MA35D1 Buildroot Quick Start

Agenda

- Overview
- Environment Setup
- Programming
- System Boot

Overview

Overview

- This slide provides instructions on how to quickly build an MA35D1 image
- PC specification standard
  - CPU: Intel i5-10400
  - Memory: 16 GB DDR RAM
  - Storage: 1 TB SSD Disk (200 GB of which is empty space)
  - Operation System: Linux OS or Linux Virtual Machine (VMware provide by Nuvoton)
  - A MA35D1 Docker container

    If you used VMware provide by Nuvoton, you have already created a MA35D1 Docker container
- Software Tool
  - Programming – NuWriter

VMware Setting

Start up with VMware

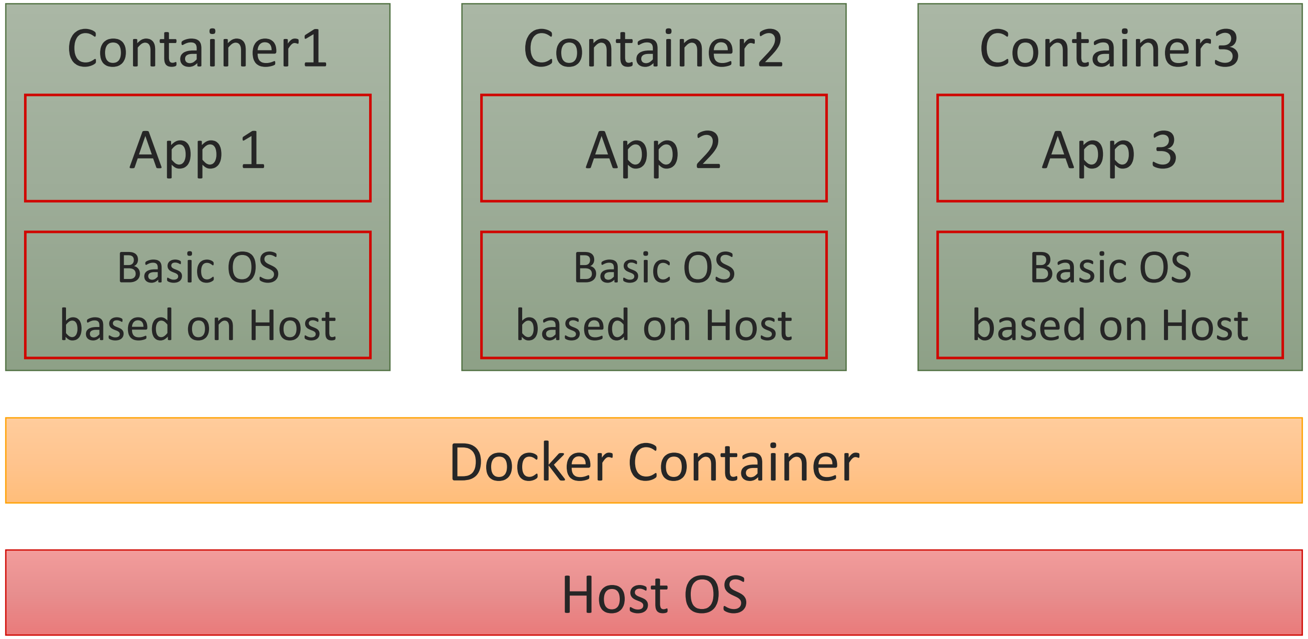
- This VMware Image offers a Linux development environment for MA35D1. If you utilize the VMware Image, you can bypass the Docker steps for building the Image
- User Name: user
  Password: user
- Buildroot:

```
$ cd ~/buildroot/MA35D1_Buildroot
~/MA35D1_Buildroot$ git pull
```

Docker Environment Setup

Development Environment – Docker

- Docker enables the packaging of code and its dependencies into containers.
- Each container is independent and based on the host OS, ensuring they operate in isolation without impacting each other. Containers run more efficiently than virtual machines, resulting in faster performance



Environment Setup (1/4)

- The necessary packages must be installed before building
- Ubuntu and Debian

```
$ sudo apt-get install gawk wget git-core diffstat unzip texinfo gcc-multilib \
   build-essential chrpath socat cpio python python3 python3-pip python3-pexpect \
     xz-utils debianutils iputils-ping libsdl1.2-dev xterm curl
```

Environment Setup (2/4)

- This demo is under Ubuntu distribution. If you use virtual machine, ensure your RAM at least 5GB
- Update existing list of packages

```
$ sudo apt-get update
```

- Install a few prerequisite packages which let apt use packages over HTTPS

```
$ sudo apt install apt-transport-https ca-certificates curl software-properties-common
```

- Add Docker’s official GPG key for the official Docker repository to your system

```
$ curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add
```

- Set up the stable repository, add the Docker repository to APT sources

```
$ sudo add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu focal stable"
```

Environment Setup (3/4)

