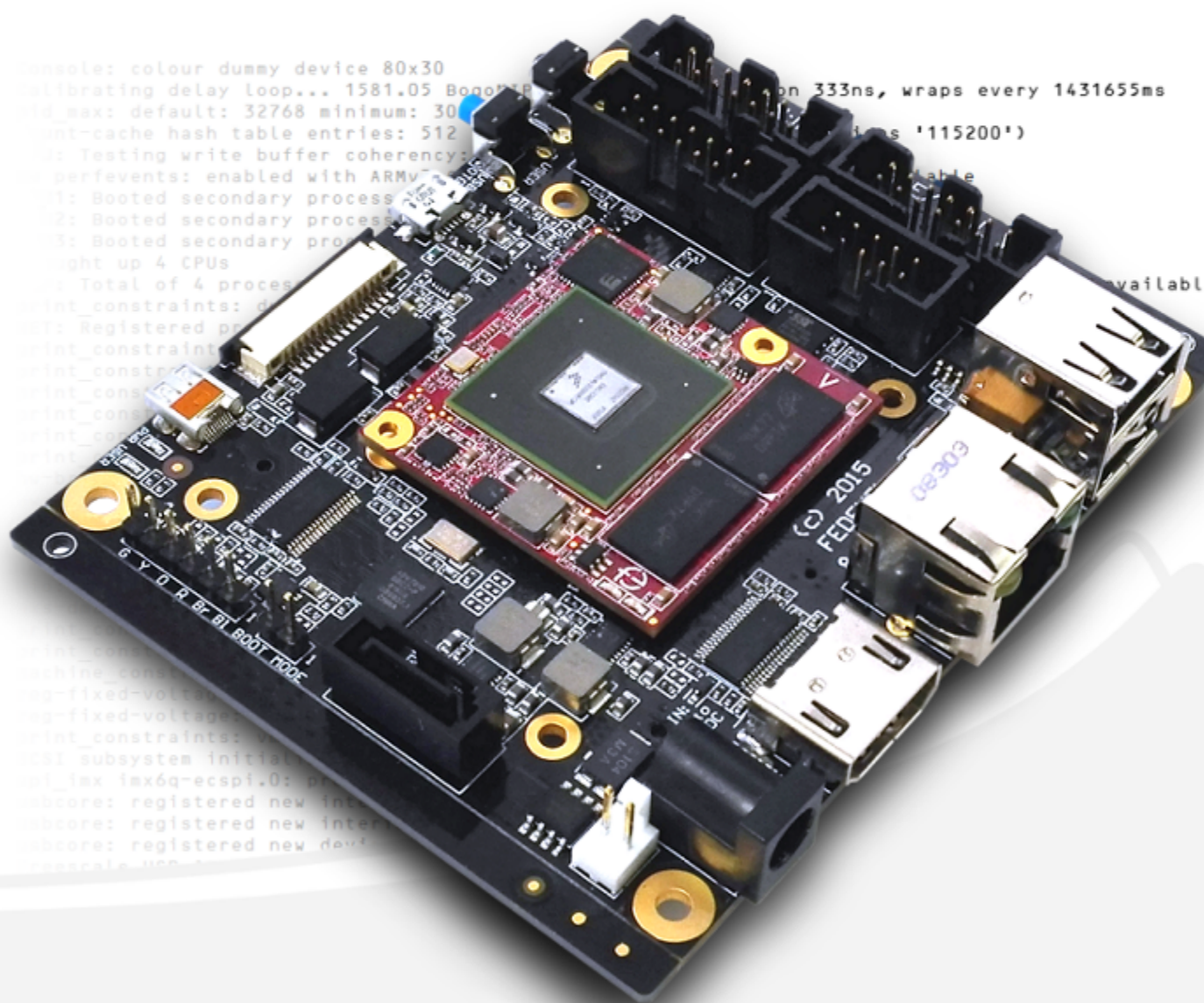


iMX6 TinyRex Development Kit

QUICK GUIDE



About Voipac iMX6 TinyRex Development Kit _____ 4

Packing List _____ 4

Connectors Locations _____ 5

Connecting the Components and Cables _____ 6

The First Steps _____ 7

Controlling the Development Kit Over Serial Line _____ 7

Using External Monitor and USB Keyboard _____ 8

Controlling the Development Kit Over Ethernet (telnet, ssh, ftp, sftp) _____ 8

MfgTool for Booting by USB OTG _____ 10

MfgTool _____ 10

SPI Bootloader _____ 13

*How to Flash SPI1 Bootloader (u-boot-imx6-tinyrex*recovery.imx)* _____ 13

Creating Bootable microSD Card _____ 16

USB Writer _____ 16

Important and Usefull information _____ 19

Product Life Cycle Phases _____ 19

CE Compliance of Voipac Products _____ 19

Technical Support _____ 19

About Voipac iMX6 TinyRex Development Kit

iMX6 TinyRex Development kit is a complete development environment designed to present the functionality, connectivity and performance of the iMX6 TinyRex Modules, ideal for evaluation and application development purposes. It is targeting skilled development teams building multimedia or other demanding and compact solutions.

This Quick Guide shows, how to flash the module using MfgTool program, load Yocto Project Linux image on microSD card using USB writer. Programs run under Windows XP/7/8/10. More information available at: [imx6 tinyrex at wiki.voipac.com](http://wiki.voipac.com/imx6_tinyrex).

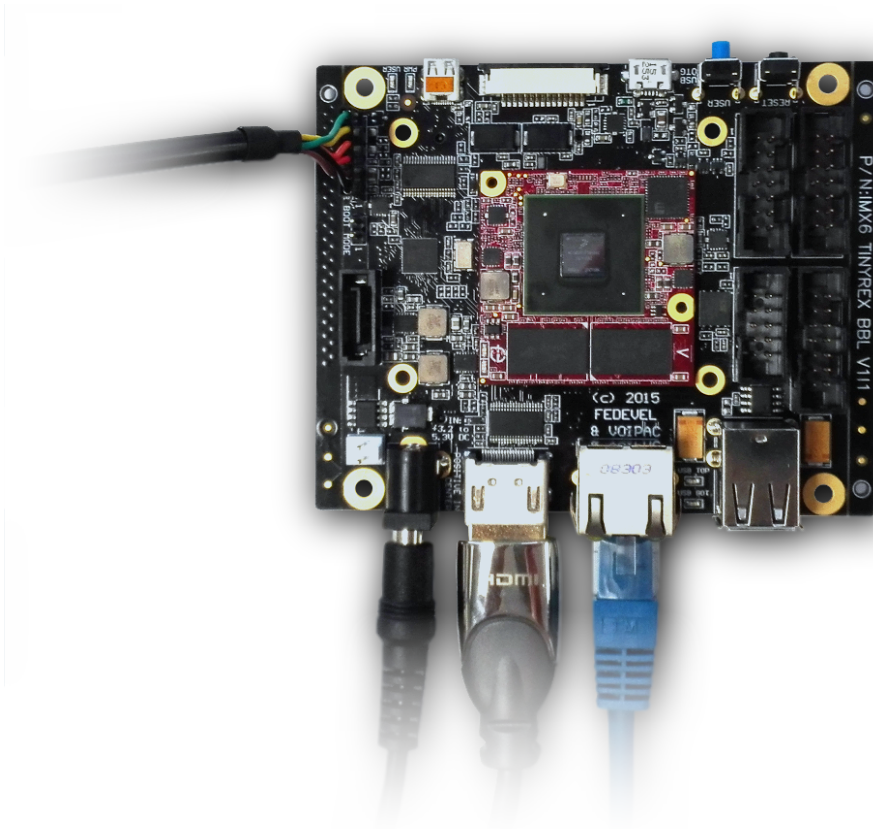
Packing List

COMPONENTS	QUANTITY
iMX6 TinyRex Base Board Lite	1
iMX6 TinyRex Module	1
8GB microSDHC Class 4 memory card	1
Aluminum 35 x 35 x 10mm heatsink	1
TTL-232R-3V3 cable	1
iMX6 TinyRex Documentation on USB clip	1
Aluminium case set	1
HDMI High Speed CAT.2 cable with Ethernet	1
SFTP CAT.6 Patch Ethernet cable	1
5V Power supply	1
100pin Header	3
Spacer with bolt and nut	2
Quick Guide brochure	1
Yocto Project Linux OS preinstalled. (Android 7.1 preinstalled upon request)	

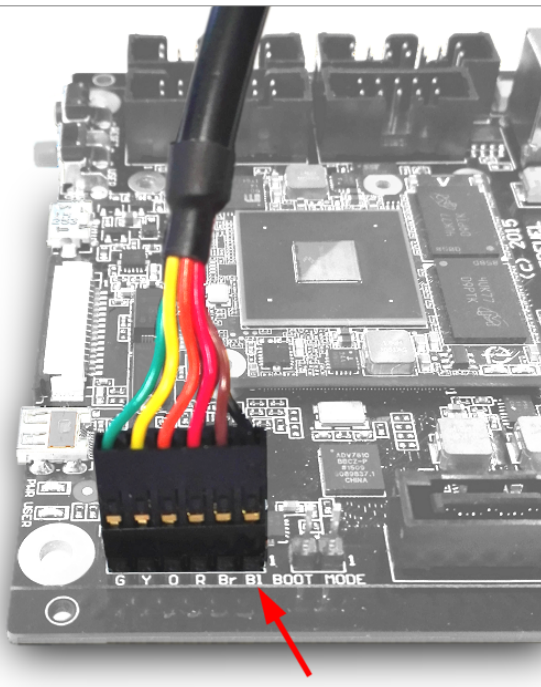


Connecting the Components and Cables

Prepare base board and plug in (bootable) microSD card, TTL-232R-3V3 FTDI cable, Ethernet cable, HDMI cable and other devices or interfaces you need. Plug the power supply connector in.

