sysWORXX Pi-AM62x

Datasheet

Disclaimer:
All data, information and technical specifications contained in this document has been subjected to a thorough examination. The information in the document is current at the time of publication as long as nothing else is explicitly stated. However, no liability is given for the correctness, completeness and topicality of the[1] contents.

Revision History

Version	Date	Changes
01	03.03.2025	Created
02	25.04.2025	Chapter 9 added

Contact	**Direct**	**Your Local Distributor**
Address:	SYS TEC electronic AG Am Windrad 2 D-08468 Heinsdorfergrund GERMANY	Please contact us for information about your local distributor. info@systec-electronic.com
Ordering Information:	+49 (0) 37 65 / 38 600-0 info@systec-electronic.com	
Technical Support:	+49 (0) 37 65 / 38 600-0 support@systec-electronic.com	
Fax:	+49 (0) 37 65 / 38 600 4100	
Web Site:	https://www.systec-electronic.com	

List of Abbreviations

ADC	Analog digital converter
AI	Analog Input
AO	Analog Output
BSP	Board Support Package
CAN	Controller Area Network (according to ISO 11898-1:2003 and ISO 11898-2:2003)
CPU	Central Processing Unit
EEPROM	Electrically Erasable Programmable Read-Only Memory
eMMC	embedded Multimedia card
EN	European Norm
ES1	Electrical Source Class 1
ETH	Ethernet
GB	Giga Byte (1024 x 1MB)
Gb	Giga bit
GbE	Gigabit Ethernet
GND	Ground Reference potential
GPIO	General Purpose Input Output
HW	Hardware
IEC	International Electro technical Commission
I/O	Input/Output
I^2C	Inter-integrated circuit
I^2S	Inter-Integrated circuit sound
kB	Kilo Byte (1024 byte)
MAC	Media Access Controller (e.g. Ethernet controller)
MCAN	Modular Controller Area Network
MCASP	Multichannel Audio Serial Port
MB	Mega Byte (1024 x 1kB)
MDI	Media Dependent Interface
nc	not connected
OS	Operating System
OTG	On-The-Go (USB interface mode)
PCB	Printed Circuit Board
PMIC	Power Management Integrated Circuit
PS1	Electrical Power Source Class 1
PWM	Pulse Width Modulation
RAM	Random Access Memory
RGMII	Reduced Gigabit Media-Independent Interface
RTC	Real Time Clock
RX	Receive
SAI	Synchronous Audio Interface
SCK	Continuous Serial Clock
SD	Serial Data (I2S), Secure Digital (SD-Card)
SGMII	Serial Gigabit Media-Independent Interface
SPI	Serial Peripheral Interface
TX	Transmit
UART	Universal Asynchronous Receiver Transmitter
w/o	without
WS	Word Select (I2S)

Contents

List of Tables

List of Figures

Reference documents

/1/ TI: AM62x Reference Manual (https://www.ti.com/product/de-de/AM623#tech-docs)
/2/ TI: AM62x Data Sheet (https://www.ti.com/product/de-de/AM623#tech-docs)
/3/ TI: SysConfig 1.20.0+3587 – PINMUX Tool (https://www.ti.com/tool/SYSCONFIG)
/4/ TI: TPS65219 Data Sheet (https://www.ti.com/product/de-de/TPS65219)
/5/ TI: TPS65219 Reference Manual (https://www.ti.com/product/de-de/TPS65219#tech-docs)
/6/ TE Connectivity: DIMM Socket (https://www.te.com/en/product-2013310-1.html)
/7/ SYS TEC: User Manual. L-2934e
/8/ SYS TEC: Datasheet Addendum sysWORXX AM62x device series. L-2960e
/9/ SYS TEC: Quick Start Guide. L-2935e

1 Introduction

The sysWORXX Pi-AM62x, designed as a Single Board Computer (SBC), is the evaluation platform for our sysWORXX ECUcore-AM62x. It offers you the opportunity to test the interaction between hardware and software and to validate the planned system concept in advance.

In contrast to the Raspberry Pi, there are **two independent** Ethernet interfaces, **eMMC and** SD card interface as storage for operating system and application, various interfaces for industrial applications on the IO connector, for example a CAN-Bus interface. That's why we say it's the Raspberry Pi for industrial use cases.

This document includes all device specific information. To get more general information on how to operate the device, please refer to the User Manual of our sysWORXX device series (see Table 1).

Note: Before using the product, the document must be read and understood in full.

2 Overview

Table 1 lists up all relevant manuals for the device.

Table 1: Overview of relevant manuals

Information about…	In which document?
Description of the device	In this document, chapter 3
Information for getting started	See in /9/
What is this part intended for?	In this document, chapter 3.3
Description of how the part and version can be identified	In this document, chapter 3.2
Description of the interfaces (pin assignment, electrical properties)	In this document, chapter 4 and 5
Information on installation, removal and disposal	In this document, chapter 8
Description of the sysWORXX-IO Channel mapping	In this document, chapter 9
Description of conformity	In this document, chapter 11
Information about usage of the device (configuration, administration, usage of different applications like Codesys, NodeRED, InfluxDB, firmware update, et cetera)	See in /9/ and /7/
Description the differences between the sysWORX Pi-AM62x and the ECUcore-AM62x	See in /8/

3 Product Description / Features

3.1 Ordering Information

Part number	Product name	Features[2]	Notes
24011100	sysWORXX Pi-AM62x	• TI Sitara™ AM6254 Quad-Core Cortex™-A53 @ 1.4 GHz • 2 GiB DDR4 RAM • 32 GiB eMMC • 2x ETH 100/1000Base-T • 1x WLAN/Bluetooth device • 2x USB 2.0 • 1x microSD-Card • 1x HDMI • 1x USB/UART console • Raspberry compatible 40pin IO header	Web-Link

Table 2: Ordering information

Figure 1: sysWORXX Pi-AM62x

[2] Please contact us for other device configurations.

3.2 Identification and Labeling

Figure 2 shows the position of stickers for MAC address and serial number. The MAC address for the WLAN module is lasered onto the module housing.

