
ADI FTC

ADI FTC

Software hands-on training kit: Customizing ADI Kuiper 2 Linux Distribution

Maria-Larisa Radu, Alisa Roman
Embedded Software Engineer, Associate Engineer

Agenda

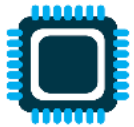
- Theoretical Concepts:
  - Introduction to ADI Kuiper Linux
  - What Does Kuiper Include
  - Unlocking the Power of Kuiper 2
    - Key Features
    - Stages Anatomy
    - Build Flow
    - Configuration File
    - Supported Versions
    - Extra scripts
- Hands-on Activity:
  - Build a Custom Kuiper 2 Image
  - Boards and accessories
  - Layout
  - Workshop Networking structure
  - Exercises carriers
  - Prepare the SD card to boot
  - Raspberry Pi 5 Networking Exercise via MQTT
  - Boot your Custom Kuiper 2 Image on Raspberry Pi 5
- Product Support and Acknowledgements

Introduction to ADI Kuiper Linux

ADI Kuiper Linux is an Open-Source Linux Distribution for Analog Devices

It is the primary distribution for product evaluation boards and reference designs

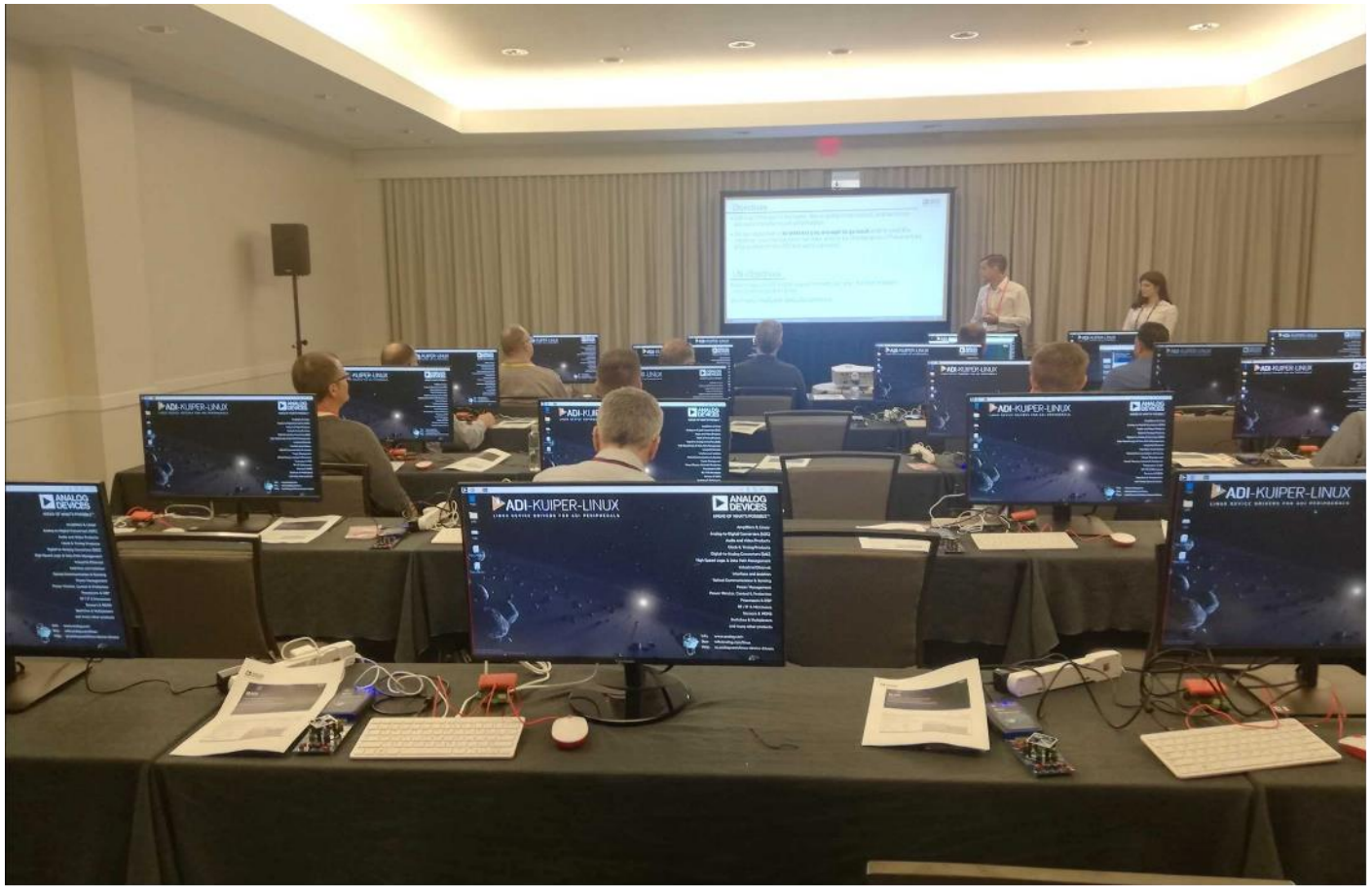
It includes pre-built boot files, device drivers and a variety of development utilities



