CutiPy™

User Manual

November 2019

Revision 1.20

(for use with Rev 1 Boards)

EMAC, inc.
EQUIPMENT MONITOR AND CONTROL
2390 EMAC Way,   Carbondale, Illinois 62902
Phone: (618) 529-4525   Fax: (618) 457-0110
http://www.emacinc.com

Table of Contents

Disclaimer

EMAC Inc. does not assume any liability arising out of the application or use of any of its products or designs. Products designed or distributed by EMAC Inc. are not intended for, or authorized to be used in, applications such as life support systems or for any other use in which the failure of the product could potentially result in personal injury, death or property damage.

If EMAC Inc. products are used in any of the aforementioned unintended or unauthorized applications, Purchaser shall indemnify and hold EMAC Inc. and its employees and officers harmless against all claims, costs, damages, expenses, and attorney fees that may directly or indirectly arise out of any claim of personal injury, death or property damage associated with such unintended or unauthorized use, even if it is alleged that EMAC Inc. was negligent in the design or manufacture of the product.

EMAC Inc. reserves the right to make changes to any products with the intent to improve overall quality, without further notification.

1 Introduction

This document describes EMAC's CutiPy Industrial IoT microcontroller. The CutiPy is a Low Power Industrial IoT device that was designed to simplify connecting devices and machines to the multitude of systems you find in the Industrial environment. This module is built around the STMicroelectronics STM32 microcontroller, which provides several of its key features.

The CutiPy provides a number of features that are native to the board (Internal) and a number of features that are accessed through two 50 Pin Header Connectors that are referred to as External.

In addition to the features listed below, the CutiPy also features a fast 32-bit core, open source software support, and a wide range of controller I/O pins.

1.1 Features

- **Low Power Industrial IoT**
- **STMicroelectronics ARM Cortex-M4 168MHz**
- **Up to 1M of Flash**
- **192 Kbytes of SRAM**
- **1x microSD Card Slot**
- **1x CAN 2.0B Port (Internal provided w/Transceiver, External provided w/o Transceiver)**
- **2x USB 2.0 Ports (1x Full Speed Device w/ PHY, 1x Full/High-Speed Device)**
- **4x Serial Ports**
  - 1x RS232 & 1x RS232/422/485 (Internal)
  - 2x TTL (External)
- **1x SPI Port**
- **2x I2C Hardware Ports**
- **25x Timer/Counters/PWM/Capture**
- **RTC with on-board battery backup**
- **Internal Temperature Sensor**
- **External Li-Ion Battery with charging from USB or 5V Power Supply Header**
- **1x Reset Button**
- **13x A/D Channels with 12-bit Resolution (3 Unique A/Ds)**
- **2x D/A Channels with 12-bit Resolution**
- **16x External Dedicated GPIOs (64x fully allocated)**
- **8x High-Drive Open Collector Outputs**
- **2x 50-Pin Expansion Headers**
- **OS/Language: MicroPython or FreeRTOS**
- **Graphic 128x32 LCD with 4x pushbuttons & 4x LEDs (optional)**

- **Redpine RS9116 (BT/Wi-Fi/Zigbee/Thread) with on board antenna (optional)**
  - **802.11 a/b/g/n Wi-Fi**
  - **802.15.1 Bluetooth**
  - **802.14.4 ZigBee & Thread**

1.2 CutiPy Specifications

- **CPU:** STM32F407IGH6 ARM Cortex-M4 w/Math Coprocessor
- **Flash:** Up to 1MB of Flash
- **RAM:** 192 KB of SRAM
- **Flash Disk:** Micro SD Card Socket
- **System Reset:** Supervisor with external Reset Button provision
- **RTC:** Battery backed Real Time Clock
- **Timer/Counters:** 25x Timer/Counter/PWM/Capture
- **Digital I/O:** 16x External Dedicated GPIOs (64x fully allocated)
- **Analog I/O:**
  - 13x A/D Channels with 12-bit Resolution (3 Unique A/Ds)
  - 2x D/A Channels with 12-bit Resolution
- **Power:**
  - 5V Power Supply Header
  - USB Device Port
- **JTAG:** Processor JTAG Supporting Programming, Trace, and Boundary Scan

1.3 LCD

LCD – 36 x 12 mm Graphic LCD (NHD-C12832A1Z-FSW-FBW-3V3):

- **Display Type:** Film Super-Twisted Nematic
- **Resolution:** 128 x 32 pixels
- **Dot pitch:** 0.25mm x 0.25mm
- **Viewing Angle:**
  - **ϴ-right:** 40°
  - **ϴ-left:** 40°
  - **ϴ-top:** 20°
  - **ϴ-bottom:** 40°
- **Backlight:** White LED

1.4 Serial Interfaces

- **UARTS:** 1x RS232, 1x RS232/422/485, 2x TTL
- **SPI:** 1x SPI Ports
- **USB:** 2x USB 2.0 Ports
  - 1x Full-Speed Device terminating to a micro USB connector
  - 1x Full-Speed/High-Speed Device terminating to a header connector
- **I2C:** 2x I2C hardware Ports

1.5 Mechanical and Environmental

- **Dimensions:** 2.3" ×3.5" (58.42mm × 88.9mm)
- **Power Supply Voltage:** 5 Vdc
- **Power Requirements (typical):**
  - 5.0 Volts @ 50mA (0.25 watts)
  - Typical Running Current Consumption: ~50mA
  - Sleep mode current: ~100µA
- **Operating Temperature:** -40 ~ 85° C (-40 ~ 185 ° F)
  - LCD: -20 ~ 70° C (optional)
- **Operating Humidity:** 0% ~ 90% relative humidity, non-condensing

2 CutiPy Product Details

2.1 Jumper Configuration & Connector Descriptions

The CutiPy comes factory configured. In the event that jumpers need to be verified or modified, this section provides the information required including instructions on setting jumpers and connecting peripherals, switches, and indicators. Be sure to read all the safety precautions before you begin any configuration procedure. See Appendix A for connector pinouts and Appendix B for Jumper Setting descriptions.

