

Run Demos from Crank Software on i.MX Developer's Kits

Embedded Artists AB

Rundelsgatan 14
SE-211 36 Malmö
Sweden

<https://www.EmbeddedArtists.com>

Copyright 2021 © Embedded Artists AB. All rights reserved.

No part of this publication may be reproduced, transmitted, transcribed, stored in a retrieval system, or translated into any language or computer language, in any form or by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise, without the prior written permission of Embedded Artists AB.

Disclaimer

Embedded Artists AB makes no representation or warranties with respect to the contents hereof and specifically disclaim any implied warranties or merchantability or fitness for any particular purpose. Information in this publication is subject to change without notice and does not represent a commitment on the part of Embedded Artists AB.

Feedback

We appreciate any feedback you may have for improvements on this document. Send your comments by using the contact form: www.embeddedartists.com/contact.

Trademarks

All brand and product names mentioned herein are trademarks, services marks, registered trademarks, or registered service marks of their respective owners and should be treated as such.

Table of Contents

1	Document Revision History	5
2	Introduction	6
2.1	Conventions.....	6
3	Crank Software.....	7
3.1	Storyboard	7
3.2	Resources.....	8
4	Setup hardware	9
4.1	7-inch HDMI display kit	9
4.2	RGB-to-HDMI adapter	10
4.3	Touch interface - USB port on carrier board.....	11
5	Run the demo	13
5.1	iMX8M Mini uCOM	13
5.1.1	Information about the demo.....	13
5.1.2	Instructions	13
5.2	iMX8M Nano uCOM	14
5.2.1	Information about the demo.....	14
5.2.2	Instructions	14
5.3	iMX8M COM.....	14
5.3.1	Information about the demo.....	14
5.3.2	Instructions	15
5.4	iMX7 Dual COM / iMX7 Dual uCOM	15
5.4.1	Information about the demo.....	15
5.4.2	Instructions	16
5.5	iMX7ULP uCOM	17
5.5.1	Information about the demo.....	17
5.5.2	Instructions	17
5.6	iMX6 Quad COM	17
5.6.1	Information about the demo.....	17
5.6.2	Instructions	18
5.7	iMX6 SoloX COM	19
5.7.1	Information about the demo.....	19
5.7.2	Instructions	19
5.8	iMX6 UltraLite COM	20
5.8.1	Information about the demo.....	20
5.8.2	Instructions	21
6	Deploying the image.....	23
6.1	USB OTG boot mode – J2 jumper	23
6.2	UUU	23
6.3	Download the bundle	23

6.4	Run the tool in Ubuntu.....	24
6.5	Run the tool in Windows.....	24
6.6	Troubleshoot	25

1 Document Revision History

<i>Revision</i>	<i>Date</i>	<i>Description</i>
A	2020-04-23	First release
A1	2021-06-24	Updated Crank logo

2 Introduction

This document provides you with step-by-step instructions for how to program and run graphical demos from Crank Software on Embedded Artists iMX6, iMX7 and iMX8 based COM boards.

Crank Software, an AMETEK company, develops the product *Storyboard* that will help you design and develop a **User Interface** for your embedded system. The demos show different kind of user interfaces developed with Storyboard.

Demos are available for:

- iMX8M Mini uCOM
- iMX8M Nano uCOM
- iMX8M Quad COM
- iMX7 Dual COM
- iMX7 Dual uCOM
- iMX7ULP uCOM
- iMX6 Quad COM
- iMX6 SoloX COM
- iMX6 UltraLite COM

Additional documentation you might need is.

- *Working with Yocto to build Linux* – found on Embedded Artists website.
- Getting started guide - <https://www.embeddedartists.com/getting-started/>
- What is Storyboard: <https://www.cranksoftware.com/storyboard>

2.1 Conventions

A number of conventions have been used throughout to help the reader better understand the content of the document.

Constant width text – is used for file system paths and command, utility and tool names.

```
$ This field illustrates user input in a terminal running on the  
development workstation, i.e., on the workstation where you edit,  
configure and build Linux
```

```
# This field illustrates user input on the target hardware, i.e.,  
input given to the terminal attached to the COM Board
```

```
This field is used to illustrate example code or excerpt from a  
document.
```

3 Crank Software

Crank Software, an AMETEK company, is both a software and service-based company with headquarters in Canada. They have extensive **GUI** design and development experience and provide the tool **Storyboard**. With this tool customers can easily develop User Interfaces for their products.

Crank Software has customers worldwide and there are several **100 million devices** in the field running user interfaces developed with Storyboard.

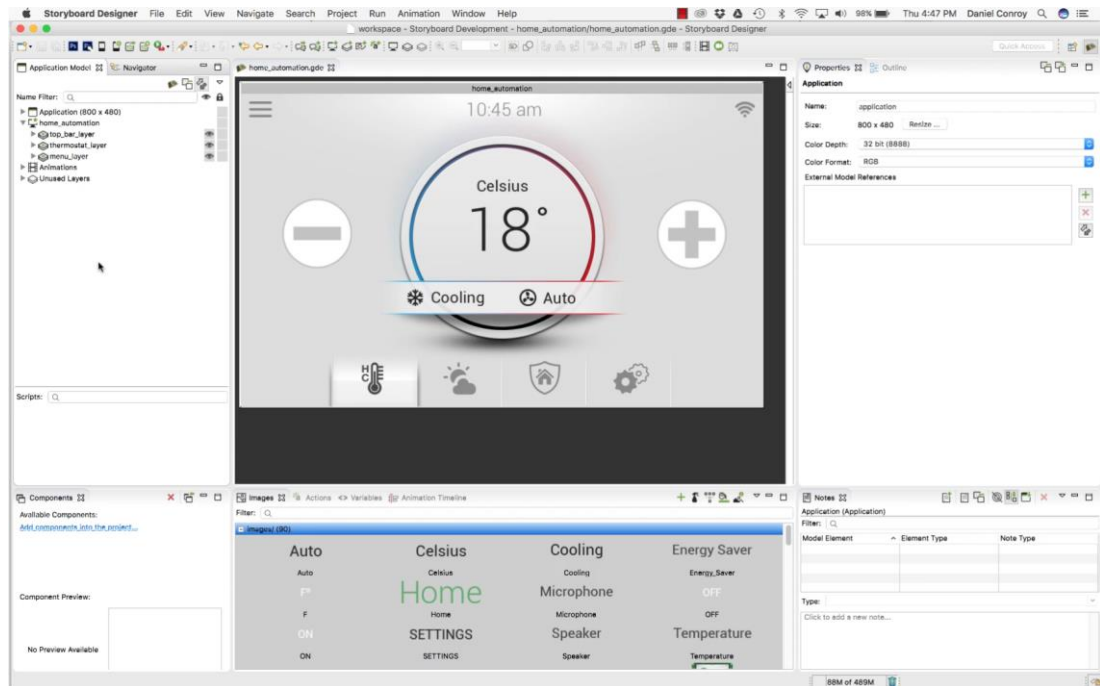


3.1 Storyboard

Storyboard is the software developed by Crank Software that will make it easier for you to develop a user interface for your product. It actually consists of two parts; **Storyboard Designer** which you use to develop your user interface and **Storyboard Engine** that runs your application. Storyboard Engine is available for many different platforms.

