

# UPX-Edge

---

UP Xtreme IoT Gateway System

User's Manual 3<sup>rd</sup> Ed

## Copyright Notice

---

This document is copyrighted, 2021. All rights are reserved. The original manufacturer reserves the right to make improvements to the products described in this manual at any time without notice.

No part of this manual may be reproduced, copied, translated, or transmitted in any form or by any means without the prior written permission of the original manufacturer. Information provided in this manual is intended to be accurate and reliable. However, the original manufacturer assumes no responsibility for its use, or for any infringements upon the rights of third parties that may result from its use.

The material in this document is for product information only and is subject to change without notice. While reasonable efforts have been made in the preparation of this document to assure its accuracy, AAEMON assumes no liabilities resulting from errors or omissions in this document, or from the use of the information contained herein

AAEMON reserves the right to make changes in the product design without notice to its users.

## Acknowledgement

---

All other product names or trademarks are properties of their respective owners.

- Microsoft Windows and Windows 10 are registered trademarks of Microsoft Corp.
- Intel® and Celeron® are registered trademarks of Intel Corporation
- Intel Core™ is a registered trademark of Intel Corporation
- ITE is a trademark of Integrated Technology Express, Inc.
- IBM, PC/AT, PS/2, and VGA are trademarks of International Business Machines Corporation.
- Ubuntu is a registered trademark of Canonical Ltd.
- Yocto and Yocto Project are registered trademarks of the Linux Foundation
- Linux is a registered trademark of Linus Torvalds
- Android is a registered trademark of Google, LLC

All other product names or trademarks are properties of their respective owners. No ownership is implied or assumed for products, names or trademarks not herein listed by the publisher of this document.

## Packing List

---

Before setting up your product, please make sure the following items have been shipped:

Item	Quantity
UPX-EDGE	1
VESA Kit	1
DIN-Rail Kit	1
SATA Cable	1
Screw Pack	1

If any of these items are missing or damaged, please contact your distributor or sales representative immediately.

## About this Document

---

This User's Manual contains all the essential information, such as detailed descriptions and explanations on the product's hardware and software features (if any), its specifications, dimensions, jumper/connector settings/definitions, and driver installation instructions (if any), to facilitate users in setting up their product.

Users may refer to the product page at [AAEON.com](http://AAEON.com) for the latest version of this document.

## Safety Precautions

---

Please read the following safety instructions carefully. It is advised that you keep this manual for future references

1. All cautions and warnings on the device should be noted.
2. Make sure the power source matches the power rating of the device.
3. Position the power cord so that people cannot step on it. Do not place anything over the power cord.
4. Always completely disconnect the power before working on the system's hardware.
5. No connections should be made when the system is powered as a sudden rush of power may damage sensitive electronic components.
6. If the device is not to be used for a long time, disconnect it from the power supply to avoid damage by transient over-voltage.
7. Always disconnect this device from any AC supply before cleaning.
8. While cleaning, use a damp cloth instead of liquid or spray detergents.
9. Make sure the device is installed near a power outlet and is easily accessible.
10. Keep this device away from humidity.
11. Place the device on a solid surface during installation to prevent falls
12. Do not cover the openings on the device to ensure optimal heat dissipation.
13. Watch out for high temperatures when the system is running.
14. Do not touch the heat sink or heat spreader when the system is running
15. Never pour any liquid into the openings. This could cause fire or electric shock.
16. As most electronic components are sensitive to static electrical charge, be sure to ground yourself to prevent static charge when installing the internal components. Use a grounding wrist strap and contain all electronic components in any static-shielded containers.

17. If any of the following situations arises, please the contact our service personnel:
  - i. Damaged power cord or plug
  - ii. Liquid intrusion to the device
  - iii. Exposure to moisture
  - iv. Device is not working as expected or in a manner as described in this manual
  - v. The device is dropped or damaged
  - vi. Any obvious signs of damage displayed on the device
18. **DO NOT LEAVE THIS DEVICE IN AN UNCONTROLLED ENVIRONMENT WITH TEMPERATURES BEYOND THE DEVICE'S PERMITTED STORAGE TEMPERATURES (SEE CHAPTER 1) TO PREVENT DAMAGE.**

### **Warning!**



This device complies with Part 15 FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received including interference that may cause undesired operation.

### **Caution:**

*There is a danger of explosion if the battery is incorrectly replaced. Replace only with the same or equivalent type recommended by the manufacturer. Dispose of used batteries according to the manufacturer's instructions and your local government's recycling or disposal directives.*

### **Attention:**

*Il y a un risque d'explosion si la batterie est remplacée de façon incorrecte. Ne la remplacer qu'avec le même modèle ou équivalent recommandé par le constructeur. Recycler les batteries usées en accord avec les instructions du fabricant et les directives gouvernementales de recyclage.*



## 产品中有毒有害物质或元素名称及含量

AAEON System

QO4-381 Rev.A0

部件名称	有毒有害物质或元素					
	铅 (Pb)	汞 (Hg)	镉 (Cd)	六价铬 (Cr(VI))	多溴联苯 (PBB)	多溴二苯 醚(PBDE)
印刷电路板 及其电子组件	×	○	○	○	○	○
外部信号 连接器及线材	×	○	○	○	○	○
外壳	○	○	○	○	○	○
中央处理器 与内存	×	○	○	○	○	○
硬盘	×	○	○	○	○	○
液晶模块	×	×	○	○	○	○
光驱	×	○	○	○	○	○
触控模块	×	○	○	○	○	○
电源	×	○	○	○	○	○
电池	×	○	○	○	○	○

本表格依据 SJ/T 11364 的规定编制。

○：表示该有毒有害物质在该部件所有均质材料中的含量均在 GB/T 26572标准规定的限量要求以下。

×：表示该有害物质的某一均质材料超出了GB/T 26572的限量要求，然而该部件仍符合欧盟指令2011/65/EU 的规范。

备注：

- 一、此产品所标示之环保使用期限，系指在一般正常使用状况下。
- 二、上述部件物质中央处理器、内存、硬盘、光驱、电源为选购品。
- 三、上述部件物质液晶模块、触控模块仅一体机产品适用。

# China RoHS Requirement (EN)

## Hazardous and Toxic Materials List

AAEON System

QO4-381 Rev.A0

Component Name	Hazardous or Toxic Materials or Elements					
	Lead (Pb)	Mercury (Hg)	Cadmium (Cd)	Hexavalent Chromium (Cr(VI))	Polybrominated biphenyls (PBBS)	Polybrominated ethers (PBDES)
PCB and Components	X	O	O	O	O	O
Wires & Connectors for Ext.Connections	X	O	O	O	O	O
Chassis	O	O	O	O	O	O
CPU & RAM	X	O	O	O	O	O
HDD Drive	X	O	O	O	O	O
LCD Module	X	X	O	O	O	O
Optical Drive	X	O	O	O	O	O
Touch Control Module	X	O	O	O	O	O
PSU	X	O	O	O	O	O
Battery	X	O	O	O	O	O

This form is prepared in compliance with the provisions of SJ/T 11364.

O: The level of toxic or hazardous materials present in this component and its parts is below the limit specified by GB/T 26572.

X: The level of toxic of hazardous materials present in the component exceed the limits specified by GB/T 26572, but is still in compliance with EU Directive 2011/65/EU (RoHS 2).

Notes:

1. The Environment Friendly Use Period indicated by labelling on this product is applicable only to use under normal conditions.
2. Individual components including the CPU, RAM/memory, HDD, optical drive, and PSU are optional.
3. LCD Module and Touch Control Module only applies to certain products which feature these components.

# Table of Contents

---

<b>Chapter 1 - Product Specifications</b> .....	<b>1</b>
1.1 Specifications .....	2
<b>Chapter 2 – Hardware Information</b> .....	<b>4</b>
2.1 Dimensions .....	5
2.2 Jumpers and Connectors.....	7
2.3 List of Jumpers and Connectors.....	10
2.3.1 Power Button (SW2) .....	12
2.3.2 RTC (CN1).....	13
2.3.3 Mini-Card (CN2).....	13
2.3.4 M.2 2230 E-Key (CN3) .....	15
2.3.5 M.2 Slot M-Key (CN4).....	17
2.3.6 SATA Connector (CN5) .....	19
2.3.7 SATA Power (CN6).....	19
2.3.8 LAN1 (CN8 i219)/ LAN2 (CN7 i210) .....	20
2.3.9 HDMI/ DP (CN10).....	21
2.3.10 DOCKING I (CN11) .....	22
2.3.11 EDP (CN13).....	24
2.3.12 USB Type A Connector (CN14/ CN15).....	25
2.3.13 USB 2.0 1x10P Wafer (CN16) .....	26
2.3.14 FAN (CN19).....	26
2.3.15 DC Jack (DC: 24~50V) (CN20) .....	27
2.3.16 HAT 40 (CN22) .....	27
2.3.17 Audio Jack (CN24) .....	29
2.3.18 Internal Header STM32 MCU (CN25).....	30
2.3.19 DOCKING II (CN26).....	31
2.3.20 Reset Pin Header (CN27) .....	32

2.4	Cable Pin Assignments.....	33
2.4.1	RS232/RS422/RS485 (COM1/COM2).....	33
2.4.2	HAT40 (GPIO1).....	34
2.4.3	External Connector STM32 MCU (GPIO2).....	35
2.4.4	Serial Port.....	36
<b>Chapter 3 – Drivers Installation.....</b>		<b>37</b>
3.1	Driver Download and Installation.....	38
<b>Appendix A – UP Framework SDK Installation.....</b>		<b>41</b>
A.1	Introduction.....	42
A.2	Installation for Windows 10.....	42
A.3	Installation for Windows IoT Core.....	45
<b>Appendix B – Cables and Connectors.....</b>		<b>47</b>
B.1	Cables and Connectors.....	48
<b>Appendix C – CEC-Client Installation.....</b>		<b>50</b>
C.1	CEC-Client Installation Manual.....	51

# Chapter 1

---

Product Specifications

## 1.1 Specifications

---

### System

CPU	8th generation Intel® Core™ i7/i5/i3/Celeron Processor SoC
Memory	Onboard DDR4 memory, Max 16GB Single Channel: 4GB Dual Channel: 8GB, 16GB
Graphics	Intel® Graphics, GEN 9
Storage	eMMC 64GB
Ethernet	<b>i7/ i5 CPU:</b> i210-AT x1; i219-LM x1 <b>i3/ Celeron CPU:</b> i210-AT x1; i219-V x1
WIFI/BT	Optional
Audio	Line out + MIC x 1
USB	USB 3.2 Gen 2 x 4
Expansion Slot	M.2 2230 E Key x 1 M.2 2280 M Key x 1 (with up to two-lane PCIe) mPCIe slot x 1 100pin connector x 1 40-pin HAT x 1 SATA III (6.0 Gbps) connector x 1 (support 2.5" up to 7mm height HDD/SSD expansion)

## I/O Placements

<b>Power</b>	DC-in Jack x 1, Power Button x 1
<b>USB</b>	USB 3.2 Gen 2 STACK Connector for 2 ports (Front) x 1 USB 3.2 Gen 2 STACK Connector for 2 ports (Rear) x 1
<b>Display Port</b>	HDMI 2.0 x 1 (Support 4K at 60 Hz) DP 1.2 x 1 16pin GPIO connectors (1 x GPIO for SMT32, 1 x GPIO for 40pin HAT)
<b>Ethernet</b>	RJ45 connector x 2
<b>COM</b>	RS-232/422/485 x 2

## Environment

<b>Power</b>	12V - 60VDC
<b>Form Factor</b>	7.48" x 5.07" x 3.1" (190mm x 129mm x 78.6mm)
<b>Gross Weight</b>	6.61 lbs. (3 kg)
<b>Operating Temperature</b>	-4°F ~ 158°F (-20°C ~ 70°C) SATA: -4°F ~ 158°F (-20°C ~ 70°C) eMMC: 32°F ~ 140°F (0°C ~ 60°C) with AI module: 32°F ~ 140°F (0°C ~ 60°C)
<b>Operating Humidity</b>	0% ~ 90% relative humidity non-condensing
<b>Certification</b>	CE/FCC Class A, RoHS Compliant, REACH
<b>OS Support</b>	Win 10, Linux Ubuntu 18.04 with Kernel 5.0, Linux Yocto 2.7 with Kernel 4.19

# Chapter 2

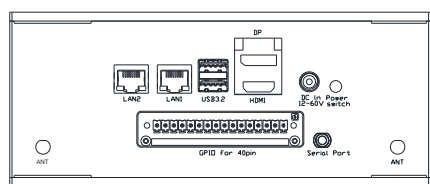
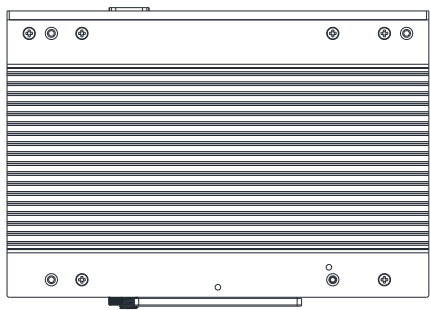
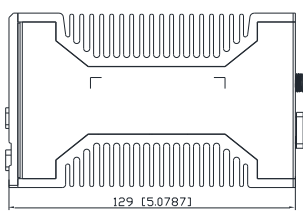
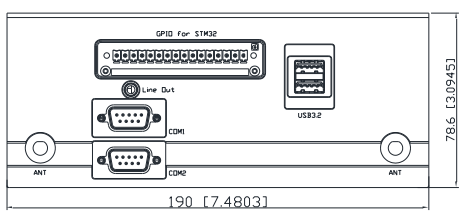
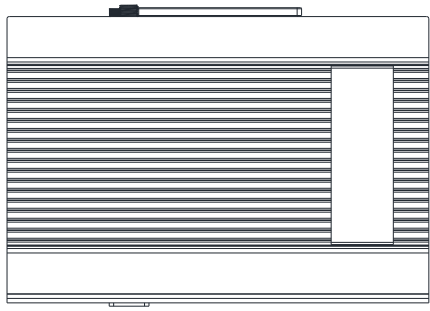
---