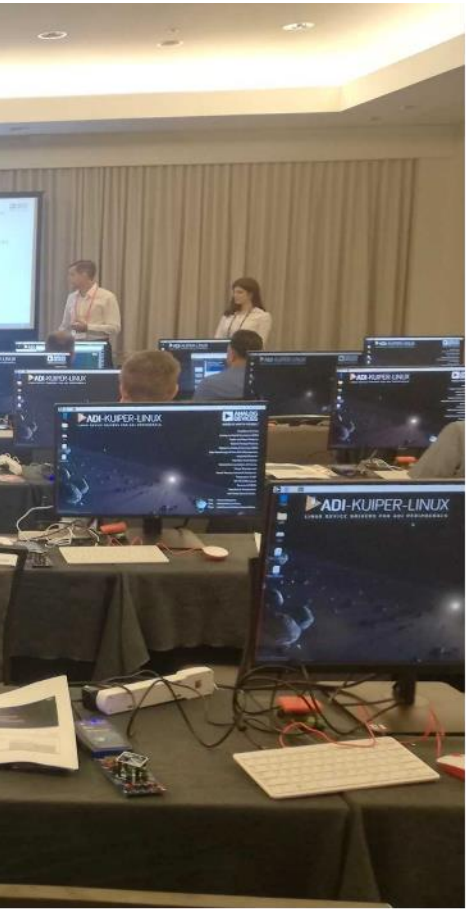
Supports over 140 FPGA-based projects and over 30 Raspberry Pi-based designs

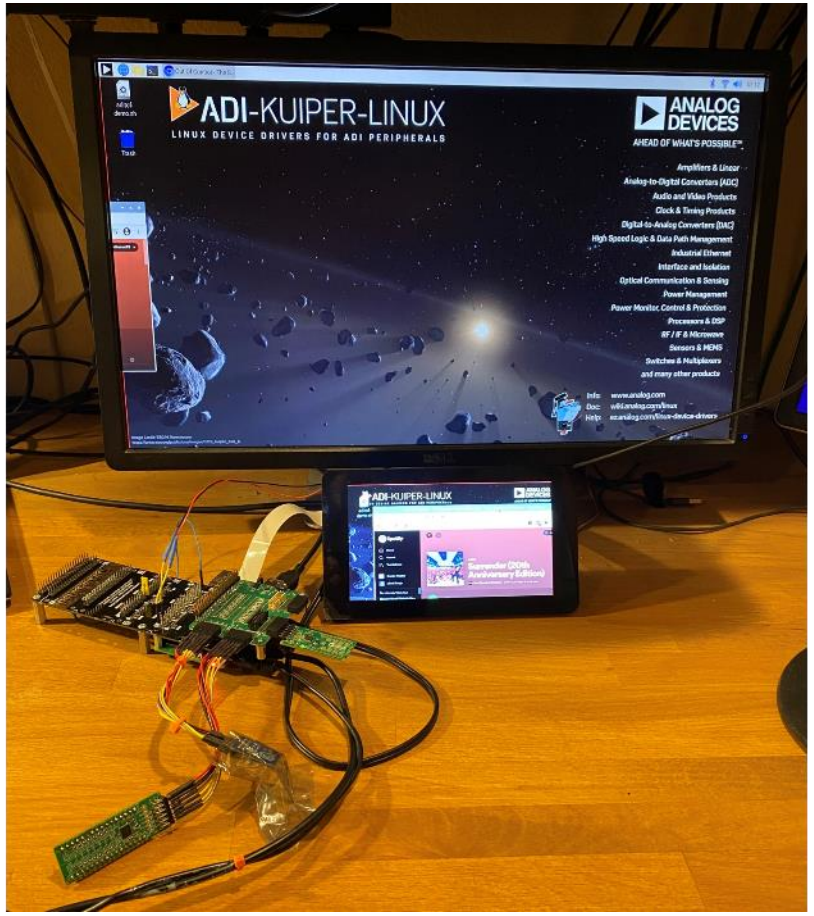
The first release was published at the beginning of 2021