Table 1: Jumpers

Label	Function	Default
JB1	Boot1 Source Selection	ROM
JB2	Boot0 Source Selection	FLS
JB3	RTC & RAM Retention Battery	OFF
JB4	CAN Termination	TRM

Table 2: Connectors

Label	Function
CN1	Main Battery Connector
CN2	Serial Port UART2
HDR1	I/O Header 1
HDR2	I/O Header 2
HDR3	Vin Power 4-Pin Male Header
HDR4	CAN 2.0 Port (CAN-1)
HDR5	Serial Port UART3
HDR6	JTAG Header
JK1	USB Connector
SOK1	MicroSD Card Socket

2.2 Power Connectors

The CutiPy provides one power connector. HDR3 is a TE Connectivity AMP 4-pin male connector (Part # 640456-4) and mates with TE Connectivity AMP 4-pin female socket (Part # 1375820-4). As an alternative, the CutiPy can also be powered through the micro USB connector located at JK1. This connector is a Wurth Electronics right angle 5 pin receptacle connector (Part# 629105136821).

The CutiPy only requires 5Vdc to operate. Vin (HDR3 pin 4) is not utilized on the CutiPy and is simply routed to Expansion Connectors HDR1 & HDR2 (pin 5). However, the pullup option for the HiDrive Outputs is routed to HDR1 (pin 6) and therefore Vin can be used to pull-up HiDrive Outputs by connecting HDR1 pin 5 and pin 6 together.

Table 3: Power Connector Pinout (HDR3)

Pin	Signal
1	+Vin ~12V
2	GND
3	GND
4	+Vin 5V

2.3 Battery Operation and Charging

The CutiPy can be operated off of a Lithium Ion Battery as a main power source or backup power source. The battery is to be connected to CN1 (connector type JST - S2B-PH-SM4-TB(LF)(SN)). Note: if a Li-Ion battery is used, the battery may limit the operation temperature of the system.

The CutiPy utilizes a Linear Technologies LTC4085 USB Power Manager & Lithium Ion Battery charger chip. This device seamlessly will take power from one of three power sources: USB, 5-volt (Mini-Floppy) power connector, or external Battery, and supply whichever power source is most appropriate to the rest of the CutiPy’s internal power distribution.

Internal power can be visualized as a diode-OR arrangement where any one of three power sources can power the CutiPy so long as any one of them is active. The junction where all three meet may have a wide voltage variation from just at or below 5 volts, all the way down to 3 volts depending on conditions. The junction of all three power sources is had by intelligent switching of Mosfets and is fed to a boost-buck switching regulator that will buck voltages higher than 3.3 volts down or boost voltages lower than 3.3V up to the normal system voltage of 3.3 volts.

While functioning in boost mode the boost-buck switcher allows the system maximum “hang-time“, when the system is running from an external battery whose voltage may be falling as the battery approaches discharge.

Internal power distribution priority is as follows:

Floppy 5V power 1st: 5V floppy power goes to LTC4085 Battery power manager and to Utility header's 5V I/O pins. 5 volts power can be provided through the Utility I/O just as easily as through the floppy power connector.

USB 5V power 2nd: if Floppy power is absent. This is not directly fed to the Utility Header's 5V I/O.

Battery power 3rd: if both Floppy and USB power is absent. This is also not directly fed to the Utility Header's 5V I/O.

Whichever of (or all three) power sources are present, they are seamlessly switched by Mosfets to a single, combined power supply node which feeds a LO-drop Lo-quiescent current switching Regulator, a Texas Instrument's TPS63070. This boost-buck switching regulator takes whatever actual voltage is present at the combined power supply node (3 to 5 volts) and provides a regulated 3.3 volts to the CutiPy's main power bus.

If a charged external battery is present at header connector CN1, the LTC4085 will charge this battery from either the USB power or the Mini-Floppy connector; HDR3. The flexibility provided with the LTC4085 along with the low basic power requirements of the CutiPy allow the system to be powered by the USB alone if only a USB host is powering the unit.

If a stand-alone 5-volt power source is present and connected to the +5 and GND pins of HDR1, the LTC4085 will preferentially take power from it instead, which normally takes precedence over the USB power source.

The USB power input is set by the LTC4085 to limit the input current to no more than 500 mA total, as 500 mA is the usual current limit providable by a USB Host. This internal limit prevents overloading of a typical USB port. Should the load current taken by the CutiPy plus the battery charge current attempt to exceed the 500 mA USB limit, the battery charge current will be automatically reduced to compensate, keeping the USB port current below 500 mA. This will of course increase the time required for full battery recharge.

The Mini-Floppy provided 5 volts has essentially higher limit for those rare cases where a CutiPy might be connected to a larger system that uses higher currents, but this is an exceptional case. The Mini-Floppy power input is limited to 1.1 amps with a polyfuse.

EMAC can provide a 3.7 volt @ 1.2 Amp-hour Lithium-Ion rechargeable battery (EMAC Part#: PER-PWR-0101PR0). This battery also has a 2mm pitched two position receptacle that perfectly mates to the External Battery connector CN1. Because the CutiPy and the LTC4085 can and expects to charge a Li-Ion battery, only an Li-Ion battery should ever be connected to the External Battery connector CN1. Although Li-Ion batteries similar to PER-PWR-0101PR0 (Li-Ion single cell) with higher or lower Amp-Hour capacities can be used, **never connect a battery of different voltage or another kind of rechargeable or primary battery to CN1!**

PER-PWR-0101PR0, does not have any internal battery protection circuitry, which is recommended for production use. However, the LTC4085 Battery Charger chip does automatically reduce the Battery charge current such that the sum of the load current and charge current does not exceed the programmed input current limit (the battery charge current goes to zero when load current exceeds the programmed input current limit) and has Current Limit & Charger Undervoltage Lockouts.

The LTC4085 has been programmed to recharge the external battery (when used) with a current of 250 mA, given the PER-PWR-0101PR0's 1200 mAh (1.2 Amps) capacity a typical recharge from near dead to fully charged takes around 4 and ¾ hours for this battery. If another Li-Ion battery of different capacity is used instead, one can easily recalculate that battery's charge time by dividing the battery capacity in mAh by 250.

The LTC4085 will terminate the battery charge cycle when the battery reaches 4.2 volts. Logic pins are connected to the CutiPy's MPU so the CutiPy can know the status of the charge cycle.

Temperature monitoring of the battery is not provided by the CutiPy.

A voltage measurement analog point is provided by other CutiPy circuitry to measure the external battery's voltage, it can be used to predict battery charge lifetime.

Table 4: Power Connector Pinout (CN1)

Pin	Signal
1	Battery +
2	Battery -

2.4 Serial Ports

The CutiPy is equipped with four serial ports. One RS-232 terminates to a male DB9 connector (UART2), one RS-232/422/485 terminating to 10-pin header connector (UART3), and two TTL accessed through Utility I/O header 2 (HDR2).