Hardware Information

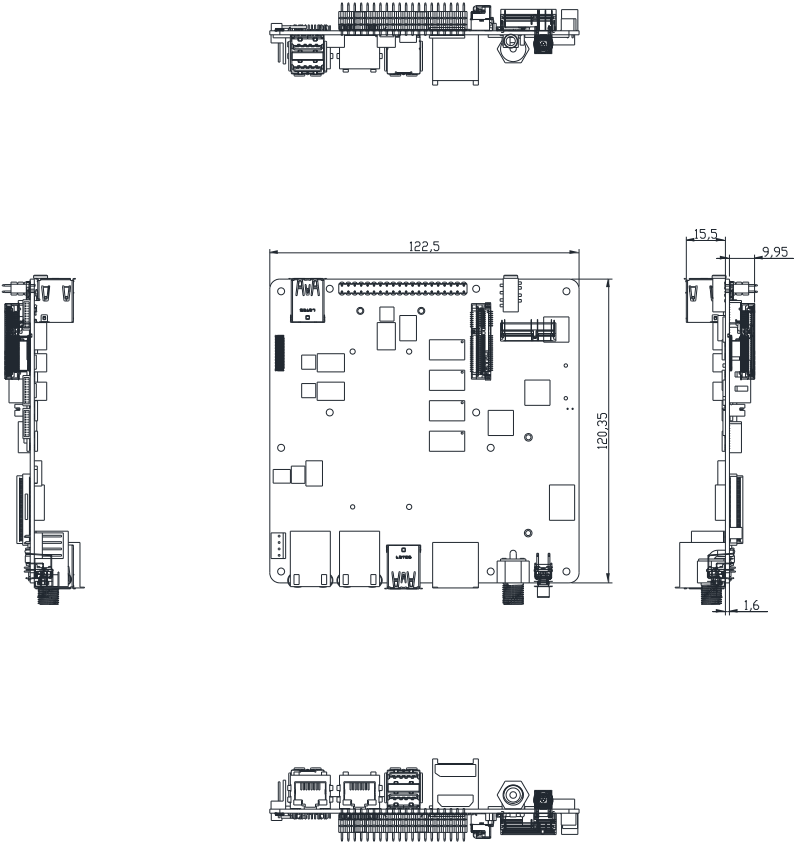


## 2.1 Dimensions

### System

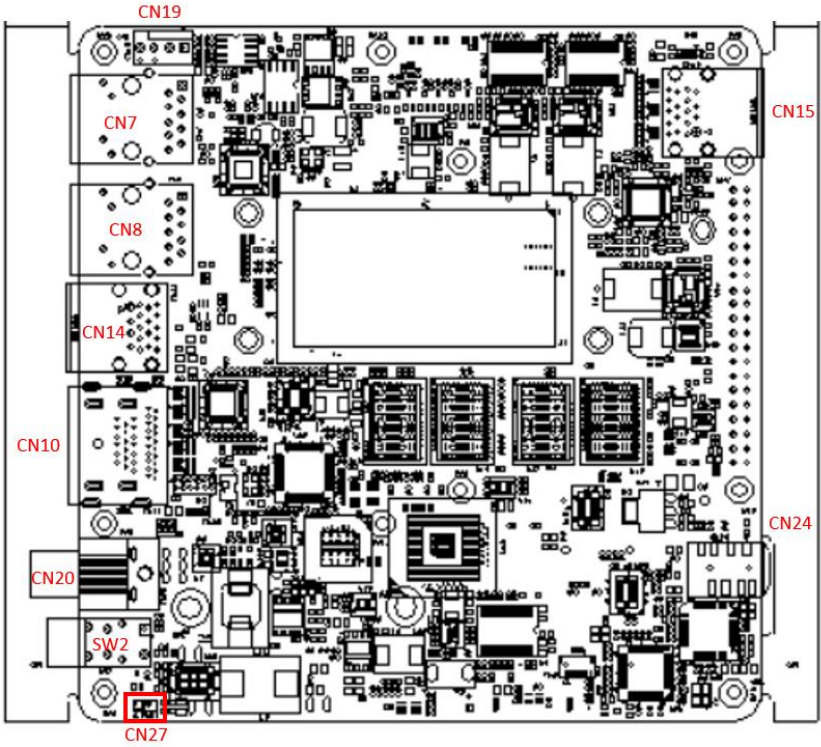


# Board

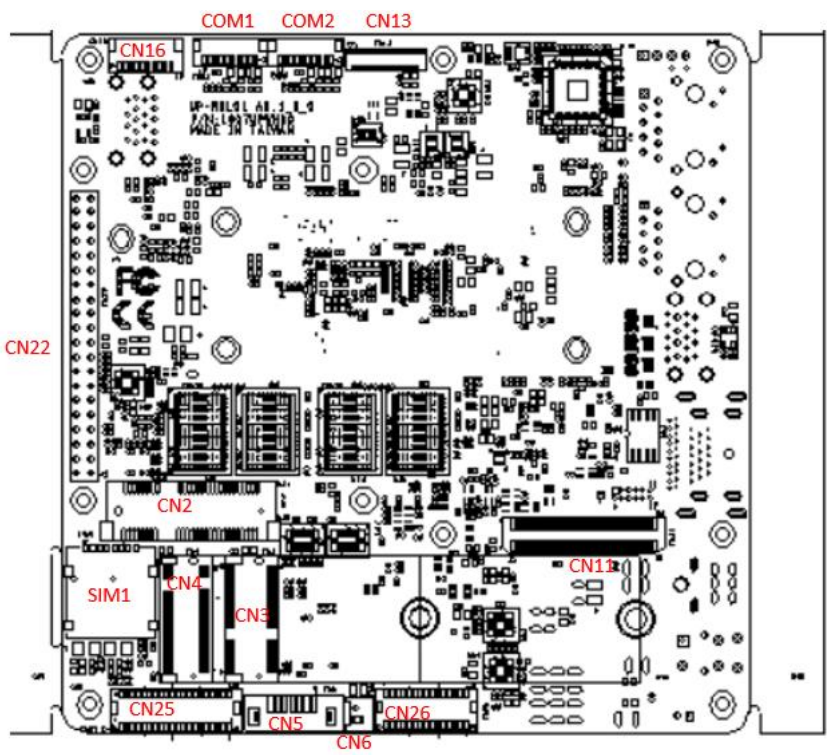


## 2.2 Jumpers and Connectors

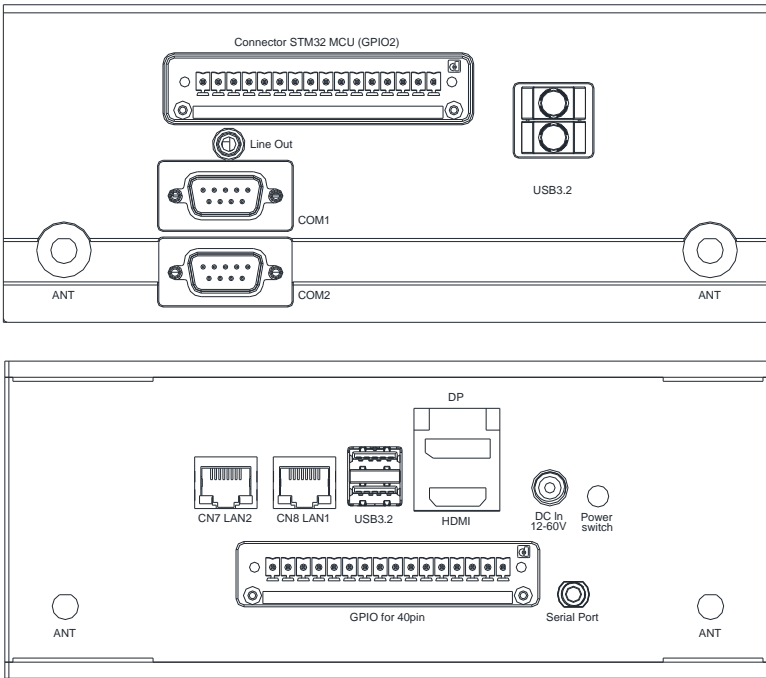
Board Top:



Board Bottom:



## External Connections



## 2.3 List of Jumpers and Connectors

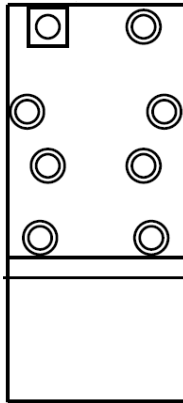
Please refer to the table below for all of the board's jumpers that you can configure for your application

Label	Function	Connector Type
SW2	PWR button	(TF)Push Button Switch.8PDIP180D.w/ Red & Blue LED.Highlyelec.PB6156RSL-14-101
CN1	RTC	(TF)WAFER BOX.2P180D(M).DIP1.25mm.PINREX.712-71-02TW01
CN2	MINI CARD	(TF)MiniCard SLOT.52P90D.(F).SMD.FOXCONN.AS0B226-S68Q-7H
CN3	M.2_2230_E-KEY	(TF)M.2 Key-E Slot.H=4.0mm conn.75P90D(F).BLACK.SMD.FOXCONN.AS0BC21-S40BE-7H
CN4	M.2 SLOT_M key	TF)M.2 Key-M Slot.75P90D(F).Standard type.BLACK.SMD.H=8.5mm conn.FOXCONN.2E0BC21-S85BM-7H
CN5	SATA CONN	(TF)SATA CONNECTOR.7P180D(M).SMT.TechBest.007-01-00757
CN6	SATA POWER	(TF)WAFER BOX.2P180D(M).DIP2.0mm.w/LOCK.PINREX.721-81-02TW00
CN7	LAN2 i210	(TF)RJ45.14P90D(F)W/TF(10/100/1000Base). & LED(L-G/O,R-Y).DIP.SPEEDTECH.RK7L8A-K WH1-F30-0R
CN8	LAN1 i219	(TF)RJ45.14P90D(F)W/TF(10/100/1000Base). & LED(L-G/O,R-Y).DIP.SPEEDTECH.RK7L8A-K WH1-F30-0R
CN10	HDMI/DP	(TF)HDMI/DP combo Port conn..39P90D(F).DIP.FOXCONN.3VD11203-HHJ0-4H

Label	Function	Connector Type
CN11	DOCKING I	(TF)Board-Board Connector.100P:180D(M).Pitch=0.5mm.H=5.3mm.SMD.Panasonic.AXK6S00647YG
CN13	eDP	(TF)FPC/FFC Conn.41P:90D(F).SMD.0.6mm.Hirose.FH35C-41S-0.3SHW(50)
CN14	USB Type A Connector 1	(TF)USB CONNECTOR DUAL PORT.8P:TechBest.KS-002D-ANB(2.0)-L
CN15	USB Type A Connector 1	(TF)USB CONNECTOR DUAL PORT.8P:TechBest.KS-002D-ANB(2.0)-L
CN16	USB 2.0 1x10P Wafer	(TF)Wafer Box.10P:90D.1.0mm.CATCH.1204-700-10RM
CN19	FAN	(TF)WAFER.3P:180D(M).DIP2.54mm.PINREX.742-81-03TW00
CN20	DC JACK	(TF)DC Power Jack.3P:90D(F).DIP.COXOC.416AEDCD020105B
CN22	HAT 40	(TF)PIN HEADER.20*2P:180D(M).DIP2.54mm.JVE.21N22564-40S20B-01G-5.5/3.3
CN24	AUDIO JACK	(TF)Audio Jack.8P:90D(F).SMD.Astron.E35S16AA-8S-R
CN25	Internal Header STM32 MCU	(TF)WAFER BOX.40P:180D(M).SMD.1mm.W/Cap.ASTRON.28-3602-40-R
CN26	DOCKING II	(TF)BOX Header.15*2P:180D(M).SMD.1.0mm.CATCH.1204-710-30SMP
CN27	Reset Pin Header	
COM1	1701100180	(TF)COM Cable.D-SUB 9P(M).10P:1.0mm Housing.15cm
COM2	1701100180	(TF)COM Cable.D-SUB 9P(M).10P:1.0mm Housing.15cm
GPIO1	1703401301	(TF)WIRE.16Pto 40Pin.130mm.Housing.for FWAA-1218

Label	Function	Connector Type
GPIO2	1703401600	(TF)WIRE.16Pin.to 40 Pin.160mm.Housing.for FWAA-1217
Serial port	1709103100	(TF)WAFER Cable.AUDIO JACK.to 10Pin.310mm.Housing.for FWAA-1216

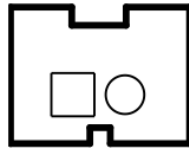
### 2.3.1 Power Button (SW2)



Pin	Signal	Pin	Signal
1	PWR_SW#_CTL_R	2	PWR_SW#_CTL_R
3	GND	4	GND
5	GND	6	GND
L1	SW1_LED_P	L2	SW1_LED_N

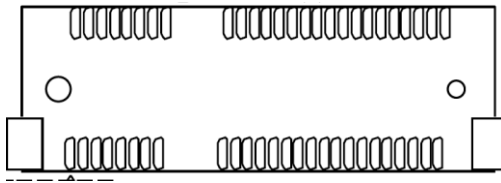


### 2.3.2 RTC (CN1)



Pin	Signal
1	RTC_VCC
2	GND

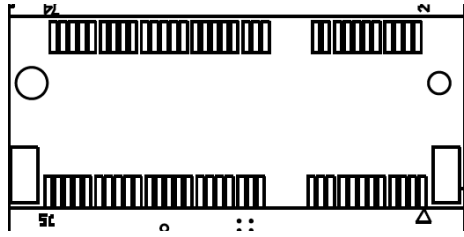
### 2.3.3 Mini-Card (CN2)



Pin	Signal	Pin	Signal
1	WAKE_MINI_N	2	VCC3_MINIPCIE
3	NC	4	GND
5	NC	6	V1.5S
7	NC	8	P_UIM_PWR
9	GND	10	P_UIM_DAT
11	PCIE_REFCLK5_P2_N	12	P_UIM_CLK
13	PCIE_REFCLK5_P2_P	14	P_UIM_RST
15	GND	16	P_UIM_VPP
17	NC	18	GND
19	NC	20	3G_EN

Pin	Signal	Pin	Signal
21	GND	22	3G_RST
23	PERN0_MSATA_R+	24	VCC3_MINIPCIE
25	PERP0_MSATA_R-	26	GND
27	GND	28	V1.5S
29	GND	30	NC
31	PETN0_MSATA_T-	32	NC
33	PETP0_MSATA_T+	34	GND
35	GND	36	USB2_DN_R
37	GND	38	USB2_DP_R
39	VCC3_MINIPCIE	40	GND
41	VCC3_MINIPCIE	42	NC
43	MSATA_PCIE_SEL_C	44	NC
45	NC	46	NC
47	NC	48	V1.5S
49	NC	50	GND
51	NC	52	VCC3_MINIPCIE

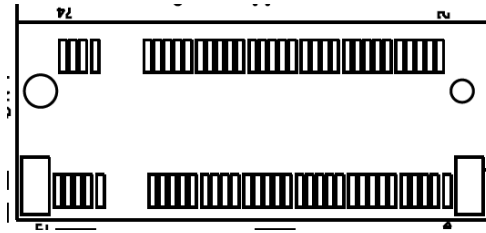
### 2.3.4 M.2 2230 E-Key (CN3)



Pin	Signal	Pin	Signal
1	GND	2	+3.3VAUX_WIFI
3	BT_DP	4	+3.3VAUX_WIFI
5	BT_DM	6	NC
7	GND	8	NC
9	CNV_WR_LANE1_DN	10	CNV_RF_RST#
11	CNV_WR_LANE1_DP	12	NC
13	GND	14	CNV_PCMTOUT_CLKREQ_R
15	CNV_WR_LANE0_DN	16	NC
17	CNV_WR_LANE0_DP	18	GND
19	GND	20	NC
21	CNV_WR_CLK_DN	22	CNV_RGI_RSP_R
23	CNV_WR_CLK_DP	24	NC
25	NC	26	NC
27	NC	28	NC
29	NC	30	NC
31	NC	32	GPP_F6_CNV_RGI_DT_UAR
33	GND	34	CNV_BRI_DT_R
35	PCIE_C_TXP10	36	CNV_BRI_DT
37	PCIE_C_TXN10	38	NC

Pin	Signal	Pin	Signal
39	GND	40	NC
41	PCIE_P10_RX_DP	42	NC
43	PCIE_P10_RX_DN	44	NC
45	GND	46	NC
47	PCIE_REFCLK5_P1_P	48	NC
49	PCIE_REFCLK5_P1_N	50	SUS_CLK_CPU
51	GND	52	WIFI_RST#
53	PCIE_M2_CLKREQ#	54	BT_EN
55	WAKE_M2_N	56	WIFI_EN
57	GND	58	NC
59	CNV_WT_LANE1_DN	60	NC
61	CNV_WT_LANE1_DP	62	NC
63	GND	64	CLKIN_XTAL_LCP_R
65	CNV_WT_LANE0_DN	66	NC
67	CNV_WT_LANE0_DP	68	NC
69	GND	70	NC
71	CNV_WT_CLK_DN	72	3.3VAUX_WIFI
73	CNV_WT_CLK_DP	74	3.3VAUX_WIFI
75	GND	76	GND
77	GND	78	

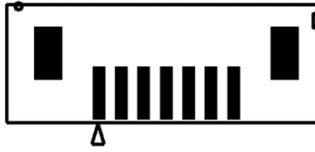
### 2.3.5 M.2 Slot M-Key (CN4)



Pin	Signal	Pin	Signal
1	GND	35	PCIE_P15_SATA_P1B_TXN
2	+3.3V	36	NC
3	GND	37	PCIE_P15_SATA_P1B_TXP
4	+3.3V	38	VPU_EN
5	NC	39	GND
6	FULL_CARD_PWR_OFF#	40	NC
7	NC	41	PCIE_TXN0
8	NC	42	NC
9	NC	43	PCIE_TXP0
10	NC	44	NC
11	NC	45	GND
12	+3.3V	46	NC
13	NC	47	PCIE_RXN0
14	+3.3V	48	NC
15	GND	49	PCIE_RXP0
16	+3.3V	50	PLT_RST#
17	NC	51	GND
18	+3.3V	52	PCIE_CLKREQ#
19	NC	53	PCIE_CLK_N

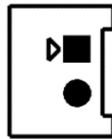
Pin	Signal	Pin	Signal
20	NC	54	WAKE_M2
21	GND	55	PCIE_CLK_P
22	NC	56	NC
23	NC	57	GND
24	NC	58	NC
25	NC	67	NC
26	NC	68	NC
27	GND	69	NC
28	NC	70	+3.3V
29	PCIE_P15_SATA_P1B_RXN	71	GND
30	NC	72	+3.3V
31	PCIE_P15_SATA_P1B_RXP	73	GND
32	NC	74	+3.3V
33	GND	75	GND
34	NC		

### 2.3.6 SATA Connector (CN5)



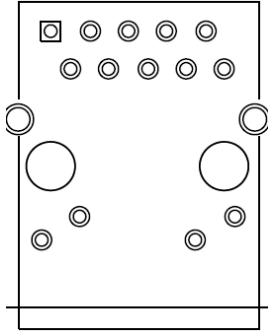
Pin	Signal	Pin	Signal
1	GND	2	SATA_TXP0_C
3	SATA_TXN0_C	4	GND
5	SATA_RXN0_C	6	SATA_RXP0_C
7	GND		

### 2.3.7 SATA Power (CN6)



Pin	Signal	Pin	Signal
1	+V5S	2	GND

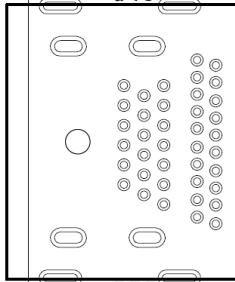
### 2.3.8 LAN1 (CN8 i219)/ LAN2 (CN7 i210)



Pin	Signal	Pin	Signal
1	LAN1_TMDI0+	5	LAN1_TMDI2-
2	LAN1_TMDI0-	6	LAN1_TMDI1-
3	LAN1_TMDI1+	7	LAN1_TMDI3+
4	LAN1_TMDI2+	8	LAN1_TMDI3-