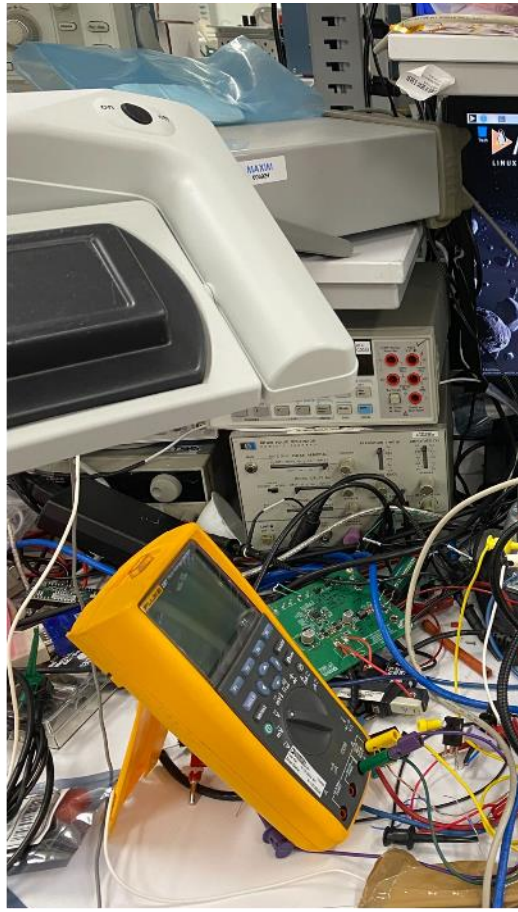
Kuiper is Everywhere!