6. Update the package database with the Docker packages from the newly added repo

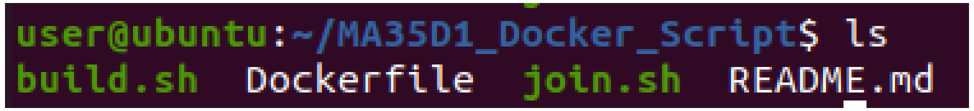
```
$ sudo apt-get update
```

7. Install Docker

```
$ sudo apt-get install docker-ce docker-ce-cli containerd.io
```

8. Download the Docker Script for MA35D1

```
$ git clone https://github.com/OpenNuvoton/MA35D1_Docker_Script.git
```



```
user@ubuntu:~/MA35D1_Docker_Script$ ls
build.sh  Dockerfile  join.sh  README.md
```

Environment Setup (4/4)

9. Enter docker-yocto folder, build docker image.It may take one hour to get about 710 files

```
$ ./build.sh
```

10. Enter docker image, and your command line head will be like nuvoton@a24d9e06abe3:~$

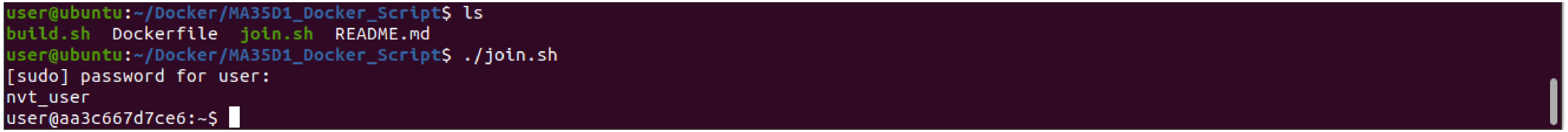
```
$ ./join.sh
ma35d1_user
nuvoton@a24d9e06abe3:~$
```

Environment Setup

Download and Update Buildroot

- Enter Docker container

```
$ ./join.sh
```

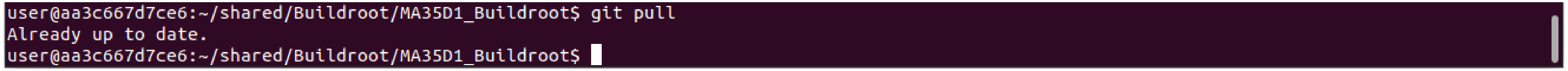


```
user@ubuntu:~/Docker/MA35D1_Docker_Script$ ls
build.sh  Dockerfile  join.sh  README.md
user@ubuntu:~/Docker/MA35D1_Docker_Script$ ./join.sh
[sudo] password for user:
nvt_user
user@aa3c667d7ce6:~$ 
```

- Download MA35D1 Buildroot (No need to clone with VMware provided by Nuvoton)

```
$ git clone https://github.com/OpenNuvoton/MA35D1_Buildroot.git
```

```
$ git pull
```

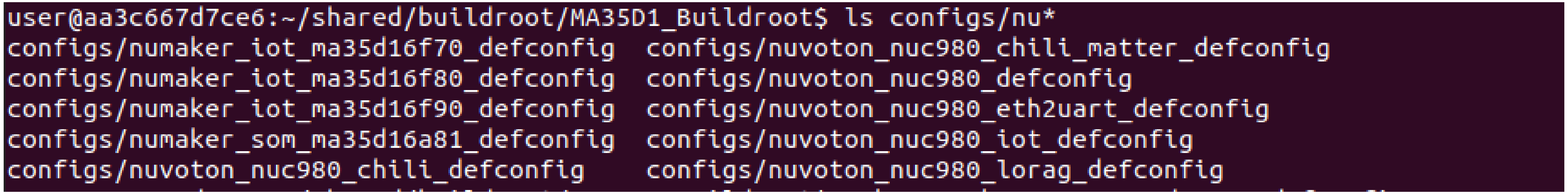


```
user@aa3c667d7ce6:~/shared/Buildroot/MA35D1_Buildroot$ git pull
Already up to date.
user@aa3c667d7ce6:~/shared/Buildroot/MA35D1_Buildroot$ 
```

Download and Update Buildroot

- List all default configurations provide by Nuvoton and choose your target board configuration

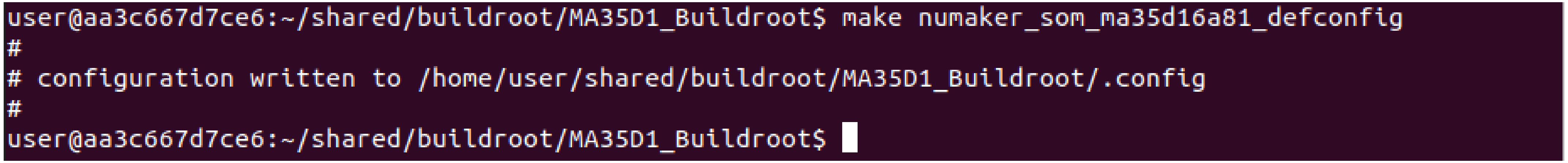
```
$ ls configs/n*
```



```
user@aa3c667d7ce6:~/shared/buildroot/MA35D1_Buildroot$ ls configs/nu*
configs/numaker_iot_ma35d16f70_defconfig  configs/nuvoton_nuc980_chili_matter_defconfig
configs/numaker_iot_ma35d16f80_defconfig  configs/nuvoton_nuc980_defconfig
configs/numaker_iot_ma35d16f90_defconfig  configs/nuvoton_nuc980_eth2uart_defconfig
configs/numaker_som_ma35d16a81_defconfig  configs/nuvoton_nuc980_iot_defconfig
configs/nuvoton_nuc980_chili_defconfig    configs/nuvoton_nuc980_lorag_defconfig
```