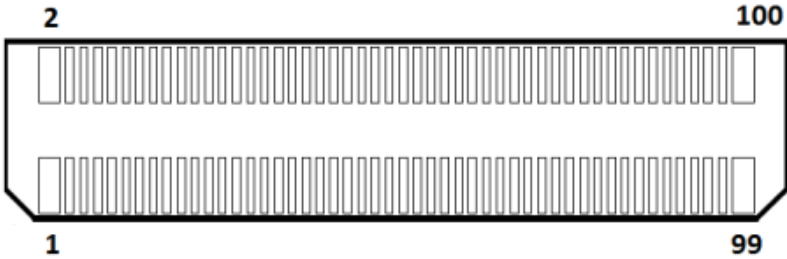
### 2.3.9 HDMI/ DP (CN10)



Pin	Signal	Pin	Signal
P1	DDIO_TXP_DP_0	P2	GND
P3	DDIO_TXN_DP_0	P4	DDIO_TXP_DP_1
P5	GND	P6	DDIO_TXN_DP_1
P7	DDIO_TXP_DP_2	P8	GND
P9	DDIO_TXN_DP_2	P10	PORT0_CLK+
P11	GND	P12	PORT0_CLK-
P13	CONFIG1	P14	CONFIG2
P15	DP_AUX_P	P16	GND
P17	DP_AUX_N	P18	DDIO_TYPE_C_HPD
P19	GND	P20	3.3V
P21	DDI1_TXP_HDMI_0	P22	GND
P23	DDI1_TXN_HDMI_0	P24	DDI1_TXP_HDMI_1
P25	GND	P26	DDI1_TXN_HDMI_1
P27	DDI1_TXP_HDMI_2	P28	GND
P29	DDI1_TXN_HDMI_2	P30	DDI1_CLK+_HDMI
P31	GND	P32	DDI1_CLK-_HDMI
P33	HDMI1_CEC_D	P34	NC
P35	DDC_CLK	P36	DDC_DATA

Pin	Signal	Pin	Signal
P37	GND	P38	5V
DDI1_TYPE_C_HPDP			

### 2.3.10 DOCKING I (CN11)

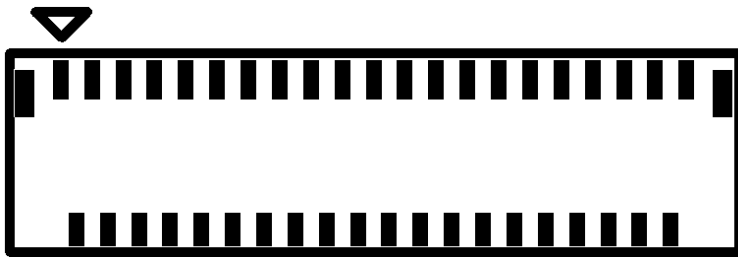


Pin	Signal	Pin	Signal
1	+12V	51	PCIE_REFCLK2_N
2	+12V	52	PCIE_P3_TXN
3	+12V	53	GND
4	+12V	54	GND
5	+12V	55	PCIE_TXP2
6	+12V	56	PCIE_P3_RXP_SWITCH
7	+12V	57	PCIE_TXN2
8	+12V	58	PCIE_P3_RXN_SWITCH
9	NC	59	GND
10	NC	60	GND
11	GND	61	PCIE_RXP2
12	GND	62	SATA_P1_USB3_P5_RXN
13	NC	63	PCIE_RXN2
14	NC	64	SATA_P1_USB3_P5_RXP
15	NC	65	GND

Pin	Signal	Pin	Signal
16	NC	66	GND
17	GND	67	NC
18	GND	68	SATA_P1_USB3_P5_TXN
19	NC	69	NC
20	NC	70	SATA_P1_USB3_P5_TXP
21	NC	71	GND
22	NC	72	GND
23	GND	73	NC
24	GND	74	SATA_RXN0
25	NC	75	NC
26	NC	76	SATA_RXP0
27	NC	77	GND
28	NC	78	GND
29	GND	79	USB2_DP3
30	GND	80	SATA_TXP0
31	PCIE_REFCLK0_P	81	USB2_DN3
32	PCIE_REFCLK1_P	82	SATA_TXN0
33	PCIE_REFCLK0_N	83	GND
34	PCIE_REFCLK1_N	84	GND
35	GND	85	NC
36	GND	86	NC
37	PCIE_RXP0	87	NC
38	PCIE_RXP1	88	NC
39	PCIE_RXN0	89	GND
40	PCIE_RXN1	90	GND
41	GND	91	NC
42	GND	92	SATA_LED_N

Pin	Signal	Pin	Signal
43	PCIE_TXP0	93	PLTRST_PNLBKLEN
44	PCIE_TXP1	94	NC
45	PCIE_TXN0	95	PCIE_WAKE0_N
46	PCIE_TXN1	96	PCIE_CLKREQ0#
47	GND	97	PCIE_WAKE1_N
48	GND	98	PCIE_CLKREQ1#
49	PCIE_REFCLK2_P	99	PCIE_WAKE2_N
50	PCIE_P3_TXP	100	PCIE_CLKREQ2#

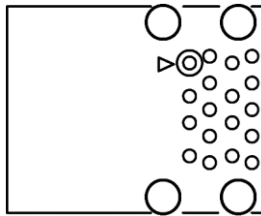
### 2.3.11 EDP (CN13)



Pin	Signal	Pin	Signal
1	NC	2	NC
3	GND	4	NC
5	NC	6	GND
7	NC	8	NC
9	GND	10	NC
11	NC	12	GND
13	NC	14	NC
15	GND	16	data positive 0
17	data negative 0	18	GND

Pin	Signal	Pin	Signal
19	data positive 1	20	data negative 1
21	GND	22	data positive 2
23	data negative 2	24	GND
25	data positive 3	26	data negative 3
27	GND	28	AUX positive
29	AUX negative	30	GND
31	Hot Plug Detect	32	Back Light Control
33	VDD Enable	34	Back Light Enable
35	I2C_CLK	36	I2C_DAT
37	3.3V	38	3.3V
39	3.3V	40	3.3V
41	3.3V		

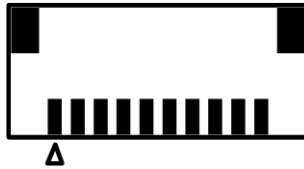
### 2.3.12 USB Type A Connector (CN14/ CN15)



Pin	Signal	Pin	Signal
1	5V	2	USB2_D1-
3	USB2_D1+	4	GND
5	USB3_RX1-	6	USB3_RX1+
7	GND	8	USB3_TX1-
9	USB3_TX1+	10	5V

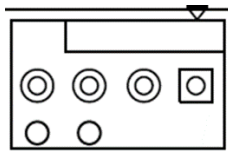
Pin	Signal	Pin	Signal
11	USB2_D2-	12	USB2_D2+
13	GND	14	USB3_RX2-
15	USB3_RX2+	16	GND
17	USB3_TX2-	18	USB3_TX2+

### 2.3.13 USB 2.0 1x10P Wafer (CN16)



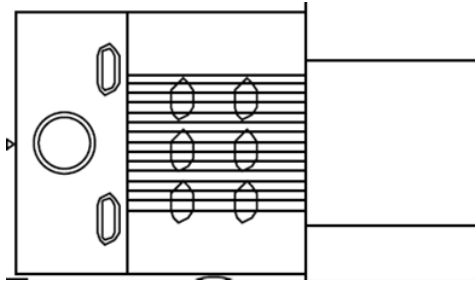
Pin	Signal	Pin	Signal
1	5V	2	USB2_D1-
3	USB2_D1+	4	GND
5	NC	6	NC
7	NC	8	NC
9	UART_RX	10	UART_TX

### 2.3.14 FAN (CN19)



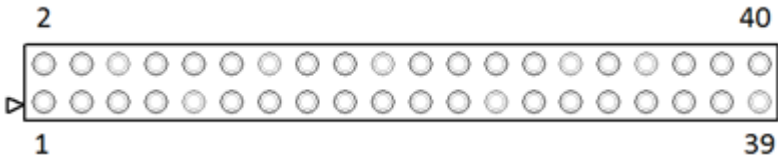
Pin	Signal	Pin	Signal
1	GND	2	VCC_FAN_CPU_CON
3	FAN_TAC_CPU_CON	4	FAN_CTL_CPU_CON

### 2.3.15 DC Jack (DC: 24~50V) (CN20)



Pin	Signal	Pin	Signal
1	DC_IN	2	GND
3	GND		

### 2.3.16 HAT 40 (CN22)



Pin	Signal	BIOS
1	3V3	
2	5V	
3	GPIO0/I2C1_SDA	GPIO1
4	5V	
5	GPIO1/I2C1_SCL	GPIO2
6	GND	
7	GPIO2/ADC_in1	GPIO3

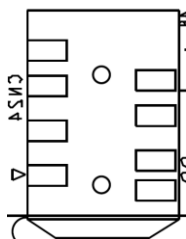
Pin	Signal	BIOS
8	GPIO15/UART_TXD	GPIO16
9	GND	
10	GPIO16/UART_RXD	GPIO17
11	GPIO3/UART_RTX/ADC_in2	GPIO4
12	GPIO17/I2S_BCLK	GPIO18
13	GPIO4/ADC_in3	GPIO5
14	GND	
15	GPIO5/ADC_in4	GPIO6
16	GPIO18	GPIO19
17	3V3	
18	GPIO19	GPIO20
19	GPIO6/SPI_1_TXD	GPIO7
20	GND	
21	GPIO7/SPI_1_RXD	GPIO8
22	GPIO20	GPIO21
23	GPIO8/SPI_1_CLK	GPIO9
24	GPIO21/SPI_1_FSO	GPIO22
25	GND	
26	GPIO22/SPI_1_FS1	GPIO23
27	GPIO9/I2C0_SDA	GPIO10
28	GPIO23/I2C0_SCL	GPIO24
29	GPIO10	GPIO11
30	GND	
31	GPIO11	GPIO12
32	GPIO24	GPIO25
33	GPIO12	GPIO13
34	GND	



Pin	Signal	BIOS
35	GPIO13/I2S_WS_SYNC	GPIO14
36	GPIO25/UART_CTS	GPIO26
37	GPIO14	GPIO15
38	GPIO26/I2S_SDI	GPIO27
39	GND	
40	GPIO27/I2S_SDO	GPIO28

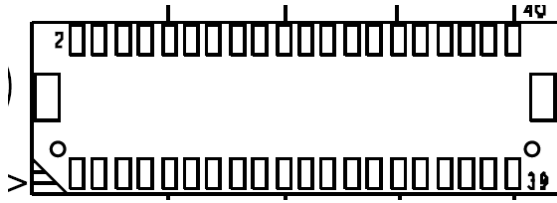
**Note:** Open-drain I2C bus. Needs the pullup resistors to external 3.3V.

### 2.3.17 Audio Jack (CN24)



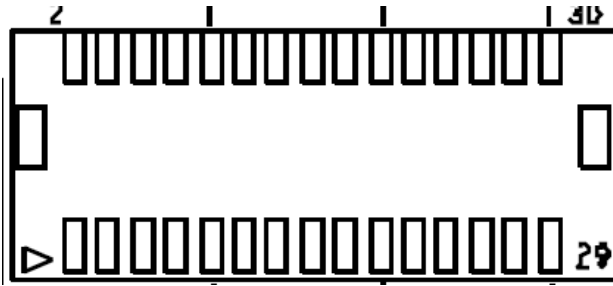
Pin	Signal	Pin	Signal
1	MIC_LR_CN	2	AGND
3	LOUT_R	4	NC
5	NC	6	NC
7	NC	8	LOUT_L

### 2.3.18 Internal Header STM32 MCU (CN25)



Pin	Signal	Pin	Signal
1	+3V_MCU	2	+3V_MCU
3	GND	4	GND
5	STM32_PB1	6	STM32_VBAT
7	STM32_PB7	8	STM32_PC8
9	STM32_PB6	10	STM32_PB9
11	STM32_PB5	12	STM32_PC15
13	STM32_PB4	14	MCU_NRST_R
15	STM32_PB3	16	MCU_WKUP_R
17	USB_RENUMn	18	STM32_PA1
19	SWDCLK_R	20	STM32_PA2
21	SWDIO_R	22	STM32_PA3
23	NC	24	STM32_PA4
25	NC	26	STM32_PA5
27	STM32_PA10	28	STM32_PA6
29	STM32_PA9	30	STM32_PA7
31	STM32_PA8	32	HDMI1_CEC_D
33	STM32_PB15	34	STM32_PB2
35	STM32_PB14	36	HOST_I2C_SCL
37	STM32_PB13	38	HOST_I2C_SDA
39	MCU_INT_R	40	+V5S

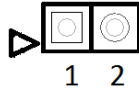
### 2.3.19 DOCKING II (CN26)



Pin	Signal	Pin	Signal
1	SLP_S3#	2	V1.8S
3	STACK_GPIO1	4	V1.8S
5	STACK_GPIO2	6	LPC_AD0
7	STACK_GPIO3	8	LPC_AD1
9	STACK_GPIO4	10	LPC_AD2
11	STACK_GPIO5	12	LPC_AD3
13	STACK_GPIO6	14	LPC_FRAME_R
15	BUF_PLT_RST#	16	LPC_R_CLKOUT1
17	STACK_GPIO7	18	SMB_CLK_A
19	I2C_SCL4_1V8	20	SMB_DATA_A
21	I2C_SDA4_1V8	22	INT_SERIRQ_3P3
23	GND	24	+V3.3S
25	GND	26	+V3.3S
27	GND	28	+V5S
29	GND	30	+V5S

### 2.3.20 Reset Pin Header (CN27)

---



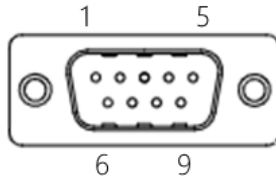
Pin	Signal
1	PMU_RSTBTN_N
2	GND

## 2.4 Cable Pin Assignments

---

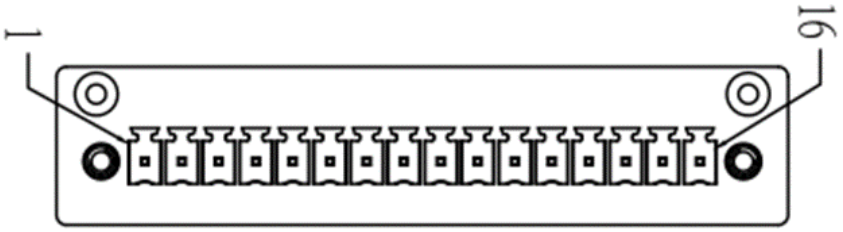
### 2.4.1 RS232/RS422/RS485 (COM1/COM2)

---



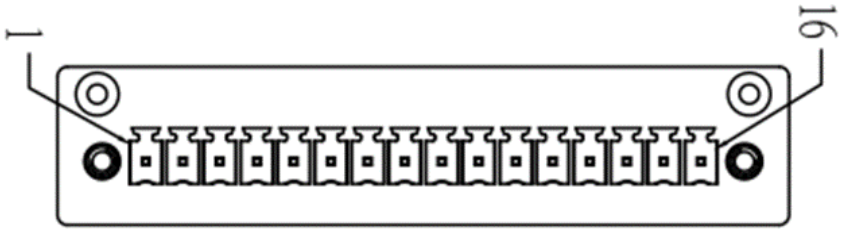
Pin	Signal
1	DCDA / RS422TX- /
2	RXA / RS422TX+ / RS485+
3	TXA / RS422RX+
4	DTRA / RS422RX-
5	GND
6	DSRA
7	RTSA
8	CTSA
9	RIA

## 2.4.2 HAT40 (GPIO1)



Pin	Signal	Pin	Signal
1	PIN1 (3.3V)	2	PIN2 (5V)
3	PIN3 (I2C_SDA)	4	PIN5 (I2C_SCL)
5	PIN6 (GND)	6	PIN9 (GND)
7	PIN12 (GPIO)	8	PIN13 (GPIO)
9	PIN15 (GPIO)	10	PIN16 (GPIO)
11	PIN18 (GPIO)	12	PIN19 (GPIO)
13	PIN21 (GPIO)	14	PIN22 (GPIO)
15	PIN32	16	PIN33

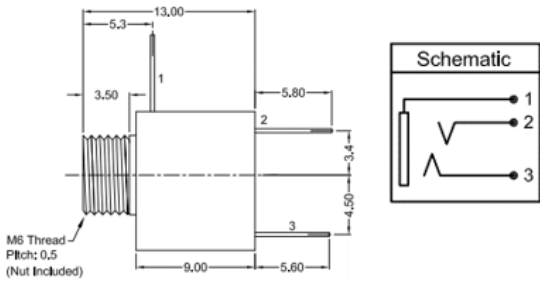
### 2.4.3 External Connector STM32 MCU (GPIO2)



**Note:** GPIO2 connector is wired internally to the CN25 header and supplies the following signals:

Pin	Signal	Pin	Signal
1	PIN1 3.3V	2	PIN3 GND
3	PIN7 PB7/ I2C_SDA	4	PIN9 PB6/ I2C_SCL
5	PIN18 PA1/ADC12_IN1	6	PIN20 PA2/ADC12_IN2
7	PIN22 PA3/ADC12_IN3	8	PIN24 PA4/ADC12_IN4
9	PIN26 PA5/ADC12_IN5	10	PIN28 PA6/ADC12_IN6
11	PIN30 PA7/ADC12_IN7	12	PIN18 PA1/ADC12_IN1
13	PIN5 PB1/ADC12_IN9	14	PIN12 PC15/GPIO
15	PIN8 PB8/CANRX	16	PIN10 PB9/CANTX

## 2.4.4 Serial Port



Pin	Signal
1	pin 8 (GND)
2	pin10 (TXD_3V3)
3	pin 9 (RXD_3V3)



# Chapter 3

---

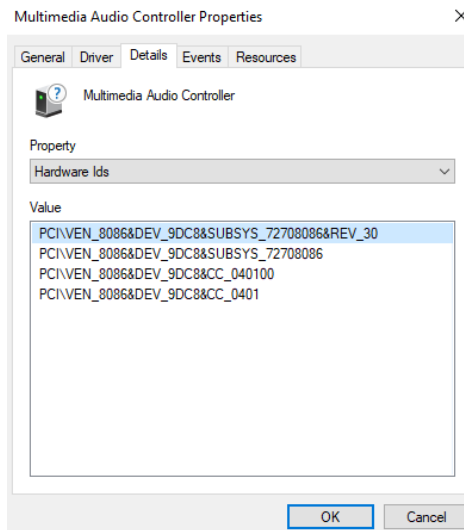
Drivers Installation

### 3.1 Driver Download and Installation

Access <https://www.up-community.org> and go to the Downloads Section> UP Xtreme to find the relevant drivers.