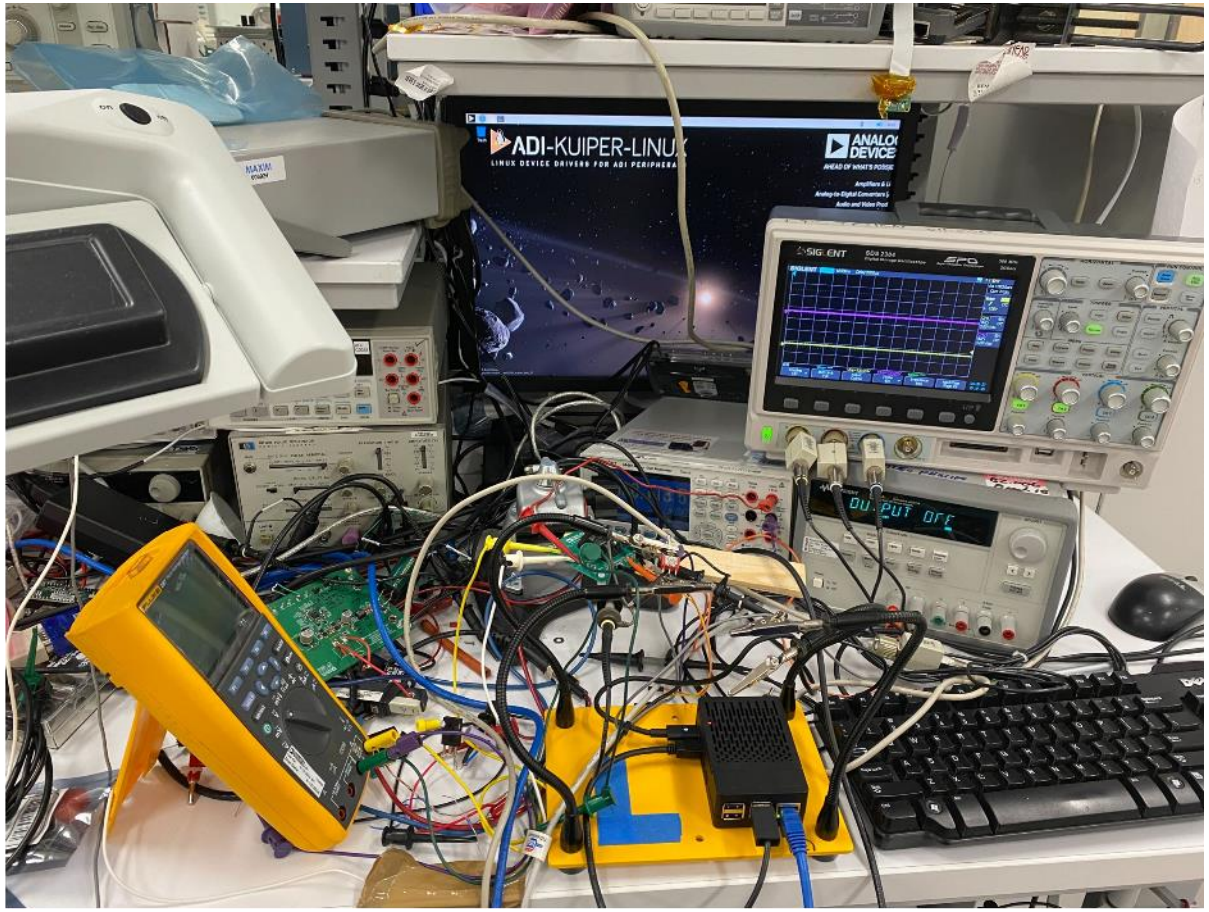
Kuiper is Everywhere!







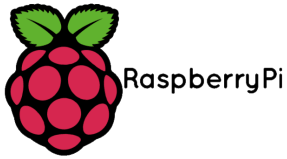
Kuiper is Everywhere!



What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

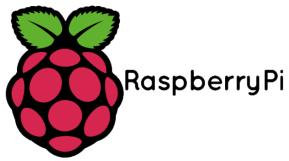


What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

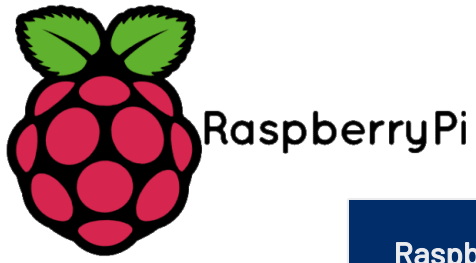
Cyclone5
Arria10
Stratix10



What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs



RaspberryPi
2 / 3 / 4 / 5

intel FPGA

XILINX
AMD

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

RaspberryPi

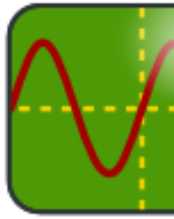
RaspberryPi
2 / 3 / 4 / 5

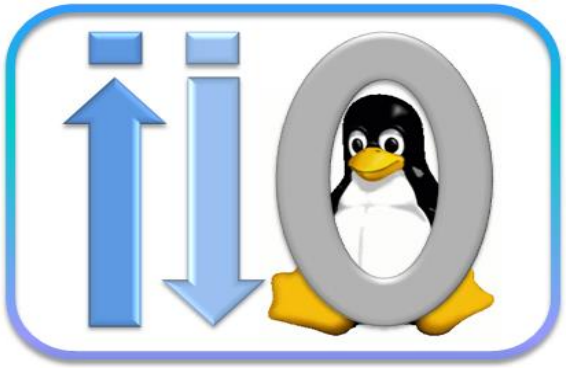
intel FPGA

XILINX
AMD

Libraries and Applications

- A variety of development utilities, software libraries and examples for different projects