- The RS-232 serial port (UART2) terminates to a male DB9 connector located at CN2. The RS-232 Transceiver (U7) limits the maximum throughput to 250 kb/s. The transceiver can be placed in standby mode by holding GPIO PE1 low. In standby mode the transceiver will typically draw 1µA. When the device is powered down the receivers remain active and the drivers are placed in a high-impedance state.

- RS-232/422/485 (software-configurable) serial port (UART3) terminates to a 10-pin header located at HDR5. By default, the RS-232 transceiver (U11) and the RS-422/485 transceiver (U10) are in standby mode. In standby mode, the RS-232 transceiver typically draws 1µA and the RS-422/485 transceiver typically draws 0.1µA. Both devices' driver and receiver output(s) will be high-impedance while in standby mode. To configure RS-232 mode, PE2 should be held high. The maximum throughput for RS-232 is 250 Kb/s. To configure RS-422/485 mode, PF11 should be held high and the maximum throughput in this mode is 2.62Mb/s (Oversampling by 16) and 5.25 Mb/s (Oversampling by 8).

- The TTL serial ports (UARTs 1 & 6) terminates to Utility I/O header 2 (HDR2). UART1 provides for bidirectional communication offering Tx and Rx lines with a maximum bit rate of 10.5 Mb/s. UART6 provides for bidirectional communication offering Tx, Rx, CTS, RTS, and CK lines with Synchronous Mode and Hardware Flow Control Mode. The maximum bit rate for this is 10.5Mb/s. These also support LIN, Smartcard Protocol, and IrDA.

Table 5: Serial Port-UART2 (CN2)

Pin	DB9 Connector: Description
1	NC
2	RXD
3	TXD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	NC

Table 6: Serial Port-UART3 (HDR5)

Pin	10-Pin Header: Description (RS-232)	10-Pin Header Description (RS-485/422)
1	NC	TX-
2	NC	NC
3	RXD	TX+
4	RTS	NC
5	TXD	RX+
6	CTS	NC
7	NC	RX-
8	NC	NC
9	GND	GND
10	NC	NC

Table 7: Serial Port-UART1 & UART6 TTL (HDR2)

Pin	Utility I/O 2 Pinout
38	USART1_TXD_3V
39	USART1_RXD_3V
40	USART6_CLK_3V
41	USART6_TXD_3V
42	USART6_RXD_3V
43	USART6_CTS_3V
44	USART6_RTS_3V

2.5 USB Ports

The CutiPy provides for two USB 2.0 ports. The STM32 features one OTG FS core, one OTG HS core, and one OTG FS PHY where only 1 core can utilize the internal OTG FS PHY at a time. If both USB interfaces are to be utilized, an external PHY will need to be implemented. Additionally, the HS mode will require an external HS PHY connected to the STM32's ULPI interface.

One port is a USB 2.0 Full-Speed (FS) Device and can be accessed via a micro USB 2.0 connector located at JK1. This USB Full-Speed port supports 12Mb/s transfers in device mode.

The second port is a USB 2.0 High-Speed (HS) OTG Port which can also be used as a Full Speed OTG Port and is accessed through the Utility I/O Header 1 located at HDR1. To use the HS USB as a Full-Speed Port all that is required is the connection of a USB Connector to pins 45 – 48 (see below). However, to use the HS USB as High-Speed Port an external HS USB PHY is required as well as a USB Connector. This port could be used as OTG (Device or Host), but would require external VBUS switch circuitry. Both USB Ports support the Sessions Request Protocol (SRP) and soft disconnect features.

The Table below calls out the necessary connections for the USB_HS_OTG Port.

Table 8: USB HS Port Pin Assignments (HDR1 & HDR2)

Pin	CutiPy Pin Name	USB ULPI Pin Name
HDR2: 34 or 15	PI11 or ADC123_IN12 (PC2)	OTG_HS_ULPI_DIR
HDR2: 16 or 24	ADC123_IN13 (PC3) or I2C2_SCL (PH4)	OTG_HS_ULPI_NXT
HDR2: 14	ADC123_IN10 (PC0)	OTG_HS_ULPI_STP
HDR2: 21	DAC_OUT2 (PA5)	OTG_HS_ULPI_CK
HDR2: 7	ADC123_IN3 (PA3)	OTG_HS_ULPI_D0
HDR2: 18	ADC12_IN8 (PB0)	OTG_HS_ULPI_D1
HDR2: 19	ADC12_IN9 (PB1)	OTG_HS_ULPI_D2
HDR1: 16	TIM2_CH3 (PB10)	OTG_HS_ULPI_D3
HDR1: 17	TIM2_CH4 (PB11)	OTG_HS_ULPI_D4
HDR2: 32	SPI-1_MOSI (PB5)	OTG_HS_ULPI_D7
HDR1: 45	CAN2_RX (PB12)	OTG_HS_ULPI_D5
HDR1: 46	CAN2_TX (PB13)	OTG_HS_ULPI_D6
HDR1: 47	USB_OTG_HS_D_N (PB14)	*OTG_HS_DM
HDR1: 48	USB_OTG_HS_D_P (PB15)	*OTG_HS_DP

* indicates Full-Speed lines

2.6 MicroSD Card Socket

The CutiPy provides a high capacity MicroSD socket. This socket is hot-swappable and can accept a wide variety of Flash Cards. A card that is written to by the CutiPy can be read by another computer using a MicroSD card reader. The MicroSD interface is compatible with Standard and High Capacity MicroSD cards.

2.7 LCD (Optional)

The CutiPy can be equipped with an LCD that includes 4x user pushbuttons and LEDs on the top side of the board. The LCD is a 128x32 pixel Graphic Newhaven Display Intl NHD-C12832A1Z-FSW-FBW-3V3 with 4-Line SPI MCU interfaces.

2.8 CAN Port

The CutiPy provides a CAN 2.0B port utilizing the TI TCAN334GDCNT Transceiver chip. The CAN port is accessible via a 3-pin header located at HDR4 (TE Connectivity PN 640456-3). Jumper JB4 should be placed in the TRM position to Terminate the CAN network if the Node exists at either network endpoint. In standby mode, the transceiver will typically draw ~17.5µA. During standby mode, the CAN driver and main receiver are turned off and bi-directional CAN communication is not possible. The low power receiver and bus monitor are enabled to allow for RXD Wake Requests via the CAN bus. MCU pin PA10 should be set High to enable standby mode while PA9 should be set High to enable shutdown mode. Setting the same pins Low will disable the low power mode for the CAN transceiver chip. In MicroPython, the standby mode is disabled/enabled automatically by initializing/de-initializing the CAN class. MicroPython CAN instructions can be found in the Software Section in this manual.