- Set configuration to the target board

```
$ make numaker_som_ma35d16a81_defconfig
```



```
user@aa3c667d7ce6:~/shared/buildroot/MA35D1_Buildroot$ make numaker_som_ma35d16a81_defconfig
#
# configuration written to /home/user/shared/buildroot/MA35D1_Buildroot/.config
#
user@aa3c667d7ce6:~/shared/buildroot/MA35D1_Buildroot$ 
```

Install Qt5 Package

- Open Buildroot configuration

```
$ make menuconfig
```

```
Target packages  --->
   Graphic libraries and applications   --->
     [*] directfb
     [*] Qt5  --->
         -*-  qt5base
         [*]     Compile and install examples (with code)
         -*-     gui module
         [*]     widgets module
         -*-     linuxfb support
         [*]     directfb support
         [*]     GIF support
         [*]     JPEG support
         [*]     PNG support
         [*]     Enable Tslib support
         [*]     qt5declarative
         [*]     quick module
         [*]     qt5multimedia
         [*]     qt5script
         [*]     qt5sensors
         [*]     qt5serialbus
         -*-     qt5serialport
```

Install Gstreamer Related Package

```
Target packages  --->
  Audio and video applications  --->
   [*] alsa-utils  --->
     [*]   alsactl
     [*]   alsamixer
   -*- ffmpeg  --->
   [*]   Build ffmpeg
     -*-   Build libswscale
     (all) Enabled encoders
     (all) Enabled decoders
     (all) Enabled muxers
     (all) Enabled demuxers
     (all) Enabled parsers
     (all) Enabled bitstreams
     (all) Enabled protocols
     (all) Enabled filters
   [*]   Enable input devices
[*]   Enable output devices
[*] gstreamer 1.x
-*-   enable unit test libraries
[*]   enable command-line parser
[*]   enable tracing subsystem
[*]   enable gst-debug trace support
[*]   enable plugin registry
[*]   install tools
[*]   gstreamer1-mm
-*-   gst1-plugins-base  --->
  -*-   app
  [*]   audioconvert
  [*]   audiomixer
  [*]   audiorate
  [*]   audiotestsrc
  [*]   videoconvert
  [*]   playback
[*]   audioresample
[*]   tcp
-*-   typefind
[*]   videotestsrc
[*]   videorate
[*]   videoscale
[*]   volume
[*]   alsa
[*]   ogg (*.ogg audio/video)
```

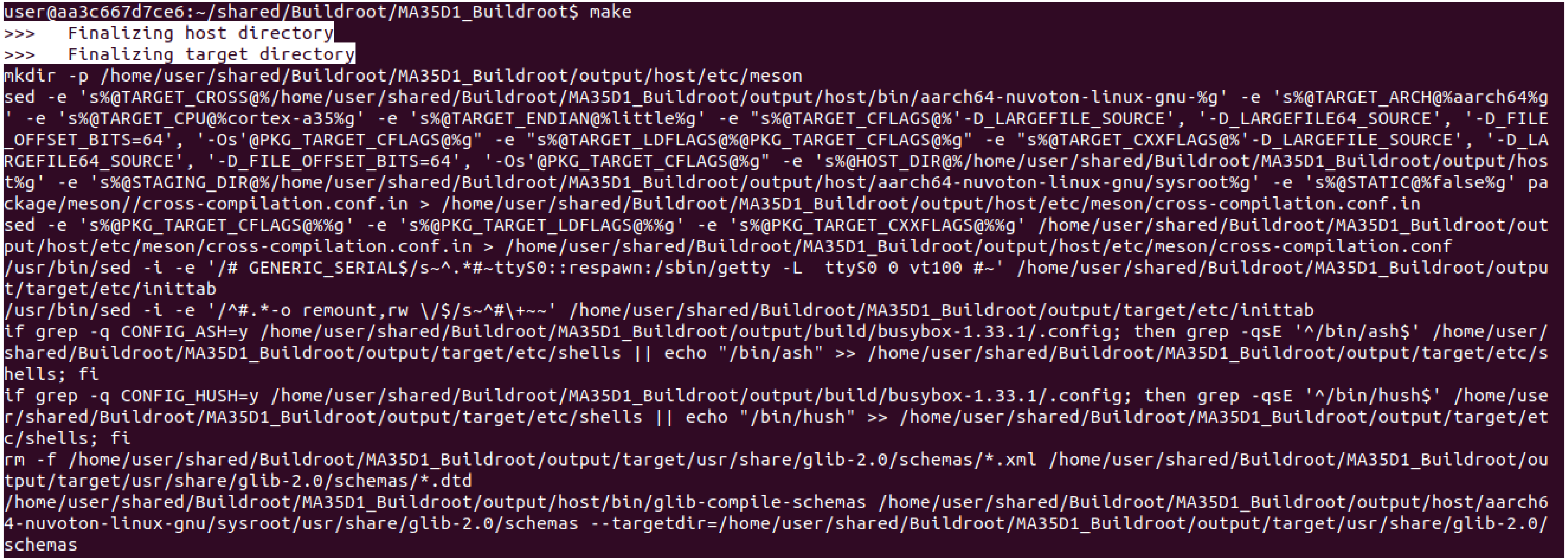
Install Gstreamer Related Package

```
Target packages  --->
  Audio and video applications  --->
   -*-   gst1-plugins-good  --->
      [*]   jpeg (JPEG support)
      [*]   png (PNG support)
      [*]   audiofx
      [*]   audioparsers
      [*]   auparse
      [*]   autodetect
      [*]   avi (*.avi video)
      [*]   isomp4
      -*-   rtp
      -*-   rtpmanager
      [*]   rtsp
      -*-   udp
      [*]   videobox
      [*]   videocrop
      [*]   videofilter
      [*]   videomixer
      [*]   wavparse (*.wav audio)
      [*]   mpg123 (*.mp3 audio)
      [*]   ossaudio (OSS audio)
      [*]   v4l2
      [*]   v4l2-probe (m2m)
   [*]   gst1-plugins-bad  --->
      [*]   autoconvert
      [*]   videoparsers
   [*]   gst1-plugins-ugly  --->
      [*]   mpeg2dec
      [*]   x264
   [*]   gst1-libav
   [*]   gst1-rtsp-server
   -*- mpg123
```