LibIIO

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

RaspberryPi

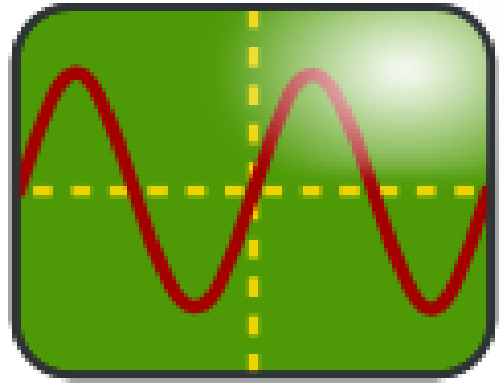
RaspberryPi
2 / 3 / 4 / 5

intel FPGA

XILINX
AMD

Libraries and Applications

- A variety of development utilities, software libraries and examples for different projects



IIO Oscilloscope

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

RaspberryPi

RaspberryPi 2/3/4/5

intel FPGA

XILINX AMD

Libraries and Applications

- A variety of development utilities, software libraries and examples for different projects

PYTHON FOR

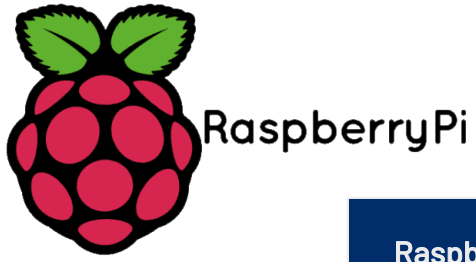
C++ CODE ADI-LIBM2k

INTERFACE LIBARY FOR THE ADALM2000

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs



RaspberryPi

RaspberryPi
2 / 3 / 4 / 5

intel FPGA

XILINX
AMD

Libraries and Applications

- A variety of development utilities, software libraries and examples for different projects

PyADI-IIO
PYTHON FOR ADI INDUSTRIAL I/O DEVICES

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

RaspberryPi

RaspberryPi
2 / 3 / 4 / 5

intel FPGA

XILINX
AMD

Libraries and Applications

- A variety of development utilities, software libraries and examples for different projects

GNURadio
THE FREE & OPEN SOFTWARE RADIO ECOSYSTEM

PyADI-IIO

What does Kuiper Include

Boot Files

- More than 1,300 Linux device drivers for Analog Devices products
- Pre-built boot files for over 140 FPGA-based projects and over 30 RPi-based designs

RaspberryPi

RaspberryPi
2 / 3 / 4 / 5

intel FPGA

XILINX
AMD

Libraries and Applications

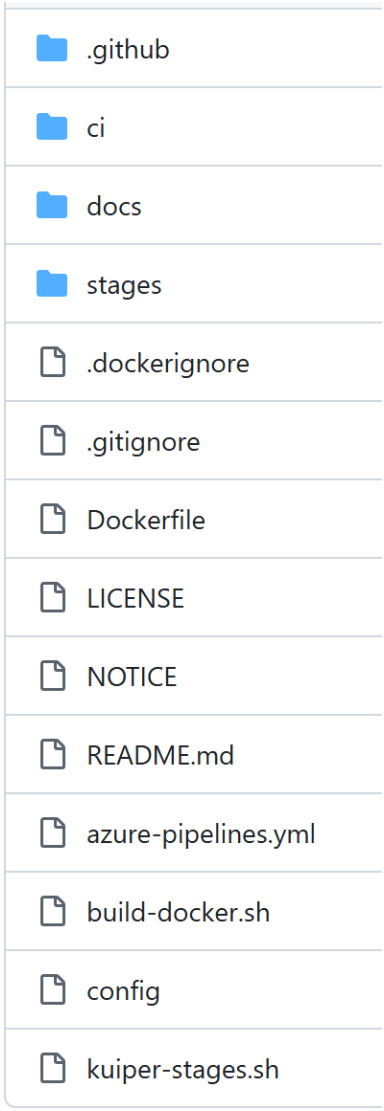
- A variety of development utilities, software libraries and examples for different projects