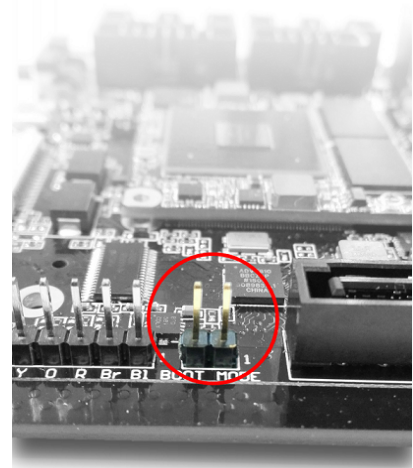


IMPORTANT! Be careful when connecting TTL-232R FTDI cable to the base board. Check if the cable conductor 1 (black wire) is connected to Pin 1 (Header connector J10 – TTL-232R FTDI) on the base board.



PIN 1

To boot from microSD Card, make sure that BOOT_MODE jumper is not present.



The First Steps

The development kit is supplied with bootloader and Yocto Project Linux distribution preinstalled on microSD card by default. The development kit can be controlled over:

Controlling the Development Kit over Serial Line

Recommended HW:

- a) PC with USB port
- b) Voipac iMX6 TinyRex Development Kit
- c) [TTL-232R-3V3 \(FTDI \) cable](#)

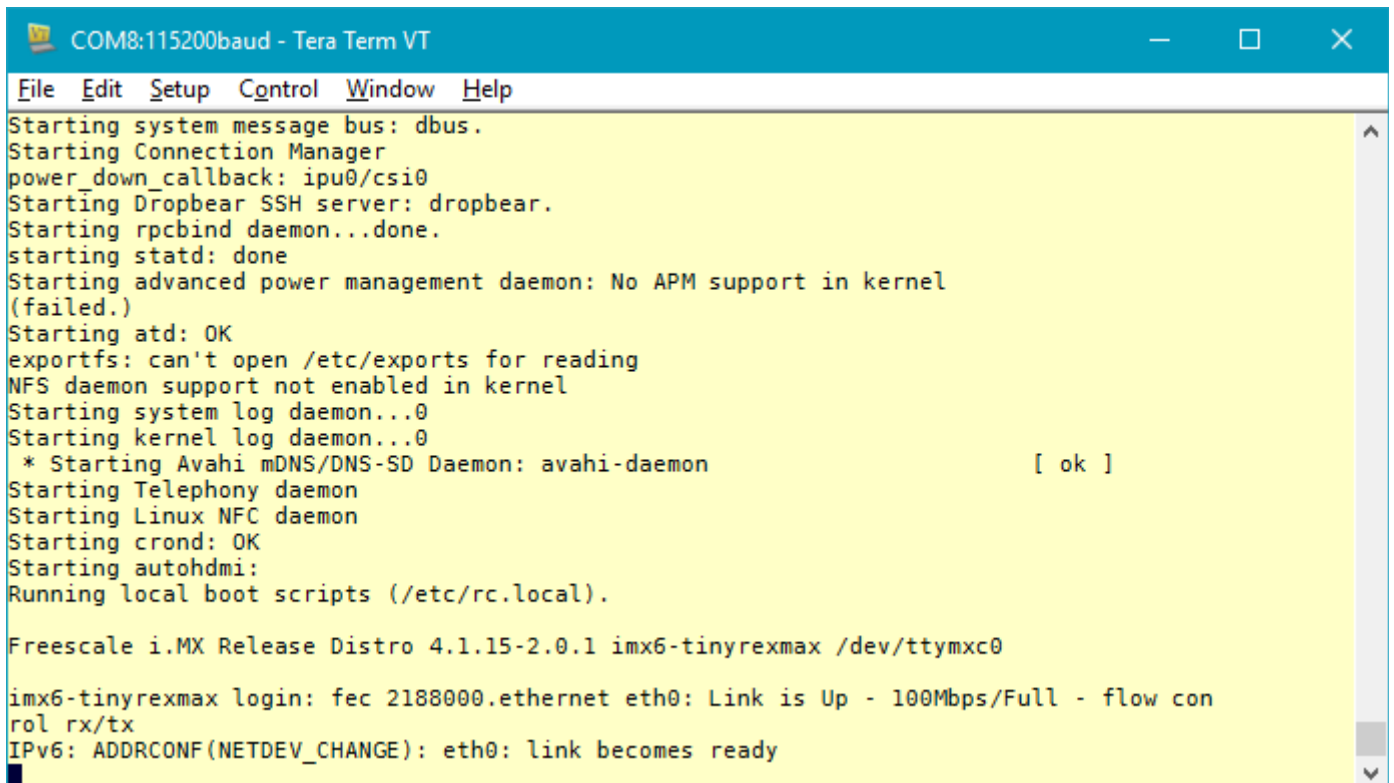
Recommended SW:

Serial line terminal (PUTTY, Minicom, Ckermit, Hyperterminal, TeraTerm, ...)

Default serial port settings:

Speed (baud):	115200
Data bits:	8
Stop bits:	1
Parity:	None
Flow control:	None

Controlling the development kit using TeraTerm



```
COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
Starting system message bus: dbus.
Starting Connection Manager
power_down_callback: ipu0/csi0
Starting Dropbear SSH server: dropbear.
Starting rpcbind daemon...done.
starting statd: done
Starting advanced power management daemon: No APM support in kernel
(failed.)
Starting atd: OK
exportfs: can't open /etc/exports for reading
NFS daemon support not enabled in kernel
Starting system log daemon...0
Starting kernel log daemon...0
* Starting Avahi mDNS/DNS-SD Daemon: avahi-daemon [ ok ]
Starting Telephony daemon
Starting Linux NFC daemon
Starting crond: OK
Starting autohdmi:
Running local boot scripts (/etc/rc.local).

Freescale i.MX Release Distro 4.1.15-2.0.1 imx6-tinyrexmax /dev/ttymx0

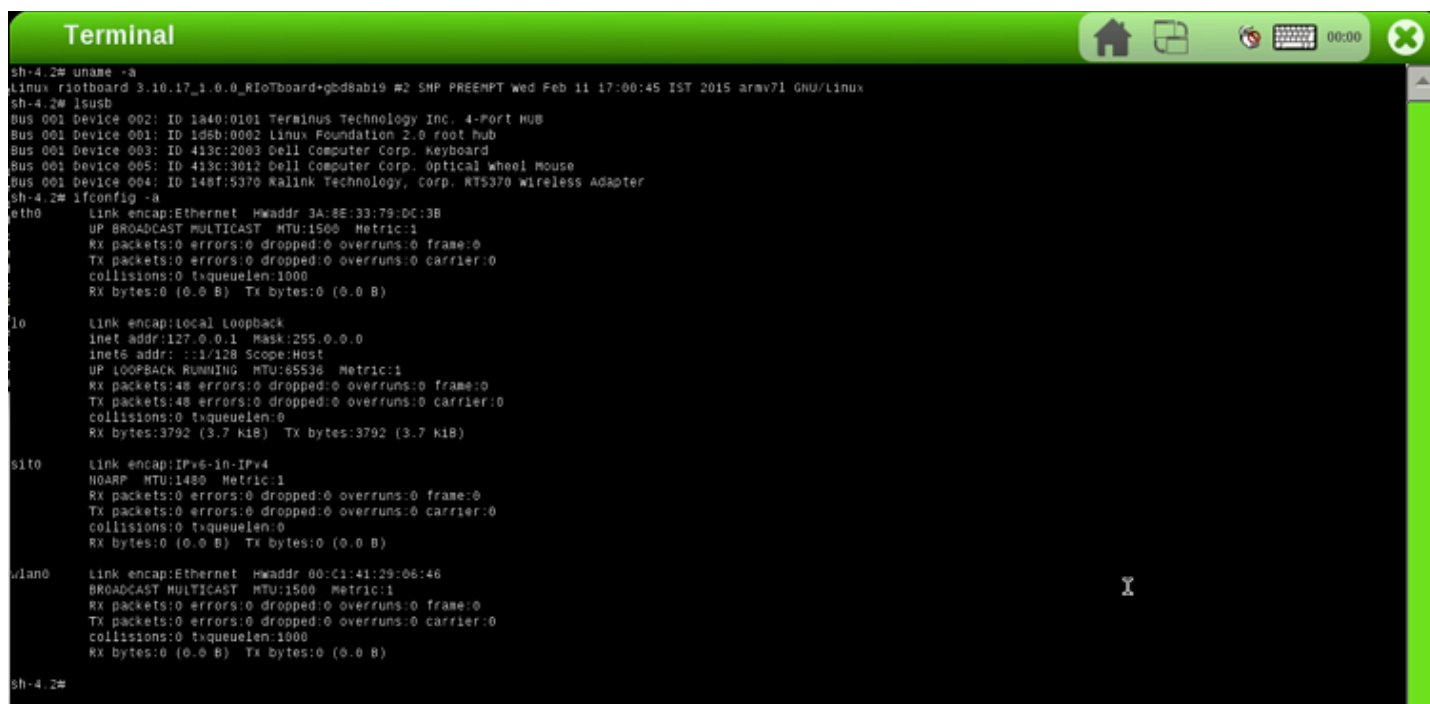
imx6-tinyrexmax login: fec 2188000.ethernet eth0: Link is Up - 100Mbps/Full - flow con
rol rx/tx
IPv6: ADDRCONF(NETDEV_CHANGE): eth0: link becomes ready
```