Install Package to MA35D1 Image

- If you do not need to install any additional packages to the MA35D1 image, you can start building it

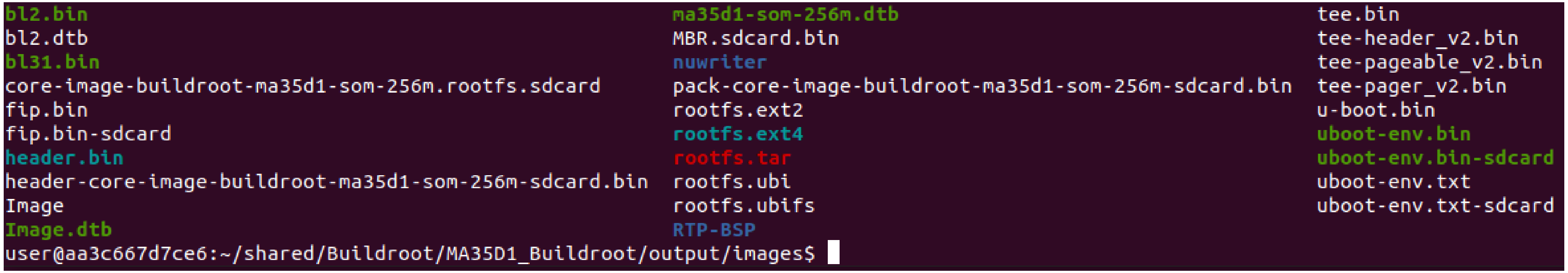
```
$ make
```



```
user@aa3c667d7ce6:~/shared/Buildroot/MA35D1_Buildroot$ make
>>>   Finalizing host directory
>>>   Finalizing target directory
mkdir -p /home/user/shared/Buildroot/MA35D1_Buildroot/output/host/etc/meson
sed -e 's%@TARGET_CROSS@%/home/user/shared/Buildroot/MA35D1_Buildroot/output/host/bin/aarch64-nuvoton-linux-gnu-%g' -e 's%@TARGET_ARCH@%aarch64%g
' -e 's%@TARGET_CPU@%cortex-a35%g' -e 's%@TARGET_ENDIAN@%little%g' -e "s%@TARGET_CFLAGS@%'-D_LARGEFILE_SOURCE', '-D_LARGEFILE64_SOURCE', '-D_FILE
_OFFSET_BITS=64', '-Os'@PKG_TARGET_CFLAGS@%g" -e "s%@TARGET_LDFLAGS@%@PKG_TARGET_CFLAGS@%g" -e "s%@TARGET_CXXFLAGS@%'-D_LARGEFILE_SOURCE', '-D_LA
RGEFILE64_SOURCE', '-D_FILE_OFFSET_BITS=64', '-Os'@PKG_TARGET_CFLAGS@%g" -e 's%@HOST_DIR@%/home/user/shared/Buildroot/MA35D1_Buildroot/output/hos
t%g' -e 's%@STAGING_DIR@%/home/user/shared/Buildroot/MA35D1_Buildroot/output/host/aarch64-nuvoton-linux-gnu/sysroot%g' -e 's%@STATIC@%false%g' pa
ckage/meson//cross-compilation.conf.in > /home/user/shared/Buildroot/MA35D1_Buildroot/output/host/etc/meson/cross-compilation.conf.in
sed -e 's%@PKG_TARGET_CFLAGS@%%g' -e 's%@PKG_TARGET_LDFLAGS@%%g' -e 's%@PKG_TARGET_CXXFLAGS@%%g' /home/user/shared/Buildroot/MA35D1_Buildroot/out
put/host/etc/meson/cross-compilation.conf.in > /home/user/shared/Buildroot/MA35D1_Buildroot/output/host/etc/meson/cross-compilation.conf
/usr/bin/sed -i -e '/# GENERIC_SERIAL$/s~^.*#~ttyS0::respawn:/sbin/getty -L  ttyS0 0 vt100 #~' /home/user/shared/Buildroot/MA35D1_Buildroot/outpu
t/target/etc/inittab
/usr/bin/sed -i -e '/^#.*-o remount,rw \/$/s~^#\+~~' /home/user/shared/Buildroot/MA35D1_Buildroot/output/target/etc/inittab
if grep -q CONFIG_ASH=y /home/user/shared/Buildroot/MA35D1_Buildroot/output/build/busybox-1.33.1/.config; then grep -qsE '^/bin/ash$' /home/user/
shared/Buildroot/MA35D1_Buildroot/output/target/etc/shells || echo "/bin/ash" >> /home/user/shared/Buildroot/MA35D1_Buildroot/output/target/etc/s
hells; fi
if grep -q CONFIG_HUSH=y /home/user/shared/Buildroot/MA35D1_Buildroot/output/build/busybox-1.33.1/.config; then grep -qsE '^/bin/hush$' /home/use
r/shared/Buildroot/MA35D1_Buildroot/output/target/etc/shells || echo "/bin/hush" >> /home/user/shared/Buildroot/MA35D1_Buildroot/output/target/et
c/shells; fi
rm -f /home/user/shared/Buildroot/MA35D1_Buildroot/output/target/usr/share/glib-2.0/schemas/*.xml /home/user/shared/Buildroot/MA35D1_Buildroot/ou
tput/target/usr/share/glib-2.0/schemas/*.dtd
/home/user/shared/Buildroot/MA35D1_Buildroot/output/host/bin/glib-compile-schemas /home/user/shared/Buildroot/MA35D1_Buildroot/output/host/aarch6
4-nuvoton-linux-gnu/sysroot/usr/share/glib-2.0/schemas --targetdir=/home/user/shared/Buildroot/MA35D1_Buildroot/output/target/usr/share/glib-2.0/
schemas
```