Some of the features of Storyboard:

- Decouples front-end UI from back-end logic
- Data model and events API connects them together
- Hardware agnostic framework – working across many platforms
- Enables collaborative workflow for designers and developers



3.2 Resources

Here are some further resources related to Crank Software and Storyboard.

Quickstart Video Tutorials

<https://www.cranksoftware.com/resources/quickstart-video-tutorials>

Knowledge Base

<https://support.cranksoftware.com/hc/en-us/categories/360005511011-Knowledge-base>

Community Forum

<https://support.cranksoftware.com/hc/en-us/community/topics>

Professional Services

<https://www.cranksoftware.com/professional-services>

4 Setup hardware

4.1 7-inch HDMI display kit

All the instructions in this document are based on using the 7-inch HDMI display kit (**EAD00363** or **EAD00369**) from Embedded Artists.



Figure 1 - 7-inch HDMI display

The 7-inch HDMI display kit comes with the following content:

- 7-inch display
- 2x micro-B to A USB cables
- 1x HDMI cable
- 1x RGB-to-HDMI adapter board (**Note:** only included with **EAD00369**).
- Mounting material for frame and holder

See Figure 2 below for the location of important connectors on the display.

Number	Connector	Description
1	HDMI connector	Connect to the HDMI connector on the carrier board or to RGB-to-

		HDMI adapter for boards that require this adapter (see section 4.2 .
2	Power	Connect micro-B to A USB cable to an external USB hub or your computer for powering.
3	Touch	Connect micro-B to A USB cable to USB port on carrier board, see section 4.3 for details.

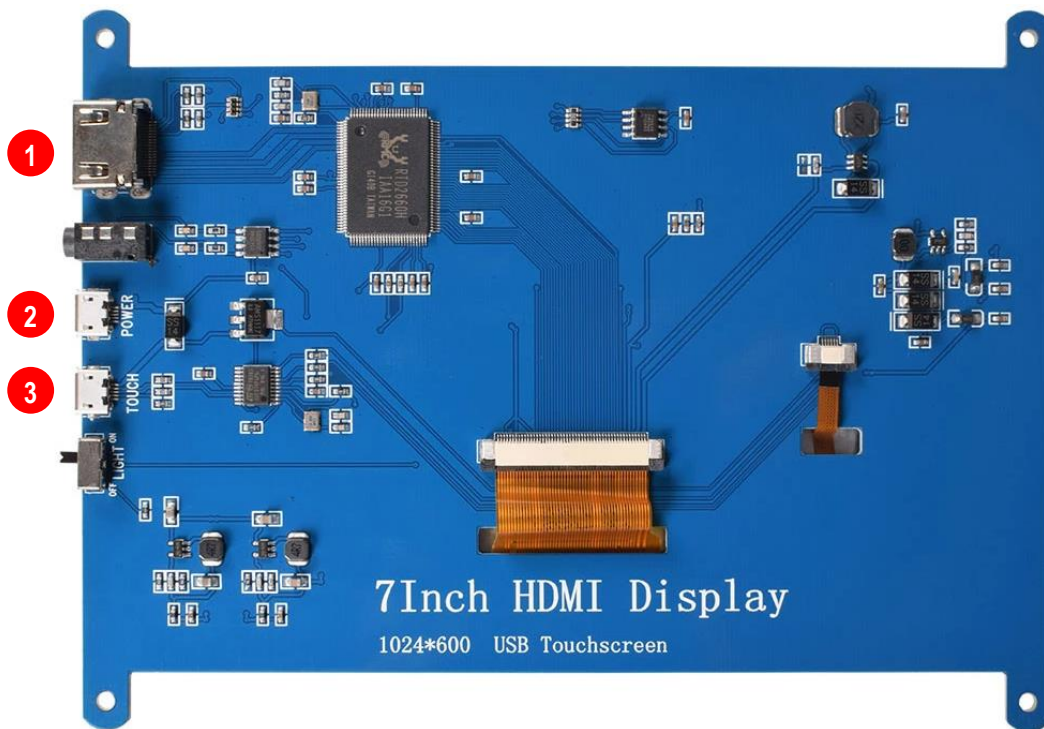


Figure 2 - Connectors on 7-inch HDMI display

4.2 RGB-to-HDMI adapter

The following COM boards require you to use the RGB-to-HDMI adapter (included with **EAD00369**) between the Developer's Kit and the display.

- iMX7 Dual COM
- iMX7 Dual uCOM
- iMX6 SoloX COM
- iMX6 UltraLite COM

First connect the flat cable to the **J39 connector** on the COM Carrier Board V2 as shown in Figure 3.

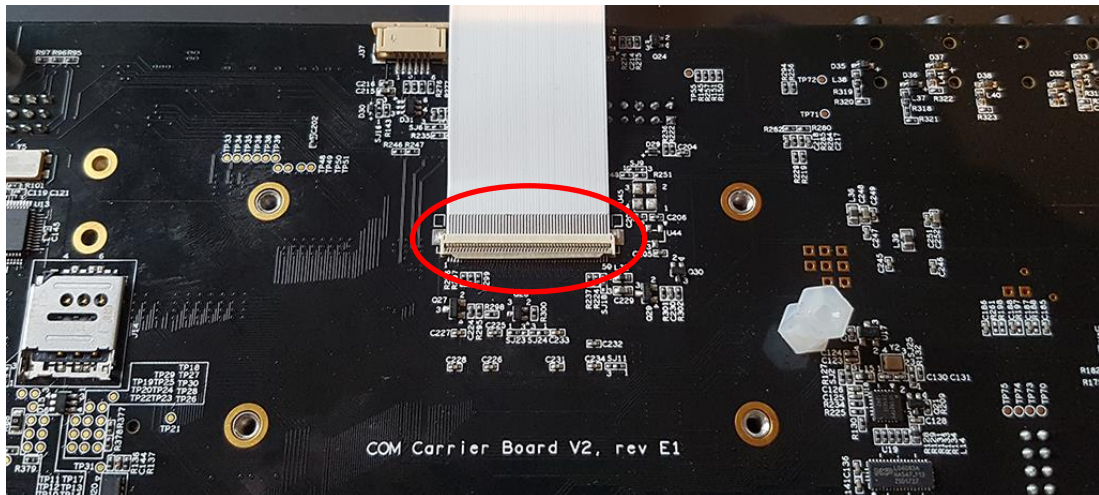


Figure 3 - Flat cable connected to J39 on Carrier Board

Secondly connect the flat cable to the RGB-to-HDMI adapter as shown in Figure 4.