Using External Monitor and USB Keyboard

Recommended HW:

- External monitor with HDMI connector (HDMI to VGA adapter is required for VGA monitor)
- Voipac iMX6 TinyRex Development Kit
- [HDMI High Speed cable](#)
- USB keyboard and USB mouse

Controlling the development kit using external monitor and USB keyboard



```
Terminal
sh-4.2# uname -a
Linux riotboard 3.10.17_1.0.0_RIoTboard-gbd8ab19 #2 SMP PREEMPT wed Feb 11 17:00:45 IST 2015 arav71 GNU/Linux
sh-4.2# lsusb
BUS 001 Device 002: ID 1a40:0101 Terminus Technology Inc. 4-Port HUB
BUS 001 Device 001: ID 1d5b:0002 Linux Foundation 2.0 root hub
BUS 001 Device 003: ID 413c:2003 Dell Computer Corp. keyboard
BUS 001 Device 005: ID 413c:3012 Dell Computer Corp. Optical wheel Mouse
BUS 001 Device 004: ID 148f:5370 Ralink Technology, Corp. RT5370 wireless Adapter
sh-4.2# ifconfig -a
eth0      Link encap:Ethernet  HWaddr 3A:8E:33:79:DC:3B
          UP BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  Tx bytes:0 (0.0 B)

lo        Link encap:local loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:48 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:3792 (3.7 KiB)  Tx bytes:3792 (3.7 KiB)

sit0      Link encap:IPv6-in-IPv4
          NOARP  MTU:1480  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 B)  Tx bytes:0 (0.0 B)

vlan0     Link encap:Ethernet  HWaddr 00:C1:41:29:06:46
          BROADCAST MULTICAST  MTU:1500  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  Tx bytes:0 (0.0 B)

sh-4.2#
```

Illustration Photo

Controlling the Development Kit over Ethernet (telnet, ssh, ftp, sftp)

Recommended HW:

- PC with Ethernet
- Voipac iMX6 TinyRex Development Kit
- [Ethernet cable](#)

Recommended SW:

- Telnet client (Telnet, PUTTY, ...)
- SSH client (SSH,PUTTY, ...)
- FTP client (FTP, Filezilla, BareFTP, ...)
- SFTP client (Filezilla, PUTTY, WinSCP, ...)



IMPORTANT!

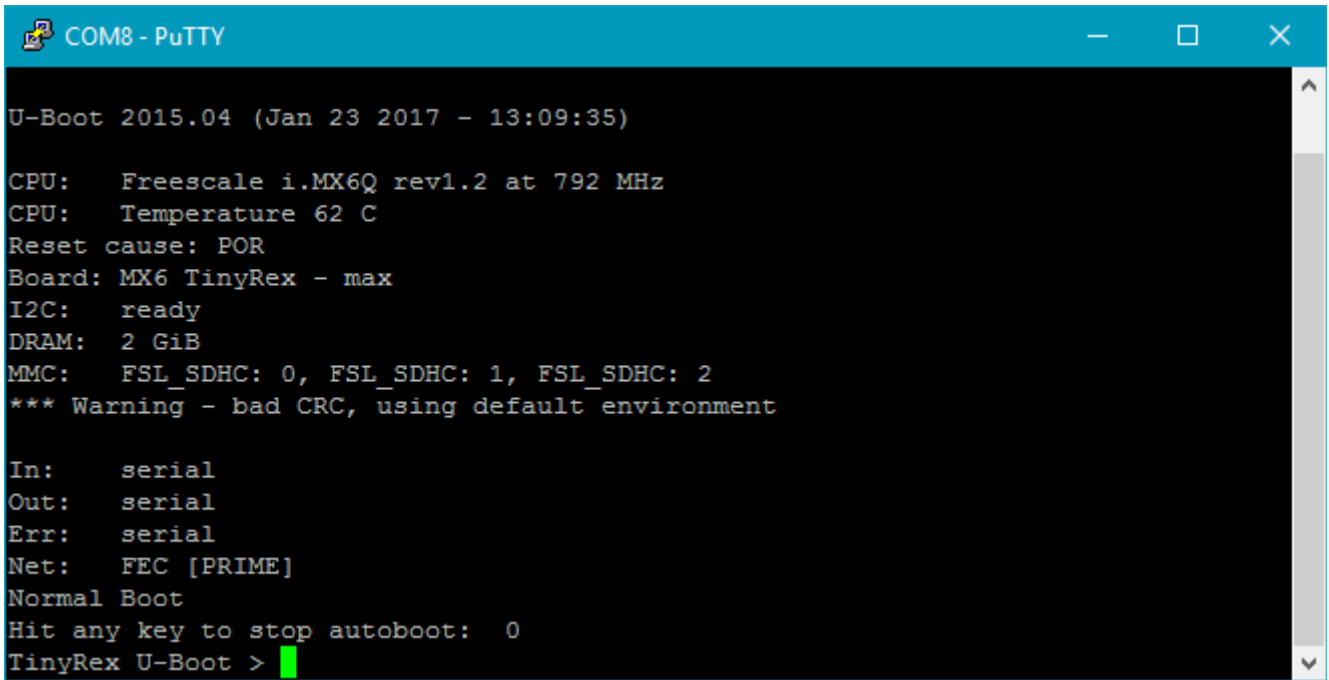
iMX6 TinyRex Development Kit is shipped with empty password.

Login is: root

The default IP address is dynamic and setup by your dhcp server upon boot.

SSH, SFTP require root password to be set up. ("passwd" command)
FTP, SFTP are recommended only for data transfers (binary mode is recommended).