Building Finishing

- Once the build is complete, you can find the resulting image at /output/image



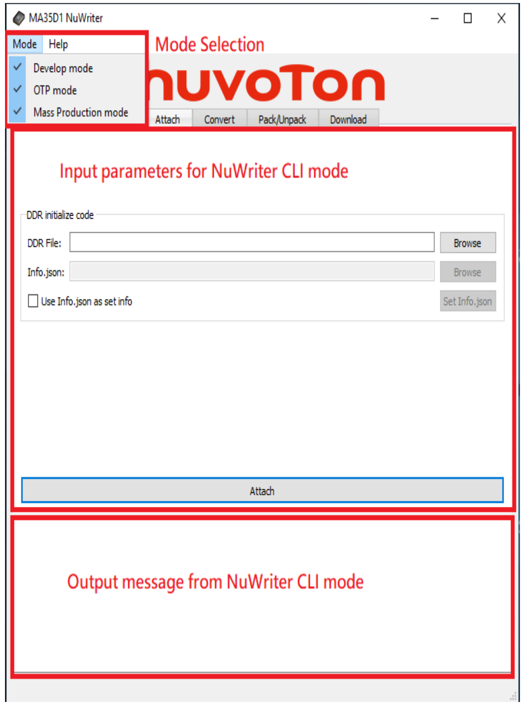
```
bl2.bin                                                  ma35d1-som-256m.dtb                                   tee.bin
bl2.dtb                                                  MBR.sdcard.bin                                        tee-header_v2.bin
bl31.bin                                                 nuwriter                                              tee-pageable_v2.bin
core-image-buildroot-ma35d1-som-256m.rootfs.sdcard       pack-core-image-buildroot-ma35d1-som-256m-sdcard.bin  tee-pager_v2.bin
fip.bin                                                  rootfs.ext2                                           u-boot.bin
fip.bin-sdcard                                           rootfs.ext4                                           uboot-env.bin
header.bin                                               rootfs.tar                                            uboot-env.bin-sdcard
header-core-image-buildroot-ma35d1-som-256m-sdcard.bin  rootfs.ubi                                            uboot-env.txt
Image                                                    rootfs.ubifs                                          uboot-env.txt-sdcard
Image.dtb                                                RTP-BSP
user@aa3c667d7ce6:~/shared/Buildroot/MA35D1_Buildroot/output/images$
```

- The core-image-buildroot-ma35d1-som-256m.rootfs.sdcard can be used to directly program to an SD card
- Alternatively, pack-core-image-buildroot-ma35d1-som-256-sd.bin can be used to program an SD card through NuWriter

Programming

Image Programming

- Usually, you would use NuWriter for MA35 to program the image, but for debugging purposes, you can also program the image to an SD card using an writing tool by PC
- One open-source utility that can be used for writing image files is balenaEtcher, which you can use to program the image