ANALOG DEVICES
SCOPY

Unlocking the Power of Kuiper 2: Key Features

- Kuiper 2 is a **Debian based** Distribution
- Contains a **configuration file**, where every setting can be done without modifying the source code
- Supported architectures are **armhf** (32 bits) and **arm64** (64 bits)
- Analog Devices **libraries can be added individually**, making the image completely customizable for each project
- Supports installation of **different versions of boot files** for Xilinx and Intel carriers and Raspberry Pi platforms
- Can **configure the image at build time for a specific hardware setup**, preparing the image to be transferred on an SD card and booted on a board, without the need to manually copy boot files
- Supports running a **custom script** at build time, to cover even more custom settings that are not standard in the config file
- Supports **exporting the source files** of all the packages used in the Kuiper 2 image at build time

Unlocking the Power of Kuiper 2: Stages Anatomy



Support for automatic builds in GitHub Actions

Support for GitHub Documentation

01.bootstrap
02.set-locale-and-timezone
03.system-tweaks
04.configure-desktop-env
05.adi-tools
06.boot-partition
07.extra-tweaks
08.export-stage

BSD-3-Clause License file

Shell script used to automate the process of preparing the Docker environment

Set of variables that control tool installation

Shell script used to build the Kuiper 2 image

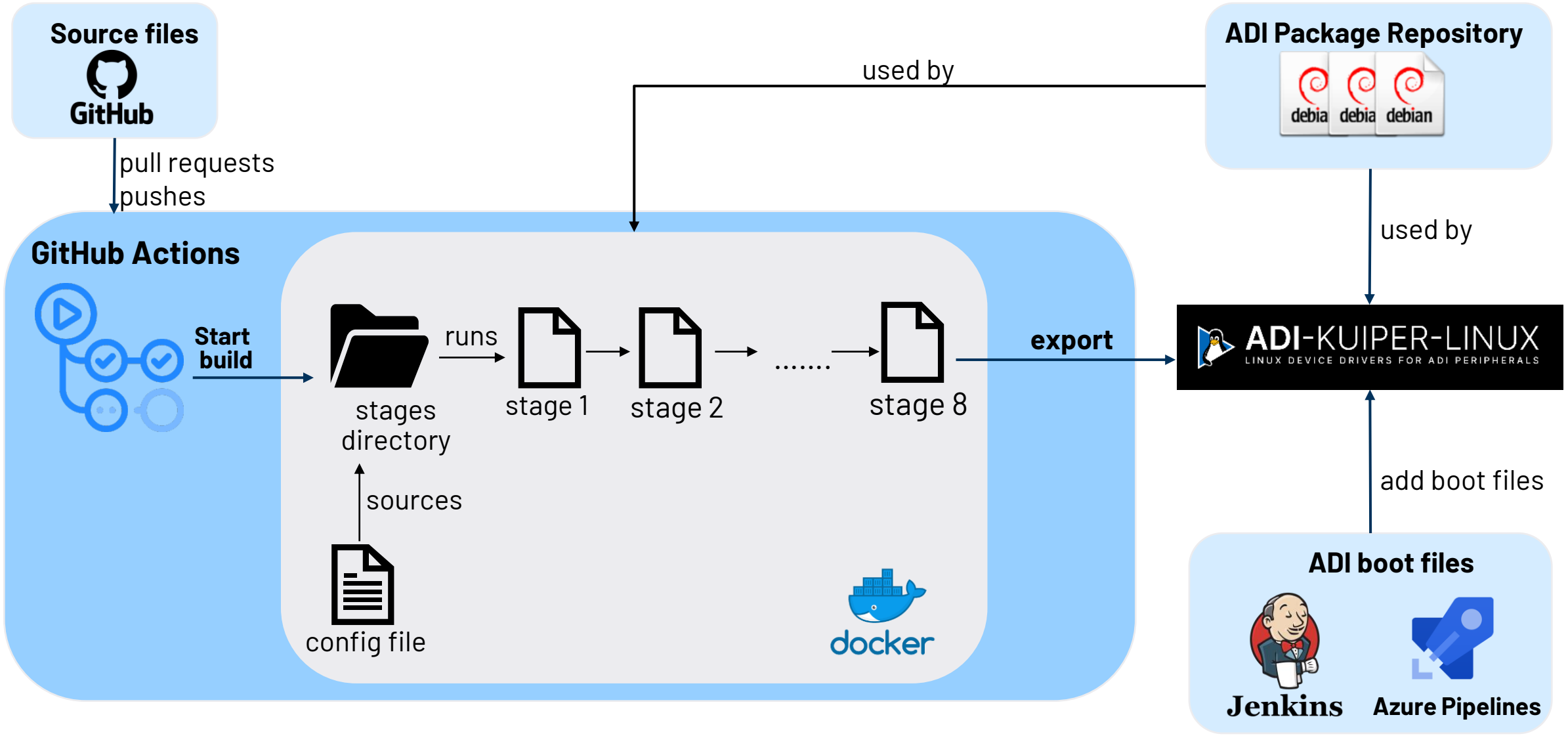
Pros

- Adds logical clarity
- Adds modularity
- Simplifies maintenance
- Simplifies customization

Initial complexity

Cons

Unlocking the Power of Kuiper 2: Build Flow