Table 9: Pinout for the CAN-1 2.0 Port (HDR4)

Pin	Signal
1	CAN1_HI
2	CAN1_LO
3	GND

2.9 Real-Time Clock

The CutiPy is equipped with a battery-backed (B1) Real Time Clock (RTC) located in the STM32 microcontroller. Jumper JB3 should be placed in the ON position in order to retain system time when powered down.

2.10 Temperature Sensor

The CutiPy is equipped with an internal temperature sensor located in the STM32 microcontroller that can be used to measure the ambient temperature of the device. ADC1_IN16 channel is used to convert the sensor output voltage to a digital value.

2.11 Reset

The CutiPy is equipped with a reset button located at PB5. Pressing this button will cause the system to reset.

2.12 Wireless (Optional)

The CutiPy, Redpine RS9116 Wireless Radio Module, provides both an internal on-board antenna and an external antenna provision. This module by default utilizes the on-board chip antenna but offers a U.FL connector for external antenna connection with a Software option to select either Antenna option. EMAC can provide an antenna kit that plugs into the chip's antenna U.FL jack.

The Redpine Wireless Module offers high throughput and extended range along with Wi-Fi and Bluetooth (BT) coexistence in a power-optimized design capable of running Tread and Zigbee. Some of the features include:

Wi-Fi:

- Compliant to single-spatial stream IEEE 802.11 b/g/n, 802.11j with single band support
- Support for 20 MHz and 40 MHz channel bandwidths
- Operating Frequency Range: 2412 MHz - 2484 MHz
- Transmit power up to +20dBm with integrated PA
- Receive sensitivity as low as -97 dBm

Bluetooth:

- Compliant to dual-mode Bluetooth 5
- Operating Frequency Range: 2.402 GHz - 2.480 GHz
- Receive sensitivity as low as -104 dBm

RF Features:

- Integrated baseband processor with calibration memory, RF transceiver, high-power amplifier, balun and T/R switch
- Integrated Antenna and u.FL connector

2.13 I/O Expansion

The CutiPy provides access to a number of I/O lines on connectors HDR1 and HDR2. The 50-pin dual row headers contain GPIO lines, USB, UART, SPI bus, I²C bus, CAN bus, A/D lines, interrupts, high drive outputs, and power pins. The tables below list the pinouts, MCU pin designations, and the corresponding signals for both I/O headers.

Table 10: I/O Header (HDR1)

Pin#	Signal	MCU Pin Name	MCU Pin Number	Pin#	Signal	MCU Pin Name	MCU Pin Number
1	GND	N/A	N/A	**2**	GND	N/A	N/A
3	3P3_VCC	N/A	N/A	**4**	5V_JACK	N/A	N/A
5	12V_VIN	N/A	N/A	**6**	V_HIDRV	N/A	N/A
7	TIM1_ETR	PE7	R8	**8**	TIM1_CH1N	PE8	P8
9	TIM1_CH1	PE9	P9	**10**	TIM1_CH2N	PE10	R9
11	TIM1_CH2	PE11	P10	**12**	TIM1_CH3N	PE12	R10
13	TIM1_CH3	PE13	N11	**14**	TIM1_CH4	PE14	P11
15	TIM1_BKIN	PE15	R11	**16**	TIM2_CH3	PB10	R12
17	TIM2_CH4	PB11	R13	**18**	TIM3_CH2	PC7	G15
19	TIM4_CH2	PD13	M15	**20**	TIM4_ETR	PE0	A4
21	TIM4_CH3	PD14	M14	**22**	GND	N/A	N/A
23	TIM5_CH1	PH10	L13	**24**	TIM4_CH4	PD15	L14
25	TIM5_CH3	PH12	K12	**26**	TIM5_CH2	PH11	L12
27	TIM8_CH3N	PH15	D13	**28**	TIM8_CH2N	PH14	E13
29	TIM8_CH1	PI5	C4	**30**	TIM8_BKIN	PI4	D4
31	TIM8_CH3	PI7	C2	**32**	TIM8_CH2	PI6	C3
33*	PG1	PG1	M7	**34***	PG0	PG0	N7
35*	PG3	PG3	K15	**36***	PG2	PG2	L15
37*	PG5	PG5	K13	**38***	PG4	PG4	K14
39*	PG10	PG10	B10	**40***	PG6	PG6	J15
41	PF12	PF12	P6	**42**	PF13	PF13	N6
43	PF14	PF14	R7	**44**	PF15	PF15	P7
45	CAN2_RX	PB12	P12	**46**	CAN2_TX	PB13	P13
47	USB_HS_D_N	PA11	R14	**48**	USB_HS_D_P	PA12	R15
49	GND	N/A	N/A	**50**	GND	N/A	N/A

**High drive outputs. An MCU signal feeds the high drive open collector darlington driver, which in turn outputs to the indicated header Pin#*

Table 11: I/O Connector (HDR2)

