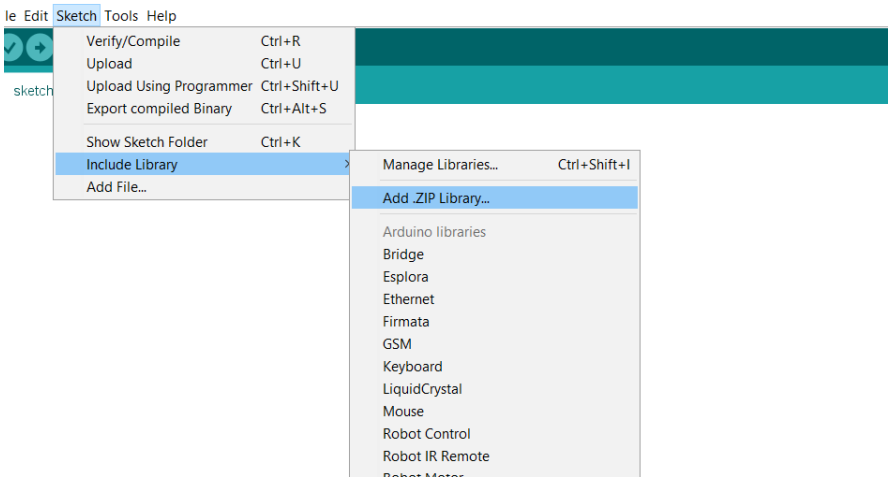


Download the MARS library and add it to the Arduino IDE:

MARS Library (https://github.com/VX-inc/VX_MARS)

If you are using the EV1 Board, download the EV1 Library (https://github.com/VX-inc/VX_EV1) as well. Note, that both libraries must be installed to use the EV1.

Download a zip of the repository from Github and install it in the application as shown below, or clone the library directly to your libraries folder.



Additional Resources

More resources can be found on the VX website here:

www.vx-inc.com/technical-resources

Or at our GitHub:

https://github.com/VX-inc

For questions or project inquiries please email us:

info@vx-inc.com

Or fill out our contact form:

www.vx-inc.com/about/#contact

Important Notice – Please Read Carefully

***

No license, express or implied, to any intellectual property right, is granted by VX herein.

VX Inc. reserves the right to make changes, modifications, and corrections to VX products and/or to this document at any time without notice. Information in this document supersedes and replaces information previously supplied in any prior versions of this document.

Customers are solely responsible for the use of VX products and VX assumes no liability for application assistance or the design of Customers' products.

Storage: 22°C at 50% relative humidity is recommended. Prolonged storage is not recommended.

Resale of VX products shall void any warranty granted by VX for such products.

This product shall not be used in life-support devices or other medical systems. Customer to independently verify information and shall test for all required certifications, including but not limited to, RoHS, ANSI Z87, and FDA.

***

Augmented Reality
Design
Displays
Integration