Controlling the development kit using PUTTY connected to Serial Line.



```
COM8 - PuTTY
U-Boot 2015.04 (Jan 23 2017 - 13:09:35)

CPU:   Freescale i.MX6Q rev1.2 at 792 MHz
CPU:   Temperature 62 C
Reset cause: POR
Board: MX6 TinyRex - max
I2C:   ready
DRAM:  2 GiB
MMC:   FSL_SDHC: 0, FSL_SDHC: 1, FSL_SDHC: 2
*** Warning - bad CRC, using default environment

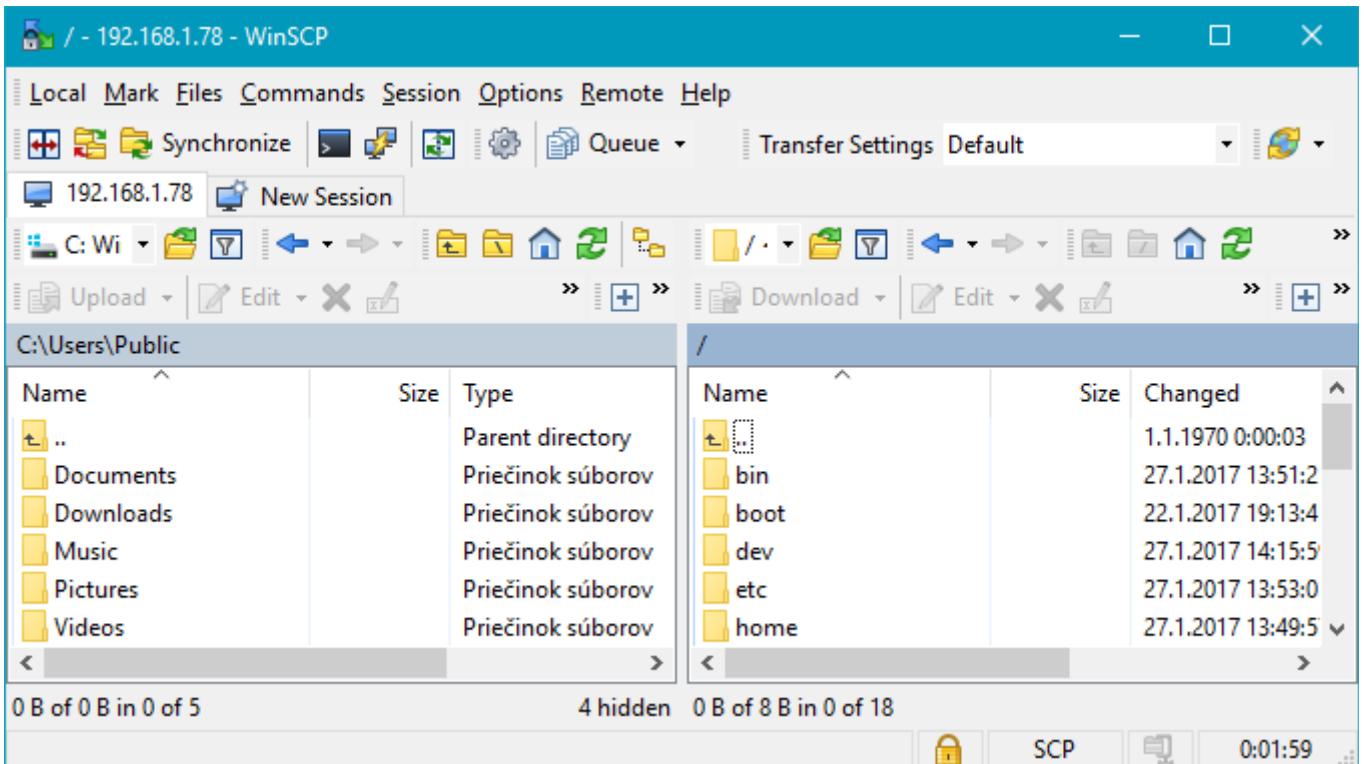
In:    serial
Out:   serial
Err:   serial
Net:   FEC [PRIME]
Normal Boot
Hit any key to stop autoboot:  0
TinyRex U-Boot > █
```

Controlling the development kit using PUTTY SSH client.



```
192.168.1.78 - PuTTY
login as: root
root@imx6-tinyrexmax:~# █
```

Controlling the development kit using WinSCP.



WinSCP interface showing local and remote file systems. The local side shows the C:\Users\Public directory with subdirectories like Documents, Downloads, Music, Pictures, and Videos. The remote side shows the root directory with subdirectories like bin, boot, dev, etc, and home.

Name	Size	Type
..		Parent directory
Documents		Priečnik súborov
Downloads		Priečnik súborov
Music		Priečnik súborov
Pictures		Priečnik súborov
Videos		Priečnik súborov

Name	Size	Changed
bin		1.1.1970 0:00:03
boot		27.1.2017 13:51:2
dev		22.1.2017 19:13:4
etc		27.1.2017 14:15:5
home		27.1.2017 13:53:0
		27.1.2017 13:49:5

MfgTool for Booting by USB OTG

MfgTool

U-boot is a bootloader responsible for hardware initialization, loading and booting Linux kernel. It is also used for module flashing. Following example is for iMX6 TinyRex Module in Max configuration.

Recommended HW:

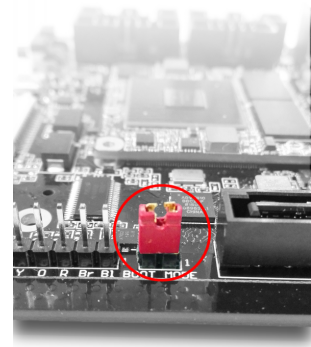
- a) PC with USB port
- b) Voipac iMX6 TinyRex Development Kit
- c) USB to Micro-USB cable

Recommended SW:

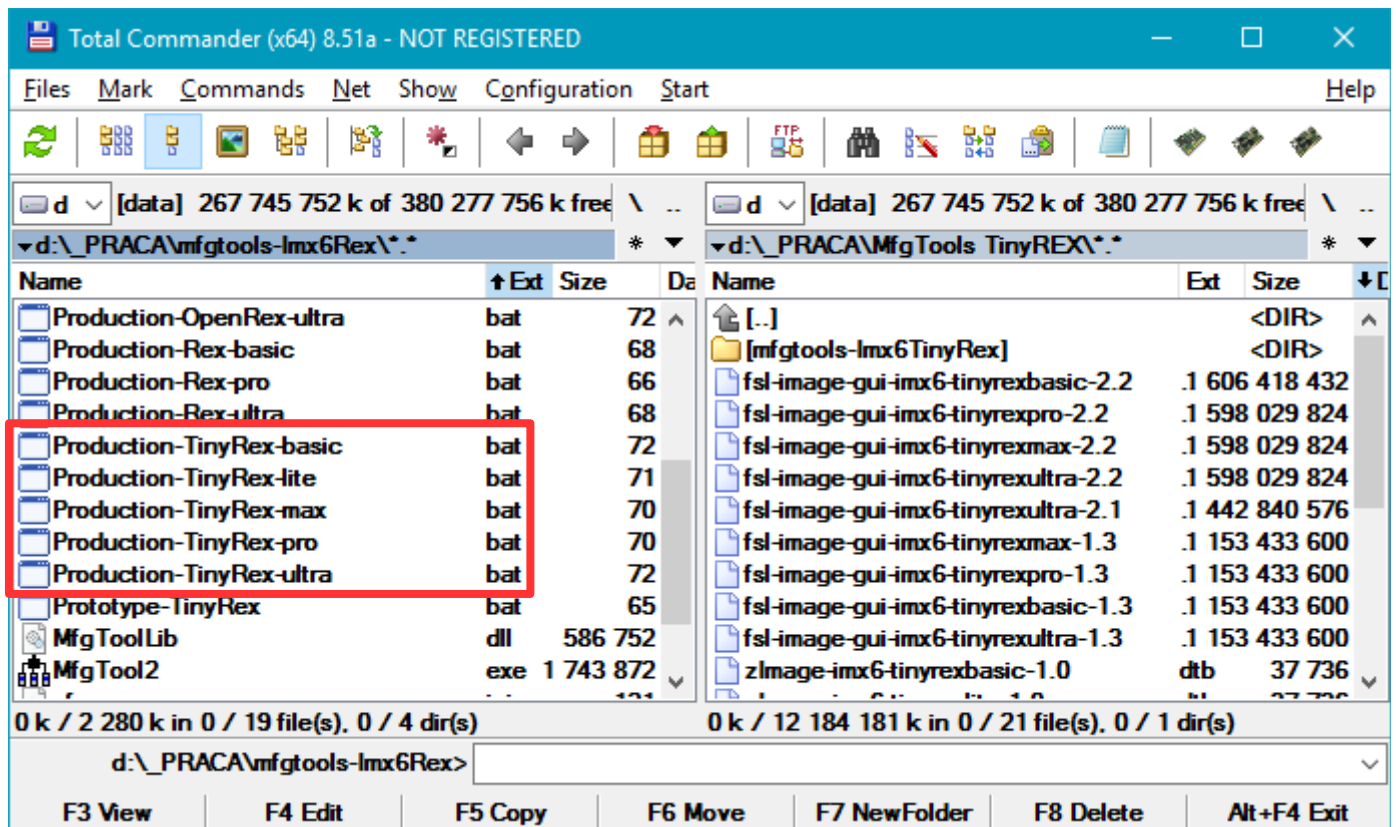
- [MfgTool](#)
- Serial line terminal

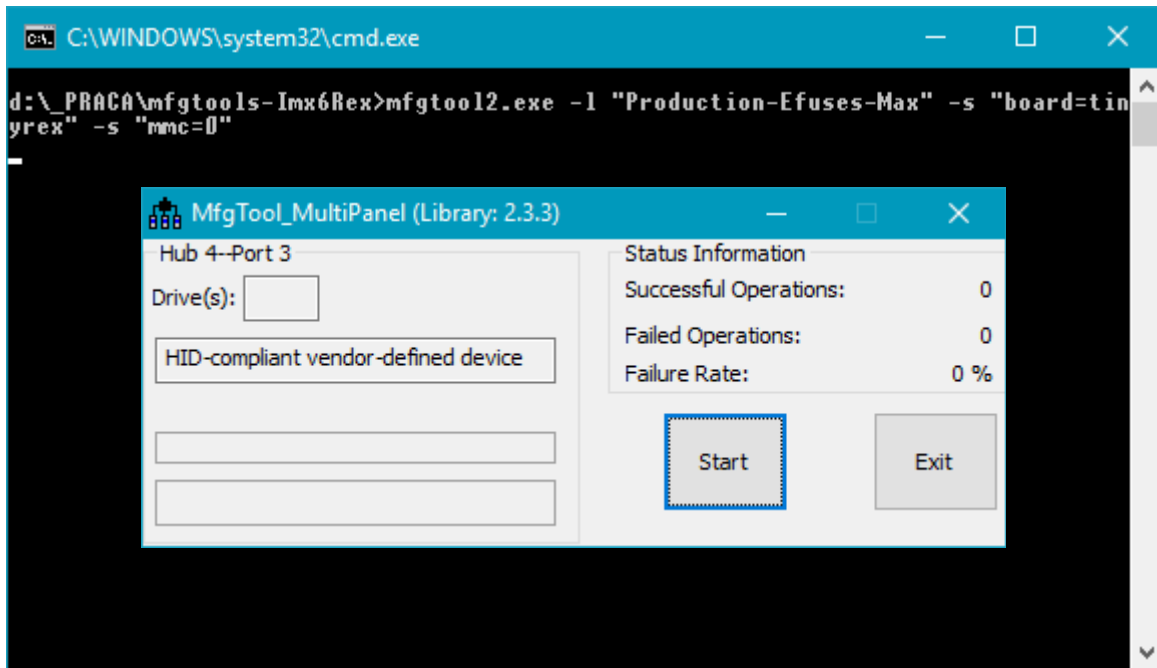
1 STEP Connect USB cable to USB port on your computer and Development Kit USB OTG port.

2 STEP Short BOOT_MODE jumper on iMX6 TinyRex Base Board Lite and Power on.



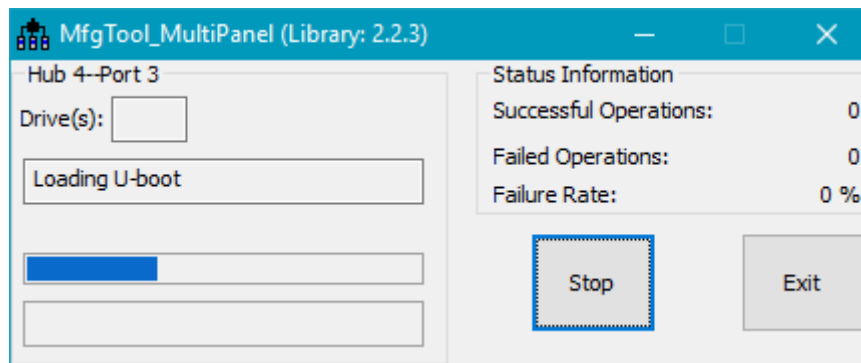
3 STEP Execute Manufacturing toolkit at host PC. Open an appropriate BAT file.





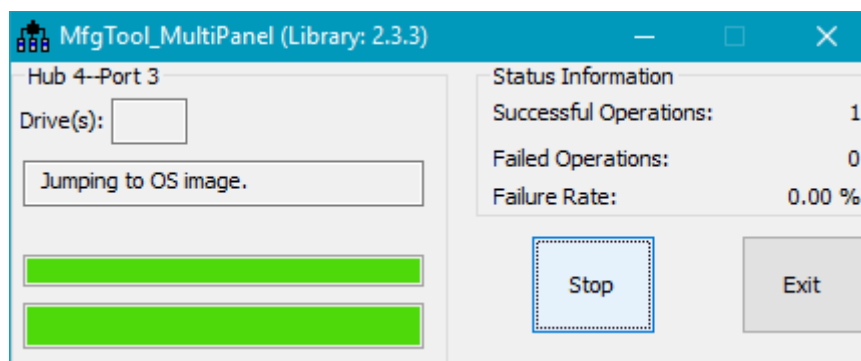
4 Press Start button (The iMX6 TinyRex Module will boot firmware loaded over USB).

STEP



5 Wait until firmware (u-boot) is booted (serial terminal).

STEP




```

COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
U-Boot 2015.04 (Jan 23 2017 - 13:09:35)

CPU: Freescale i.MX6Q rev1.2 at 792 MHz
CPU: Temperature 44 C
Reset cause: POR
Board: MX6 TinyRex - max
I2C: ready
DRAM: 2 GiB
MMC: FSL_SDHC: 0, FSL_SDHC: 1, FSL_SDHC: 2
*** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
Net: FEC [PRIME]
Normal Boot
Hit any key to stop autoboot: 0
TinyRex U-Boot >

```

 **These additional steps are not required for the modules supplied as standard !**

6 STEP Burn efuses over serial terminal. **Only for “VIRGIN” Modules !**

Efuses Boot From SD3:

```

fuse prog 0 5 0x00003040
fuse prog 0 6 0x00000010

```

MAC Address (For example 00:0D:15:00:D1:75):

```

fuse prog 4 3 0x000d
fuse prog 4 2 0x1500d175

```

Efuses Boot From SD3 + SPI1 recovery:

```

fuse prog 0 5 0x48003040
fuse prog 0 6 0x00000010

```

```

COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
Hit any key to stop autoboot: 0
TinyRex U-Boot > fuse prog 0 5 0x48003040
Programming bank 0 word 0x00000005 to 0x48003040...
Warning: Programming fuses is an irreversible operation!
        This may brick your system.
        Use this command only if you are sure of what you are doing!

Really perform this fuse programming? <y/N>
y
TinyRex U-Boot > fuse prog 0 6 0x00000010
Programming bank 0 word 0x00000006 to 0x00000010...
Warning: Programming fuses is an irreversible operation!
        This may brick your system.
        Use this command only if you are sure of what you are doing!

Really perform this fuse programming? <y/N>
y
TinyRex U-Boot >

```




This operation is not reversible and should be executed carefully. The iMX6 TinyRex COM must be replaced in the case of error.



**BE AWARE THAT E-FUSES PROGRAMING IS A NON REVERSAL PROCESS !
WARRANTY CLAIM CAUSED BY IMPROPER E-FUSES PROGRAMMING WILL
NOT BE ACCEPTED !**

7
STEP

Power off iMX6 TinyRex Base Board Lite.

8
STEP

Remove BOOT_MODE jumper.

SPI Bootloader

*How to Flash SPI1 Bootloader (u-boot-imx6-tinyrex*recovery.imx)*

This part of QuickGuide shows how to flash u-boot and configure the module to run, in addition to the SD card, from iMX6 TinyRex Base Board Lite SPI flash. Updating of existing modules with burned efuses possible.

Recommended HW:

- a) PC with USB port
- b) Voipac iMX6 TinyRex Development Kit
- c) USB to Micro-USB cable

Recommended SW:

- [MfgTool](#) (only for “VIRGIN” modules)
- Serial line terminal
- [Appropriate files](#)

* Select appropriate file for specified configuration:

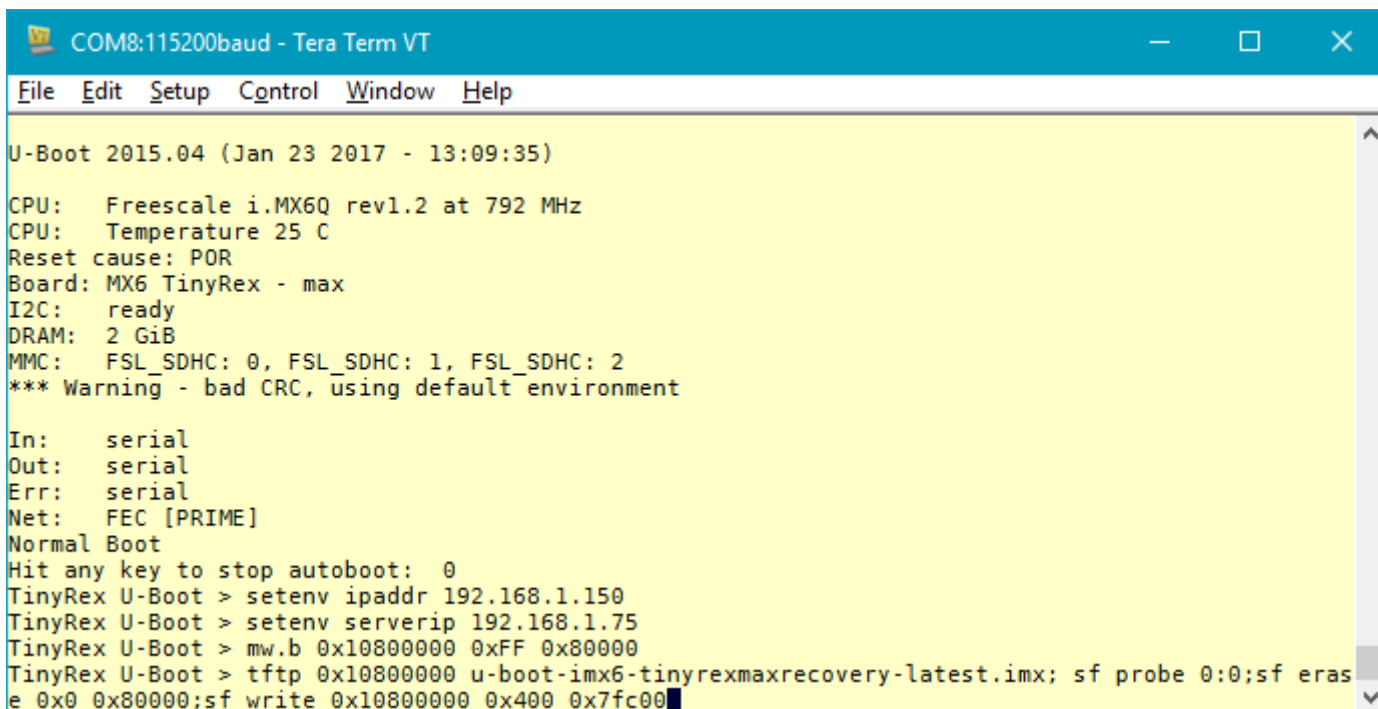
- | | |
|------------------------------|---|
| - iMX6 TinyRex Module Ultra: | u-boot-imx6-tinyrexultrarecovery-latest.imx |
| - iMX6 TinyRex Module Max: | u-boot-imx6-tinyrexmaxrecovery-latest.imx |
| - iMX6 TinyRex Module Pro: | u-boot-imx6-tinyrexprorecovery-latest.imx |
| - iMX6 TinyRex Module Basic: | u-boot-imx6-tinyrexbasicrecovery-latest.imx |

1
STEP

Open MfgTool appropriate BAT file to load bootloader over USB OTG port. Use this step only for “VIRGIN” module as described in the above [MfgTool Chapter](#) of this document.

2 **STEP** Stop autoboot in your serial line terminal. Type or paste commands separately to download appropriate bootloader file from TFTP server where the bootloader file is located. Following example is for iMX6 TinyRex Max Module.

```
setenv ipaddr 192.168.1.150
setenv serverip 192.168.1.75
mw.b 0x10800000 0xFF 0x80000
tftp 0x10800000 u-boot-imx6-tinyrexmaxrecovery-latest.imx; sf probe 0:0;sf erase
0x0 0x80000;sf write 0x10800000 0x400 0x7fc00
```

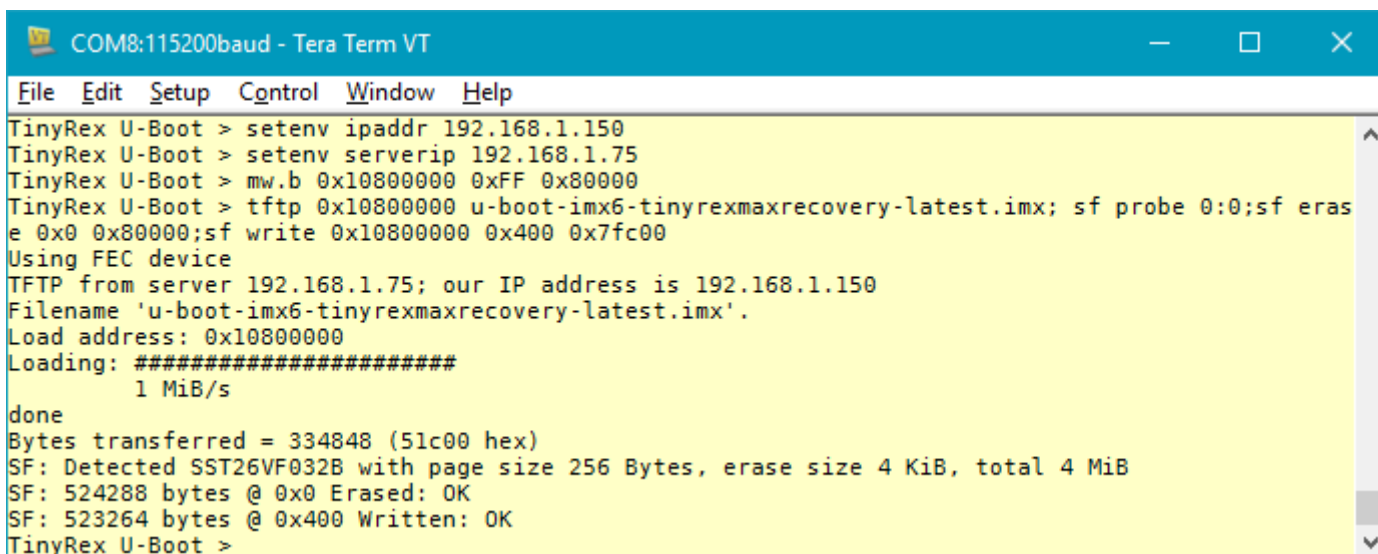


```
COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
U-Boot 2015.04 (Jan 23 2017 - 13:09:35)

CPU: Freescale i.MX6Q rev1.2 at 792 MHz
CPU: Temperature 25 C
Reset cause: POR
Board: MX6 TinyRex - max
I2C: ready
DRAM: 2 GiB
MMC: FSL_SDHC: 0, FSL_SDHC: 1, FSL_SDHC: 2
*** Warning - bad CRC, using default environment

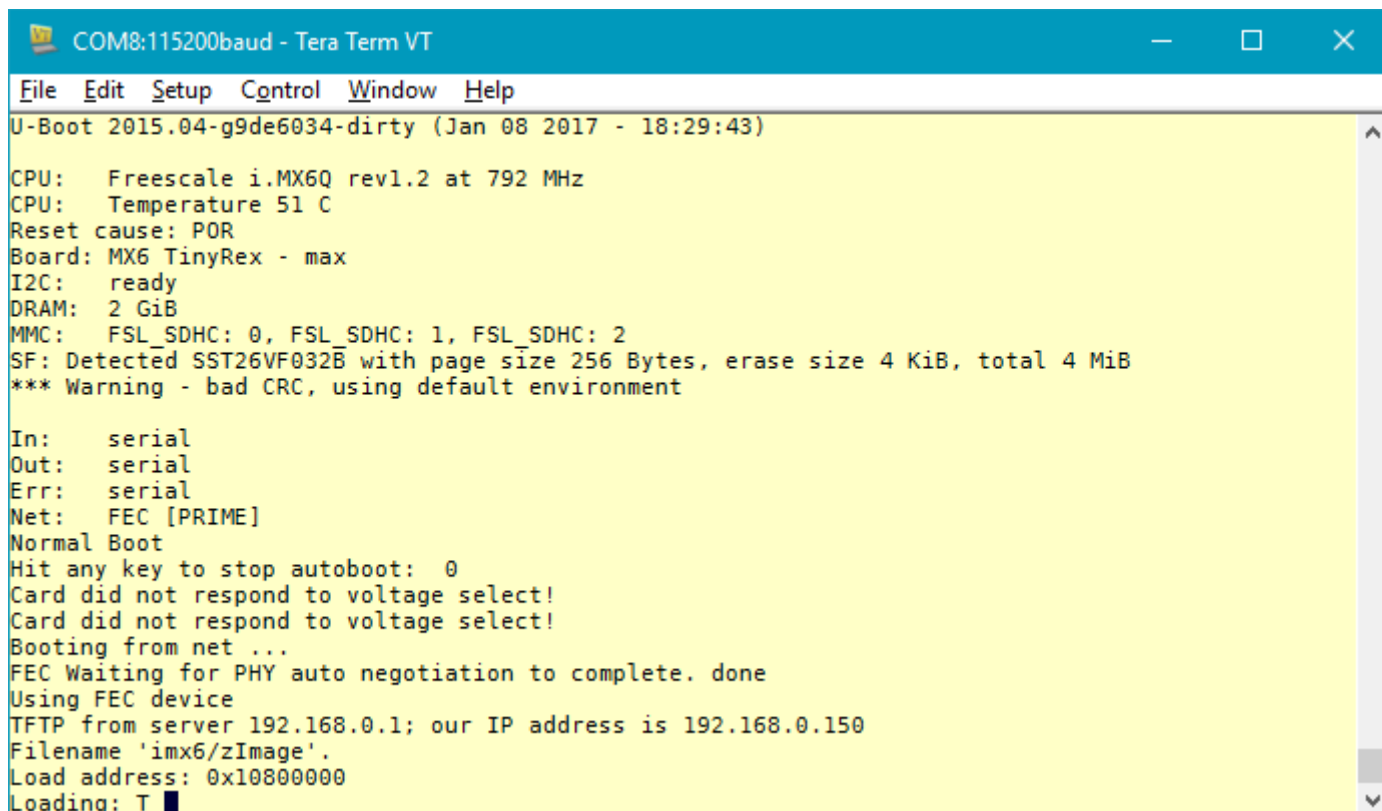
In: serial
Out: serial
Err: serial
Net: FEC [PRIME]
Normal Boot
Hit any key to stop autoboot: 0
TinyRex U-Boot > setenv ipaddr 192.168.1.150
TinyRex U-Boot > setenv serverip 192.168.1.75
TinyRex U-Boot > mw.b 0x10800000 0xFF 0x80000
TinyRex U-Boot > tftp 0x10800000 u-boot-imx6-tinyrexmaxrecovery-latest.imx; sf probe 0:0;sf eras
e 0x0 0x80000;sf write 0x10800000 0x400 0x7fc00
```