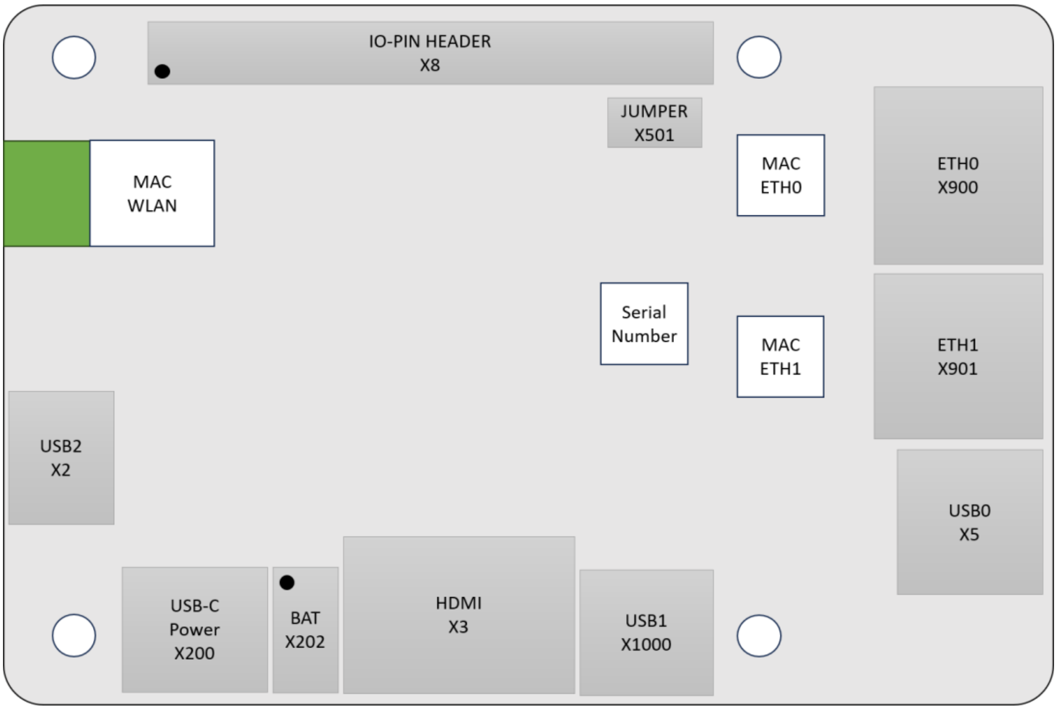

Figure 2: Position of MAC-Address and Serial Number labels

3.3 Intended Use

The sysWORXX Pi-AM62x is the evaluation platform for our sysWORXX ECUcore-AM62x. It offers you the opportunity to test the interaction between hardware and software and to validate the planned system concept in advance. The product is intended exclusively for customer- and application-specific testing by professionals exclusively in research and development facilities for such purposes.

3.4 Features

Table 3: Features sysWORXX Pi-AM62x

Core

CPU	TI Sitara™ AM6254 Quad-Core Cortex™-A53 @ 1.4 GHz and with 3D Graphics Processing Unit (GPU)
Real-time Co-Pro	Cortex™-M4 @ 400 MHz
RAM	2 GiB DDR4
eMMC	32 GiB
SD-Card	Card types: SD, SDHC, SDXC, SDUC Bus speed: DS (SDR12), HS (SDR25)
RTC	on-board (optional external buffering with battery or capacitor)
Boot Mode	SD Card / eMMC (selectable per jumper)

Connectivity

ETH	2 (100BASE-TX, 1000Base-T)
USB Host	1 USB-A (USB 2.0 high-speed)
USB OTG	1 USB-C (USB 2.0 high-speed)
Display	1 HDMI (1080p)
WLAN	802.11 a/b/g/n/ac
Bluetooth	5.2
IO-pin header	max. 1x SPI max. 2x I2C max. 1x UART max. 2x CAN max. 2x PWM max 1x MCASP max. 28 GPIO (interruptable)
Linux console	Serial (on-board via USB/UART bridge and USB-C connector)

Power Supply

Main Power Supply	5V
Power Consumption	≤ 15 W

3.5 Block Diagram

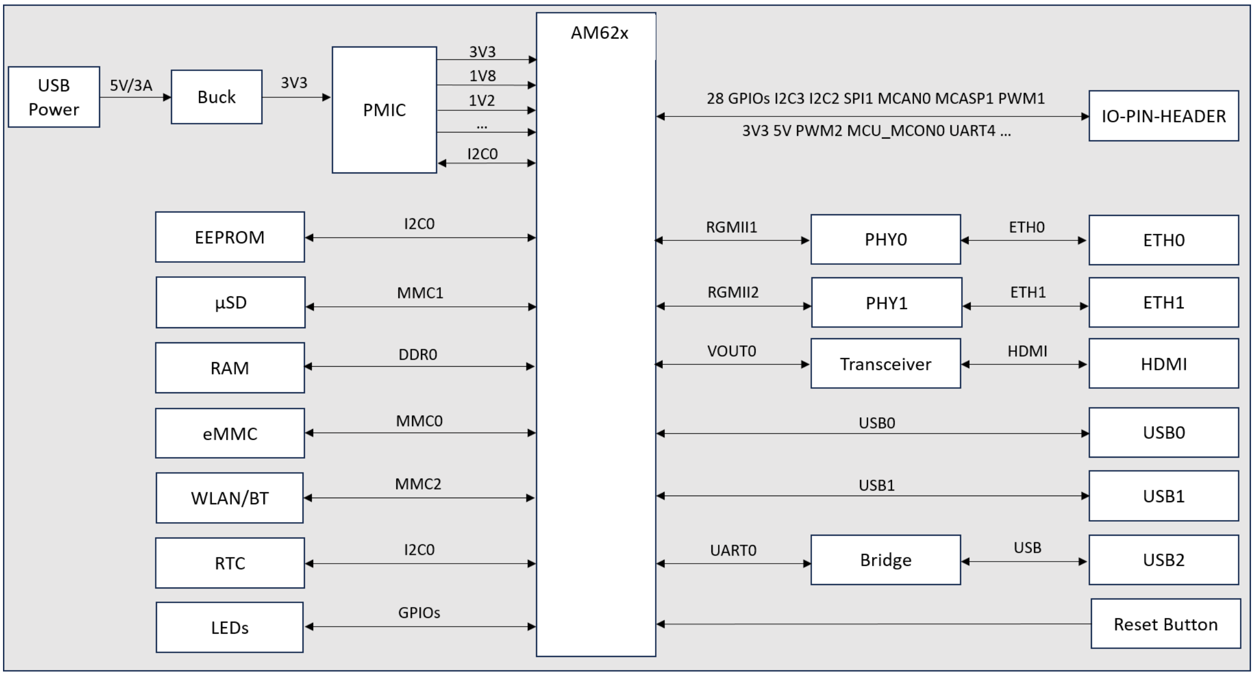

Figure 3: Block Diagram of sysWORXX Pi-AM62x

3.6 Hardware Comparison sysWORXX Pi-AM62x and ECUcore-AM62x

The sysWORXX Pi-AM62x board is designed as an evaluation board for the ECUcore-AM62x module. The sysWORXX Pi-AM62x offers the opportunity to test the interaction between hardware and software in advance and to validate a planned system concept for the ECUcore-AM62x carrier board in advance.

Therefore, the hardware on both, the sysWORXX Pi-AM62x and the ECUcore-AM62x, are almost identical. Similarities and differences in the hardware design are presented in /8/.