Pin#	Signal	MCU Pin Name	MCU Pin Number	Pin#	Signal	MCU Pin Name	MCU Pin Number
1	GND	N/A	N/A	**2**	GND	N/A	N/A
3	3P3_VIN	N/A	N/A	**4**	5V_JACK	N/A	N/A
5	12V_IN	N/A	N/A	**6**	NC	N/A	N/A
7	ADC123_IN3	PA3	R2	**8**	ADC3_IN4	PF6	K2
9	ADC3_IN5	PF7	K1	**10**	ADC3_IN6	PF8	L3
11	ADC3_IN7	PF9	L2	**12**	ADC3_IN8	PF10	L1
13	ADC3_IN9	PF3	J2	**14**	ADC123_IN10	PC0	M2
15	ADC123_IN12	PC2	M4	**16**	ADC123_IN13	PC3	M5
17	ADC12_IN6	PA6	P3	**18**	ADC12_IN8	PB0	R5
19	ADC12_IN9	PB1	R4	**20**	DAC_OUT1	PA4	N4
21	DAC_OUT2	PA5	P4	**22**	GND	N/A	N/A
23	AVDD	N/A	R1	**24**	I2C2_SCL	PH4	H4
25	I2C2_SDA	PH5	J4	**26**	I2C2_SMBA	PH6	M11
27	I2C3_SCL	PH7	N12	**28**	I2C3_SDA	PH8	M12
29	I2C3_SMBA	PH9	M13	**30**	SPI-1_SCK	PB3	A10
31	SPI-1_MISO	PB4	A9	**32**	SPI-1_MOSI	PB5	A6
33	SPI-1_NSS	PA15	A13	**34**	PI11	PI11	E4
35	PI10	PI10	E3	**36**	PH2	PH2	F4
37	PH3	PH3	G4	**38**	USART1_TX	PB6	B6
39	USART1_RX	PB7	B5	**40**	USART6_CK	PG7	J14
41	MDM_TXD_3V	PC6	H15	**42**	MDM_RXD_3V	PG9	C10
43	MDM_CTS_3V	PG15	B7	**44**	MDM_RTS_3V	PG8	H14
45	PH13	PH13	E12	**46**	PI9	PI9	D3
47	WKUP	PA0	N3	**48**	RTC_TAMP/TS	PI8	D2
49	GND	N/A	N/A	**50**	GND	N/A	N/A

2.14 I2C

The CutiPy provides for two I2C hardware ports located on I/O header 2 (HDR2). Both can support a bit rate up to 100 kb/s (Standard mode) and 400 kb/s (Fast mode) with 7/10 bit addressing mode and a 7-bit addressing mode (as slave). By default, the I2C interface operates in Slave mode.

Table 12: I2C-2 (HDR2)

Pin#	Port Line	Description
24	I2C2_SCL	I2C2 Clock signal
25	I2C2_SDA	I2C2 Data signal
26	I2C2_SMBA	I2C2 System Management Bus Alert signal
27	I2C3_SCL	I2C3 Clock signal
28	I2C3_SDA	I2C3 Data signal
29	I2C3_SMBA	I2C3 System Management Bus Alert signal

2.15 SPI

The CutiPy is equipped with one Serial Peripheral Interface bus communicating up to 42 Mbits/s. This provides for half/full duplex synchronous transfers with external devices with an 8 or 16-bit transfer frame format selection.

Table 13: Serial Peripheral Interface SPI-1 (HDR2)

Pin#	Port Line	Description
30	SPI-1_SCK	SPI 1 Serial Clock
31	SPI-1_MISO	SPI 1 Master In Slave Out
32	SPI-1_MOSI	SPI 1 Master Out Slave In
33	SPI-1_NSS	SPI 1 Slave Select

2.16 Analog to Digital Converter

The CutiPy has 13 A/D input channels with 3 unique A/Ds available on the Utility I/O header 2 (HDR2). Voltages applied to the inputs must be in the range of 0 - 3.3V with reference to ground. For additional information please reference the STM32F407IGH6 User Manual. See Table 9 in the I/O Expansion section for pinout details.

2.17 Digital to Analog Converter

The CutiPy comes equipped with 2 D/A converters. These converters run directly from the ST Processor and are routed to Utility header 2 (HDR2). The output of the converters is 0 to 3.3V depending on digital count written to the D/A converter. For additional information please reference the STM32F407IGH6 User Manual. See Table 9 in the I/O Expansion section for pinout details.

2.18 Timers/Counters

The CutiPy is equipped with 25 timers and counters and are located on Utility I/O header 1 (HDR1). TIM1 and TIM8 are advanced control timers that are 16-bit auto-reload counter driven by a programmable prescaler. TIM2 - TIM5 are general purpose timers that consist of 16-bit or 32-bit auto reload counter driven by a programmable prescaler. These timers can be used for generating output waveforms or for measuring pulse lengths of input signals and are completely independent and do not share any resources. See Table 10 in the I/O expansion section for pinout details.

2.19 JTAG/SWD

The CutiPy provides for JTAG/SWD capabilities through the STM32 MCU. A 10-pin male connector located at HDR6 provides access to the JTAG/SWD lines which is a Sullins Electronics GRPB052VWVN-RC. This connection will allow the ability to program and debug software.

Table 14: JTAG/SWD (HDR6)

Pin#	Signal	Description	MCU Pin Name	MCU Pin Number
1	3P3_VCC	3.3 Volts	N/A	N/A
2	SWDIO	Serial Wire Data Input/Output	PA13	A15
3	GND	Ground	N/A	N/A
4	SWDCLK	Serial Wire Clock	PA14	A14
5	GND	Ground	N/A	N/A
6	SWO	Serial Wire Output	PB3	A10
7	GND	Ground	N/A	N/A
8	TDI	JTAG Test Data Input	PA15	A13
9	GND	Ground	N/A	N/A
10	nRESET	JTAG Test nReset	NRST	J1

3 Software

The CutiPy offers the ability to use different operating systems to meet different customer needs. There are available board support packages for the CutiPy from EMAC that uses MicroPython and FreeRTOS. EMAC provides a fully functional MicroPython and FreeRTOS BSPs loaded on the CutiPy at no additional charge. Middleware has been added to compliment the already available middleware supplied by the developers at STM to make these packages available for easy integration into user developed applications targeted for the CutiPy.

For more information on software support, please visit the EMAC Wiki CutiPy Section at:

http://wiki.emacinc.com/wiki/Cutipy_Getting_Started

3.1 MicroPython

MicroPython is one of the packages that is offered for the CutiPy. It is an implementation of the Python 3 programming language optimized to run on microcontrollers in a constrained environment. The CutiPy can be preloaded with the MicroPython BSP at no charge.