After downloading and installing the drivers you will see three unknown devices in Windows Device Manager.

#### Unknown Device 1: Multimedia Audio Controller



To clear this, go to CBR setup and navigate to the following menu:

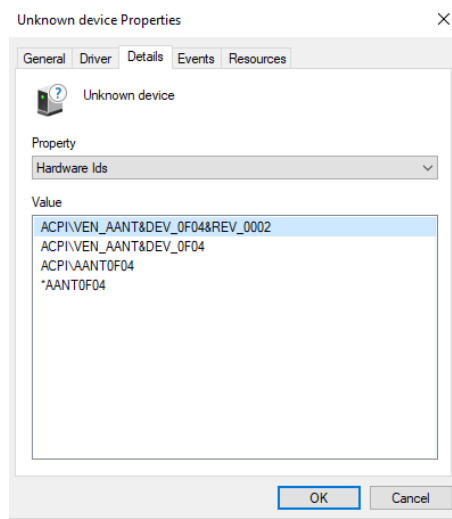
CRB Chipset > PCH-IO Configuration > HD Audio Configuration

Change "Audio DSP Compliance Mode" to [UAA (HAD Inbox/IntelSST)]

#### HD Audio Subsystem Configuration Settings

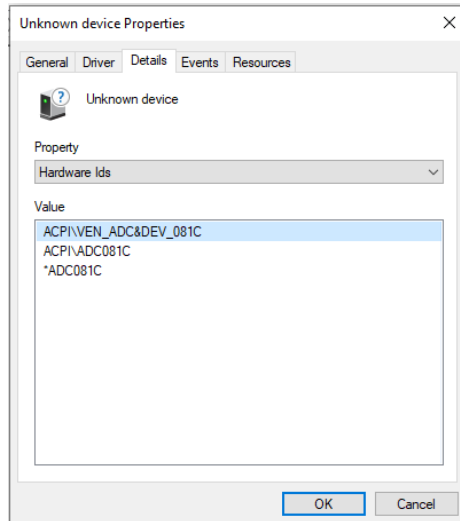
HD Audio	[Enabled]
Audio DSP	[Enabled]
Audio DSP Compliance Mode	[UAA (HDA Inbox/IntelSST)]

## Unknown Device 2: AANT0F04



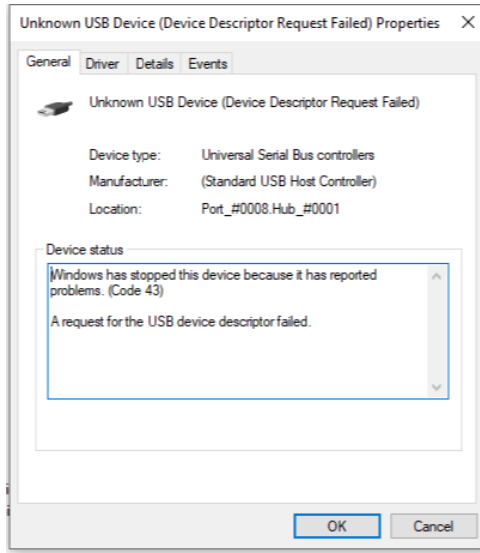
AANT0F04 is the 40pin HAT CPLD, there is no Windows driver for this device.

## Unknown Device 3: ADC081C



ADC081C is ADC, there is no Windows driver for this device

## Unknown USB Device



Some users may see an "Unknown USB Device (Device Descriptor Request Failed)" in Windows Device Manager. This is a common occurrence in Windows. The issue occurs because there is no corresponding device firmware or driver available for the USB port connecting to ST stm32f microcontroller (CN25). The device needs to be developed further by user or by project basis, otherwise the device is not supported on Windows.

# Appendix A

---

UP Framework SDK Installation

## A.1 Introduction

---

This section provides instructions for the installation of the UP Framework SDK. Instructions are provided for Windows 10 and Windows IoT Core. You can download the latest version of UP Framework SDK from the UP community:

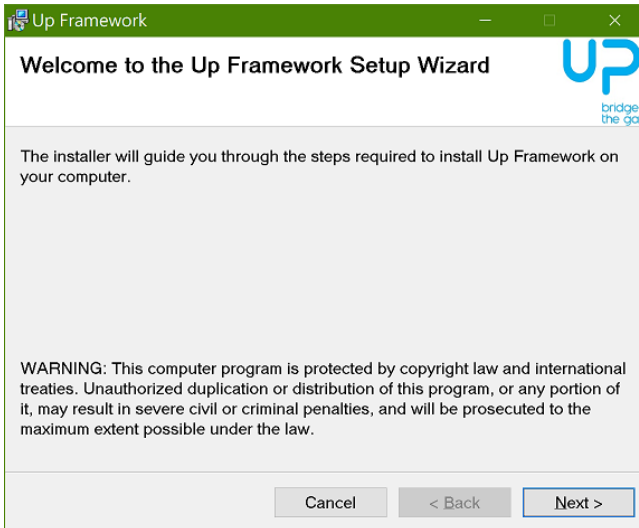
<https://downloads.up-community.org/download/up-sdk-for-windows-10-and-windows-iot/>

## A.2 Installation for Windows 10

---

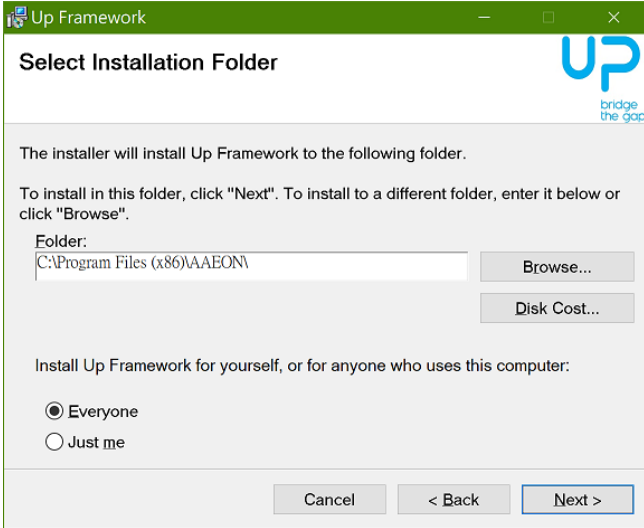
### Step 1

Locate the downloaded file UpFrameworkSetup.msi and run the installer. Press “Next” to begin the setup process.



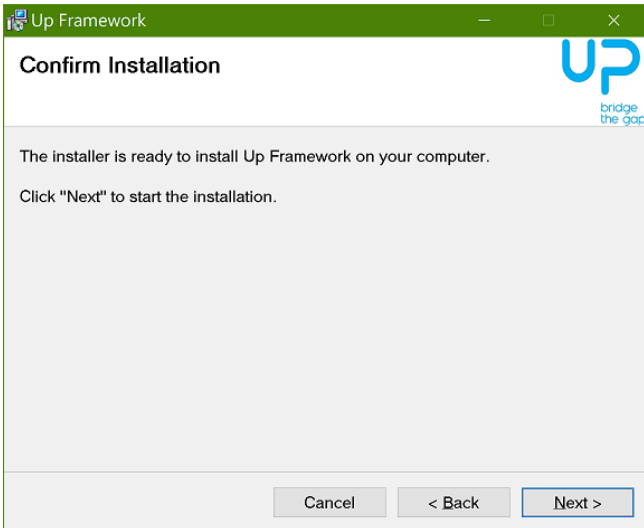
## Step 2

Select the installation folder. Default destination path is C:\Program Files(x86)\AAEON\  
You may also choose to install the UP Framework SDK for all users or only the current user. Press "Next" to continue installation.



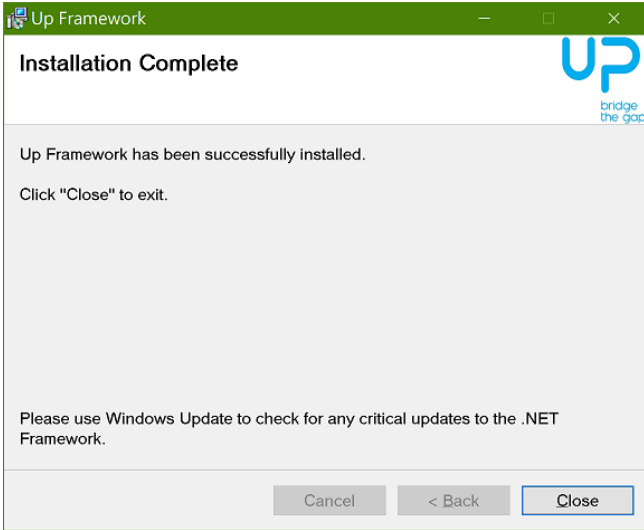
## Step 3

Press "Next" to confirm the installation.



## Step 4

Press "Close" to exit once setup is complete.





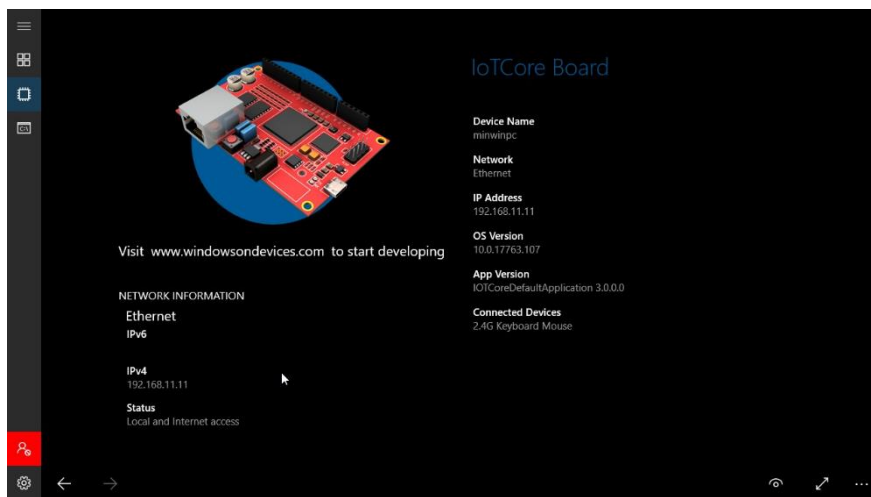
## A.3 Installation for Windows IoT Core

Before you begin, make sure you have downloaded and installed the latest version of the Windows IoT Core image from the UP community.

Installation requires using a connected PC with the UP Framework SDK software downloaded and saved. **Note:** Make sure the UP IoT Core device is connected to the same network as the PC you are using to install the software from.

### Step 1

Turn on your UP IoT Core device and note the IP address at the home screen.



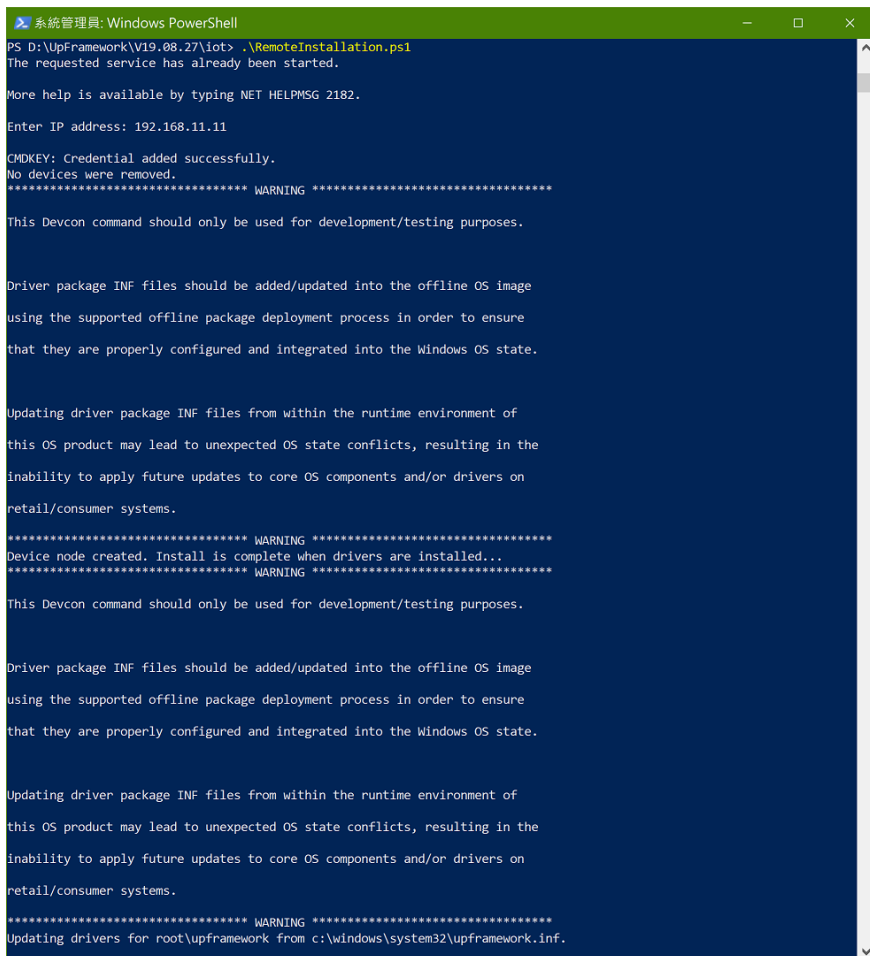
## Step 2

Download the UP Framework SDK to your PC and unzip the files.

Open PowerShell as an Administrator. Run the command

`RemoteInstallation.ps1` to install the UP Framework SDK.

Enter the IP address of the UP IoT Core device when prompted.



```
系統管理員: Windows PowerShell
PS D:\UpFramework\19.08.27\iot> .\RemoteInstallation.ps1
The requested service has already been started.

More help is available by typing NET HELPMSG 2182.

Enter IP address: 192.168.11.11

CMDKEY: Credential added successfully.
No devices were removed.
***** WARNING *****
This Devcon command should only be used for development/testing purposes.

Driver package INF files should be added/updated into the offline OS image
using the supported offline package deployment process in order to ensure
that they are properly configured and integrated into the Windows OS state.

Updating driver package INF files from within the runtime environment of
this OS product may lead to unexpected OS state conflicts, resulting in the
inability to apply future updates to core OS components and/or drivers on
retail/consumer systems.

***** WARNING *****
Device node created. Install is complete when drivers are installed...
***** WARNING *****
This Devcon command should only be used for development/testing purposes.

Driver package INF files should be added/updated into the offline OS image
using the supported offline package deployment process in order to ensure
that they are properly configured and integrated into the Windows OS state.

Updating driver package INF files from within the runtime environment of
this OS product may lead to unexpected OS state conflicts, resulting in the
inability to apply future updates to core OS components and/or drivers on
retail/consumer systems.

***** WARNING *****
Updating drivers for root\upframework from c:\windows\system32\upframework.inf.
```

# Appendix B

---

Cables and Connectors

## B.1 Cables and Connectors

This table provides detailed information about the cables and connectors used by the UP Xtreme (UP-WHL01). If you have any questions about the configuration of your board, please contact your AAEON sales representative.