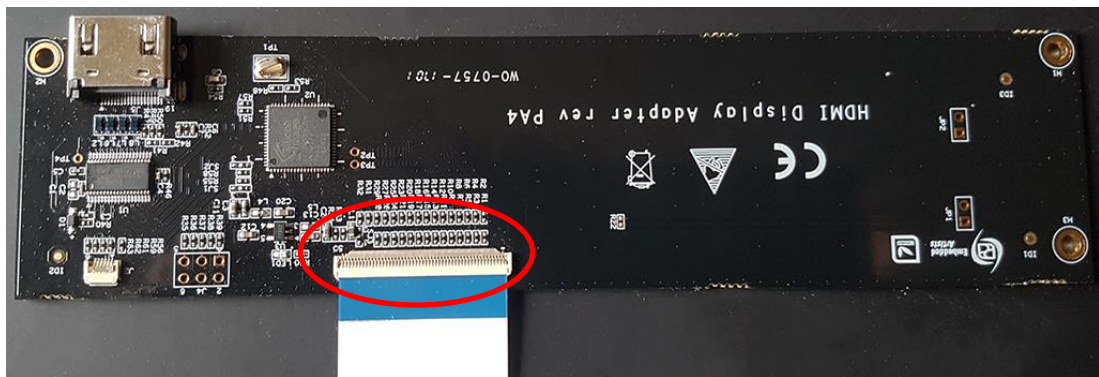


Figure 4 - Flat cable connected to RGB-to-HDMI adapter

You can now connect the HDMI cable between the HDMI connector (J5) on the adapter and the HDMI connector on the display.

4.3 Touch interface - USB port on carrier board

The touch interface should for most of the COM boards be connected to the USB Host Type A connector (J12) on the carrier board, see Figure 5.

For iMX7ULP uCOM and for uMX8M Nano uCOM you however have to use connector J11 instead since these boards only have one USB interface. To be able to use J11 you need a micro-B (male) to USB A (female) adapter similar to the one shown in Figure 6.

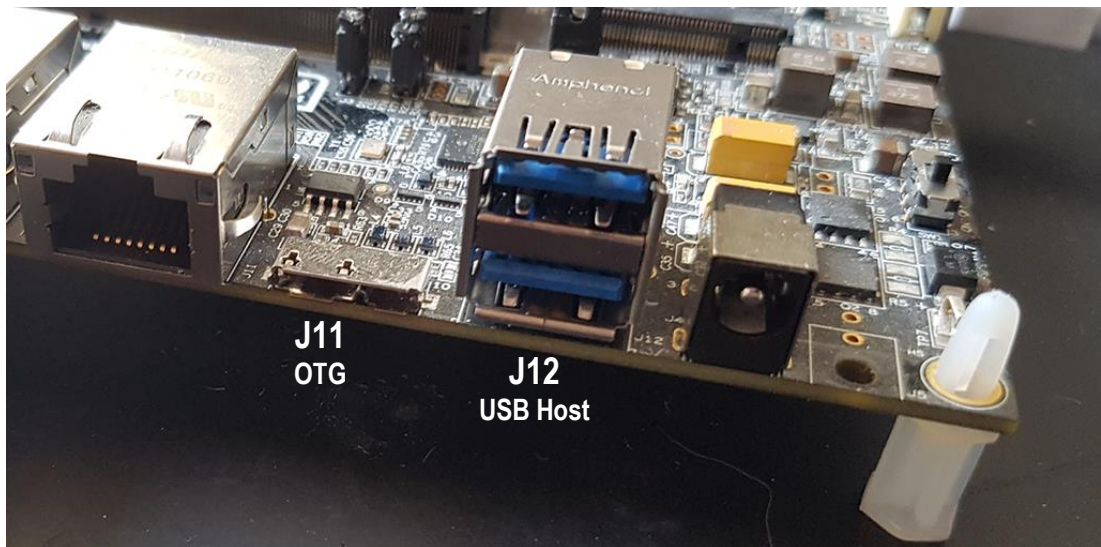


Figure 5 - USB connectors on carrier board



Figure 6 - USB micro-B (male) to USB A (female) adapter

5 Run the demo

This chapter contains one subsection with instructions for each supported COM board. Go to the subsection for the COM board you are using.

5.1 iMX8M Mini uCOM

5.1.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 1280x720
- Rendering: OpenGL ES 2.0
- Demos: Oven (Appliances), 3D Robotic Arm (Industrial), Vitals Monitor (Medical)
- Storyboard version: 6.0.0

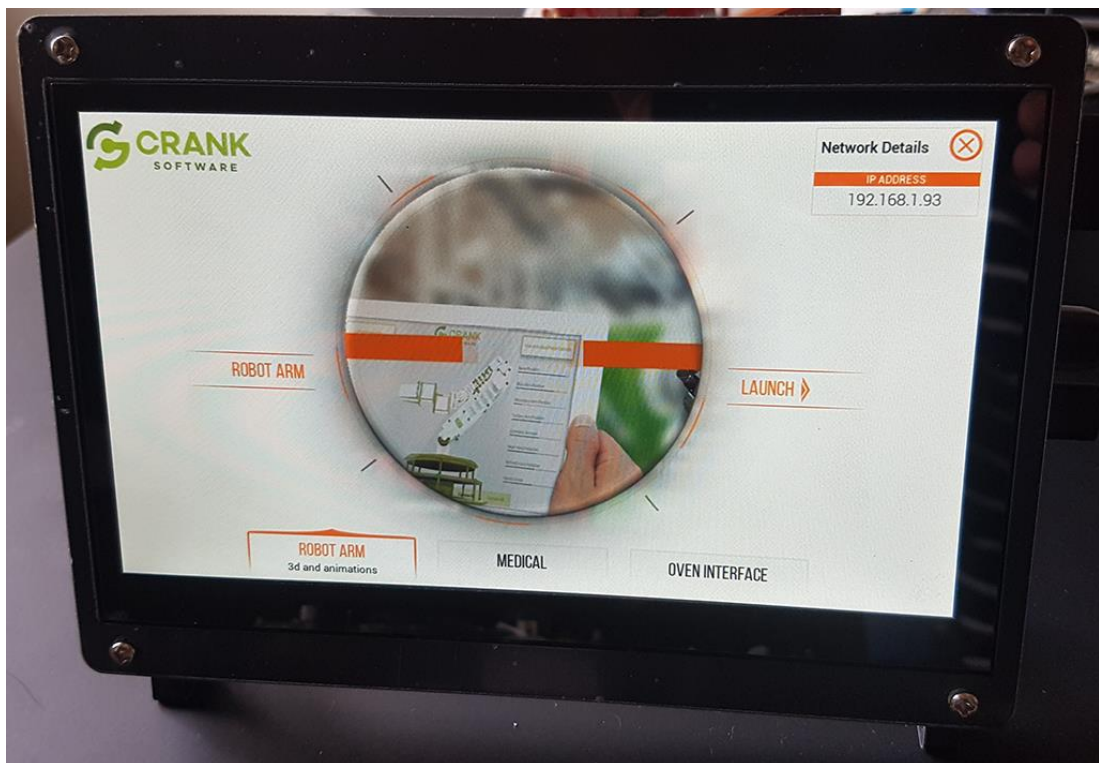


Figure 7 - Demo on iMX8M Mini

5.1.2 Instructions

For this board there is a MIPI-to-HDMI bridge on the uCOM adapter so you can connect the HDMI cable between the HDMI connector (J13) on the carrier board and the HDMI connector on the display.