For more information on MicroPython support, please visit the EMAC Wiki MicroPython Section at:

http://wiki.emacinc.com/wiki/Micropython

3.1.1 Mu IDE

EMAC utilizes the Mu IDE for the MicroPython development which is a simple Python editor that is quick and easy to learn. The minimal setup makes it easy to use so the user can jump right in to developing. It has a built in REPL (Read Evaluate Print Loop) console window for easy command input and debugging.

https://codewith.mu/en/

3.2 FreeRTOS

FreeRTOS is one of the many packages available for application development that can be included when using the STM32CubeIDE code generation tool. Optionally, the FreeRTOS package can be downloaded from https://www.freertos.org/. FreeRTOS has their own board support package for many of the available STM32 development platforms. When using the STM32CubeIDE code generator, a wrapper is provided for most of the FreeRTOS functionality to make development even more simple.

https://www.freertos.org/FreeRTOS-Plus/BSP_Solutions/ST/index.html

Amazon provides Amazon Web Services (AWS) cloud support for FreeRTOS. For further information go to: https://aws.amazon.com/freertos/getting-started/

3.2.1 STM32CubeIDE

EMAC utilizes the STM32CubeIDE all-in-one multi OS development tool which is an advanced C/C++ development platform with IP configuration, code generation, code compilation, and debug features for STM32 microcontrollers. The ST32CubeIDE integrates all STM32CubeMX functionalities to offer an all-in-one tool experience while saving installation and development time. The CutiPy board support package includes the STM32CubeIDE project that can be used to tailor the FreeRTOS build to the customer's exact needs.

https://www.st.com/en/development-tools/stm32cubeide.html

3.2.1.1 STM32CubeMX

The STM32CubeMX is used to configure the processor for the CutiPy board. This tool allows for the configuration of all processor pins as well as adding, removing, and configuring middleware packages that are available through the STM32CubeMX code generation tool. One of the benefits of using the code generation tool is that it will automatically pull in and use the most current software updates from the upstream repository. Some of the middleware packages that are available and are supported by the CutiPy are as follows;

- FatFS
- FreeRTOS
- LwIP
- MBEDTLS
- USB Device
- USB Host
- Redpine APIs

http://www.st.com/en/development-tools/stm32cubemx.html

3.2.1.2 STM32CubeMX Expansion Software

Additional software and expansion packs are available through STMicro for use with the STM32CubeMX code generation utility. Software expansion packs may need to be tailored specifically to work with the CutiPy board, however, they are a useful starting point for incorporating various sensors and peripherals to the CutiPy board.

http://www.st.com/en/ecosystems/stm32cube-expansion-software.html?querycriteria=productId=SC2005

4 Appendix A: Connector Pinouts

4.1 Main Battery Connector (CN1)

Pin	Signal
1	BATT_VOLT
2	GND

4.2 Serial Port UART2 (CN2)

Pin	DB9 Connector: Description
1	NC
2	RXD
3	TXD
4	NC
5	GND
6	NC
7	RTS
8	CTS
9	NC

4.3 I/O Header 1 (HDR1)

Pin#	Signal	MCU Pin Name	MCU Pin Number	Pin#	Signal	MCU Pin Name	MCU Pin Number
1	GND	N/A	N/A	2	GND	N/A	N/A
3	3P3_VCC	N/A	N/A	4	5V_JACK	N/A	N/A
5	12V_VIN	N/A	N/A	6	V_HIDRV	N/A	N/A
7	TIM1_ETR	PE7	R8	8	TIM1_CH1N	PE8	P8
9	TIM1_CH1	PE9	P9	10	TIM1_CH2N	PE10	R9
11	TIM1_CH2	PE11	P10	12	TIM1_CH3N	PE12	R10
13	TIM1_CH3	PE13	N11	14	TIM1_CH4	PE14	P11
15	TIM1_BKIN	PE15	R11	16	TIM2_CH3	PB10	R12
17	TIM2_CH4	PB11	R13	18	TIM3_CH2	PC7	G15
19	TIM4_CH2	PD13	M15	20	TIM4_ETR	PE0	A4
21	TIM4_CH3	PD14	M14	22	GND	N/A	N/A
23	TIM5_CH1	PH10	L13	24	TIM4_CH4	PD15	L14
25	TIM5_CH3	PH12	K12	26	TIM5_CH2	PH11	L12
27	TIM8_CH3N	PH15	D13	28	TIM8_CH2N	PH14	E13
29	TIM8_CH1	PI5	C4	30	TIM8_BKIN	PI4	D4
31	TIM8_CH3	PI7	C2	32	TIM8_CH2	PI6	C3
33*	PG1	PG1	M7	34*	PG0	PG0	N7

35*	PG3	PG3	K15	**36***	PG2	PG2	L15
37*	PG5	PG5	K13	**38***	PG4	PG4	K14
39*	PG10	PG10	B10	**40***	PG6	PG6	J15
41	PF12	PF12	P6	**42**	PF13	PF13	N6
43	PF14	PF14	R7	**44**	PF15	PF15	P7
45	CAN2_RX	PB12	P12	**46**	CAN2_TX	PB13	P13
47	USB_HS_D_N	PA11	R14	**48**	USB_HS_D_P	PA12	R15
49	GND	N/A	N/A	**50**	GND	N/A	N/A

**High drive outputs. An MCU signal feeds the high drive open collector darlington driver, which in turn outputs to the indicated header Pin#*

4.4 I/O Header 2 (HDR2)

Pin#	Signal	MCU Pin Name	MCU Pin Number	Pin#	Signal	MCU Pin Name	MCU Pin Number
1	GND	N/A	N/A	**2**	GND	N/A	N/A
3	3P3_VIN	N/A	N/A	**4**	5V_JACK	N/A	N/A
5	12V_IN	N/A	N/A	**6**	NC	N/A	N/A
7	ADC123_IN3	PA3	R2	**8**	ADC3_IN4	PF6	K2
9	ADC3_IN5	PF7	K1	**10**	ADC3_IN6	PF8	L3
11	ADC3_IN7	PF9	L2	**12**	ADC3_IN8	PF10	L1
13	ADC3_IN9	PF3	J2	**14**	ADC123_IN10	PC0	M2
15	ADC123_IN12	PC2	M4	**16**	ADC123_IN13	PC3	M5
17	ADC12_IN6	PA6	P3	**18**	ADC12_IN8	PB0	R5
19	ADC12_IN9	PB1	R4	**20**	DAC_OUT1	PA4	N4
21	DAC_OUT2	PA5	P4	**22**	GND	N/A	N/A
23	AVDD	N/A	R1	**24**	I2C2_SCL	PH4	H4
25	I2C2_SDA	PH5	J4	**26**	I2C2_SMBA	PH6	M11
27	I2C3_SCL	PH7	N12	**28**	I2C3_SDA	PH8	M12
29	I2C3_SMBA	PH9	M13	**30**	SPI-1_SCK	PB3	A10
31	SPI-1_MISO	PB4	A9	**32**	SPI-1_MOSI	PB5	A6
33	SPI-1_NSS	PA15	A13	**34**	PI11	PI11	E4
35	PI10	PI10	E3	**36**	PH2	PH2	F4
37	PH3	PH3	G4	**38**	USART1_TX	PB6	B6
39	USART1_RX	PB7	B5	**40**	USART6_CK	PG7	J14
41	MDM_TXD_3V	PC6	H15	**42**	MDM_RXD_3V	PG9	C10
43	MDM_CTS_3V	PG15	B7	**44**	MDM_RTS_3V	PG8	H14
45	PH13	PH13	E12	**46**	PI9	PI9	D3
47	WKUP	PA0	N3	**48**	RTC_TAMP/TS	PI8	D2
49	GND	N/A	N/A	**50**	GND	N/A	N/A