The bootloader is written to the base board SPI Flash after automatic download from the TFTP Server.



```
COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
TinyRex U-Boot > setenv ipaddr 192.168.1.150
TinyRex U-Boot > setenv serverip 192.168.1.75
TinyRex U-Boot > mw.b 0x10800000 0xFF 0x80000
TinyRex U-Boot > tftp 0x10800000 u-boot-imx6-tinyrexmaxrecovery-latest.imx; sf probe 0:0;sf eras
e 0x0 0x80000;sf write 0x10800000 0x400 0x7fc00
Using FEC device
TFTP from server 192.168.1.75; our IP address is 192.168.1.150
Filename 'u-boot-imx6-tinyrexmaxrecovery-latest.imx'.
Load address: 0x10800000
Loading: #####
          1 MiB/s
done
Bytes transferred = 334848 (51c00 hex)
SF: Detected SST26VF032B with page size 256 Bytes, erase size 4 KiB, total 4 MiB
SF: 524288 bytes @ 0x0 Erased: OK
SF: 523264 bytes @ 0x400 Written: OK
TinyRex U-Boot >
```

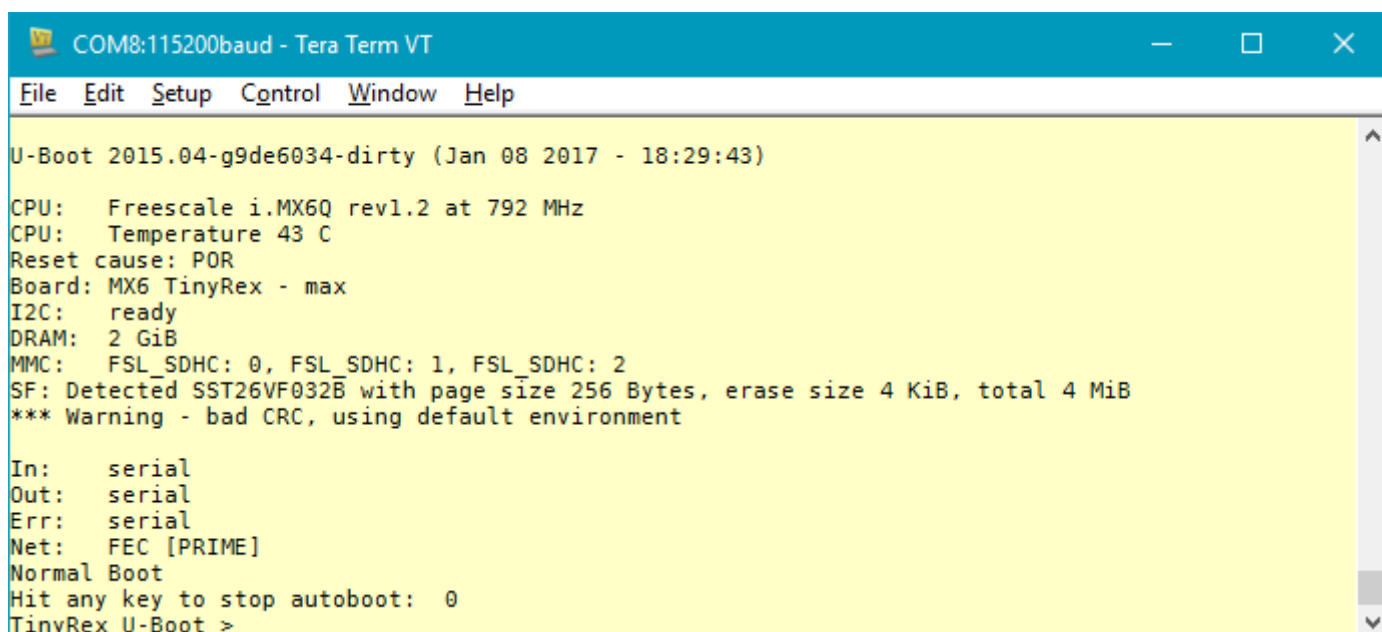
- 3** **STEP** Reset the base board. iMX6 TinyRex Module will start booting from SPI Flash. The bootloader will defaultly start to download Image from the TFTP Server. To change bootloader environment, stop autoboot.



```
COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
U-Boot 2015.04-g9de6034-dirty (Jan 08 2017 - 18:29:43)

CPU: Freescale i.MX6Q rev1.2 at 792 MHz
CPU: Temperature 51 C
Reset cause: POR
Board: MX6 TinyRex - max
I2C: ready
DRAM: 2 GiB
MMC: FSL_SDHC: 0, FSL_SDHC: 1, FSL_SDHC: 2
SF: Detected SST26VF032B with page size 256 Bytes, erase size 4 KiB, total 4 MiB
*** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
Net: FEC [PRIME]
Normal Boot
Hit any key to stop autoboot: 0
Card did not respond to voltage select!
Card did not respond to voltage select!
Booting from net ...
FEC Waiting for PHY auto negotiation to complete. done
Using FEC device
TFTP from server 192.168.0.1; our IP address is 192.168.0.150
Filename 'imx6/zImage'.
Load address: 0x10800000
Loading: T
```



```
COM8:115200baud - Tera Term VT
File Edit Setup Control Window Help
U-Boot 2015.04-g9de6034-dirty (Jan 08 2017 - 18:29:43)

CPU: Freescale i.MX6Q rev1.2 at 792 MHz
CPU: Temperature 43 C
Reset cause: POR
Board: MX6 TinyRex - max
I2C: ready
DRAM: 2 GiB
MMC: FSL_SDHC: 0, FSL_SDHC: 1, FSL_SDHC: 2
SF: Detected SST26VF032B with page size 256 Bytes, erase size 4 KiB, total 4 MiB
*** Warning - bad CRC, using default environment

In: serial
Out: serial
Err: serial
Net: FEC [PRIME]
Normal Boot
Hit any key to stop autoboot: 0
TinyRex U-Boot >
```

Creating Bootable microSD Card

USB Writer

Following example is for iMX6 TinyRex Max Module.

Recommended HW:

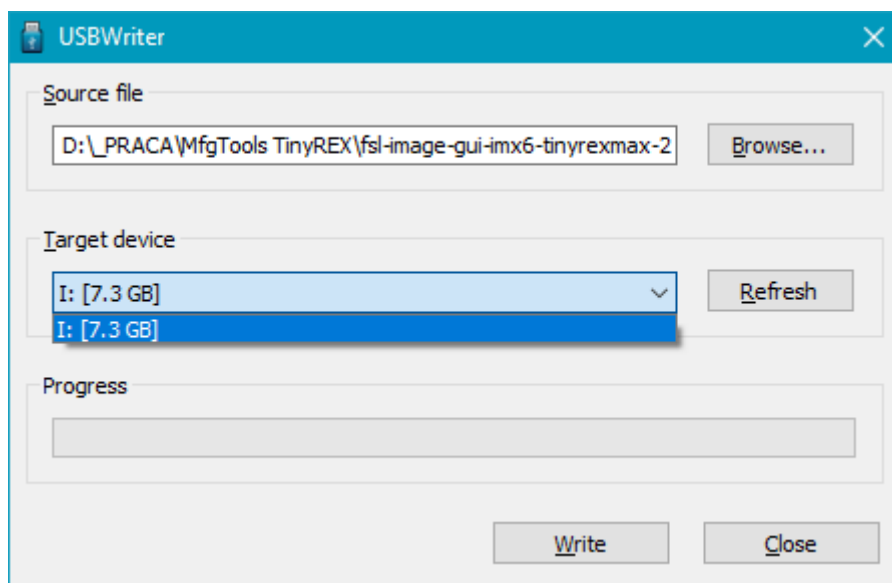
- a) PC with microSD port
- b) microSD card

Recommended SW:

- [USBWriter](#)
- [Appropriate Image files](#)