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Let the board boot into Linux.
4. Login using user: `root` and password: `pass`.

5. The demo will now be unpacked and then the board will automatically reboot.
6. When the board has rebooted you should see the demo on the display.

5.2 iMX8M Nano uCOM

5.2.1 Information about the demo

This is the same demo as for iMX8M Mini uCOM. Below are a few highlighted features about the demo.

- Resolution: 1280x720
- Rendering: OpenGL ES 2.0
- Demos: Oven (Appliances), 3D Robotic Arm (Industrial), Vitals Monitor (Medical)
- Storyboard version: 6.0.0

5.2.2 Instructions

For this board there is a MIPI-to-HDMI bridge on the uCOM adapter so you can connect the HDMI cable between the HDMI connector (J13) on the carrier board and the HDMI connector on the display.

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Let the board boot into Linux.
4. Login using user: `root` and password: `pass`.
5. The demo will now be unpacked and then the board will automatically reboot.
6. When the board has rebooted you should see the demo on the display.

5.3 iMX8M COM

5.3.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 1280x720
- Rendering: OpenGL ES 2.0
- Demos: Smart Home (IoT), Movie Kiosk (Consumer) , 3D cluster (Automotive), Oven (Appliances)
- Storyboard version: 6.0.0

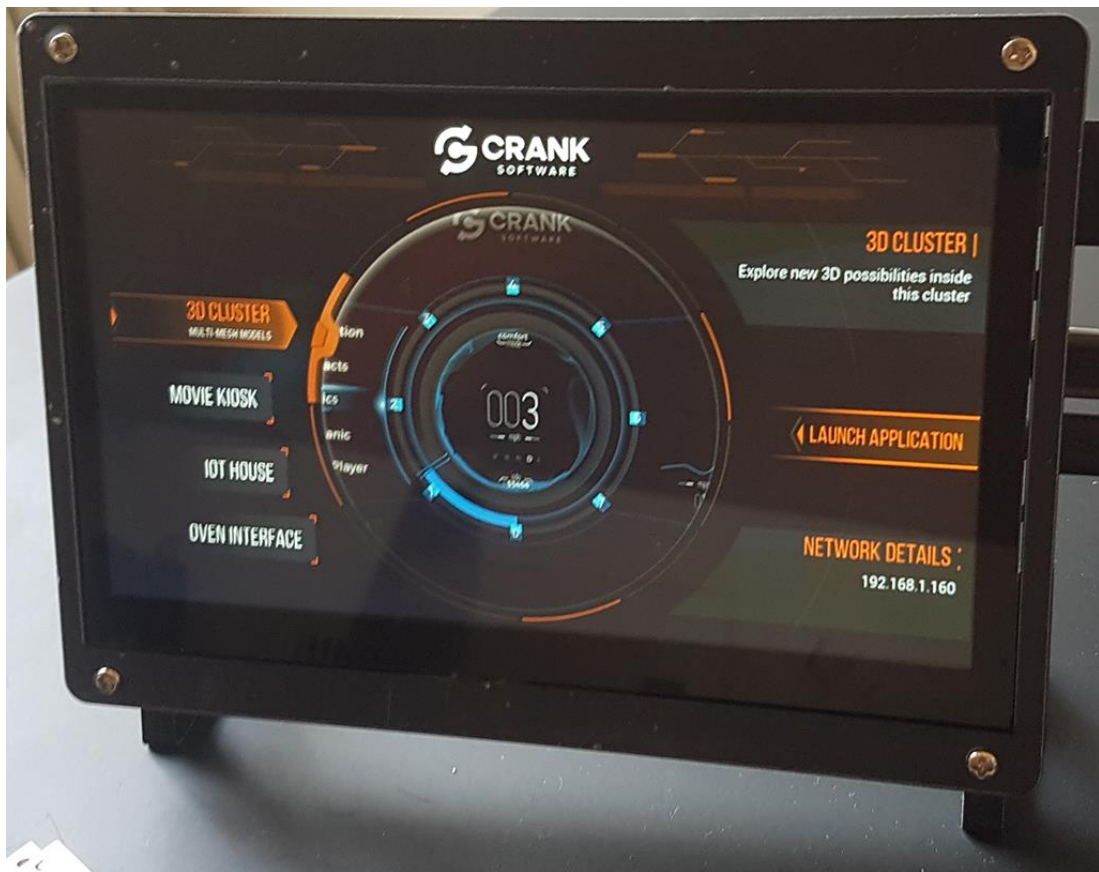


Figure 8 - Demo on iMX8M COM

5.3.2 Instructions

This board has direct support for HDMI display interfaces. Connect the HDMI cable between the HDMI connector (J13) on the carrier board and the HDMI connector on the display.

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Let the board boot into Linux.
4. Login using user: `root` and password: `pass`.
5. The demo will now be unpacked and then the board will automatically reboot.
6. When the board has rebooted you should see the demo on the display.