4 Functional Overview

4.1 Sockets and Connectors

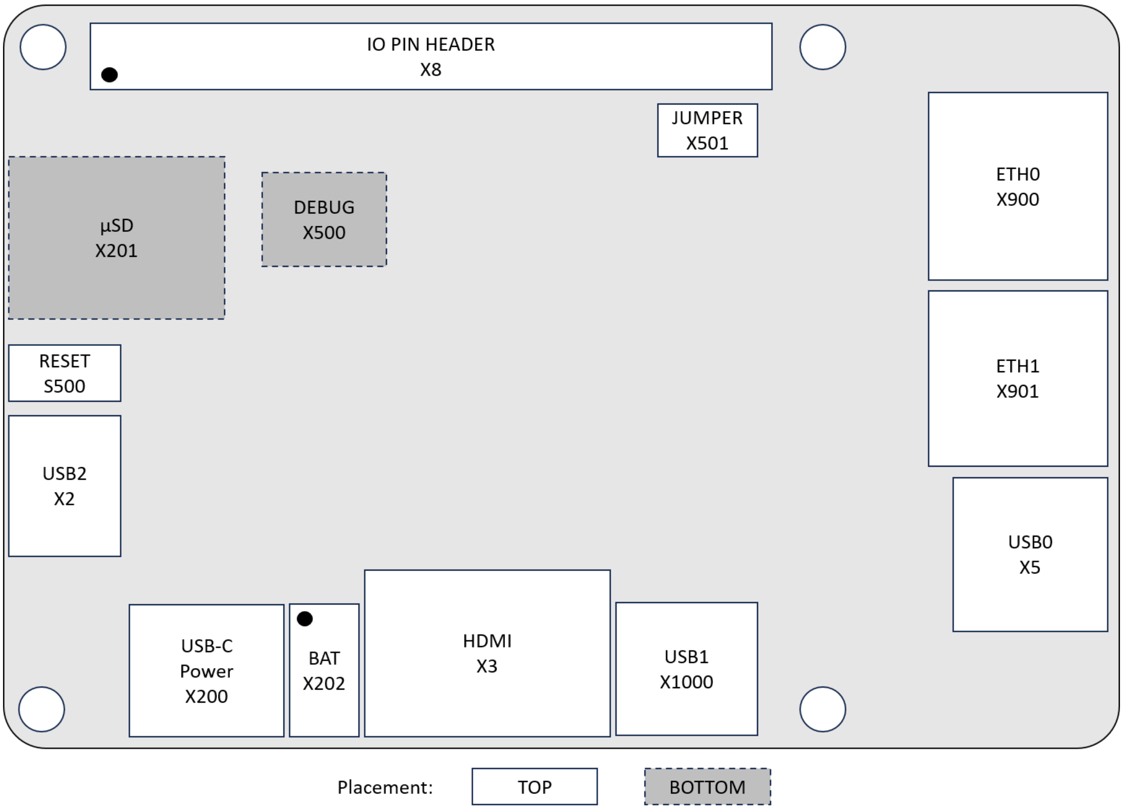

Figure 4: Connector arrangement of sysWORXX-Pi-AM62x

Table 4: Connector description

Connector	Designator	Interface	Connector type
ETH0	X900	100BASE-TX, 1000BASE-T	RJ45
ETH1	X901	100BASE-TX, 1000BASE-T	RJ45
USB0	X5	USB-Host 2.0	USB Type A
USB1	X1000	USB-OTG 2.0	USB Type C
USB2	X2	USB-Device	USB Type C
HDMI	X3	Display interface	HDMI Type A
USB-C-Power	X200	USB-C Power	USB Type C
RESET-Button	S500	Button to RESET the device	Push button
µSD	X201	Connector for microSD-Cards	Push-Push-Connector
Debug	X500	Optional connector for JTAG	Not available
IO Pin Header	X8	Pin header for application specific HATs	Pin header 2x20 pins 2.54mm pitch
Jumper	X501	Boot-Jumper	2 pins, 2.54mm pitch
BAT	X202	Interface for RTC voltage buffering	3 connection holes

4.2 Power supply

The sysWORXX Pi-AM62x is powered by a single power rail. The power supply can be via the USB-C connector or via the 5V0 pin on the IO connector.

Note: The sysWORXX Pi-AM62x may only be operated with a Class 1 electrical energy source (ES1 according to DIN EN 62368-1) and a Class 1 electrical power source (PS1 according to DIN EN 62368-1).

The sysWORXX Pi-AM62x is equipped with the power management IC (PMIC). The CPU communicates via I2C0 to the PMIC. The I2C device address is 0x30.

4.3 Boot Jumper

Using the boot jumper, you can choose between an SD card (jumper not inserted) and an eMMC (jumper inserted) as the boot device. This requires that a valid image is present on the boot device. After selecting the boot device, the RESET button must be pressed or a power cycle must be performed.

4.4 Reset button

The sysWORXX Pi-AM62x is equipped with a reset button S500, which resets the AM62x and on-board peripherals when pressed.

4.5 IO-Pin header

The sysWORXX Pi-AM62x IO pin header is compatible with the Raspberry Pi 40-pin header. Table 5 lists the connected GPIO signals of the CPU (GPIOs – general-purpose input/output) for the individual pins of the IO pin header. The signal names correspond to the designations used in the AM62x manufactures data sheet (/2/) for *MUX MODE 7*. All signals are 3.3 V signals.

In Linux, the desired function of a pin is selected via the Device Tree. To provide different functionalities for the IO pins in a Linux image, the Device Tree overlay mechanism is used. For some HATs, corresponding Device Trees already exist and are stored in the Linux image. The desired Device Tree for the respective HAT is then applied by changing the configuration. In this case, a new image does not need to be created.

The individual hardware-specific IO-pin configuration and the corresponding Device Tree are described in the HAT documentation. Regardless of a specific HAT, there is a default variant for a Device Tree (see Table 5).

Table 5: IO-Pin Header (X8) with AM62x signal names and default IO configuration

Default State[3]	Function and IO-Label	Pin name	Pin	Pin	Pin name	Function and IO-Label	Default State[3]
-	**3V3 power**	**3V3 power**	**1**	**2**	**5V power**	**5V power**	**-**
-	I2C3_SDA	GPIO1_23	**3**	**4**	**5V power**	**5V power**	**-**
-	I2C3_SCL	GPIO1_22	**5**	**6**	**Ground**	**Ground**	**-**
PD	DI_0	GPIO0_0	**7**	**8**	GPIO0_72	UART4_TX	High
-	**Ground**	**Ground**	**9**	**10**	GPIO0_71	UART4_RX	PU
low	DO_0	GPIO0_1	**11**	**12**	GPIO0_35	DO_1	-
PD	DI_1	GPIO0_2	**13**	**14**	**Ground**	**Ground**	**-**
low	DO_2	GPIO0_3	**15**	**16**	MCU_ GPIO0_13	DO_3	Low
-	**3V3 power**	**3V3 power**	**17**	**18**	MCU_ GPIO0_14	DI_2	PD
-	SPI1_MOSI	GPIO0_9	**19**	**20**	**Ground**	**Ground**	**-**
-	SPI1_MISO	GPIO0_10	**21**	**22**	GPIO0_40	DO_4	Low
-	SPI1_CLK	GPIO0_8	**23**	**24**	GPIO0_7	SPI1_CS0	-
-	**Ground**	**Ground**	**25**	**26**	GPIO0_13	DI_3	PD
PD	DI_4	GPIO0_44	**27**	**28**	GPIO0_43	DI_5	PD
PD	DI_6	GPIO1_24	**29**	**30**	**Ground**	**Ground**	**-**
PD	DI_7	GPIO1_25	**31**	**32**	GPIO1_9	DI_8	PD
PD	DI_9	GPIO1_28	**33**	**34**	**Ground**	**Ground**	**-**
PD	DI_10	GPIO0_37	**35**	**36**	GPIO0_41	DO_5	Low
low	DO_6	GPIO0_4	**37**	**38**	GPIO0_34	DI_11	PD
-	**Ground**	**Ground**	**39**	**40**	GPIO0_33	DI_12	PD

Default State:
PD: Indicates the presence of an internal pulldown resistor of CPU, enabled via software
PU: Indicates the presence of an internal pullup resistor of CPU, enabled via software
Low: The output buffer is active and drives the state low.
High: The output buffer is active and drives the state high.