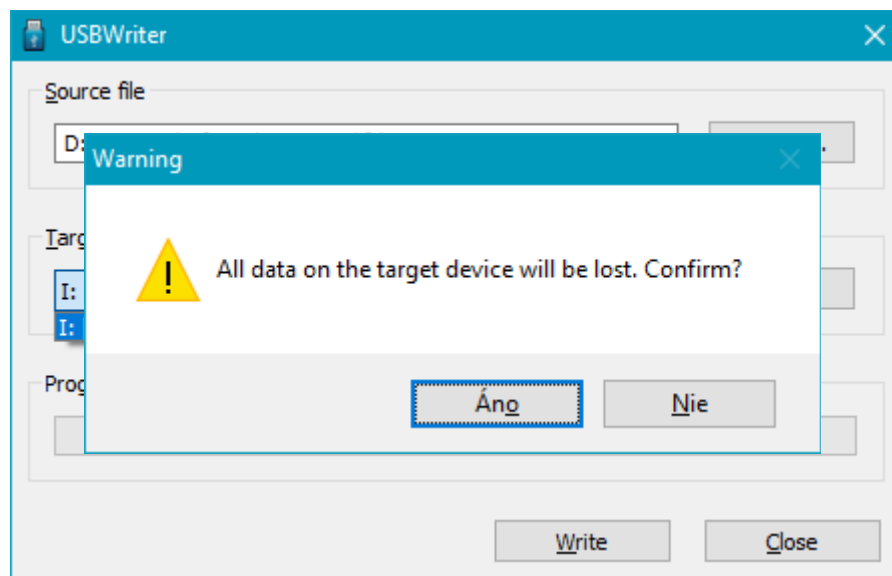
1
STEP

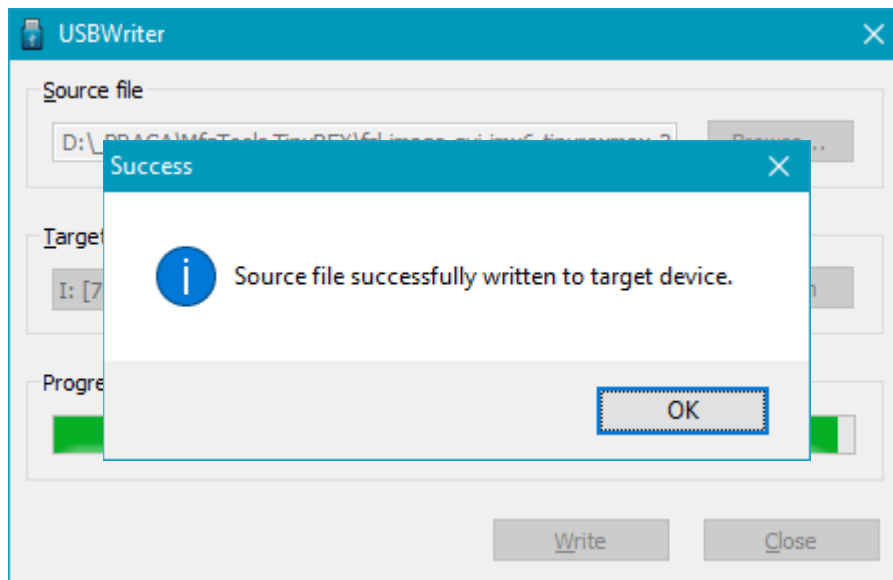
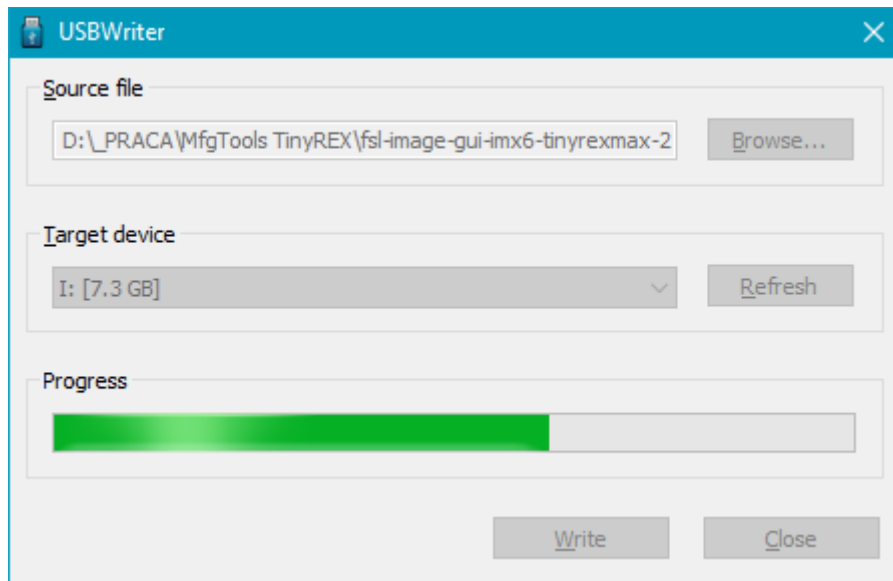
Open USBWriter. Browse source file (appropriate fsl-image). Select target device.



2
STEP

Click on Write button and confirm the procedure.





Bootable microSD Card is now created and prepared for use.

Notes

Important and Usefull information

Product Life Cycle Phases

Voipac products life cycles are divided into 4 phases:

- **INTRODUCTION PHASE**, approximately the first 6-12 months.

The last software issues are still being resolved.
Product in this stage is the most suitable for new designs.

- **ACTIVE PHASE**, the first 1-3 years following the product introduction.

Product software packages are stable, additional functions, OS and GUI are being released.
Product in this stage is suitable for new designs.

- **MATURITY PHASE**, approximately the first 4-6 years after the introduction.

Products are shipped in volumes, additional functions additions declines.
Product in this stage is no longer recommended for new designs.

- **EOL PHASE**, approximately 7-10 years after the introduction.

Used components availability decreases, although product may still be purchased under specific circumstances.
The Last Time Buy notification is send to all product users app. 6 months prior to product discontinuation.
Components stocking service for discontinued products and manufacturing of further production batches is available.

To find out the specific product life cycle phase, visit its [product](#) page and check the title color.

CE Compliance of Voipac Products

The CE label is a mandatory conformity mark for complex electronic devices placed on the market in the European Economic Area and each product sold within the EU needs a CE Certificate of Conformance that ensures that the product conforms to the essential requirements of the applicable EC directives.

However, if such complex electronic devices are produced for further processing by the industry, skilled development teams or system integrators, they do not need to observe the above mentioned CE requirements and consequently do not need any identification either. This applies to the Voipac Computers On Module, because these are not used as stand-alone devices by the general public.

To make sure that Voipac COMs can be used in CE marked devices, they are designed to obey the EC directives and the standard configuration COMs manufactured by Voipac are tested for Electromagnetic Interference and operating temperature ranges plugged in corresponding Base Board and enclosed in a standard Aluminium case provided with Voipac development kits.

Technical Support

HW & SW support: support@voipac.com

Warranty claims: warranty.claim@voipac.com

All of the relevant communication between the customer and Voipac should be executed via e-mails preferably.
Response time is up to 48 hours, except state holidays and weekends.
Voipac working hours are: 8:00 - 17:00, Monday - Friday.

Before contacting support, please read the following for the basic information about how to work with a development kit:
www.voipac.com/#Downloads
http://www.voipac.com/downloads/imx/iMX6_TinyRex
<http://wiki.voipac.com/xwiki/bin/view/imx6+tinyrex/>
<http://www.imx6rex.com>

We provide paid support for your new designs when it comes to the special drivers for the peripherals not included in the Voipac development kits, design of your own base boards, prototyping, or even new products development.

Please contact: support@voipac.com for more info.

Warranty:

VOIPAC TECHNOLOGIES s.r.o. Does Not Bear Responsibility for the Following:

- Failure of a product resulting from misuse, accident, modification, unsuitable operating environment, or improper maintenance by user
- Unless otherwise agreed in written, a product does not include technical support and the customer may be able to purchase technical support under separate agreement
- Any technical or other support provided under warranty by VOIPAC TECHNOLOGIES s.r.o. such as assistance, set-up and installation is provided WITHOUT WARRANTY OF ANY KIND.

Disclaimer:

VOIPAC TECHNOLOGIES s.r.o. reserves the right to make changes, without notice, to any product, including circuits and/or software described or contained in this datasheet. VOIPAC TECHNOLOGIES s.r.o. assumes no responsibility or liability for the use of the described product(s), conveys no license or title under any patent, copyright, or mask work rights to these products, and makes no representations or warranties that these products are free from patent, copyright, or mask work right infringement, unless otherwise specified.

Trademark Acknowledgment:

Brand and product names are trademarks or registered trademarks of their respective owners.