5.4 iMX7 Dual COM / iMX7 Dual uCOM

5.4.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 800x480 (the display resolution is 800x600)
- Rendering: Framebuffer (fbdev)
- Demos: Washing Machine (White goods), Home Automation, Medical
- Storyboard version: 5.0.0

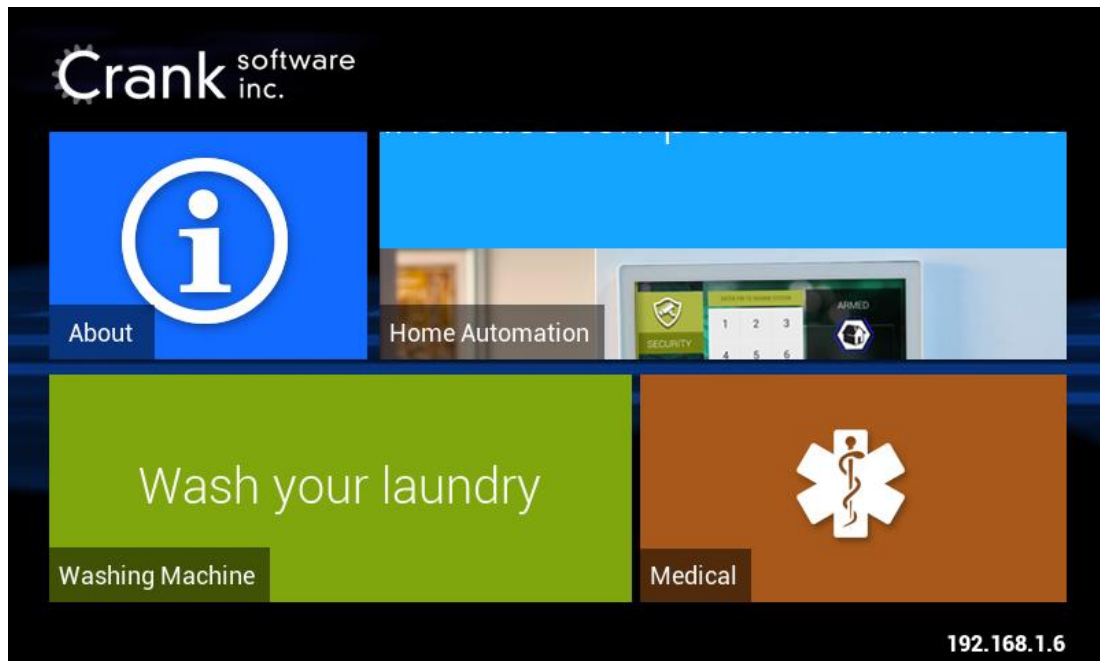


Figure 9 - Demo on iMX7 Dual uCOM

5.4.2 Instructions

This board supports RGB display interface which means that you must use the RGB-to-HDMI adapter as described in section 4.2

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Boot into u-boot and enter the commands below.
 - a. Enable the sii902x device (RGB-to-HDMI bridge).

```
=> setenv cmd_custom fdt set i2c0/sii902x status okay\;fdt set /sii902x-reset status okay
```

- b. Enable the RGB display interface.

```
=> eadisp enable rgb
```

- c. Add a new display configuration.

```
=> eadisp add rgb
"DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0"
```

- d. Check which number the new configuration got (in this case it was 8).

```
=> eadisp

Available display configurations:
0)  rgb  Innolux-AT070TN:24:33336667,800,480,89,164,75,75,10,10,0,0,1,0
1)  rgb  nhd-4.3-480272ef:24:9009009,480,272,2,2,2,2,41,10,0,0,1,0
2)  rgb  nhd-5.0-800480tf:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
```



```

3)  rgb  nhd-7.0-800480ef:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
4)  rgb  umsh-8864:24:9061007,480,272,20,20,20,3,3,0,0,1,0
5)  rgb  umsh-8596-30t:24:33264586,800,480,128,120,20,20,8,5,0,0,1,1
6)  rgb  umsh-8596-33t:24:32917475,800,480,200,200,45,45,1,1,0,0,1,1
7)  rgb  rogin-rx050a:24:32917475,800,480,200,200,45,45,1,1,0,0,1,0
8)  rgb  DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0

```

e. Change to the new display configuration.

```
=> eadisp conf rgb 8
```

f. Save the changes.

```
=> saveenv
```

g. Boot into Linux.

```
=> boot
```

4. Let the board boot into Linux.
5. Login using user: `root` and password: `pass`.
6. The demo will now be unpacked and then the board will automatically reboot.
7. When the board has rebooted you should see the demo on the display.

5.5 iMX7ULP uCOM

5.5.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 640x480
- Rendering: Framebuffer (fbdev)
- Demos: Home Automation
- Storyboard version: 5.0.0

5.5.2 Instructions

For this board there is a MIPI-to-HDMI bridge on the uCOM adapter so you can connect the HDMI cable between the HDMI connector (J13) on the carrier board and the HDMI connector on the display.

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Let the board boot into Linux.
4. Login using user: `root` and password: `pass`.
5. The demo will now be unpacked and then the board will automatically reboot.
6. When the board has rebooted you should see the demo on the display.

5.6 iMX6 Quad COM

5.6.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 1280x720
- Rendering: OpenGL ES 2.0
- Demos: Smart home (IoT), Movie Kiosk (Consumer), Vitals Monitor (Medical)
- Storyboard version: 5.0.0

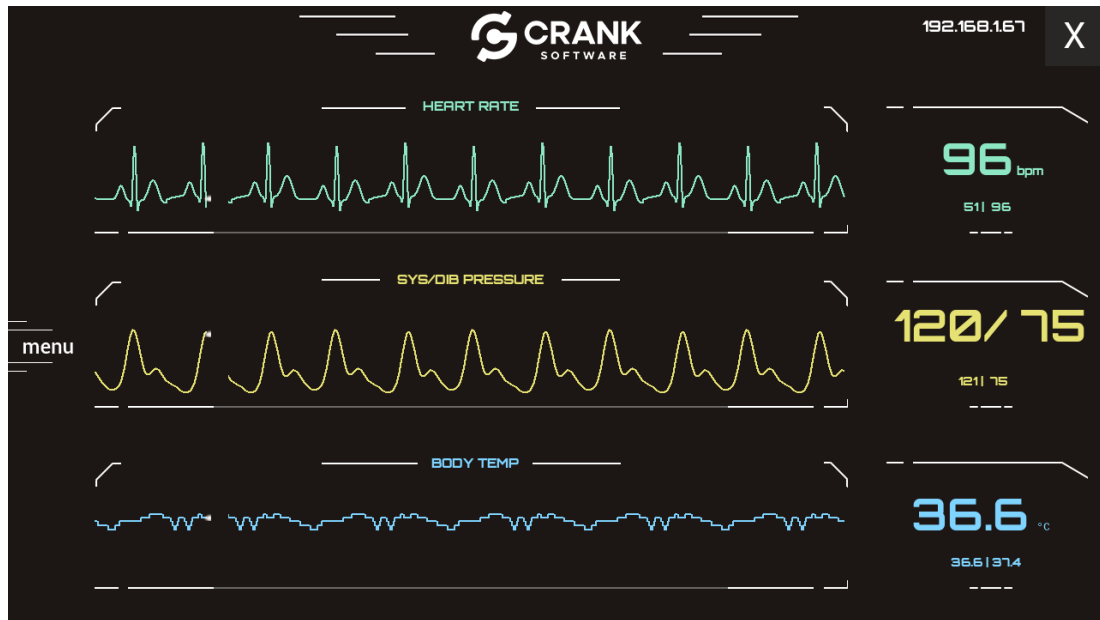


Figure 10 - Vitals demo on iMX6 Quad COM

5.6.2 Instructions

This board has direct support for HDMI display interfaces. Connect the HDMI cable between the HDMI connector (J13) on the carrier board and the HDMI connector on the display.