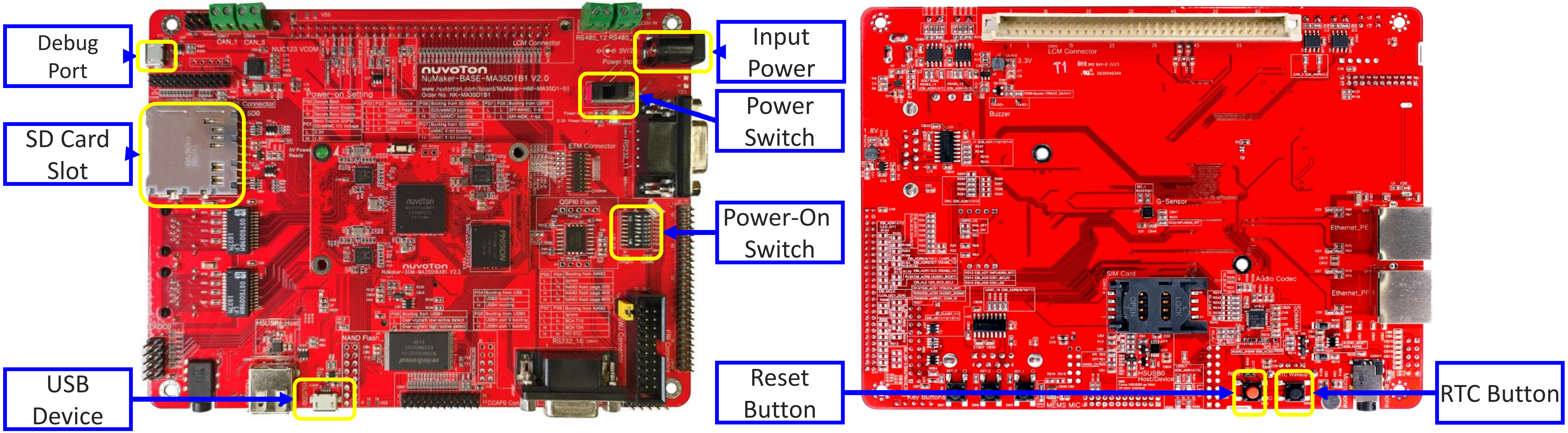
Mode Selection

Input parameters for NuWriter CLI mode

Output message from NuWriter CLI mode

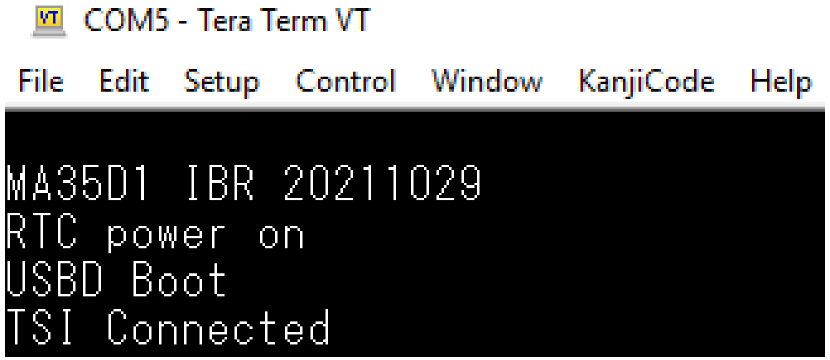
Evaluation Board Introduction

- Debug Port: show the debug message
- USB Device: used to program image by NuWriter
- Power-On Switch: Switch booting source
- Reset Button: reset MA35D1
- RTC Button: wake up RTC



Start Programming

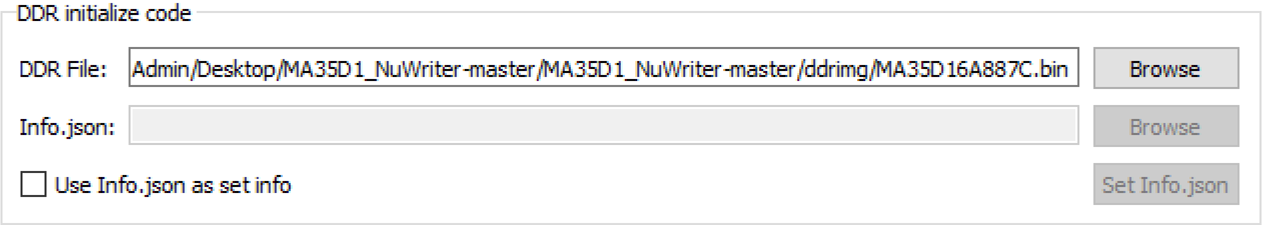
- To program an image to an SD card, please ensure the following steps are followed:
  - Connect the power supply
  - Connect the debug port
  - Connect the USB device
  - Insert the SD card
  - Set the power switch to enable USBD Boot
    - [PG 0] [PG 2] [PG 3] High
- Click the RTC button, reset button, and you will see the debug message



```
MA35D1 IBR 20211029
RTC power on
USBD Boot
TSI Connected
```