[3] The default Pull State is configured by the Device Tree.

Function:
DI: GPIO is config as digital input
DO: GPIO is config as push pull output.
I2C3: GPIO is config as I2C bus master.
SPI1: GPIO is config as SPI bus master.
UART4: GPIO is config as UART.

If a special configuration of the IOs is required, a specific device tree can be created for this purpose. The pins can be configured as either general-purpose input (DI), general-purpose output (DO) or as a special alternate function. Table 6 shows which alternative functions are generally possible. The functions and properties of the peripherals are described in the AM62x manufacturer's data sheet (/2/) and reference manual (/1/). Additional, TI provide a pin mux tool (/3/).

Table 6: IO-Pin Header (X8) description of alternate functions

MUX MODE7	MUX MODE0	MUX MODE1	MUX MODE02	MUX MODE3	MUX MODE4	MUX MODE5	MUX MODE6	MUX MODE8
GPIO0_0	OSPI0_CLK	-	-	-	-	-	-	-
GPIO0_1	OSPI0_LBCLKO	-	-	-	-	UART5_RTSn	-	-
GPIO0_2	OSPI0_DQS	-	-	-	-	UART5_CTSn	-	-
GPIO0_3	OSPI0_D0	-	-	-	-	-	-	-
GPIO0_4	OSPI0_D1	-	-	-	-	-	-	-
GPIO0_7	-	**SPI1_CS0**	MCASP1_AXR1	-	-	-	-	-
GPIO0_8	-	**SPI1_CLK**	MCASP1_AXR0	-	-	-	-	-
GPIO0_9	-	**SPI1_D0**	MCASP1_ACLKX	-	-	-	-	-
GPIO0_10	-	**SPI1_D1**	MCASP1_AFSX	-	-	-	-	-
GPIO0_13	OSPI0_CSn2	**SPI1_CS1**	OSPI0_RESET_OUT1	MCASP1_AFSR	MCASP1_AXR2	UART5_RXD	-	-
GPIO0_33	-	-	**MCASP1_AXR1**	-	PR0_PRU0_GPO10	PR0_PRU0_GPI10	-	-
GPIO0_34	-	-	**MCASP1_AXR0**	-	PR0_PRU0_GPO11	PR0_PRU0_GPI11	-	-
GPIO0_35	-	-	**MCASP1_ACLKX**	-	PR0_PRU0_GPO12	PR0_PRU0_GPI12	-	-
GPIO0_37	-	-	**MCASP1_AFSX**	-	PR0_PRU0_GPO14	PR0_PRU0_GPI14	-	-
GPIO0_40	-	PR0_ECAP0_IN_APWM_OUT	-	MCASP2_AXR13	PR0_PRU0_GPO16	PR0_PRU0_GPI16	-	
GPIO0_41	-	-	-	MCASP2_AXR14	PR0_PRU0_GPO17	PR0_PRU0_GPI17	-	-
GPIO0_43	-	**I2C2_SCL**	MCASP1_AXR4	*UART4_RXD*	PR0_PRU0_GPO19	PR0_PRU0_GPI19	-	MCASP1_AFSR
GPIO0_44	-	**I2C2_SDA**	-	*UART4_TXD*	MCASP1_AXR5	-	-	MCASP1_ACLKR
GPIO0_71	-	MCASP1_ACLKX	-	**UART4_RXD**	-	-	-	-
GPIO0_72	-	MCASP1_AFSX	-	**UART4_TXD**	-	-	-	-
MCU_GPIO0_13	**MCU_MCAN0_TX**	-	-	-	-	-	-	-
MCU_GPIO0_14	**MCU_MCAN0_RX**	MCU_TIMER_IO0	-	-	-	-	-	-
GPIO1_9	MCASP0_AXR1	-	ECAP1_IN_APWM_OUT	-	-	PR0_UART0_RXD	**EHRPWM1_A**	-
GPIO1_22	-	-	**I2C3_SCL**	UART2_RXD	TIMER_IO6	AUDIO_EXT_REFCLK0	PR0_ECAP0_SYNC_OUT	MCASP2_AFSX
GPIO1_23	-	-	**I2C3_SDA**	UART2_TXD	TIMER_IO7	AUDIO_EXT_REFCLK1	PR0_ECAP0_IN_APWM_OUT	MCASP2_ACLKX
GPIO1_24	**MCAN0_TX**	UART5_RXD	TIMER_IO2	SYNC2_OUT	-	-	PR0_UART0_RXD	MCASP2_AXR0
GPIO1_25	**MCAN0_RX**	UART5_TXD	TIMER_IO3	SYNC3_OUT	-	-	PR0_UART0_TXD	MCASP2_AXR1
GPIO1_28	-	-	TIMER_IO0	-	-	-	-	**EHRPWM2_A**

MUX MODE: The signals found in this column defines the pin multiplex pin function. The MUX MODE 7 is the default pin state after power-on RESET.
Please refer to AM62x manufacturer's data sheet and reference manual for description of functions and properties of the peripherals.

4.6 Ethernet

The sysWORXX Pi-AM62x is equipped with two independent Ethernet ports. A separate MAC address is labeled and stored for each port (see chapter 3.2). For The shield of the socket used is connected to GND potential of the board.

4.7 HDMI

The sysWORXX Pi-AM62x is equipped with a HDMI interface accessible via socket X3. A HDMI Transmitter (SII9022ACNU) translates the Video Output Data from the Display Parallel Interface (DPI) and MCASP0 data from the AM62x into the HDMI interface.