4.5 Vin Power Header (HDR3)

Pin	Signal
1	+Vin ~12V
2	GND
3	GND
4	+Vin 5V

4.6 CAN 2.0 Port (HDR4)

Pin	Signal
1	CAN1_HI
2	CAN1_LO
3	GND

4.7 Serial Port UART3 (HDR5)

Pin	10-Pin Header: Description (RS-232)	10-Pin Header Description (RS-485/422)
1	NC	TX-
2	NC	NC
3	RXD	TX+
4	RTS	NC
5	TXD	RX+
6	CTS	NC
7	NC	RX-
8	NC	NC
9	GND	GND
10	NC	NC

4.8 JTAG/SWD Header (HDR6)

Pin#	Signal	Description	MCU Pin Name	MCU Pin Number
1	3P3_VCC	3.3 Volts	N/A	N/A
2	SWDIO	Serial Wire Data Input/Output	PA13	A15
3	GND	Ground	N/A	N/A
4	SWDCLK	Serial Wire Clock	PA14	A14
5	GND	Ground	N/A	N/A
6	SWO	Serial Wire Output	PB3	A10
7	GND	Ground	N/A	N/A
8	TDI	JTAG Test Data Input	PA15	A13
9	GND	Ground	N/A	N/A
10	nRESET	JTAG Test nReset	NRST	J1

4.9 USB Micro Port (JK1)

Pin	Signal	MCU Pin Number	MCU Pin Name
1	USB_DCIN	N/A	N/A
2	USB_FS_D_N	C15	PA11
3	USB_FS_D_P	B15	PA12
4	NC	N/A	N/A
5	GND	N/A	N/A
6	FRAME_GND	N/A	N/A
7	FRAME_GND	N/A	N/A

4.10 MicroSD Card Socket (SOK1)

Pin	Signal	MCU Pin Number	MCU Pin Name
1	SDIO_D2	B14	PC10
2	SDIO_D3	B13	PC11
3	SDIO_CMD	D12	PD2
4	3P3_VCC	N/A	N/A
5	SDIO_CLK	A12	PC12
6	GND	N/A	N/A
7	SDIO_D0	G14	PC8
8	SDIO_D1	F14	PC9
9	CARD_DET	D1	PC13
10	GND	N/A	N/A

5 Appendix B: Microcontroller Unit Pinout

5.1 STM32F407IGH6 – MCU Pinout (U3)

MCU Pin#	MCU Pin Name	Signal	MCU Pin#	MCU Pin Name	Signal
D6	BOOT0	BOOT0	**M3**	PC1	LCD_Csn
L4	BYPASS_REG	Strapped to GND	**B14**	PC10	SDIO_D2
J1	NRST	nRESET	**B13**	PC11	SDIO_D3
N3	PA0/WKUP (PA0)	WKUP	**A12**	PC12	SDIO_CLK
N2	PA1	BATT_SENV	**D1**	PC13	CARD_DET
D15	PA10	CAN1_SBY	**E1**	PC14/OSC32_IN (PC14)	E1 (Ext_OSC32)
C15	PA11	USB_FS_D_N	**F1**	PC15/OSC32_OUT (PC15)	F1 (Ext_OSC32)
B15	PA12	USB_FS_D_P	**M4**	PC2	ADC123_IN12
A15	PA13 (JTMS-SWDIO)	SWDIO	**M5**	PC3	ADC123_IN13
A14	PA14 (JTCK/SWCLK)	SWDCLK	**N5**	PC4	LCD_A0
A13	PA15 (JTDI)	TDI/SPI-1_NSS	**P5**	PC5	LCD_RSTn
P2	PA2	CHARGINGn	**H15**	PC6	MDM_TXD_3V
R2	PA3	ADC123_IN3	**G15**	PC7	TIM3_CH2
N4	PA4	DAC_OUT1	**G14**	PC8	SDIO_D0
P4	PA5	DAC_OUT2	**F14**	PC9	SDIO_D1
P3	PA6	ADC12_IN6	**B12**	PD0	CAN1_RX
R3	PA7	USR_PB1	**C12**	PD1	CAN1_TX
F15	PA8	USR_PB2	**N15**	PD10	PB4_LEDKn
E15	PA9	CAN1_SDN	**N14**	PD11	COMB_CTS
R5	PB0	ADC12_IN8	**N13**	PD12	COMB_RTS
R4	PB1	ADC12_IN9	**M15**	PD13	TIM4_CH2
R12	PB10	TIM2_CH3	**M14**	PD14	TIM4_CH3
R13	PB11	TIM2_CH4	**L14**	PD15	TIM4_CH4
P12	PB12	CAN2_RX	**D12**	PD2	SDIO_CMD
P13	PB13	CAN2_TX	**D11**	PD3	COMA_CTS
R14	PB14	USB_HS_D_N	**D10**	PD4	COMA_RTS
R15	PB15	USB_HS_D_P	**C11**	PD5	COMA_TXD
M6	PB2/BOOT1 (PB2)	BOOT1	**B11**	PD6	COMA_RXD
A10	PB3 (JTDO/TRACESWO)	SWO/SPI_SCK	**A11**	PD7	no connect
A9	PB4 (NJTRST)	SPI-1_MISO	**P15**	PD8	COMB_TXD
A6	PB5	SPI-1_MOSI	**P14**	PD9	COMB_RXD
B6	PB6	USART1_TX	**C6**	PDR_ON	3P3_VCC
B5	PB7	USART1_RX	**A4**	PE0	TIM4_ETR
A5	PB8	LCD_BKLEN	**A3**	PE1	RS232_A_SHUTDN#
B4	PB9	RADIO_ULP_WAKEUP	**R9**	PE10	TIM1_CH2N
M2	PC0	ADC123_IN10	**P10**	PE11	TIM1_CH2

STM32F407IGH6 – MCU (U3) (cont.)