NuWriter Programming

- Choose DDR file ddrimg/MA35D16A887C.bin



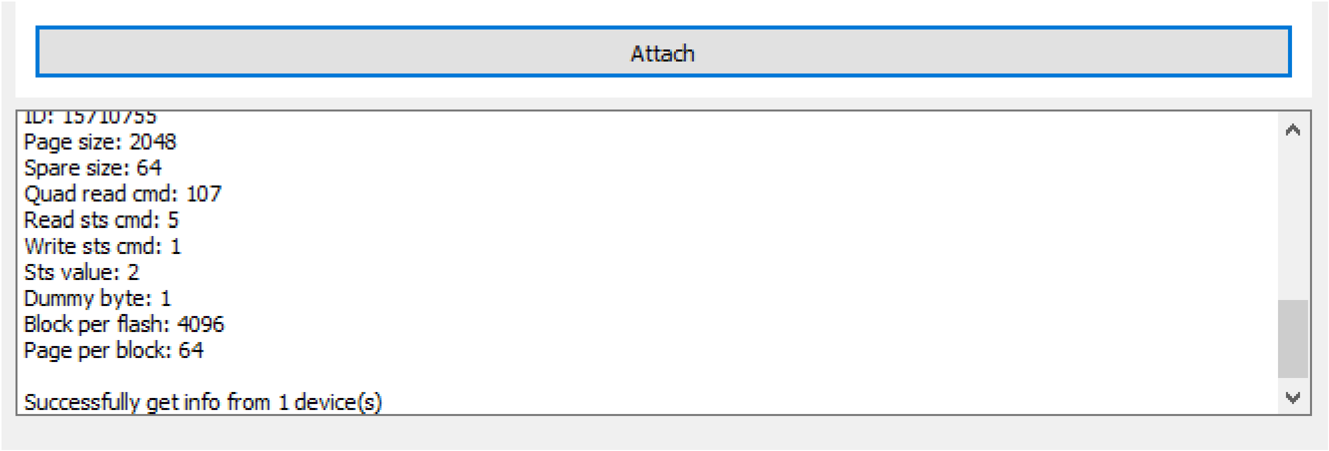
DDR initialize code

DDR File: Admin/Desktop/MA35D1_NuWriter-master/MA35D1_NuWriter-master/ddrimg/MA35D16A887C.bin

Info.json:

□ Use Info.json as set info

- Click Attach button, you will see the message below



```
ID: 15710755
Page size: 2048
Spare size: 64
Quad read cmd: 107
Read sts cmd: 5
Write sts cmd: 1
Sts value: 2
Dummy byte: 1
Block per flash: 4096
Page per block: 64

Successfully get info from 1 device(s)
```

NuWriter Programming

- Switch to Download and choose SD/EMMC
- Browse pack-core-image-buildroot-ma35d1-som-256m-sdcard.bin
- Enable Pack mode
- Click write button to start programming

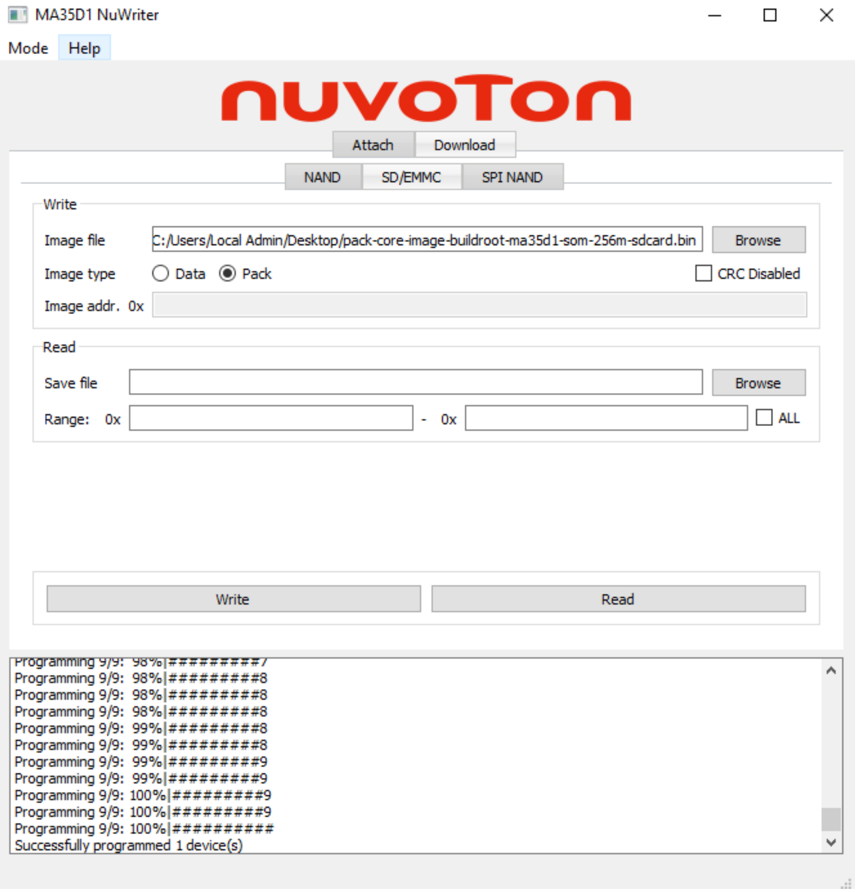


Image file C:/Users/Local Admin/Desktop/pack-core-image-buildroot-ma35d1-som-256m-sdcard.bin