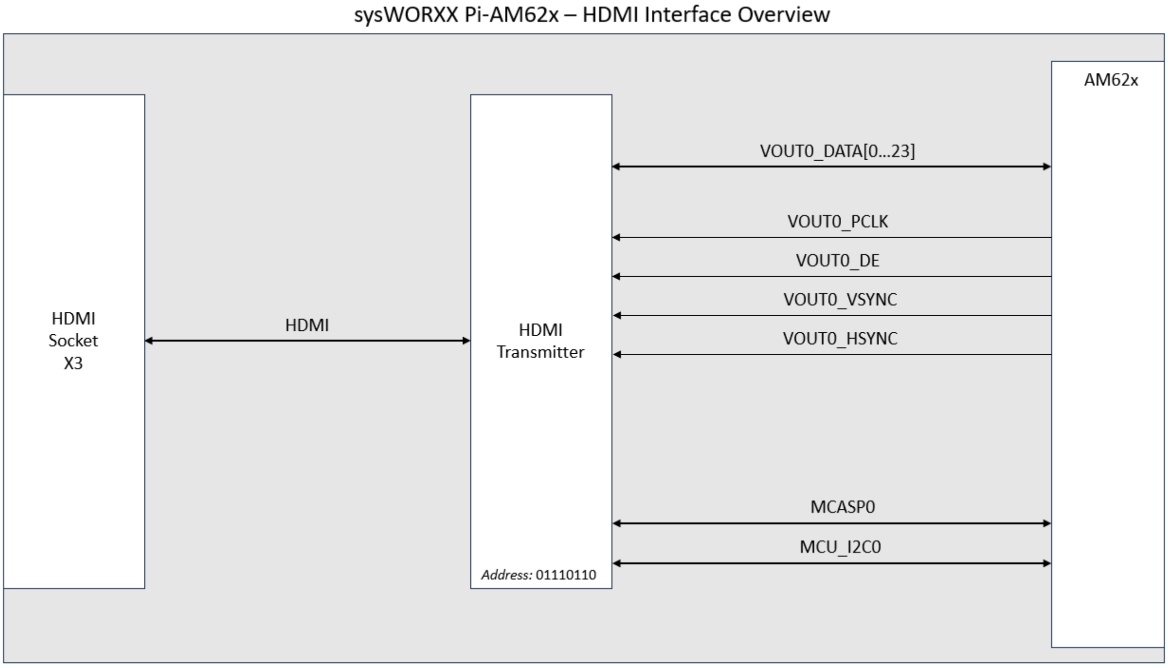

Figure 5: HDMI interface overview on sysWORXX Pi-AM62x

The HDMI Transmitter can be configured via MCU_I2C0 interface.

4.8 USB

The sysWORXX Pi-AM62x board provides several USB interfaces with different application purposes.

4.8.1 USB0 (USB Host) (Type-A socket)

The sysWORXX Pi-AM62x board provides a USB 2.0 Host via a USB Type A socket.

4.8.2 USB1 (USB OTG - Host/Device) (Type-C socket)

The sysWORXX Pi-AM62 provides a USB Host or Device connection. The corresponding socket is X1000 with an USB-C type socket. An overview of this interface is presented in Figure 6.

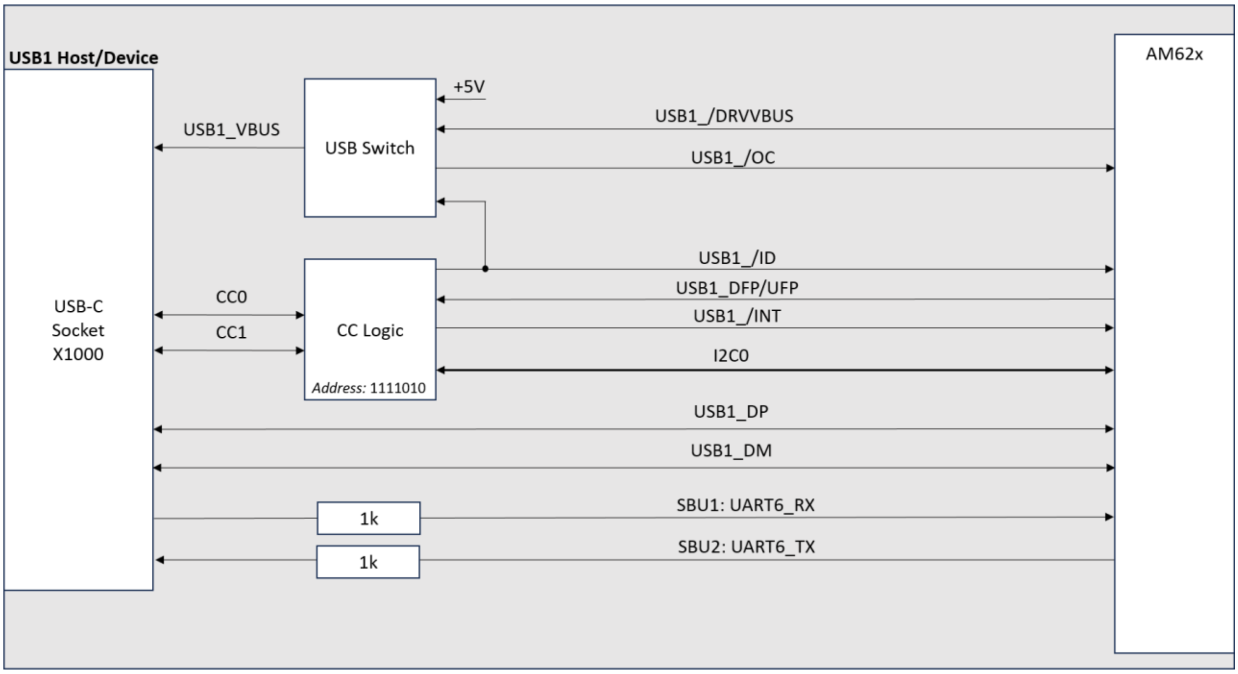

Figure 6: sysWORXX Pi-AM62x USB1 interface overview

The Pinout of the USB1 connector is shown in Table 7.

USB Switch
The USB switch enables the USB1 VBUS voltage in USB-Host mode. If an overcurrent is detected the VBUS voltage will be switched-off.

CC-Logic
In order to configure the USB1 interface in dual role (host or device mode), a configuration channel control logic is implemented. This control logic can work autonomously, or can be controlled through the I2C0 interface on the sysWORXX Pi-AM62x. By default, the USB1 interface is set to DRP-mode (dual role port)

SBU
The pins SBU1 and SBU2 of the USB-C socket X1000 are connected to AM62x UART signals according to Table 7.

Table 7: USB1 (OTG) socket signals pinout and connections

USB-C Socket Pin	USB-C Socket Signal name	Signal assignment	Notes
X1000-A8	SBU1	UART6_RX	1k-resistor in series
X1000-B8	SBU2	UART6_TX	1k-resistor in series
X1000-A6 X1000-B6	DP1	USB1_DP	
X1000-A7 X1000-B7	DN1	USB1_DN	
X1000-A5	CC1	CC1	
X1000-B5	CC2	CC2	
X1000-A4 X1000-B9 X1000-B4 X1000-A9	VBUS	5V	
X1000-A1 X1000-B12 X1000-B1 X1000-A12	GND	GND	

4.8.3 USB2 (Linux Console) (Type-C socket)

The USB interface provides a service connection for Linux console outputs. The service interface serves to administer the device. The USB data is translated into a UART0 signal of AM62x via the USB-bridge CP2102N-A01. The pinout of the USB2 socket is shown in Table 8.