Unlocking the Power of Kuiper 2: Configuration File

Unlocking the Power of Kuiper 2: Configuration File

Target architecture

Extra script

Export sources

Prepare boot files in the boot partition

- ❑*board*
- ❑*carrier*

Debian version

Boot files

- ❑*release*
- ❑*source of installation (SW Downloads or ADI repository)*

ADI library or tool

- ❑*cmake arguments (if the case)*
- ❑*github branch or tag*

Unlocking the Power of Kuiper 2: Supported Versions

Version	Variable in the configuration file	Stage/substage
Basic image	No variable	01.bootstrap 02.set-locale-and-timezone 03.system-tweaks 05.adi-tools/14.write-git-logs 06.boot-partition 08.export-stage/01.extend-rootfs 08.export-stage/03.generate-license 08.export-stage/04.export-image
With desktop environment	CONFIG_DESKTOP	04.configure-desktop-env
With ADI tool	CONFIG_[ADI tool] CONFIG_[ADI tool]_CMAKE_ARGS BRANCH_[ADI tool] (if the case)	05.adi-tools/xx.install-[ADI tool]
With exported sources	EXPORT_SOURCES	08.export-stage/02.export-sources
With custom script	EXTRA_SCRIPT	07.extra-tweaks/01.extra-scripts
With boot partition ready for a specific hardware setup	ADI_EVAL_BOARD CARRIER	08.export-stage/04.export-image

Unlocking the Power of Kuiper 2: Extra Scripts

Kuiper allows you to run additional scripts during the build process to customize the resulting image. This feature enables advanced customization beyond the standard configuration options.

Unlocking the Power of Kuiper 2: Extra Scripts

Kuiper allows you to run additional scripts during the build process to customize the resulting image. This feature enables advanced customization beyond the standard configuration options.