Connector Label	Connector PN	Description	Mating Cable PN	Mating Cable Description
CN1	1655902034	RTC Battery Connector	175011301K	Lithium Battery.CR2032H.3V.240 mAH.w/cable 90mm. DIP.Battery power.BP-CR2032-M90-001
CN2	1654226303	mini-PCIe (PCIe x1)	N/A	
CN3	1654207533	m.2 2230 E Key	N/A	
CN4	165420753B	m.2 2280 M Key	N/A	
CN5	1654907009	SATA	N/A	
CN6	1655302025	SATA Power (DC 5V)	N/A	
CN7	1652814207	GbE RJ-45	N/A	
CN8	1652814207	GbE RJ-45	N/A	
CN10	1654403931	HDMI 2.0 + DP 1.2	N/A	
CN13	1654904130	eDP Connector	N/A	
CN11	1654010006	100-pin Docking Connector	N/A	
CN14	1654801832	2x USB 3.2 Gen2 (10Gbps)	N/A	
CN15	1654801832	2x USB 3.2 Gen2 (10Gbps)	N/A	
CN16	1655810131	10-pin USB 2.0+HSUART Connector	1709103100	(TF)WAFER Cable.AUDIO JACK.to 10Pin.310mm.Housing.fo r FWAA-1216

Connector Label	Connector PN	Description	Mating Cable PN	Mating Cable Description
CN19	165500401A	CPU Smart Fan Connector (3-pin)	175920001F	CPU Cooler.Fan+Heat Sink.12V.3P.for UPWH
CN20	165250320K	12 ~ 60 VDC Input (Lockable, OD/ID: 5.1/2.5 mm)	N/A	
CN22	165302020L	40-pin HAT Connector	1703401301	(TF)WIRE.16P.to 40Pin.130mm.Housing.fo r FWAA-1218
CN25	1655840030	STM32 I/O Header	1703401600	(TF)WIRE.16Pin.to 40 Pin.160mm.Housing.for FWAA-1217
CN26	1653215200	STM32 I/O Header	N/A	
CN27	165300210C	Reset Pin Header 2-pin	N/A	
COM1	1655901000	10-pin RS-232/422/485 Header	1701100180	COM Cable.D-SUB 9P(M).10P.1.0mm Housing.15cm
COM2	1655901000	10-pin RS-232/422/485 Header	1701100180	COM Cable.D-SUB 9P(M).10P.1.0mm Housing.15cm
SIM1	1654900800	Micro-SIM Card Slot	N/A	
SW2	1601615600	Power Button with LED	N/A	

# Appendix C

---

CEC-Client Installation

## C.1 CEC-Client Installation Manual

---

The CEC-Client Installation Manual is attached to this manual. It provides information for setting up the CEC-Client to allow the UPX-Edge system to communicate with CEC enabled devices through the HDMI port. If you have any questions, please contact your AAEON Sales Representative.

# CEC Client Installation

---

Supplemental Appendix for UPX-Edge



## Table of Contents

---

Chapter 1	Introduction.....	4
1.1	About this Document .....	4
Chapter 2	CEC-Client Installation.....	4
2.1	Introduction .....	4
2.2	Windows Installation .....	4
2.2.1	Kodi Installation on Windows.....	8
2.3	libCEC and USB-CEC Installation for Ubuntu on UPX-Edge .....	8
2.4	Flashing DFU Firmware on STM Board – Windows.....	9
2.5	Installation Procedure for DfuSe Demo.....	10
2.6	Installing dfu-util on Windows .....	15
2.7	Switching Device in DFU Mode .....	16
2.8	Flashing DFU firmware file on STM board on Linux.....	17
2.8.1	STM32f042k6.....	17
2.8.2	STM32f103c8.....	17
2.9	Flashing Firmware File on STM Board using STM32CubeProgrammer...17	
2.9.1	Installing STM32CubeProgrammer on Windows.....	18
2.9.2	Installing STM32CubeProgrammer on Linux.....	19
2.9.3	Flashing Bootloader or Combined Image .....	20
Chapter 3	API .....	22
Chapter 4	libCEC .....	25
4.1	libCEC-AAEON Adapter.....	25
4.2	Demo Application cec-client.exe .....	27
4.2.1	CEC-Client.....	27
4.2.2	Testing CEC-Client.....	28
Chapter 5	Supported Commands .....	29
5.1	Limitations.....	29
5.1.1	One Touch Play.....	34

5.1.2	Routing Control.....	35
5.1.3	System Standby.....	35
5.1.4	One Touch Record .....	35
5.1.5	Timer Programming.....	36
5.1.6	System Information .....	38
5.1.7	Deck Control.....	40
5.1.8	Tuner Control .....	41
5.1.9	Vendor Specific Commands.....	43
5.1.10	OSD Display.....	43
5.1.11	Device OSD Name Transfer .....	44
5.1.12	Device Menu Control.....	44
5.1.13	Remote Control Passthrough.....	45
5.1.14	Give Device Power Status .....	46
5.1.15	System Audio Control.....	46
5.1.16	Audio Rate Control.....	49

# Chapter 1 Introduction

---

## 1.1 About this Document

---

This document is included as a supplement to the product manual for the UPX-Edge. If you do not have the product manual for your device, please visit the product page on [AAEON.com](http://AAEON.com) to download the full version.

This document is intended for use only with the UPX-Edge product manual and is not for use as a separate document nor for use with any other product. If you have any questions regarding this document or your AAEON product, please contact your AAEON sales representative for assistance.

# Chapter 2 CEC-Client Installation

---

## 2.1 Introduction

---

CEC-Client is required to allow your PC to send custom CEC commands to a connected TV using a CEC adaptor. This section details installation instructions for CEC-Client.

## 2.2 Windows Installation

---

CEC-Client is an exe file included in the libCEC-AAEON library. Microsoft Visual Studio (for C++) is required to install and run CEC-Client on Windows. During installation, libCEC-AAEON Setup will check for Microsoft Visual Studio during installation. If Microsoft Visual Studio is not detected on your device, the required packages will be installed. Otherwise, only a test application will be installed.

Refer to the following images to perform setup and installation for Windows.

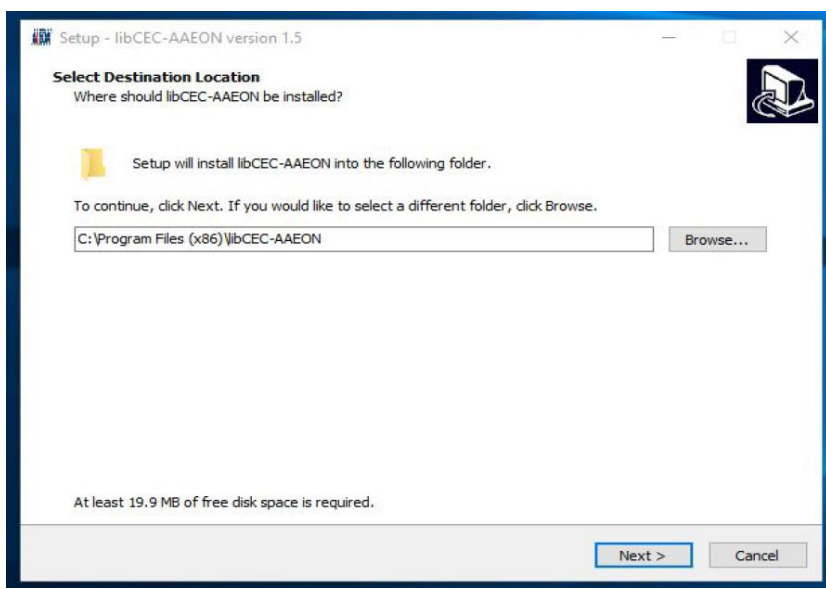


Image 1: Select Installation Path (default path shown)

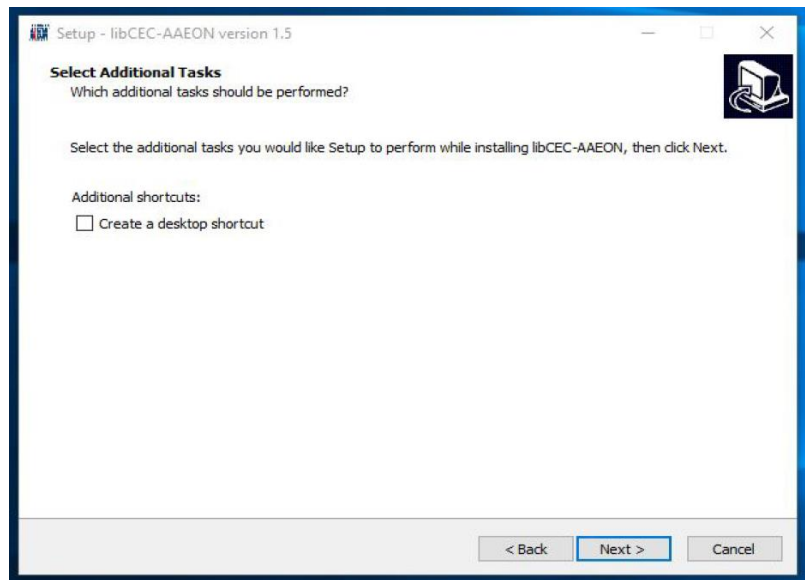


Image 2: Create Shortcut

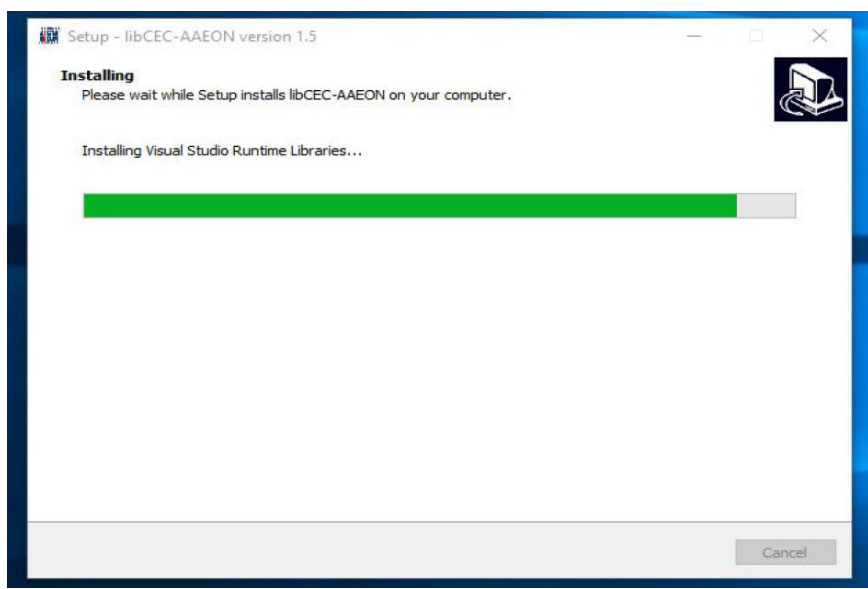


Image 3: Installing Visual Studio Runtime Libraries

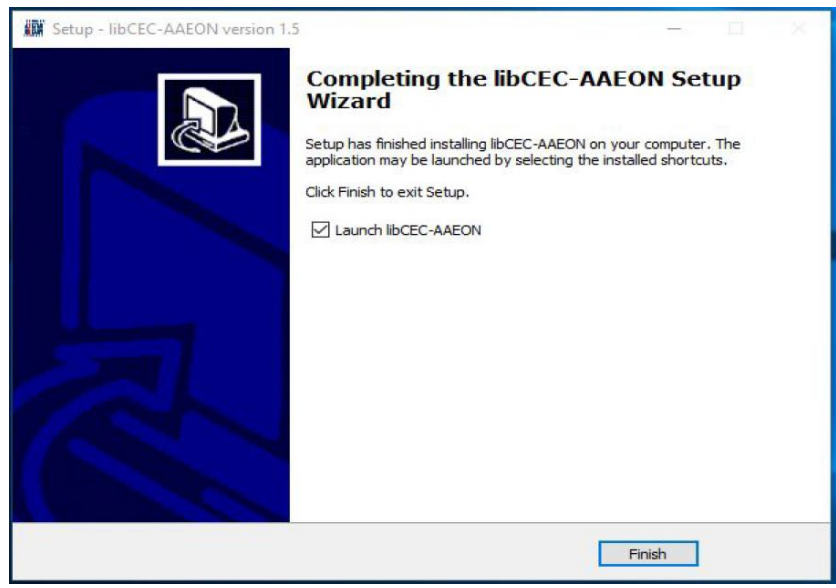


Image 4: Finish Installation

After installation is complete, open the folder location where you installed libCEC-AAEON. The default path location is shown in the image below.

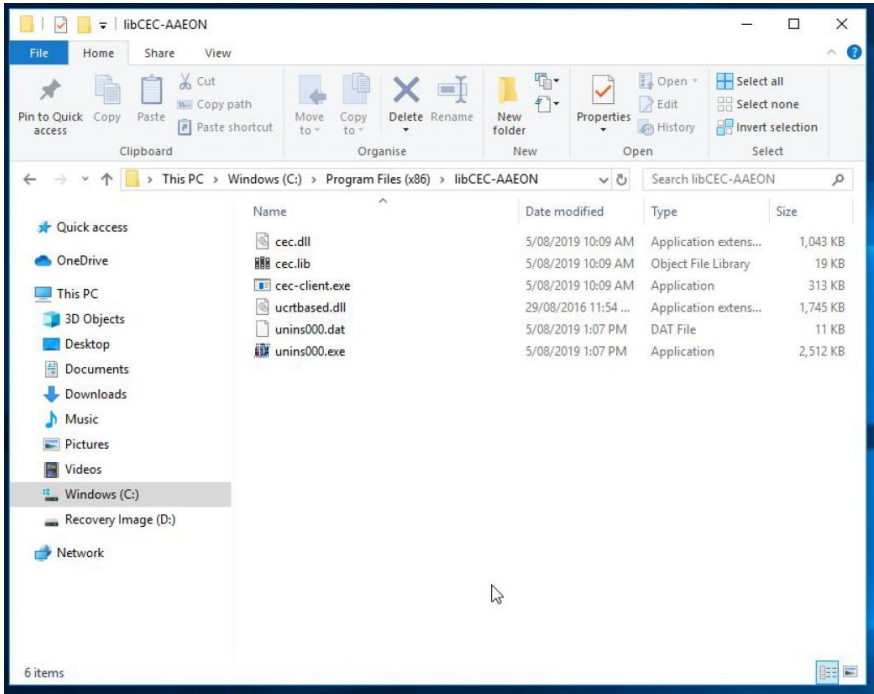


Image 5: CEC-Client File Location

Open a Command Prompt window in the file location. Run `cec-client.exe -h` to display all program options.

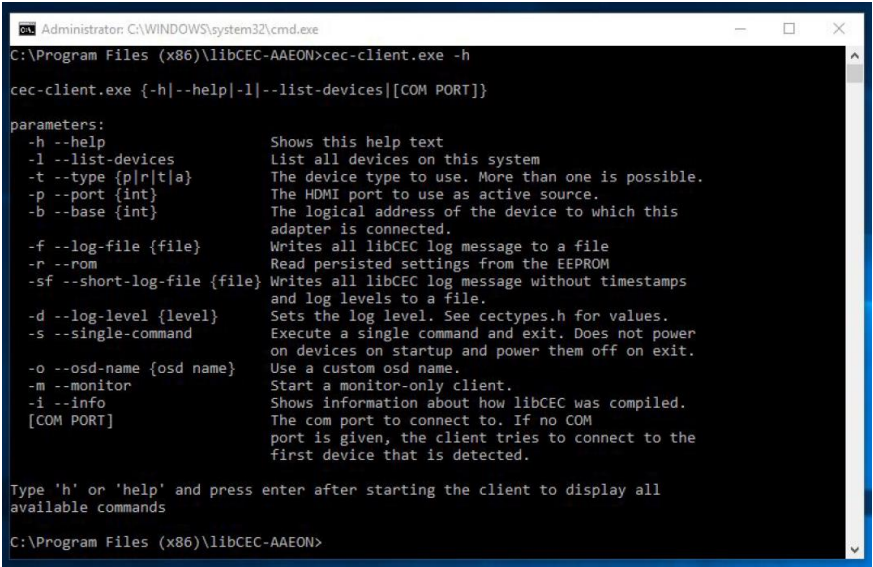


Image 6: CEC-Client Command Prompt

## 2.2.1 Kodi Installation on Windows

Before running Kodi on Windows, copy `cec.dll` from the CEC-Client folder (Image 5) to the Kodi installation folder. **Note:** You must have Kodi version 18.0 or later to use `libCEC-AAEON`.

## 2.3 libCEC and USB-CEC Installation for Ubuntu on UPX-Edge

Follow the instructions below to install CEC-Client and `libCEC-AAEON` library on Ubuntu.

CEC-Client and libCEC-AAEON require installing GDebi in order to run on Ubuntu. This step is performed in Terminal (Command Line). In Terminal, enter the command:

```
sudo apt-get install gdebi
```

After the process is completed, libCEC can be installed. Run the installer **libcec4\_4.0.4.1~bionic\_amd64.deb** from the deliverables folder. You may be required to enter the administrator password, after which installation will occur automatically.