Table 8: USB2 (Linux Console) socket signals pinout and connections

USB-C Socket Pin	USB-C Socket Signal name	Signal assignment
X2-A8	SBU1	-
X2-B8	SBU2	-
X2-A6 X2-B6	DP1	USB_DP (of USB-UART-Bridge)
X2-A7 X2-B7	DN1	USB_DN (of USB-UART-Bridge)
X2-A5	CC1	5.1k to GND
X2-B5	CC2	5.1k to GND
X2-A4 X2-B9 X2-B4 X2-A9	VBUS	5V
X2-A1 X2-B12 X2-B1 X2-A12	GND	GND

4.8.4 USB Power input (Type-C socket)

The USB-C connector X200 is a power input connector and serves the only purpose of powering the sysWORXX Pi-AM62x board. No data is transferred via this USB-C connector. Be sure to connect a power supply with sufficient power to the board. A 5V/15W power supply is the recommended minimum.

Table 9: USB Power Input socket signals pinout and connections

USB-C Socket Pin	USB-C Socket Signal name	Signal assignment
X2-A8	SBU1	-
X2-B8	SBU2	-
X2-A6 X2-B6	DP1	-
X2-A7 X2-B7	DN1	-
X2-A5	CC1	5.1k to GND
X2-B5	CC2	5.1k to GND
X2-A4 X2-B9 X2-B4 X2-A9	VBUS	5V
X2-A1 X2-B12 X2-B1 X2-A12	GND	GND

Note: If the module is supplied via the IO-Pin header, no supply may be connected here.

4.9 WLAN & Bluetooth

The sysWORXX Pi-AM62x is optionally equipped with a WLAN and Bluetooth module (Enzurio; Manuf.-No.: 453-00046R; Datasheet). The connection between the AM62x and the WiFi-module is shown in Table 10. The module is equipped with a PCB antenna.

Table 10: Connection between AM62x and WLAN module

AM62x	WLAN Modul		Notes
Function	Function	Pin	
MMC2_CLK	SDIO_CLK	15	10k resistor to 3.3V
MMC2_DAT0	SDIO_DATA0	9	10k resistor to 3.3V
MMC2_DAT1	SDIO_DATA1	10	10k resistor to 3.3V
MMC2_DAT2	SDIO_DATA2	13	10k resistor to 3.3V
MMC2_DAT3	SDIO_DATA3	11	10k resistor to 3.3V
MMC2_CMD	SDIO_CMD	12	10k resistor to 3.3V
MMC2_SDCD	WL_REG_ON	33	10k resistor to 3.3V
MMC2_SDWP	BT_REG_ON	32	10k resistor to 3.3V
MCU_UART0_RXD	BT_UART_TXD	43	-
MCU_UART0_TXD	BT_UART_RXD	45	-
MCU_UART0_/RTS	BT_UART_CTSN	44	-
MCU_UART0_/CTS	BT_UART_RTSN	46	-
WKUP_CLKOUT0	SUSCLK	34	32kHz clock

The WLAN and Bluetooth modules can be enabled or disabled according to the voltage levels on pins WL_REG_ON (WLAN enable) and BT_REG_ON (Bluetooth enable). Both can be disabled with a Low Level signal and enabled with a High Level signal. Note that internal 10k pull-ups on the module enable WLAN and Bluetooth by default.

4.10 I2C Interfaces

Several I2C interfaces are available on the sysWORXX Pi-AM62x board. An overview is presented in Figure 7.

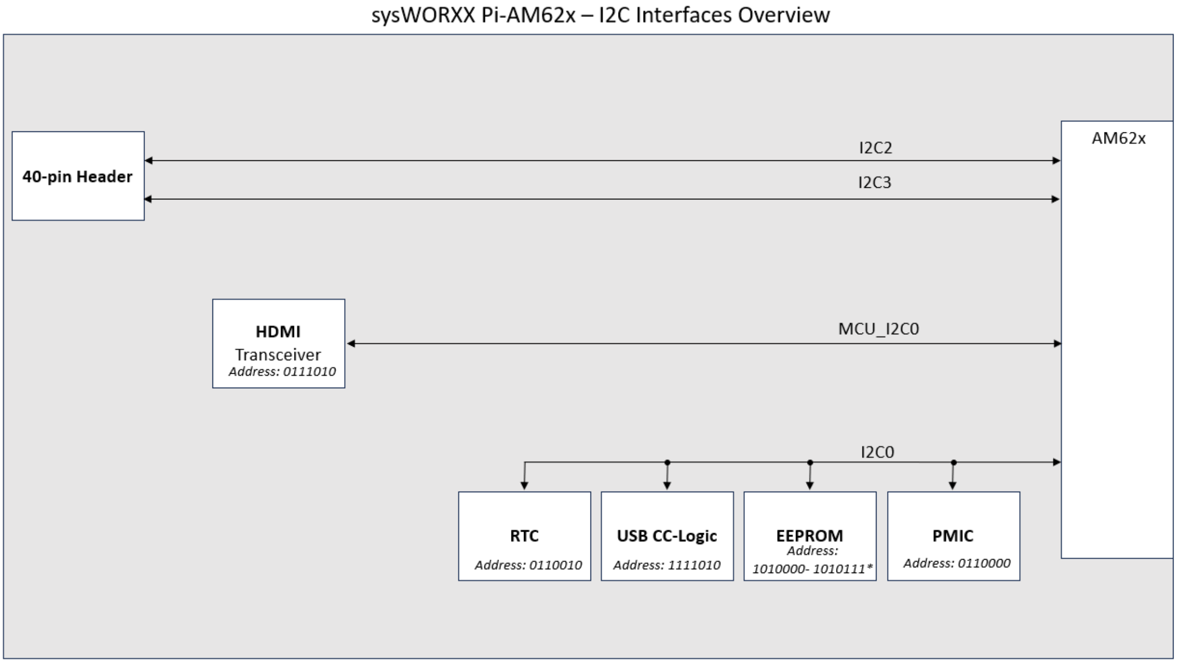

Figure 7: sysWORXX Pi-AM62x I2C interfaces overview

The I2C0 and MCU_I2C0 interfaces are only used internally on the sysWORXX Pi-AM62x board and are not routed to the IO-pin header or any other peripheral connector. Therefore, these I2C interfaces cannot be used by any peripheral application.

Two I2C interfaces are routed to the IO-pin header for user specific peripheral applications. Please note that the sysWORXX Pi-AM62x board does not provide pull-up resistors for the I2C2 and I2C3 interfaces. Therefore, additional pull-up resistors of at least 1kOhm must be considered when designing a peripheral application for the sysWORXX Pi-AM62x via these I2C interfaces.

4.11 Real-Time-Clock

The sysWORXX Pi-AM62x has an on-board Real-Time-Clock which can be accessed and configured via the I2C0 interface of the AM62x. The corresponding I2C address of the RTC is 0x32 (0110010).
The sysWORXX Pi-AM62x board does not contain a battery to back up the RTC in the event of a power failure. However, a suitable battery can be connected to connector X202, with the positive terminal connected to X202-2 and the negative terminal connected to X202-3. A capacitor can also be used instead of a battery. This can be charged via pin X202-1.