Image type ○ Data ● Pack □ CRC Disabled

Image addr. 0x

Save file

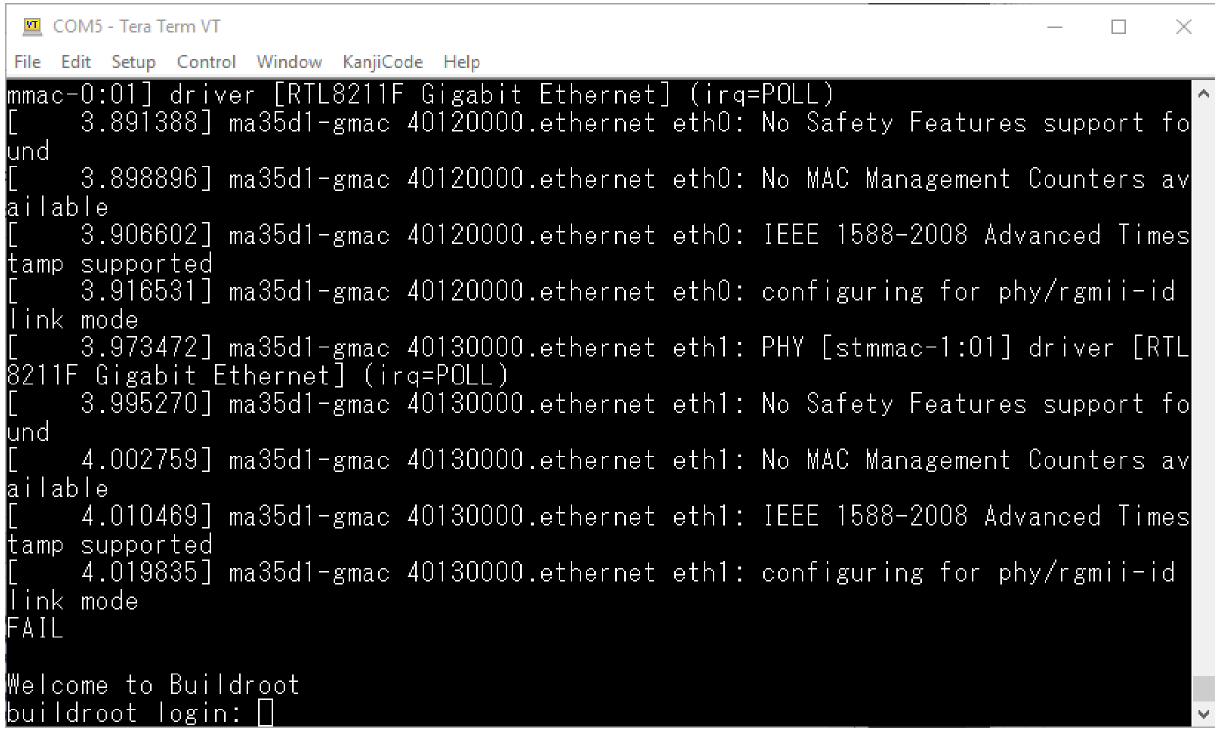
Range: 0x - 0x □ ALL

```
Programming 9/9:  98%|#########7
Programming 9/9:  98%|#########8
Programming 9/9:  98%|#########8
Programming 9/9:  98%|#########8
Programming 9/9:  99%|#########8
Programming 9/9:  99%|#########8
Programming 9/9:  99%|#########9
Programming 9/9:  99%|#########9
Programming 9/9: 100%|#########9
Programming 9/9: 100%|#########9
Programming 9/9: 100%|##########
Successfully programmed 1 device(s)
```

System Boot

System Boot

- Set the power switch to enable SD 1 Boot and click reset Button and you will see MA35D1 booting
  - [PG 0] [PG 2] High
- Enter “root” to login



```
mmac-0:01] driver [RTL8211F Gigabit Ethernet] (irq=POLL)
[    3.891388] ma35d1-gmac 40120000.ethernet eth0: No Safety Features support fo
und
[    3.898896] ma35d1-gmac 40120000.ethernet eth0: No MAC Management Counters av
ailable
[    3.906602] ma35d1-gmac 40120000.ethernet eth0: IEEE 1588-2008 Advanced Times
tamp supported
[    3.916531] ma35d1-gmac 40120000.ethernet eth0: configuring for phy/rgmii-id
link mode
[    3.973472] ma35d1-gmac 40130000.ethernet eth1: PHY [stmmac-1:01] driver [RTL
8211F Gigabit Ethernet] (irq=POLL)
[    3.995270] ma35d1-gmac 40130000.ethernet eth1: No Safety Features support fo
und
[    4.002759] ma35d1-gmac 40130000.ethernet eth1: No MAC Management Counters av
ailable
[    4.010469] ma35d1-gmac 40130000.ethernet eth1: IEEE 1588-2008 Advanced Times
tamp supported
[    4.019835] ma35d1-gmac 40130000.ethernet eth1: configuring for phy/rgmii-id
link mode
FAIL

Welcome to Buildroot
buildroot login:
```

Joy of innovation

nuvoTon

谢谢

謝謝

Děkuji

Bedankt

Thank you

Kiitos

Merci

Danke

Grazie

ありがとう

감사합니다

Dziękujemy

Obrigado

Спасибо

Gracias

Teşekkür ederim

Cảm ơn