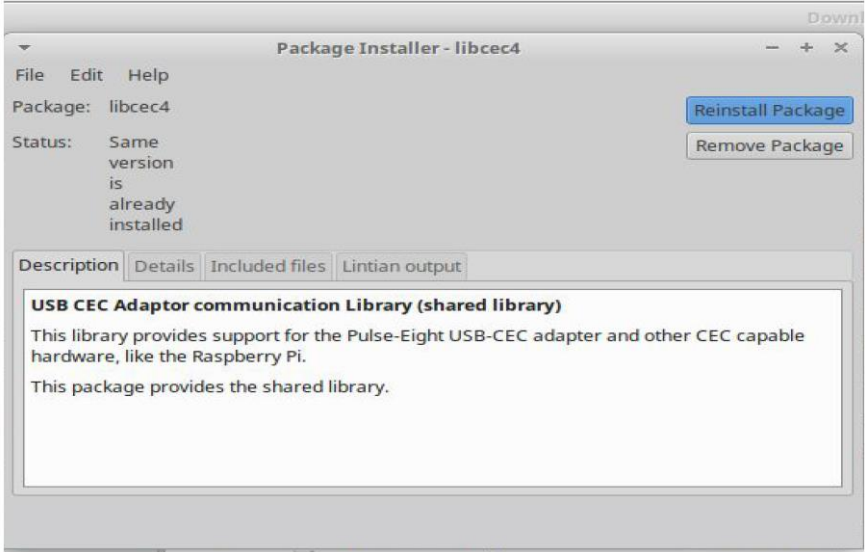


Image 7: Successful Installation

To use libCEC with Kodi, simply install Kodi with the command:

```
sudo apt-get install kodi
```

## 2.4 Flashing DFU Firmware on STM Board – Windows

This section details the procedures for dropping the .dfu (Device Firmware Update) firmware file on board and MCU's firmware update for delivered firmware on Windows platforms.



Download the .dfu file from the deliverables folder. Extract the files, then connect the board to the PC with a USB-C cable.

Flashing the firmware file on board requires it to be transferred using a DFU Bootloader program. The steps show below use **DfuSe USB Device Firmware Upgrade** to perform these actions.

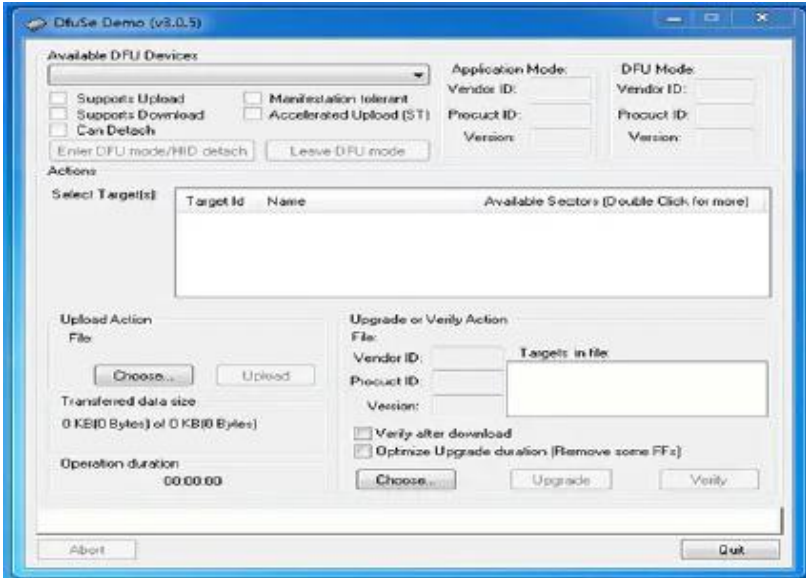


Image 8: DfuSe Utility Program

**Note:** Firmware can only be placed via the built-in bootloader on the STM32f042k6 board.

## 2.5 Installation Procedure for DfuSe Demo

This section details how to install DfuSe Demo. First, download and run the installer from <https://www.st.com/en/development-tools/stsw-stm32080.html>

InstallShield Wizard should start automatically and guide you through the installation procedure. When the software is successfully installed, click "Finish".

The DfuSe application drivers for Windows 10 should be installed automatically. If they are not, follow the procedures below to install DfuSe drivers. When starting driver installation for target device, the “Found New Hardware Wizards” should appear as shown:



*Image 9: New Hardware Wizard*

Choose “Install from a list or specific location (Advanced)” and then press “Next”.

In the next window, select "Don't search. I will choose the driver to install."

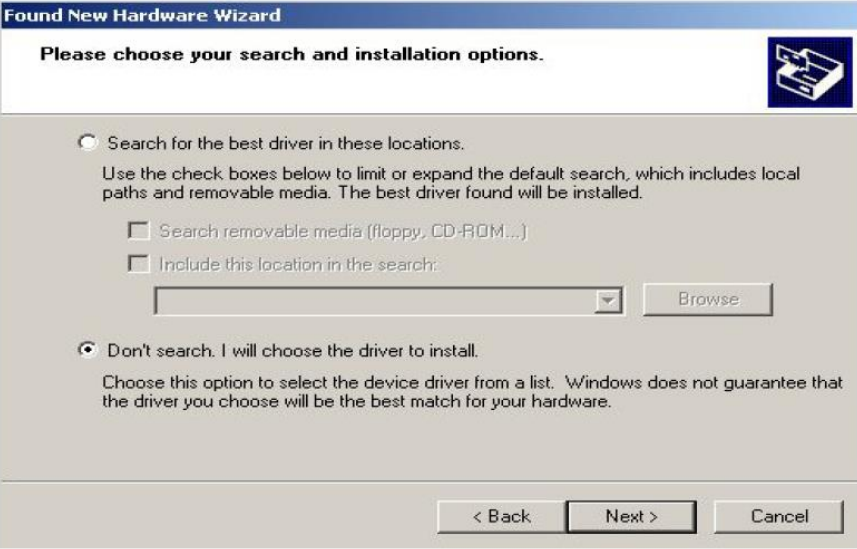


Image 10: Choose Install Options

If the DfuSe drivers are installed, you will see a list of compatible hardware models.

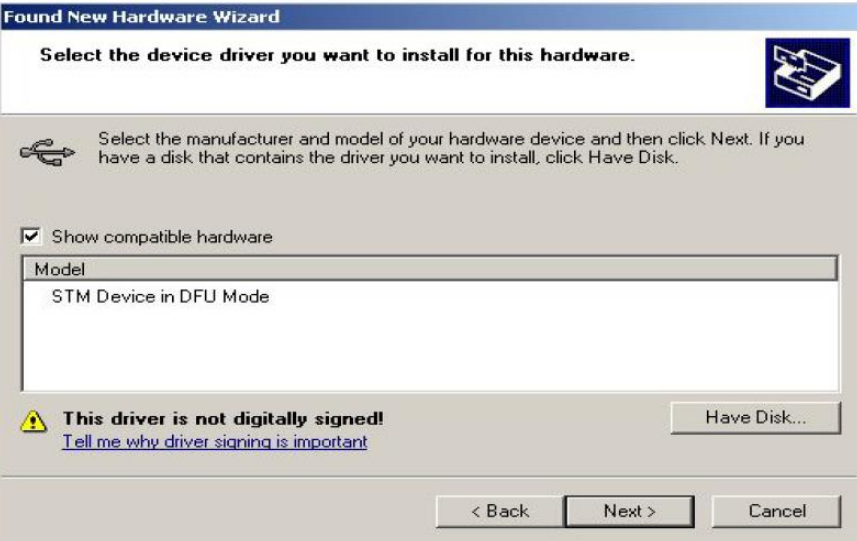


Image 11: DfuSe Drivers are installed

If there are no models listed, the drivers are not installed. Click on "Have Disk..." to select the drivers. In the Install from Disk dialog box, select browse and select the folder where the drivers are located and press "OK". Default location is **C:/Program Files/STMicroelectronics/DfuSe/Driver**

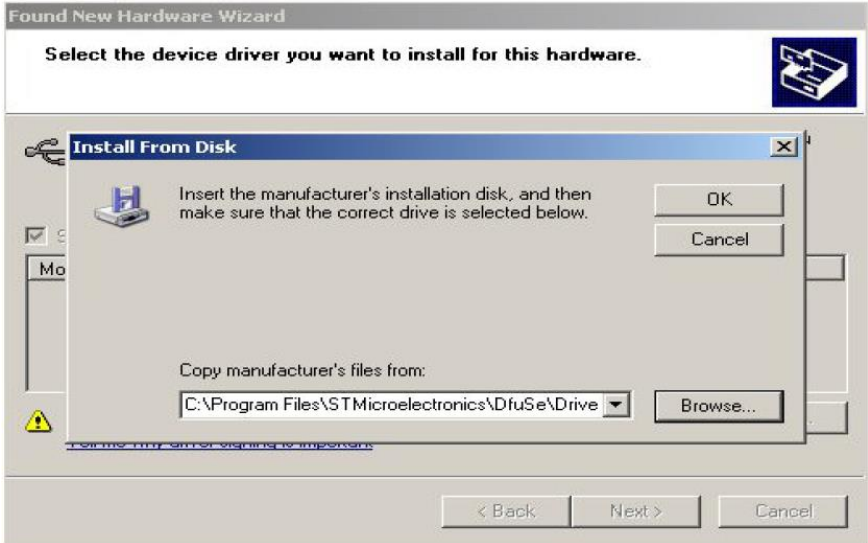


Image 12: Install from Disk

The installer will automatically find and select the .INF and the hardware model will be displayed on the list. Click "Next" to proceed.



Image 13: Installation

During installation, the following error message will appear. Click "Continue Anyway" to finish installing DfuSe drivers.



Image 14: Continue Anyway



Image 15: Installation Finished

DfuSe Installation will complete automatically. Click "Finish" once it is complete.

## 2.6 Installing dfu-util on Windows

This section details how to install dfu-util for programming the flash memory. It can be used to program both the Device OS and application firmware, as well as save and restore configurations. If you are using Windows, you can use Windows CLI Installer to automatically install dfu-util and the CLI. Otherwise, follow the instructions below.

First, download dfu-util from the release site:

<http://dfu-util.sourceforge.net/releases/dfu-util-0.9-win64.zip>

Extract the files. For this installation you will only need:

- dfu-prefix.exe
- dfu-suffix.exe
- dfu-util-static.exe

First, rename **dfu-util-static.exe** to **dfu-util.exe**. Next, create a location to save the dfu-util installation. The recommended location is **C:\Program Files\dfu-util**

Next, open Windows 10 Settings. In the top search field, type “environment” and select “Edit the system environment variables” when it appears.

Click the “**Environment Variables...**” button at the bottom of the page. In the Environment Variables window, select “**Path**” in the System Variables list (bottom list) and click “**Edit**”. In the Edit window, click “**New**” then add the file path **C:\Program Files\dfu-util** as a new row. Your system must be restarted after editing.

To test it has been completed, open a Command Prompt window and enter:

```
dfu-util -l
```

The descent command for STM32f042k6 firmware:

```
dfu-util -vvv -a 0 -e -t 2048 -s :force:mass-erase:leave -D  
STM32F042K6_Firmware.dfu
```

The descent command for STM32f103c8 firmware:

```
dfu-util -a 0 -D STM32F103C8_Firmware.bin
```

## 2.7 Switching Device in DFU Mode

---

To switch device to DFU mode, send the command “**bootloader**” through the CDC interface. The device should be restarted after this step.

Open DfuSe. The program will automatically detect STM32 board has been connected via USB to the PC and powered up in DFU mode.

In the “**Upgrade or Verify Action**” area, click on “**Choose**” and select the appropriate DFU firmware file which has been delivered, and then click on “**Upgrade**”.

DfuSe will automatically coordinate the firmware process until the entire firmware file has been successfully loaded onto the MCU. After uploading the DFU file to the device, the device will need to be reset.

## 2.8 Flashing DFU firmware file on STM board on Linux

---

### 2.8.1 STM32f042k6

---

This section details how to load the DFU firmware file onto STM32f042k6 MCU on Linux. First, install the required software tool from Terminal using the command:

```
sudo apt-get install dfu-util
```

After installing the dfu-util software tool, use the following procedure to conduct the firmware upgrade and load the upgraded firmware file using dfu-util onto the board's MCU on Linux software platform.

Switch the device to DFU Mode using the command "**bootloader**" sent through the CDC interface. Restart the device.

Use the following command to drop the DFU firmware file onto the STM board:

```
sudo dfu-util -vvv -a 0 -e -t 2048 -s :force:mass-erase:leave -D STM32F042K6_Firmware.dfu
```

After uploading the file, it is necessary to reset the device.

### 2.8.2 STM32f103c8

---

An ST-Link debugger needs to be put in place to lower the firmware. Firmware is loaded via the bootloader using the following command:

```
sudo dfu-util -a 0 -D stm32f103c8_firmware.bin
```

## 2.9 Flashing Firmware File on STM Board using STM32CubeProgrammer

---

STM32CubeProgrammer is an all-in-one multi-OS software tool for programming STM32 products. It provides an easy-to-use and efficient environment for reading, writing and verifying device memory through both the debug interface (JTAG and SWD) and the bootloader interface (UART, USB DFU, I2C, SPI, CAN).



The following software needs to be downloaded before beginning:

1. Official Java JRE from the Oracle website, version 8 or higher:

<https://www.oracle.com/technetwork/java/javase/downloads/jre8-downloads-2133155.html>

2. STM32CubeProgrammer from the official ST site:

<https://www.st.com/en/development-tools/stm32cubeprog.html>

3. For Windows, the driver for ST-Link is also needed:

<https://www.st.com/en/development-tools/stsw-link009.html>

## 2.9.1 Installing STM32CubeProgrammer on Windows

---

Install Java JRE:

1. Run setup program "jre-8u221-windows-x64.exe"
2. Follow on screen instructions provided by installer

Install driver for ST-Link:

1. Unpack archive "en.stsw-link009.zip"
2. Run setup program "stlink\_winusb\_install.bat"
3. Follow on screen instructions provided by installer

Install STM32CubeProgrammer:

1. Unpack archive "en.stm32cubeprog.zip"
2. Run setup program "SetupSTM32CubeProgrammer-2.1.0.exe"
3. Follow on screen instructions provided by installer

## 2.9.2 Installing STM32CubeProgrammer on Linux

---

Install Java Runtime Environment:

1. Unpack Java JRE archive into your home directory

```
tar -xf jre-8u221-linux-x64.tar.gz -C ~
```

2. Rename directory "jre1.8.0\_221" to "jre"

Install STM32CubeProgrammer:

1. Unpack archive "en.stm32cubeprog.zip" with command unzip:

```
unzip en.stm32cubeprog.zip
```

2. Set permissions for executing using command chmod:

```
chmod +x SetupSTM32CubeProgrammer-2.1.0.linux
```

3. Run installer using command:

```
JAVA_HOME=~/.jre ./SetupSTM32CubeProgrammer-2.1.0.linux
```

4. Set install directory to "~/STM32CubeProgrammer"

Configure STM32CubeProgrammer before launching application:

1. Enter STM32CubeProgrammer directory

```
cd ~/STM32CubeProgrammer
```

2. As sudo copy udev rules files to the "/dev/udev/rules.d" directory:

```
sudo cp Drivers/rules/*.rules /etc/udev/rules.d
```

3. Edit file `~/STM32CubeProgrammer/util/openJFXScript.csh`

Change line 4 from:

```
JAVA_PATH=`which java`
```

to

```
JAVA_PATH~/jre/bin/java
```

4. Edit file `~/STM32CubeProgrammer/bin/STM32CubeProgrammer`

Below first line:

```
#!/bin/bash
```

Insert the line:

```
export JAVA_HOME~/jre
```

### 2.9.3 Flashing Bootloader or Combined Image

---

The steps for using STM32CubeProgrammer to flash bootloader or combined image is the same for both Windows and Linux. To flash bootloader or combined image (bootloader and firmware in single image) follow these steps:

1. Select ST-LINK from drop-down menu in the top right corner of the window
2. Click refresh button below
3. Select ST-LINK by its serial number
4. Upgrade firmware on your ST-Link if necessary using "Firmware upgrade" button below
5. Connect ST-Link to the board
6. Click on the "Connect" button in the top right corner of the window
7. On the left side of the window click "Erasing & programming" (2nd icon)
8. Select your firmware using "Browse" button in the Download section
9. Click on "Verify programming" check box (optional)

10. Click on "Start Programming" button
11. Restart is needed after flashing combined image to be operative

**Note:** If you are unable to connect to the device, try changing ST-Link configuration option **Mode** from "Normal" to "Under reset".

The screenshot displays the STM32CubeProgrammer software interface. The main window is titled "Erasing & Programming".