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Boot into u-boot and enter the commands below.
 - a. Enable the HDMI interface

```
=> eadisp enable hdmi
```

- b. Change to 32 bits per pixel (bpp). The demo won't start with 24.

```
=> setenv cmd_custom fdt set fb_hdmi default_bpp <0x20>
```

- c. Save changes and boot into Linux

```
=> saveenv
=> boot
```

4. Let the board boot into Linux.
5. Login using user: `root` and password: `pass`.

6. The demo will now be unpacked and then the board will automatically reboot.
7. When the board has rebooted you should see the demo on the display.

5.7 iMX6 SoloX COM

5.7.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 800x480 (the display resolution is 800x600)
- Rendering: OpenGL ES 2.0
- Demos: Washing Machine
- Storyboard version: 5.3.0

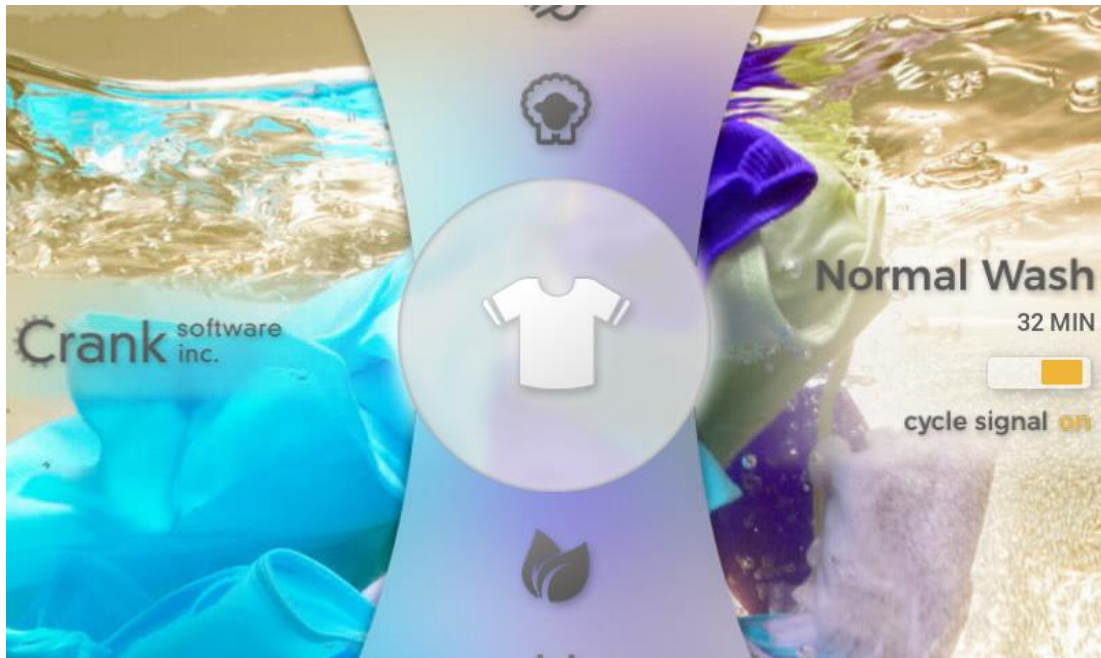


Figure 11 - Washing Machine demo on iMX6 SoloX COM

5.7.2 Instructions

This board supports RGB display interface which means that you must use the RGB-to-HDMI adapter as described in section 4.2

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Boot into u-boot and enter the commands below.
 - a. Enable the sii902x device (RGB-to-HDMI bridge).

```
=> setenv cmd_custom fdt set i2c0/sii902x status okay\;fdt set /sii902x-reset status okay
```

- b. Enable the RGB display interface.

```
=> eadisp enable rgb
```

c. Add a new display configuration.

```
=> eadisp add rgb
"DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0"
```

d. Check which number the new configuration got (in this case it was 12).

```
=> eadisp

Available display configurations:
 0) lvds0 hannstar:18:64998375,1024,768,220,40,21,7,60,10,0,0,0,0
 1) lvds0 lp101wh4_X11:24:72051300,1360,768,80,48,14,3,32,5,0,0,1,0
 2) lvds0 lp101wh4:24:72332730,1366,768,80,48,14,3,32,5,0,0,1,0
 3) lvds0 nhd-10.1-1024600af:24:51208521,1024,600,160,160,23,12,0,0,0,0,1,0
 4)   rgb Innolux-AT070TN:24:33336667,800,480,89,164,75,75,10,10,0,0,1,0
 5)   rgb nhd-4.3-480272ef:24:9009009,480,272,2,2,2,2,41,10,0,0,1,0
 6)   rgb nhd-5.0-800480tf:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
 7)   rgb nhd-7.0-800480ef:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
 8)   rgb umsh-8864:24:9061007,480,272,20,20,20,20,3,0,0,1,0
 9)   rgb umsh-8596-30t:24:33264586,800,480,128,120,20,20,8,5,0,0,1,1
10)   rgb umsh-8596-33t:24:32917475,800,480,200,200,45,45,1,1,0,0,1,1
11)   rgb rogin-rx050a:24:32917475,800,480,200,200,45,45,1,1,0,0,1,0
12)   rgb DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0
```

e. Change to the new display configuration.

```
=> eadisp conf rgb 12
```

f. Save the changes.

```
=> saveenv
```

g. Boot into Linux.

```
=> boot
```

4. Let the board boot into Linux.
5. Login using user: `root` and password: `pass`.
6. The demo will now be unpacked and then the board will automatically reboot.
7. When the board has rebooted you should see the demo on the display.

5.8 iMX6 UltraLite COM

5.8.1 Information about the demo

Below are a few highlighted features about the demo.