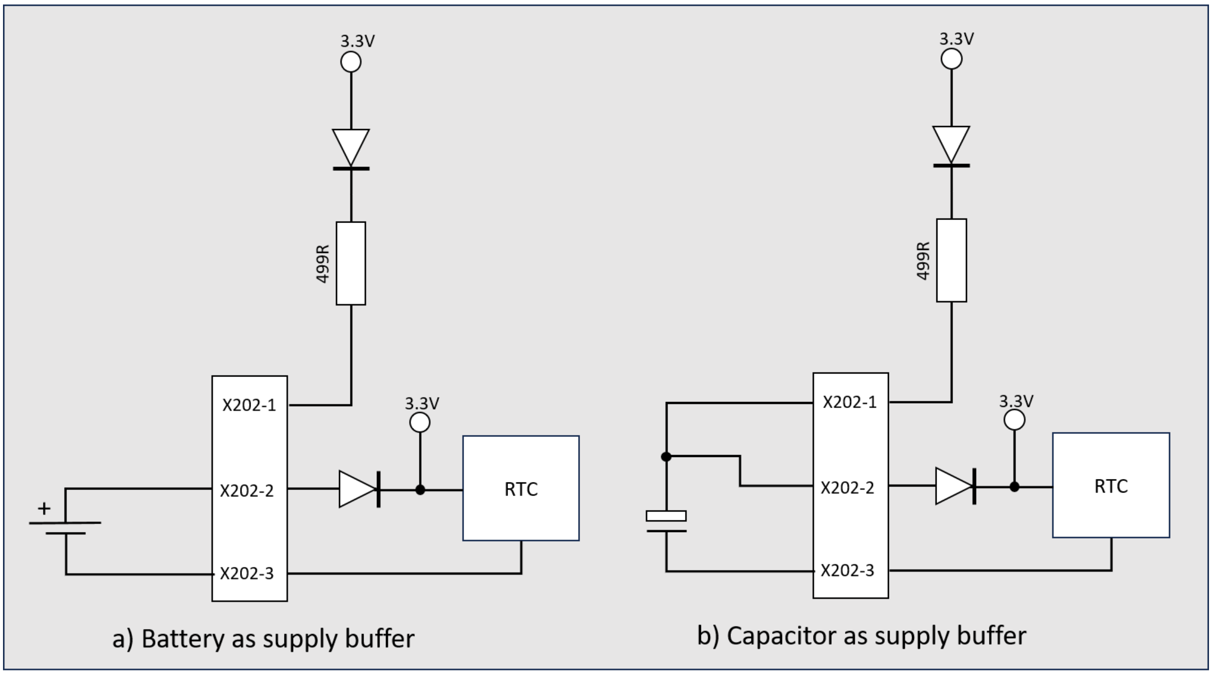

Figure 8: RTC power supply buffer concepts

4.12 Status-LEDs

The sysWORXX Pi-AM62x has an RGB status LED which can be controlled by the user via certain AM62x GPIO pins. All LEDs can be enabled by setting the according GPIO pin to a High Level.

Table 11: Connection between RGB-Status-LED and AM62x

Status LED Color	AM62x	Enable Level
Red	GPIO0_12	High
Green	GPIO0_14	
Blue	GPIO0_11	

4.13 Data Storage

4.13.1 microSD-Card

The sysWORXX Pi-AM62x has a microSD-card-holder at X201. The SD-Card can be used as a boot device. The SD card interface supports the following cards: SD, SDHC, SDXC, and SDUC. The maximum bus speed is SDR25. The UHS-I bus interface are not supported.

4.13.2 eMMC

The sysWORXX Pi-AM62x board has access to an on board embedded multi-media card (eMMC). The eMMC can be used as a boot device.

4.13.3 EEPROM

The sysWORXX Pi-AM62x has an on-board 16kBit EEPROM. The component is connected to the I2C0 bus. It has the following I2C addresses: from 0x50 to 0x57. The component is reserved for storing device-specific data and may not be modified by the user.

5 Electrical Specification and Environmental Conditions

Electrical Specification

Power Supply	
Input Voltage	4.8V … 5.25V
Input Current	3 A (maximum)
USB0, USB1	
Output voltage	5V
Current limit	500mA
I2C2, I2C3	
Power Supply	3.3V
Maximum Pull-up current at V_{OL}=0.4V	3mA
Bit Rate	100kBit/s, 400kBit/s
IO-pin header	Configured in 3.3V mode
Input Low Voltage	0 … 0.8V
Input High Voltage	2.0V … 3.6V
Input Hysteresis Voltage	≥ 150mV
Output Low Voltage	0.4V @ 5mA
Output High Voltage	2.4V @ 9mA
SD-Card	
Output voltage	3.3V (switched off by reset)
Output current	400mA
Bus speed rate	SDR12, SDR25
HDMI	
Output voltage	5V
Output current	≤ 55mA
Over current protection	PTC

Environmental conditions	
Operating height	≤ 2000m

Temperature range	
Operating	-40 °C/+70 °C (Function and load-dependent)
Storage	-55 °C/+85 °C
$T_{CPU\_junc\_max}$	85 °C

Humidity	
Operating	10 % …90 %
Storage	<95 %

Sample Time of libsysworxio	
t_{Temp_sample}[4]	2000ms
t_{ADC_sample}	-

[4] The *libsysworxx-io* provides the data via a defined interface and exchanges it cyclically with the peripherals. For more information see chapter 9

6 Thermal Design considerations

The thermal management requirements of the CPU are based on the required operating conditions. If performance requirements are low, an additional heat sink or active ventilation can be dispensed with. If the performance requirements are high and the ambient conditions are very hot, then the end user must take care of cooling.
The CPU has an internal temperature sensor to measure the junction temperature. Information on reading the internal temperature sensor can be found in the user manual. If the CPU's internal temperature rises above $T_{CPU\_junc\_max}$, the clock frequency will be throttled. In this case, a heatsink should be used to reduce the temperature.