MCU Pin#	MCU Pin Name	Signal	MCU Pin#	MCU Pin Name	Signal
R10	PE12	TIM1_CH3N	**L15**	PG2	PG2
N11	PE13	TIM1_CH3	**K15**	PG3	PG3
P11	PE14	TIM1_CH4	**K14**	PG4	PG4
R11	PE15	TIM1_BKIN	**K13**	PG5	PG5
A2	PE2	RS232_B_SHUTDN#	**J15**	PG6	PG6
A1	PE3	RADIO_RSTn	**J14**	PG7	USART6_CK
B1	PE4	RADIO_IRQ	**H14**	PG8	MDM_RTS_3V
B2	PE5	RADIO_PS_SLP_0	**C10**	PG9	MDM_RXD_3V
B3	PE6	RADIO_LP_WAKEUP	**G1**	PH0/OSC_IN (PH0)	G1 (Ext_OSC)
R8	PE7	TIM1_ETR	**H1**	PH1/OSC_OUT (PH1)	H1 (Ext_OSC)
P8	PE8	TIM1_CH1N	**L13**	PH10	TIM5_CH1
P9	PE9	TIM1_CH1	**L12**	PH11	TIM5_CH2
E2	PF0	RS4XX_B_TERM	**K12**	PH12	TIM5_CH3
H3	PF1	no connect	**E12**	PH13	PH13
L1	PF10	ADC3_IN8	**E13**	PH14	TIM8_CH2N
R6	PF11	RS4XX_B_RXEN#	**D13**	PH15	TIM8_CH3N
P6	PF12	PF12	**F4**	PH2	PH2
N6	PF13	PF13	**G4**	PH3	PH3
R7	PF14	PF14	**H4**	PH4	I2C2_SCL
P7	PF15	PF15	**J4**	PH5	I2C2_SDA
H2	PF2	BATT_SENA	**M11**	PH6	I2C2_SMBA
J2	PF3	ADC3_IN9	**N12**	PH7	I2C3_SCL
J3	PF4	USR_PB3	**M12**	PH8	I2C3_SDA
K3	PF5	USR_PB4	**M13**	PH9	I2C3_SMBA
K2	PF6	ADC3_IN4	**E14**	PI0	RADIO_CSELn
K1	PF7	ADC3_IN5	**D14**	PI1	RADIO_SPCLK
L3	PF8	ADC3_IN6	**E3**	PI10	PI10
L2	PF9	ADC3_IN7	**E4**	PI11	PI11
N7	PG0	PG0	**C14**	PI2	RADIO_MISO
M7	PG1	PG1	**C13**	PI3	RADIO_MOSI
B10	PG10	PG10	**D4**	PI4	TIM8_BKIN
B9	PG11	PB1_LEDKn	**C4**	PI5	TIM8_CH1
B8	PG12	RADIO_WAKE_IRQ	**C3**	PI6	TIM8_CH2
A8	PG13	PB2_LEDKn	**C2**	PI7	TIM8_CH3
A7	PG14	PB3_LEDKn	**D2**	PI8	RTC_TAMP/TS
B7	PG15	MDM_CTS_3V	**D3**	PI9	PI9

STM32F407IGH6 – MCU (U3) (cont.)

MCU Pin#	MCU Pin Name	Signal	MCU Pin#	MCU Pin Name	Signal
C1	VBAT	VSTDBY	**C5**	VDD	3P3_VCC
M10	VCAP_1	connected as specified in DS	**R1**	VDDA	AVDD/3P3_VCC
F13	VCAP_2	connected as specified in DS	**N1**	VREF-	GND
F3	VDD	3P3_VCC	**P1**	VREF+	3P3_VCC
G3	VDD	3P3_VCC	**F2**	VSS	GND
K4	VDD	3P3_VCC	**G2**	VSS	GND
N8	VDD	3P3_VCC	**M8**	VSS	GND
N9	VDD	3P3_VCC	**M9**	VSS	GND
N10	VDD	3P3_VCC	**H12**	VSS	GND
J12	VDD	3P3_VCC	**G12**	VSS	GND
J13	VDD	3P3_VCC	**F12**	VSS	GND
H13	VDD	3P3_VCC	**D9**	VSS	GND
G13	VDD	3P3_VCC	**D8**	VSS	GND
C9	VDD	3P3_VCC	**D7**	VSS	GND
C8	VDD	3P3_VCC	**D5**	VSS	GND
C7	VDD	3P3_VCC	**M1**	VSSA	GND

5.2 LDx Pinout

LED	MCU Number	MCU Pin Name	Signal
LD1	B9	PG11	PB1_LEDKn
LD2	A8	PG13	PB2_LEDKn
LD3	A7	PG14	PB3_LEDKn
LD4	N15	PD10	PB4_LEDKn

5.3 PBx Pinout

Push Button	MCU Number	MCU Pin Name	Signal
PB1	R3	PA7	USR_PB1
PB2	F15	PA8	USR_PB2
PB3	J3	PF4	USR_PB3
PB4	K3	PF5	USR_PB4
PB5	J1	NRST	nRESET

6 Appendix B: Jumper Settings

6.1 JB1 (Boot Mode 1 Selection)

Jumper	Position	Setting
Pins 1 & 2	RAM	Select RAM
Pins 2 & 3*	ROM	Select ROM

*Default Setting

6.2 JB2 (Boot Mode 0 Selection)

Jumper	Position	Setting
Pins 1 & 2	RXM	Select RXM
Pins 2 & 3*	FLS	Select FLS

*Default Setting

When JB2 is in the RXM position and JB1 is in the ROM position, the system will Boot from the ST bootloader. This bootloader will allow the programming of the Main Flash Memory by a number of various serial peripherals (USART, CAN, USB, I2C, SPI, etc.). Care must be exercised when using this jumper option as any program currently in Flash can be erased or corrupted. For additional information please reference the STM32F407IGH6 User Manual.

6.3 JB3 (RTC & RAM Retention Battery)

Jumper	Position	Setting
Pins 1 & 2	ON	Enable Battery
Pins 2 & 3*	OFF	Disable Battery

*Default Setting

6.4 JB4 (CAN Termination)

Jumper	Position	Setting
Pins 1 & 2*	TRM	Line Pulled HI
Pins 2 & 3	OPN	Line Pulled LO

*Default Setting

7 Appendix C: Dimensions