- Resolution: 800x480 (the display resolution is 800x600)
- Rendering: Framebuffer (fbdev)
- Demos: Washing Machine (Appliances), Coffee Machine (Consumer), Home Automation (IoT)
- Storyboard version: 5.3.0

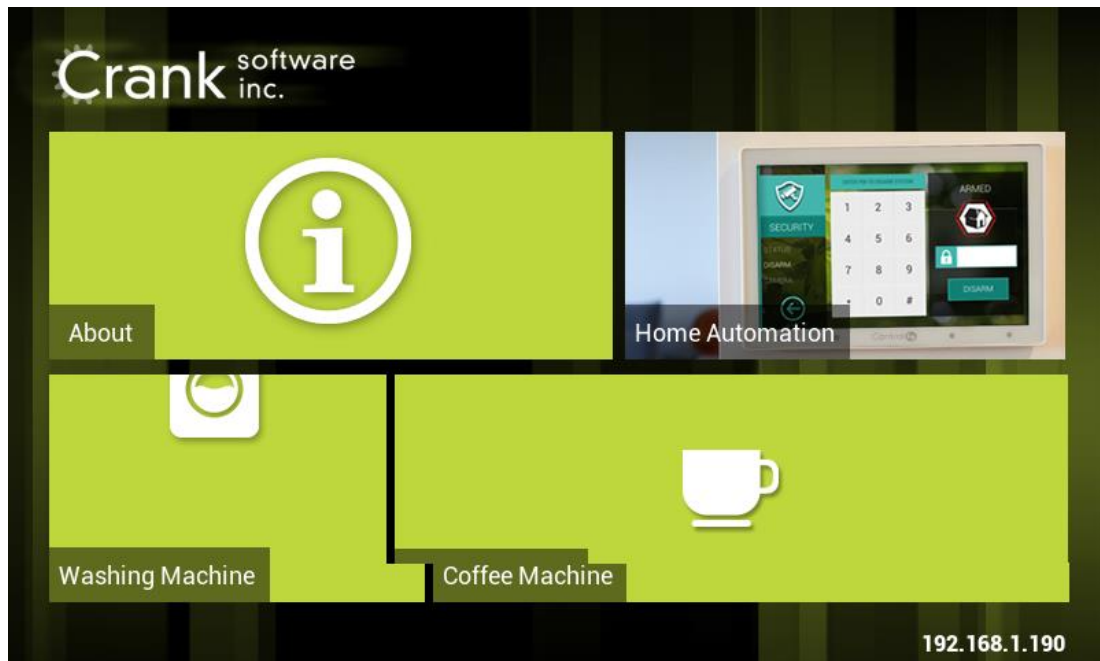


Figure 12 - Demo on iMX6 UltraLite COM

5.8.2 Instructions

This board supports RGB display interface which means that you must use the RGB-to-HDMI adapter as described in section 4.2

1. First deploy the image containing the demo to the board. See chapter 6 for instructions of how to deploy an image. The uuu bundle containing the demo can be found on <http://imx.embeddedartists.com>.
2. Set jumper J2 in open state (see Figure 13) and reset the board.
3. Boot into u-boot and enter the commands below.
 - a. Enable the sii902x device (RGB-to-HDMI bridge).

```
=> setenv cmd_custom fdt set i2c0/sii902x status okay\;fdt set /sii902x-reset status okay
```

- b. Enable the RGB display interface.

```
=> eadisp enable rgb
```

- c. Add a new display configuration.

```
=> eadisp add rgb
"DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0"
```

- d. Check which number the new configuration got (in this case it was 8).

```
=> eadisp

Available display configurations:
0)  rgb  Innolux-AT070TN:24:33336667,800,480,89,164,75,75,10,10,0,0,1,0
1)  rgb  nhd-4.3-480272ef:24:9009009,480,272,2,2,2,2,41,10,0,0,1,0
2)  rgb  nhd-5.0-800480tf:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
```

```
3)  rgb  nhd-7.0-800480ef:24:29232073,800,480,40,40,29,13,48,3,0,0,1,0
4)  rgb  umsh-8864:24:9061007,480,272,20,20,20,20,3,3,0,0,1,0
5)  rgb  umsh-8596-30t:24:33264586,800,480,128,120,20,20,8,5,0,0,1,1
6)  rgb  umsh-8596-33t:24:32917475,800,480,200,200,45,45,1,1,0,0,1,1
7)  rgb  rogin-rx050a:24:32917475,800,480,200,200,45,45,1,1,0,0,1,0
8)  rgb  DMTFT70:24:33336667,1280,720,89,164,75,75,10,10,0,0,1,0
```

e. Change to the new display configuration.

```
=> eadisp conf rgb 8
```

f. Save the changes.

```
=> saveenv
```

g. Boot into Linux.

```
=> boot
```

4. Let the board boot into Linux.
5. Login using user: `root` and password: `pass`.
6. The demo will now be unpacked and then the board will automatically reboot.
7. When the board has rebooted you should see the demo on the display.

6 Deploying the image

NXP's Manufacturing Tool called UUU is used to deploy an image to the target. You can download a bundle containing the UUU tool, scripts and pre-built Boot2Qt images from <http://imx.embeddedartists.com/>.

6.1 USB OTG boot mode – J2 jumper

Before deploying an image, the board must be put into USB OTG boot mode.

This is accomplished by closing the J2 jumper on the COM Carrier board; see Figure 13 to locate the jumper. Please note that in the figure the jumper is in open state which means that the COM board will boot from eMMC. Also note that a power cycle is needed to place the COM board in USB OTG boot mode.



Figure 13 - J2 jumper (opened state) on a COM Carrier Board V2

6.2 UUU

UUU (Universal Update Utility) is version 3 of MFGTool but it has been rewritten, is publicly available on GitHub (<https://github.com/NXPmicro/mfgtools>) and it can be run on both Windows and Linux while the older versions of MFGTool were limited to Windows only. UUU is the preferred utility to use when downloading images.

UUU can be used to write images to the board. This tool is sending files and instructions over USB and the board must be set in OTG boot mode for it to work.

Useful links:

- UUU on GitHub: <https://github.com/NXPmicro/mfgtools>
- UUU release page: <https://github.com/NXPmicro/mfgtools/releases>

6.3 Download the bundle

Download the bundle for the board you are using from <http://imx.embeddedartists.com/>

Unpack the file somewhere on your computer. Below is a description of some of the content in the file.

- `uuu` (root): Contains a README file.
- `uuu/*` .`uuu`: The different download configurations.
- `uuu/files/`: Contains pre-compiled versions of images. The tool will look in this directory when selecting images to download to the board.

6.4 Run the tool in Ubuntu

In Linux, open a terminal and navigate to the folder where the uuu file was unpacked. Make sure that the tool is executable and then execute the tool:

```
$ cd ~/uuu_imx8mm_ucom_4.14.98_crank_demo
$ chmod +x ./uuu
$ sudo ./uuu full_tar.uuu
```

The terminal will show a progress bar like illustrated below while it is running:

```
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ sudo ./uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 0   Failure 0

2:24  20/23  [=====47%                ] FBK: ucp files/ea-image-base-imx8mqea-com.tar.bz2 t:-
```

After a successful run it will look like this:

```
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ sudo ./uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 1   Failure 0

2:24  23/23  [Done                        ] FBK: DONE
```

If a problem occurs then the program will terminate and print an error message like this

```
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ sudo ./uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 0   Failure 1

2:24  1/ 1  [HID(W):LIBUSB_ERROR_IO      ] SDP: boot -f files/u-boot-imx8mqea-com.bin
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ █
```