7 Mechanical Specification

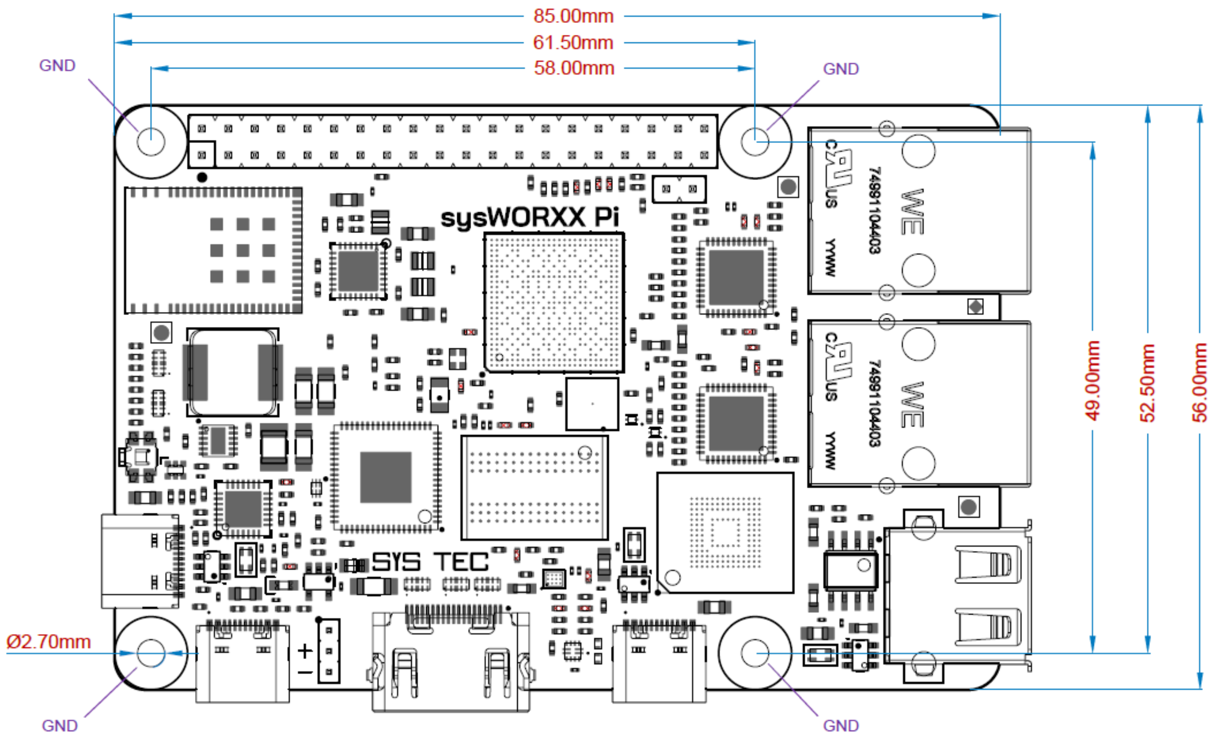

Figure 9: Board outline and size (top view)

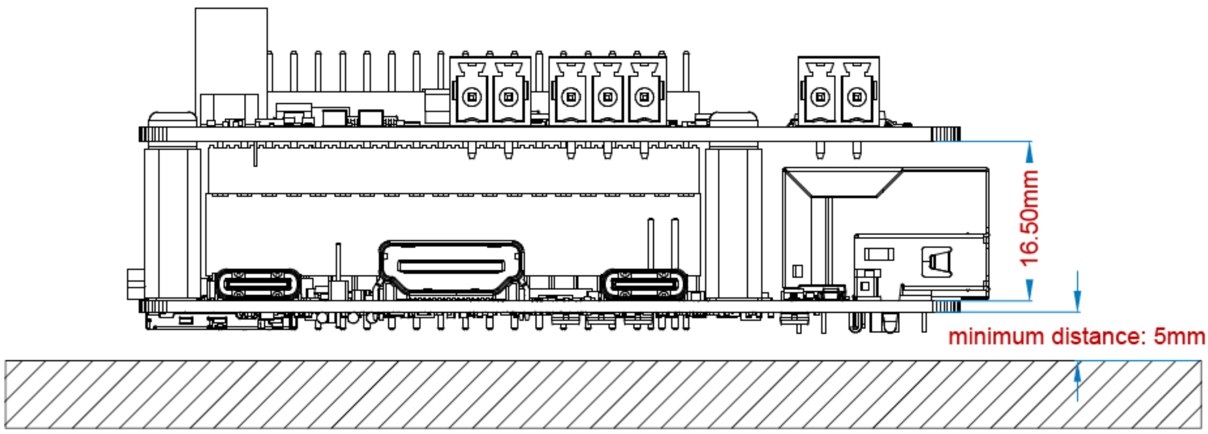

Figure 10: Minimum distances when a HAT is plugged in

8 Repair, Decommissioning and Disposal

The module must be decommissioned after its service life has expired and disposed of properly. The module must not be disposed of with household waste. Disposal of the module must be carried out in accordance with the current legal regulations of the respective country.

A defective module cannot be repaired or can only be repaired by the manufacturer.

9 libsysworxx-io

The **libsysworxx-io** library enables access to the different inputs and outputs of the device as well as additional device information. The header file with the public interface is located at “**/usr/include/sysworxx_io.h**”.

The concept of the libsysworxx-io library that encapsulates specific hardware and provides a uniform interface for different devices is based on the principles of abstraction and modularity. Essentially, this library is intended to hide the various hardware components behind a common interface, so that developers only need to work with this uniform API, regardless of the specific device (sysWORXX Pi-AM62x, CTR-6xx, CTR-8xx) they use. This significantly simplifies software development, maintenance, and extension.

There are different functions for data exchange depending on the IO type. For details about the parameter types of the functions please refer the header file “**sysworxx_io.h**”. To read or write the value of a specific channel, a channel number must be specified. The channel number is mapped by libsysworxx-io and depends on the device tree and device tree overlay used. Digital outputs are written immediately upon request, and changes to digital inputs are reported as events. Analog values are read cyclically, with the cycle time depending on the IO type and the specific device.

Table 12 shows the channel mapping for the IO-Header when the generic Device Tree Overlay is used.

Table 12: Channel Mapping of sysWORXX Pi-AM62x IO-Header

IO-Type	libsysworxx-io function	Channel	IO-Label[5]	Data exchange
Digital Inputs	IoGetInput	0	DI_0[6]	By event
		…		
		12	DI_12[6]	
Digital Outputs	IoSetOutput	0	LED_BL[7]	By request
		1	LED_RD[7]	
		2	LED_GN[7]	
		3	DO_0[6]	
		…	…	
		9	DO_6[6]	
Temperature Sensor	IoTmpGetValue	0	CPU-Temperature[7]	Cyclic (t_{Temp_sample})

[5] See chapter 4.5 for pin assignment to IO-Header

[6] This IO is configured via Device Tree Overlay and dynamically mapped at runtime by a GPIO aggregator. The IO is part of a character device with the prefix "sysworxx-." The lines of this device are added in ascending order of the GPIO chip names.

[7] This IO is mapped by the main device tree and is always available regardless of the device tree overlay used.

10 Device files /dev and interfaces

Table 13: Overview device files and interfaces

Interfaces	/dev/…	Aliases	CPU interface
Linux Console	ttyS2	serial2	&main_uart0
IO-Pin Header	ttyS6	serial6	&main_uart4
IO-Pin Header	ttyS7	serial7	&main_uart5
eMMC interface	mmcblk0	mmc0	&sdhci0
SD-Card interface	mmcblk1	mmc1	&sdhci1
WLAN-Module interface	mmcblk2	mmc2	&sdhci2
IO-Pin Header	spidev1.0	spi0	&ospi0
IO-Pin Header	i2c-3	-	&main_i2c3
ETH0	-	ethernet0	&cpsw_port1
ETH1	-	ethernet1	&cpsw_port2

11 Conformity

This chapter provides information on compliance with various directives.

No conformity assessment is required for the product with regarding to Directives 2014/30/EU (EMC) and 2014/53/EU (RED) because the product is an evaluation kit that is only used by professionals in research and development departments for such purposes.

12 Index