**Download Section:**

- File path: /home/sasa/Desktop/stm32f10368\_usb\_bootloz
- Start addr.: 0x08000000
- Buttons:  Skip flash erase before programming,  Verify programming,  Run after programming,  Full chip erase,  Download file,  Option bytes commands: --op
- Buttons: Start Programming..., Start automatic mode

**Erase flash memory Section:**

Select	Index	Start Address	Size
<input type="checkbox"/>	0	0x08000000	1K
<input type="checkbox"/>	1	0x08000400	1K
<input type="checkbox"/>	2	0x08000800	1K
<input type="checkbox"/>	3	0x08000C00	1K
<input type="checkbox"/>	4	0x08001000	1K
<input type="checkbox"/>	5	0x08001400	1K
<input type="checkbox"/>	6	0x08001800	1K
<input type="checkbox"/>	7	0x08001C00	1K
<input type="checkbox"/>	8	0x08002000	1K
<input type="checkbox"/>	9	0x08002400	1K

**ST-LINK configuration Section:**

- Serial number: STM32F10368
- Port: SWD
- Frequency (kHz): 1700
- Mode: Normal
- Access port: 1
- Reset mode: No hardware reset
- Shared: Disabled
- External loader: --
- Target voltage: 0.44 V
- Firmware version: V2.2357
- Buttons: Hardware upgrade

**Log Section:**

```

13:56:55 - Size: 7972 Bytes
13:56:55 - Address: 0x08000000
13:56:55 - Erasing memory corresponding to segment 0:
13:56:55 - Erasing internal memory sectors [0 7]
13:56:55 - Download in Progress...
13:56:56 - File download complete
13:56:56 - Time elapsed during download operation: 00:00:00.562
13:56:56 - Verifying...
13:56:56 - Read progress...
13:56:56 - Download verified successfully
  
```

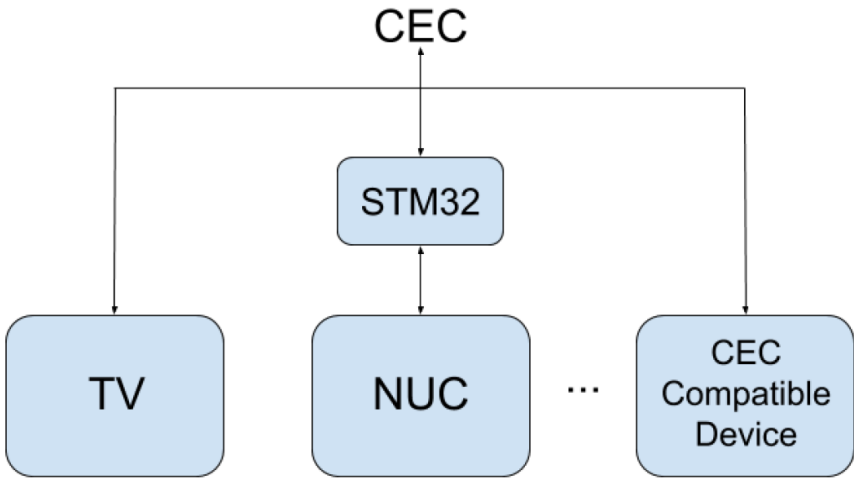


Image 17: CEC Network

STM32F10xxx device should be connected to Host CPU through USB (Communication Device Class) and HDMI CEC pin of baseboard. Firmware application should re-use already available CEC library from ST and provide adaptation layer which should enable access to CEC bus operation from USB host. The following functions should be exposed through USB API:

API Name	Description
HAL_CEC_Transmit_IT	Sends a CEC data in interrupt mode
HAL_CEC_TxCpltCallback	Tx Transfer completed callback
HAL_CEC_RxCpltCallback	Rx Transfer completed callback
HAL_CEC_ErrorCallback	CEC error callback

STM32 device should be detected on USB host as a USB CDC serial device. Each of the API functions and input/output arguments should be serialized in the following way:

- Sending a cec command is executed by sending a string in the following format via CDC:

```
send <address> <hex-string>
```

- If the command is sent successfully, the string **ACK** will be returned via CDC
- Received messages will be displayed as:

```
CEC: <hex-string>
```

- In case of error, string will be received in format:

```
ERR: <error-code>
```

All messages end with **<CR>** **<LF>** characters.

Error messages are generally received if the message was unable to be delivered on device, or if the message was sent in the wrong format.

#### Example:

Getting CEC version from the TV:

Terminal	Description
> <b>send 0 9F</b>	Send CEC get version command to address 0
<b>ACK</b>	The command was successfully sent
<b>CEC: 019E05</b>	Received from 0 to device 1: cec version 1.4

Getting power status from device 2 that is not present on the CEC network:

Terminal	Description
> <b>send 2 8F</b>	Send CEC get power status to address 2
<b>ERR: 0x00001000</b>	Sending failed: acknowledgement is not received

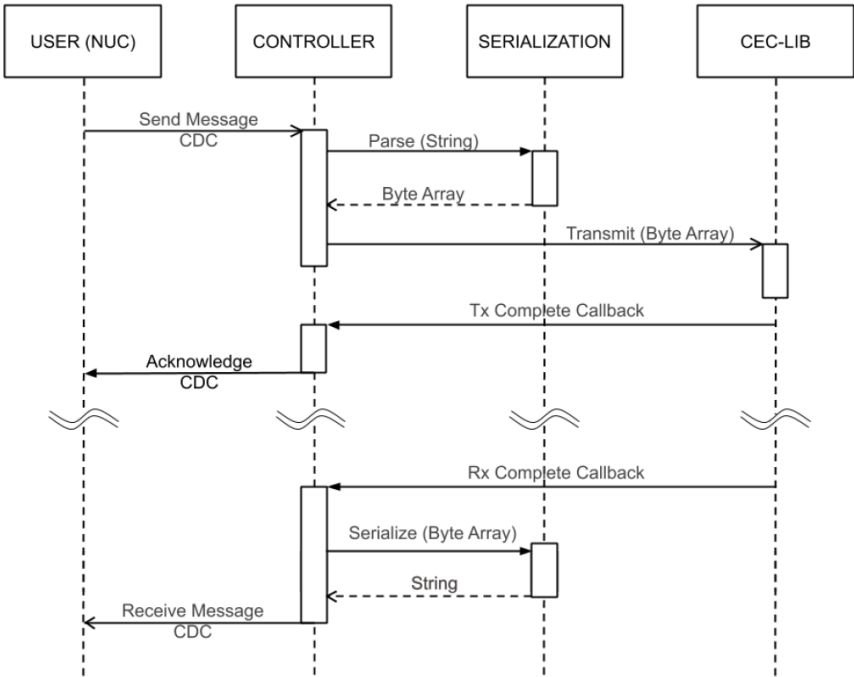


Image 18: Firmware Sending/Receiving Messages

Firmware Command	Description
<b>help</b>	View list of available commands
<b>get &lt;name&gt;</b>	Returns value of variable with specified name.†
<b>set &lt;name&gt; [value]</b>	Set value of variable.‡
<b>send &lt;address&gt; [data]</b>	Send data over CEC network to specified address.
<b>bootloader</b>	Call STM bootloader and switch device to DFU mode.

† Supported names are **logical\_address**, **physical\_address**, **firmware\_version**, **cec\_version**

‡ Currently supported names are **logical\_address** and **physical\_address**.

# Chapter 4 libCEC

---

CEC (Consumer Electronics Control) allows for control of devices through the HDMI port. With the right hardware, libCEC allows control of the device with a TV remote. libCEC is an enabling platform for the CEC bus in HDMI. It allows developers to interact with other HDMI devices without having to worry about communication overhead, handshaking, and various ways of sending a message for each vendor. libCEC communicates with all CEC Bridge branded hardware. libCEC supports Linux and Windows platforms.

## 4.1 libCEC-AAEON Adapter

---

libCEC-AAEON Adapter was developed for use with the GitHub download of libCEC (3.1.3 libCEC library) ...\\libcec\src\libcec\adapter

To add a new adapter, it is important to implement the **IAdapterCommunication** interface to communicate with the rest of the CEC library using the **IAdapterCommunicationCallback** interface. This is also necessary for detecting the device for which the new adapter is written. Four new classes will be added to the existing implementation of the libCEC library.

- **CAAEONAdapterCommunication** class is the implementation of the IAdapterCommunication interface. It is a central communication class with an adapter.
- **CAAEONAdapterDetection** class is a class that serves to find the port on which the device is located. It finds COM port based on the VID and PID in Windows through a registry, while on Linux it uses the UDEV library. In the name of the port, the prefix **AAEON:** will be added to differentiate the detection of a new type of device.
- **CAAEONAdapterMessageQueue** is a class that collects CEC messages that need to be sent to the device. It also processes the results that the device returns and manages messages based on it.
- **CAAEONAdapterMessage** is a class that represents a CEC message that contains message data and state, depending on the type of message.



- **CAAEONAdapterCommands** is a class that is used to send and receive data from the CDC, as well as for parsing and serializing data between firmware and libCEC format.

libCEC Driver for Windows is the standard driver for serial communication included with the Windows installation. For using the device, it will use detection from libCEC, which depends on Windows registers.

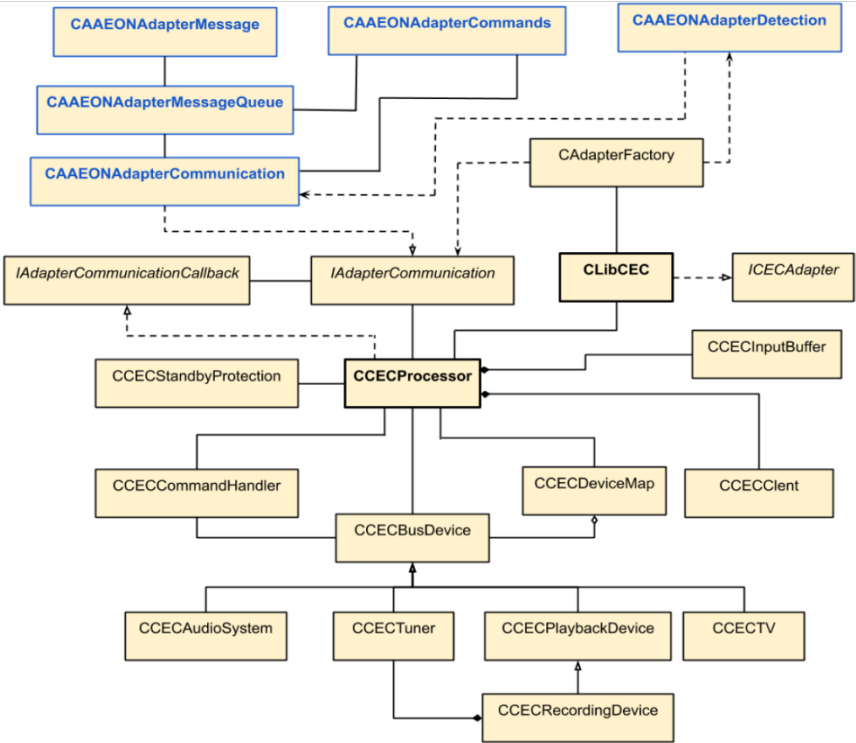


Image 19: libCEC Architecture

libCEC exports an **ICECAdapter** interface that can be used by various clients, such as **cec-client**, **Kodi**, etc. It contains methods such as: open port, transmit CEC message, set logical address, power on CEC devices, get device menu language, etc.

## 4.2 Demo Application cec-client.exe

To send custom CEC commands from PC to TV through libCEC-AAEON adapter, you will need to install the cec-client. This is an .exe file that is included in the libCEC and Windows driver package. A new CEC-Client was made with libCEC-AAEON adapter and compiled libCEC in which a new adapter was added (3.2.1 Compile libCEC)

### 4.2.1 CEC-Client

Cec-client is a test client/demo application, used to debug the device. To start the demo application and check whether the device can be detected, execute the command **"cec-client.exe"** in libcec\build\amd64 or run the installed application.

```

Administrator: C:\WINDOWS\system32\cmd.exe - cec-client.exe
No device type given. Using 'recording device'
CEC Parser created - libCEC version 4.0.4
no serial port given. trying autodetect:
path:      USB\VID_0483&PID_5740\318436634248
com port:  AAEON:COM31

opening a connection to the CEC adapter...
DEBUG: [      29] Broadcast (F): osd name set to 'Broadcast'
DEBUG: [      29] PORT: AAEON:COM31 SIZE: 6
DEBUG: [      29] ENTER: CAEONAdapterCommands::Open
DEBUG: [      81] OPEN: COM31
DEBUG: [      81] ENTER: CAEONAdapterMessageQueue::Process
NOTICE: [      81] connection opened
DEBUG: [      81] processor thread started
DEBUG: [      81] << Broadcast (F) -> TV (0): POLL
TRAFFIC: [     82] << f0
DEBUG: [      82] PUSH: send 0

DEBUG: [      96] ENTER: CAEONAdapterCommands::Write
DEBUG: [      96] SENT: send 0

DEBUG: [     111] RECEIVED: 5 bytes
DEBUG: [     112] ACK: ACK
DEBUG: [     127] QUEUE: Size=1, REQ[0]=send 0 , RES[0]=ACK
DEBUG: [     143] >> POLL sent
DEBUG: [     143] TV (0): device status changed into 'present'
DEBUG: [     143] << requesting vendor ID of 'TV' (0)
TRAFFIC: [     143] << f0:8c
DEBUG: [     143] PUSH: send 0 8C
  
```

Image 20: Run CEC-Client

## 4.2.2 Testing CEC-Client

---

CEC-Client must be run from Terminal. Type **cec-client.exe** to run it. After CEC-Client starts, it will open CEC-Client Terminal Application. From CEC-Client Terminal Application, you can send commands for testing. Some CEC-Client commands for testing libCEC are listed in the following table.

Command	Description
<b>tx 10:44:41</b>	TV change volume – volume up
<b>on 0</b>	Turn on TV
<b>standby 0</b>	Put TV in standby mode

# Chapter 5 Supported Commands

---

This section describes the message transfer and additional details for a number of common features enabled by CEC. Note that where a feature is supported, all messages within that feature should be implemented.

## 5.1 Limitations

---

### Firmware Limitations:

- We cannot guarantee that commands will be processed if sent at intervals less than 240ms.
- After calling the bootloader, the device needs to be reset. The device also needs to be reset after uploaded the DFU file to the device, after which the firmware device will be operational.
- It is necessary to disconnect ST-Link before using CEC-Client or Kodi.
- Suspend from Kodi is not supported, because TV changes its address to a non-zero number after resume. If this happens, the TV needs to be turned off and turned back on to assign itself a proper address.

### libCEC Limitations:

- Some features in libCEC have not been implemented by different brands and/or are not consistently supported across models. The following table highlights some limitations. The list is not final and may vary to support as many vendors as possible.
- Since there is no physical address detection, libCEC will only work on HDMI port 1
- Kodi is only supported with libCEC4.04.1

The tables on the following page detail support limitations. Fields marked with "-" may work but have not been verified.

	One Touch Play	Routing Control	Standby	One Touch Record	Timer Programming	System Information	Deck Control	Tuner Control	OSD String Display	Device OSD Name Transfer
Aikai	-	-	-	-	-	-	-	-	-	-
AOC	-	-	-	-	-	-	-	-	-	-
Beng	-	-	-	-	-	-	-	-	-	-
Daewoo	-	-	-	-	-	-	-	-	-	-
Grundig	-	-	-	-	-	-	-	-	-	-
Hitachi	-	-	-	-	-	-	-	-	-	-
LG	Yes	Yes	Yes†	-	-	-	Yes	-	No	No
Loewe	-	-	-	-	-	-	-	-	-	-
Marantz	-	Yes	-	-	-	-	-	-	-	-
Medion	-	-	-	-	-	-	-	-	-	-
Mitsubishi	-	-	-	-	-	-	-	-	-	-
Onkyo	Yes	Yes	Yes	-	-	-	-	-	No	-
Panasoni	Yes	Yes	Yes	-	-	-	-	-	-	No

	Device Menu Control	Remote Control Passthrough	Power Status	System Audio Control
Akai	-	-	-	-
AOC	-	-	-	-
Benq	-	-	-	-
Daewoo	-	-	-	-
Grundig	-	-	-	-
Hitachi	-	-	-	-
LG	-	Yes	-	-
Loewe	-	-	-	-
Marantz	-	-	-	-
Medion	-	-	-	-
Mitsubish	-	-	-	-
Onkyo	-	Yes	-	Yes
Panasoni	-	Yes	-	-

† Supported by all devices except TVs.

	One Touch Play	Routing Control	Standby	One Touch Record	Timer Programming	System Information	Deck Control	Tuner Control	OSD String Display	Device OSD Name Transfer
Philips	Yes	Yes	Yes	No	No	Yes	No	No	No	Yes
Pioneer	-	-	-	-	-	-	-	-	-	-
Runco	-	-	-	-	-	-	-	-	-	-
Samsung	Yes	Yes	Yes†	-	-	-	Yes	-	No	Yes
Sharp	Yes	Yes	Yes	-	-	-	-	-	-	-
Sony	Yes	Yes	Yes	-	-	-	-	-	-	-
Toshiba	Yes	Yes	Yes	-	-	-	-	-	-	-
Vizio	Yes	Yes	Yes	-	-	-	-	-	-	-
Yamaha	-	-	Yes	-	-	-	-	-	-	-

	Device Menu Control	Remote Control Passthrough	Power Status	System Audio Control
Philips	-	Yes	Yes	Yes
Pioneer	-	-	-	-
Runco	-	-	-	-
Samsung	-	-	-	-
Sharp	-	-	-	-
Sony	-	-	-	-
Toshiba	-	-	-	-
Vizio	-	-	-	-
Yamaha	-	-	-	Yes

† Some models require you to turn on “Auto-Standby” in the Anynet+ settings to enable this feature.



## 5.1.1 One Touch Play

**Command Feature:** The One Touch Play feature allows a device to be played and become the active source with a single button press.

The following messages are used for the One Touch Play feature: <Active Source>, <Image View On>, <Text View On>.