6.5 Run the tool in Windows

In Windows, open a Command Prompt, navigate to the folder where the uuu file was unpacked and then run the tool:

```
C:\> cd c:\temp\uuu_imx8mm_ucom_4.14.98_crank_demo
C:\temp\uuu_imx8mm_ucom_4.14.98_crank_demo> uuu.exe full_tar.uuu
```

The terminal will show a progress bar similar to below while it is running:

```
c:\temp\uuu_imx8mq_com_4.14.78>uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 0   Failure 0

1:23  20/23  [=====> 34%                ] FBK: ucp files/ea-image-base-imx8mqea-com.tar.bz2 t:-
```

After a successful run it will look like this:

```
c:\temp\uuu_imx8mq_com_4.14.78>uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 1   Failure 0

1:23  23/23  [Done                        ] FBK: DONE

c:\temp\uuu_imx8mq_com_4.14.78>
```

If a problem occurs then the program will terminate and print an error message like this


```
c:\temp\uuu_imx8mq_com_4.14.78>uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

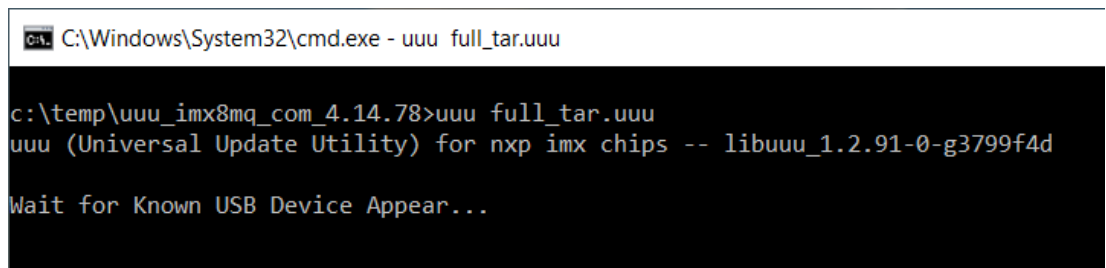
Success 0    Failure 1

1:23 20/23 [Bulk(R):LIBUSB_ERROR_PIPE ] FBK: ucp files/ea-image-base-imx8mqa-com.tar.bz2 t:-
c:\temp\uuu_imx8mq_com_4.14.78>
```

6.6 Troubleshoot

Some common problems and solutions:

- The first time you run uuu on your computer it fails.**
 This is likely because of USB driver installation. Let the driver install, reset the hardware and then run the uuu command again. In Windows it is three different drivers that are needed so this procedure might have to be repeated three times - each time the procedure gets a little bit further.
- UUU appears to hang with a "Wait for Known USB Device Appear..." message like this:**



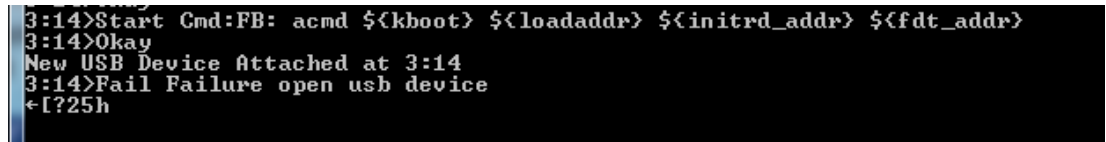
```
C:\Windows\System32\cmd.exe - uuu full_tar.uuu

c:\temp\uuu_imx8mq_com_4.14.78>uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Wait for Known USB Device Appear...
```

This means that the hardware is either not connected to the computer with the USB cable or it is not in the USB OTG boot mode. Check and follow instructions in section 6.1 and then run the uuu command again.

- Windows 7 fail to flash with an error like this:**

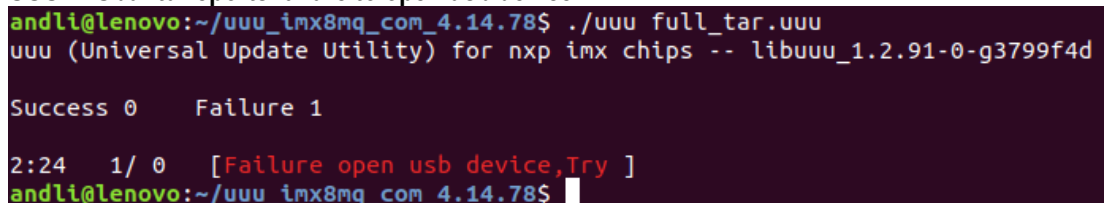


```
3:14>Start Cmd:FB: acmd ${kboot} ${loadaddr} ${initrd_addr} ${fdt_addr}
3:14>Okay
New USB Device Attached at 3:14
3:14>Fail Failure open usb device
←[?25h
```

It could be due to a driver problem. Follow instructions here:

<https://github.com/NXPmicro/mfgtools/wiki/WIN7-User-Guide>

- Windows 7 terminal does not appear as in the screenshots**
 This is because Windows 7 does not support what the UUU tool calls "VT mode" so it defaults to verbose mode which has a lot more printouts and no progress bar.
- UUU in Ubuntu reports failure to open usb device:**



```
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ ./uuu full_tar.uuu
uuu (Universal Update Utility) for nxp imx chips -- libuuu_1.2.91-0-g3799f4d

Success 0    Failure 1

2:24 1/ 0 [Failure open usb device, Try ]
andli@lenovo:~/uuu_imx8mq_com_4.14.78$
```

This happens if the uuu program is not executed with the correct rights. Either use "sudo uuu" or setup udev rules so that sudo rights are not needed. The instructions for how to create the

udev rules are built into the tool so run "uuu -udev" and then follow the steps:

```
andli@lenovo:~/uuu_imx8mq_com_4.14.78$ ./uuu -udev
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="012f", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="0129", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0076", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0054", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0061", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0063", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0071", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="007d", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="15a2", ATTRS{idProduct}=="0080", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="0128", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="0126", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="0135", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="0134", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="1fc9", ATTRS{idProduct}=="012b", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="0525", ATTRS{idProduct}=="b4a4", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="0525", ATTRS{idProduct}=="b4a4", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="0525", ATTRS{idProduct}=="b4a4", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="066f", ATTRS{idProduct}=="9afe", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="066f", ATTRS{idProduct}=="9bff", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="0525", ATTRS{idProduct}=="a4a5", MODE="0666"
SUBSYSTEM=="usb", ATTRS{idVendor}=="18d1", ATTRS{idProduct}=="0d02", MODE="0666"

1: put above udev run into /etc/udev/rules.d/99-uuu.rules
   sudo sh -c "uuu -udev >> /etc/udev/rules.d/99-uuu.rules"
2: update udev rule
   sudo udevadm control --reload-rules
andli@lenovo:~/uuu_imx8mq_com_4.14.78$
```