OpCode	Value	Description	Parameters	Response	Supported
<Active Source>	0x82	Used by a new source to indicate that it has started to transmit a stream OR used in response to a <Request Active Source>	[Physical Address]	A current active source should take appropriate Action. TV should switch to the appropriate input. Any CEC switches between source and root shall switch to the appropriate input and come out of standby if necessary.	Yes*
<Image View On>	0x04	Sent by a source device to the TV whenever it enters the active state (alternatively it may send <Text View On>).	None	Turn on (if not on). If in 'Text Display' state than the TV enters 'Image Display' state. Note: Should not change TV menu or PIP status.	Yes
<Text View On>	0x0D	As <Image View On>, but should also remove any text, menus and PIP windows from the TV's display.	None	As <Image View On>, but should remove PIPs and menus from the screen. The TV enters 'Image Display' state regardless of its previous state.	Yes

\*See Chapter 5.1 for limitations.

## 5.1.2 Routing Control

**Command Feature:** This feature is used to control the routing of the HDMI network by controlling CEC Switches.

The following messages are used for the Routing Control feature: <Active Source>, <Inactive Source>, <Request Active Source>, <Set Stream Path>, <Routing Change>, <Routing Information>.

## 5.1.3 System Standby

**Command Feature:** The broadcast message <Standby> can be used to switch all CEC devices to standby. The following message is used for the System Standby feature: <Standby>

OpCode	Value	Description	Parameters	Response	Supported
<Standby>	0x36	Switches one or all devices into standby mode. Can be used as a broadcast message or be addressed to a specific device.	None	Switch the device into Standby. Ignore the message if already in standby.	Yes

## 5.1.4 One Touch Record

**Command Feature:** This feature allows the user to easily start a recording of the source that is being displayed on the TV, just by selecting a Recording Device and giving the record command.

The following messages are used for the One Touch Record feature: <Record Off>, <Record On>, <Record Status>, <Record TV Screen>.

OpCode	Value	Description	Parameters	Response	Supported
<Record Off>	0x0B	Requests a device to stop a recording.	None	Exit 'Recording' state.	Yes

*Table Continues on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<Record On>	0x09	Attempt to record the specified source.	[Record Source]	Enter 'Recording' state and start recording if possible. Send the initiator <Record Status>.	Yes
<Record Status>	0x0A	Used by a Recording Device to inform the initiator of the message <Record On> about its status.	[Record Status Info]		Yes

### 5.1.5 Timer Programming

**Command Feature:** This feature allows a device (e.g. TV) to set a timer recording on a Recording Device. For example, it can be used to set timer blocks of a Recording Device via a TV menu or via an EPG.

The following messages are used for the Timer Programming feature: <Clear Analogue Timer>, <Clear Digital Timer>, <Clear External Timer>, <Set Analogue Timer>, <Set Digital Timer>, <Set External Timer>, <Set Timer Program Title>, <Timer Cleared Status>, <Timer Status>.

OpCode	Value	Description	Parameters	Response	Supported
<Clear Analogue Timer>	0x33	Used to clear an Analogue timer block of a device.	See <Set Analogue Timer> message.	Clear timer block if possible, then respond with <Timer Cleared Status>	Yes
<Clear Digital Timer>	0x99	Used to clear a Digital timer block of a device.	See <Set Digital Timer> message	Clear timer block if possible, then respond with <Timer Cleared Status> message	Yes
<Clear External Timer>	0xA1	Used to clear an External timer block of a device	See <Set External Timer> message	Clear timer block if possible, then respond with <Timer Cleared Status> message	Yes

*Table Continues on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<b>&lt;Set Analogue Timer&gt;</b>	0x34	Used to set a single timer block on an Analogue Recording Device.	[Day of Month] [Month of Year] [Start Time] [Duration] [Recording Sequence] [Analogue Broadcast Type] [Analogue Frequency] [Broadcast System]	<Timer Status> message	Yes
<b>&lt;Set Digital Timer&gt;</b>	0x97	Used to set a single timer block on a Digital Recording Device.	[Day of Month] [Month of Year] [Start Time] [Duration] [Recording Sequence] [Digital Service Identification]	<Timer Status> message	Yes
<b>&lt;Set External Timer&gt;</b>	0xA2	Used to set a single timer block to record from an external device.	[Day of Month] [Month of Year] [Start Time] [Duration] [Recording Sequence] [External Source Specifier] [External Plug] [External Physical Address]	<Timer Status> message	Yes

Table Continues on Next Page

OpCode	Value	Description	Parameters	Response	Supported
<Set Timer Program Title>	0x67	Used to set the name of a program associated with a timer block. Sent directly after sending a <Set Analogue Timer> or <Set Digital Timer> message. The name is then associated with that timer block.	[Program Title String]	Recording device stores title for future reference. Ignore message if it is not the immediate next message from this initiator following a <Set Analogue Timer> or <Set Digital Timer> message.	Yes
<Timer Cleared Status>	0x43	Used to give the status of a <Clear Analogue Timer>, <Clear Digital Timer> or <Clear External Timer> message	[Timer Cleared Status Data]	If the message indicates that the timer was not cleared because there was no matching entry, the device should remove the timer block locally.	Yes
<Timer Status>	0x35	Used to send timer status to the initiator of a <Set Timer> msg	[Timer Status Data]	None	Yes

## 5.1.6 System Information

**Command Feature:** This feature allows devices to automatically use the same OSD and Menu language settings as the TV and also for a TV to discover the current language when it is being installed.

The following messages are used for the System Information feature: <CEC Version>, <Get CEC Version>, <Get Menu Language>, <Give Physical Address>, <Polling Message>, <Report Physical Address>, <Set Menu Language>.

*Table on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<CEC Version>	0x9E	Used to indicate the supported CEC version, in response to a <Get CEC Version>	[CEC Version]		Yes
<Get CEC Version>	0x9F	Used by a device to enquire which version of CEC the target supports	None	The source responds with a <CEC Version> Message indicating the CEC version	Yes
<Give Physical Address>	0x83	A request to a device to return its physical address.	None	<Report Physical Address>	Yes
<Get Menu Language>	0x91	Sent by a device capable of character generation (for OSD and Menus) to a TV in order to discover the currently selected Menu language. Also used by a TV during installation to discover the currently set menu language of other devices.	None	The addressed device responds with a <Set Menu Language> message	Yes
<Polling Message>	—	Used by any device for device discovery – similar to ping in other protocols.	None	Shall set a low level ACK.	Yes
<Report Physical Address>	0x84	Used to inform all other devices of the mapping between physical and logical address of the initiator.	[Physical Address] [Device Type]		Yes
<Set Menu Language>	0x32	Used by a TV or another device to indicate the menu language.	[Language]	Set the menu language as specified, if possible.	Yes

## 5.1.7 Deck Control

**Command Feature:** This feature allows a Playback Device (a deck or disc player or recorder) to be controlled by another device (e.g. TV).

The following messages are used for the Deck Control feature: <Deck Status>, <Give Deck Status>, <Deck Control>, <Play>.

OpCode	Value	Description	Parameters	Response	Supported
<Deck Control>	0x42	Used to control a device's media functions.	[Deck Control Mode]	Perform the specified actions, or return a <Feature Abort> message. It is device dependent whether or not a Skip Forward/Wind or Skip Backward /Rewind command is legal when in the 'Deck Inactive' state. If the device is in standby and receives an eject command, it should power on and eject its media.	Yes
<Give Deck Status>	0x1A	Used to request the status of a device, regardless of whether or not it is the current active source.	[Status Request]	<Deck Status>	Yes
<Deck Status>	0x1B	Used to provide a deck's status to the initiator of the <Give Deck Status> msg	[Deck Info]		Yes

*Table Continues on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<Play>	0x41	Used to control the playback behavior of a source device.	[Play Mode]	Perform the specified actions, or return a <Feature Abort> message. If media is available the device enters 'Deck Active' state. If the device is in standby, has media available and the parameter is ["Play Forward"] it should power on.	Yes

### 5.1.8 Tuner Control

**Command Feature:** This feature allows a device (e.g. TV) to control another CEC device's tuner.

The following messages are used for the Tuner Control feature: <Give Tuner Device Status>, <Record On>, <Select Analogue Service>, <Select Digital Service>, <Tuner Step Decrement>, <Tuner Step Increment>, <Tuner Device Status>.

OpCode	Value	Description	Parameters	Response	Supported
<Give Tuner Device Status>	0x08	Used to request the status of a tuner device.	[Status Request]	Respond with a <Tuner Device Status> message, or stop reporting changes on receipt of the["Off"] message.	Yes
<Record On>	0x09	Attempt to record the specified source.	[Record Source]	Enter 'Recording' state and start recording if possible. Send the initiator <Record Status>.	Yes

*Table Continues on Next Page*



OpCode	Value	Description	Parameters	Response	Supported
<Select Analogue Service>	0x92	Directly selects an Analogue TV service	[Analogue Broadcast Type] [Analogue Frequency] [Broadcast System]	Change to the selected analogue service and stream its output on the HDMI connection. If the tuner device is not capable of selecting this service, respond with a <Feature Abort>	Yes
<Select Digital Service>	0x93	Directly selects a Digital TV, Radio or Data Broadcast Service	[Digital Service Identification]	Change to the selected digital service and stream its output on the HDMI connection. If the tuner device is not capable of selecting this service, respond with a <Feature Abort>	Yes
<Tuner Step Decrement>	0x06	Used to tune to next lowest service in a tuner's service list. Can be used for PIP.	None	Follower tunes to next lowest service in its service list.	Yes
<Tuner Step Increment>	0x05	Used to tune to the next highest service in a tuner's service list. Can be used for PIP.	None	Follower tunes to next highest service in its service list.	Yes
<Tuner Device Status>	0x07	Use by a tuner device to provide its status to the initiator of the <Give Tuner Device Status> message.	[Tuner Device Info]		Yes

## 5.1.9 Vendor Specific Commands

**Command Feature:** This feature allows a set of vendor specific commands to be used to communicate between devices.

The following messages are used for the Vendor Specific Commands feature: <Device Vendor ID>, <Give Device Vendor ID>, <Vendor Command>, <Vendor Command With ID>, <Vendor Remote Button Down>, <Vendor Remote Button Up>

OpCode	Value	Description	Parameters	Response	Supported
<Device Vendor ID>	0x87	Reports the vendor ID of this device.	[Vendor ID]	Any other interested device may store the vendor ID of the device.	Yes
<Give Device Vendor ID>	0x8C	Requests the Vendor ID from a device.	None	<Device Vendor ID>	Yes
<Vendor Command>	0x89	Allows vendor specific commands to be sent between two devices.	[Vendor Specific Data]	Vendor Specific	Yes
<Vendor Command With ID>	0xA0	Allows vendor specific commands to be sent between two devices or broadcast.	[Vendor ID] [Vendor Specific data]	Vendor Specific	Yes
<Vendor Remote Button Down>	0x8A	Indicates that a remote control button has been depressed.	[Vendor Specific RC Code]	Vendor Specific	Yes

## 5.1.10 OSD Display

**Command Feature:** This feature allows a device to transfer a text string to the TV for On Screen Display. The <Set OSD String> message is used to transfer the text string to the TV.

The following message is used for the OSD Display feature: <Set OSD String>

*Table on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<Set OSD String>	0x64	Used to send a text message to output on a TV.	[Display Control] [OSD String]	TV displays the message.	Yes

### 5.1.11 Device OSD Name Transfer

**Command Feature:** This feature is used to request the preferred name of a device to be used in any on screen display (e.g. menus), which reference that device.

The following messages are used for the Device OSD Name Transfer feature: <Give OSD Name>, <Set OSD Name>

OpCode	Value	Description	Parameters	Response	Supported
<Give OSD Name>	0x46	Used to request the preferred OSD name of a device for use in menus associated with that device.	None	<Set OSD Name>	Yes
<Set OSD Name>	0x47	Used to set the preferred OSD name of a device for use in menus associated with that device.	[OSD Name]	Store the name and use it in any menus associated with that device.	Yes

### 5.1.12 Device Menu Control

**Command Feature:** This feature allows device menus to be controlled via the TV remote control as if it was using its own remote control, and allow the TV to be aware when another device has a menu on its display.

The following messages are used for the Device Menu Control feature: <User Control Pressed>, <User Control Released>, <Menu Request>, <Menu Status>

OpCode	Value	Description	Parameters	Response	Supported
<User Control Pressed>	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another.	[UI Command]	Update display or perform an action, as required.	Yes

OpCode	Value	Description	Parameters	Response	Supported
<User Control Released>	0x45	Indicates that user released a remote control button (the last one indicated by the <User Control Pressed> message)	None	Update display or perform an action, as required.	Yes
<Menu Request>	0x8D	A request from the TV for a device to show/remove a menu or to query if a device is currently showing a menu.	[Menu Request Type]	May enter or exit the 'Device Menu Active' state if the parameter was "Activate" or "Deactivate". Send <Menu Status> to indicate the current status of the devices menu.	Yes
<Menu Status>	0x8E	Used to indicate to the TV that the device is showing/has removed a menu and requests the remote control keys to be passed though.	[Menu State]	If Menu State indicates activated, TV enters 'Device Menu Active' state and forwards those Remote control commands, shown in Table 26, to the initiator. If deactivated, TV enters 'Device Menu Inactive' state and stops forwarding remote control commands.	Yes

### 5.1.13 Remote Control Passthrough

**Command Feature:** This feature is used to pass remote control commands received by one device (typically the TV) through to another device in the network.

The following messages are used for the Remote Control Passthrough feature: <User Control Pressed>, <User Control Released>.

*Table on Next Page*

OpCode	Value	Description	Parameters	Response	Supported
<User Control Pressed>	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another.	[UI Command]	Update display or perform an action, as required.	Yes
<User Control Released>	0x45	Indicates that user released a remote control button (the last one indicated by the <User Control Pressed> message)	None	Update display or perform an action, as required.	Yes

### 5.1.14 Give Device Power Status

**Command Feature:** Several messages, such as <Image View On> and <Play>, bring another device out of standby. The <Give Device Power Status> message is used to determine the current power status of a target device. The target device responds with a <Report Power Status> message containing the Power Status operand.

The following messages are used for the Give Device Power Status feature: <Give Device Power Status>, <Report Power Status>.

OpCode	Value	Description	Parameters	Response	Supported
<Give Device Power Status>	0x8F	Used to determine the current power status of a target device.	None	<Report Power Status>	Yes
<Report Power Status>	0x90	Used to inform a requesting device of the current power status.	[Power Status]		Yes

### 5.1.15 System Audio Control

**Command Feature:** This feature allows an audio amplifier to provide the audio for a source that is being displayed on a TV. When in this mode, the amplifier uses the same source as the video and provides the volume control function, instead of the TV, which mutes its speakers.

The following messages are used for the System Audio Control feature: <Give Audio Status>, <Give System Audio Mode Status>, <Report Audio Status>, <Set System Audio Mode>, <System Audio Mode Request>, <System Audio Mode Status>, <User Control Pressed>, <User Control Released>.

OpCode	Value	Description	Parameters	Response	Supported
<Give Audio Status>	0x71	Requests an amplifier to send its volume and mute status	None	<Report Audio Status>	Yes
<Give System Audio Mode Status>	0x7D	Requests the status of the System Audio Mode	None	Amplifier sends a <System Audio Mode Status> message indicating status (On or Off)	Yes
<Report Audio Status>	0x7A	Reports an amplifier's volume and mute status	[Audio Status]		Yes
<Set System Audio Mode>	0x72	Turns the System Audio Mode On or Off.	[System Audio Status]	If set to On, the TV mutes its speakers. The TV or STB sends relevant <User Control Pressed> or <User Control Released> as necessary. If set to Off, the TV unmutes its speakers. The TV or STB stop sending the volume-related <User Control Pressed> or <User Control Released> messages.	Yes

Table Continues on Next Page

OpCode	Value	Description	Parameters	Response	Supported
<System Audio Mode Request>	0x70	A device implementing System Audio Control and which has volume control RC buttons (eg TV or STB) requests to use System Audio Mode to the amplifier	[Physical Address]	The amplifier comes out of standby (if necessary) and switches to the relevant connector for device specified by [Physical Address]. It then sends a <Set System Audio Mode> [On] Message. <System Audio Mode Request> sent without a [Physical Address] parameter requests termination of the feature. In this case, the amplifier sends a <Set System Audio Mode> [Off] message.	Yes
<System Audio Mode Status>	0x7E	Reports the current status of the System Audio Mode	[System Audio Status]	If [On], the device requesting this information can send the volume-related <User Control Pressed> or <User Control Released> messages.	Yes
<User Control Pressed>	0x44	Used to indicate that the user pressed a remote control button or switched from one remote control button to another.	[UI Command] of "Volume Up", "Volume Down" or "Mute"	Increase or Decrease the volume of the amplifier, or mute/unmute the amplifier.	Yes
<User Control Released>	0x45	Used to control audio rate from Source Device.	[Audio Rate]	Perform the specified actions, or return a <Feature Abort> msg	Yes

## 5.1.16 Audio Rate Control

---

**Command Feature:** This feature allows the audio playback rate of a Source Device to be controlled by another device, e.g. an Audio System.

The following messages are used for the Audio Rate Control Feature: <Set Audio Rate>

OpCode	Value	Description	Parameters	Response	Supported
<Set Audio Rate>	0x9A	Used to control audio rate from Source Device.	[Audio Rate]	Perform the specified actions, or return a<Feature Abort> message.	Yes