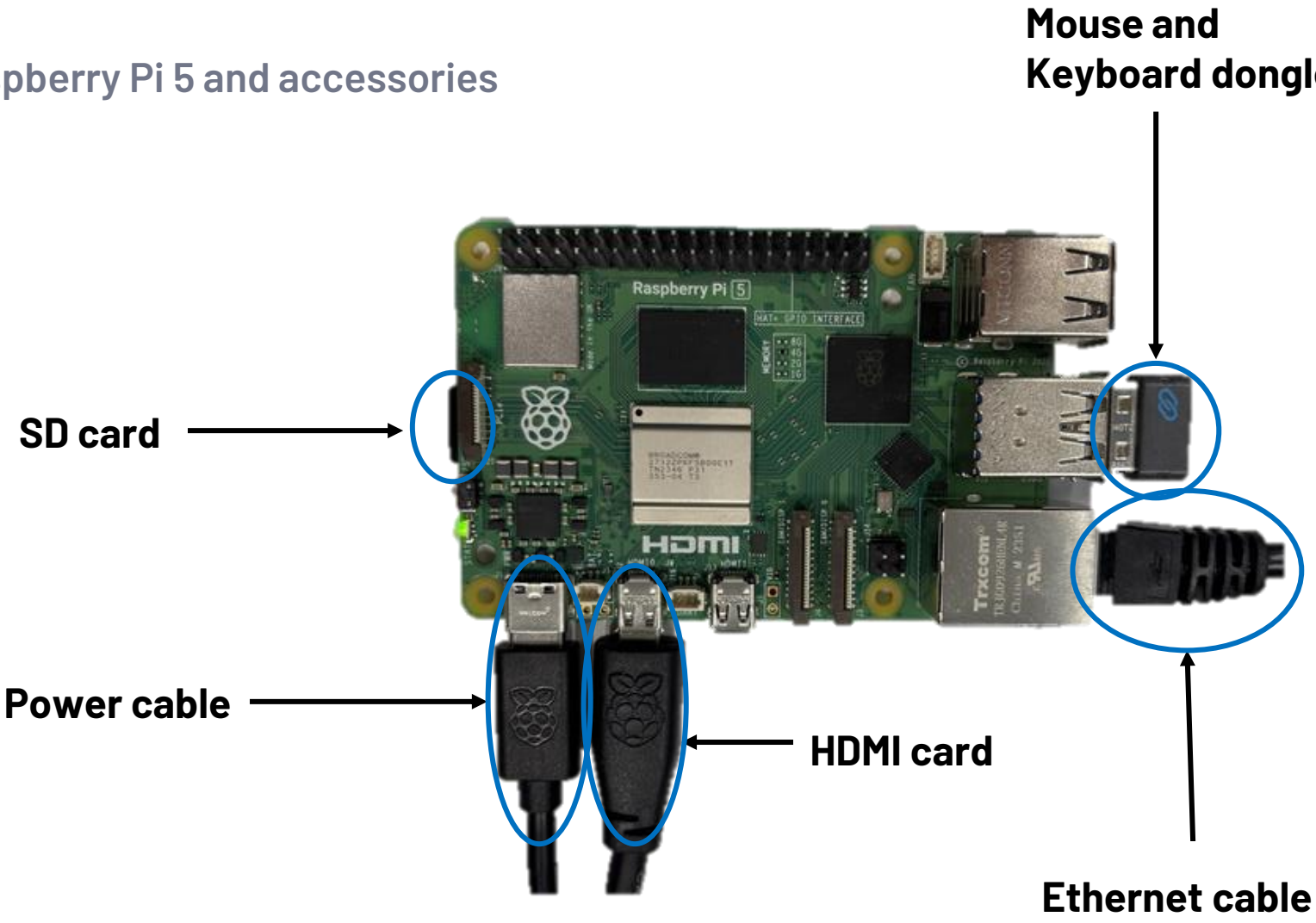
Unlocking the Power of Kuiper 2: Extra Scripts

Kuiper allows you to run additional scripts during the build process to customize the resulting image. This feature enables advanced customization beyond the standard configuration options.

You can also turn your additional configurations into a Debian package that can be stored in the ADI Package Repository and easily installed on any Debian-based Linux.

Hands-On Activities

Raspberry Pi 5 and accessories



Hands-On Activities

Build a Custom Kuiper 2 Image

1. Open a terminal on the Raspberry Pi 5 in front of you and run the following commands:

```
$ git clone https://github.com/analogdevicesinc/adi-kuiper-gen
$ cd adi-kuiper-gen
$ code .
```

2. Modify the config file according to the table:

Line	Old string	Modified string
8	TARGET_ARCHITECTURE=armhf	TARGET_ARCHITECTURE=arm64
23	CONFIG_DESKTOP=n	CONFIG_DESKTOP=y
30	CONFIG_LIBIIO=n	CONFIG_LIBIIO=y
226	EXTRA_SCRIPT=	EXTRA_SCRIPT= stages/07.extra-tweaks/01.extra-scripts/examples/extra-script-example.sh

3. Create a new file in *stages/07.extra-tweaks/01.extra-scripts/examples/* named *custom-file.txt* and write to it "*Hello FTC25*" (or anything you want).

4. Open file stages/07.extra-tweaks/01.extra-scripts/examples/extra-scripts-example.sh and add the following line at the end of the file:

```
$ cp stages/07.extra-tweaks/01.extra-scripts/examples/custom-file.txt /home/analog
```

5. Close Visual Studio Code and write to the terminal:

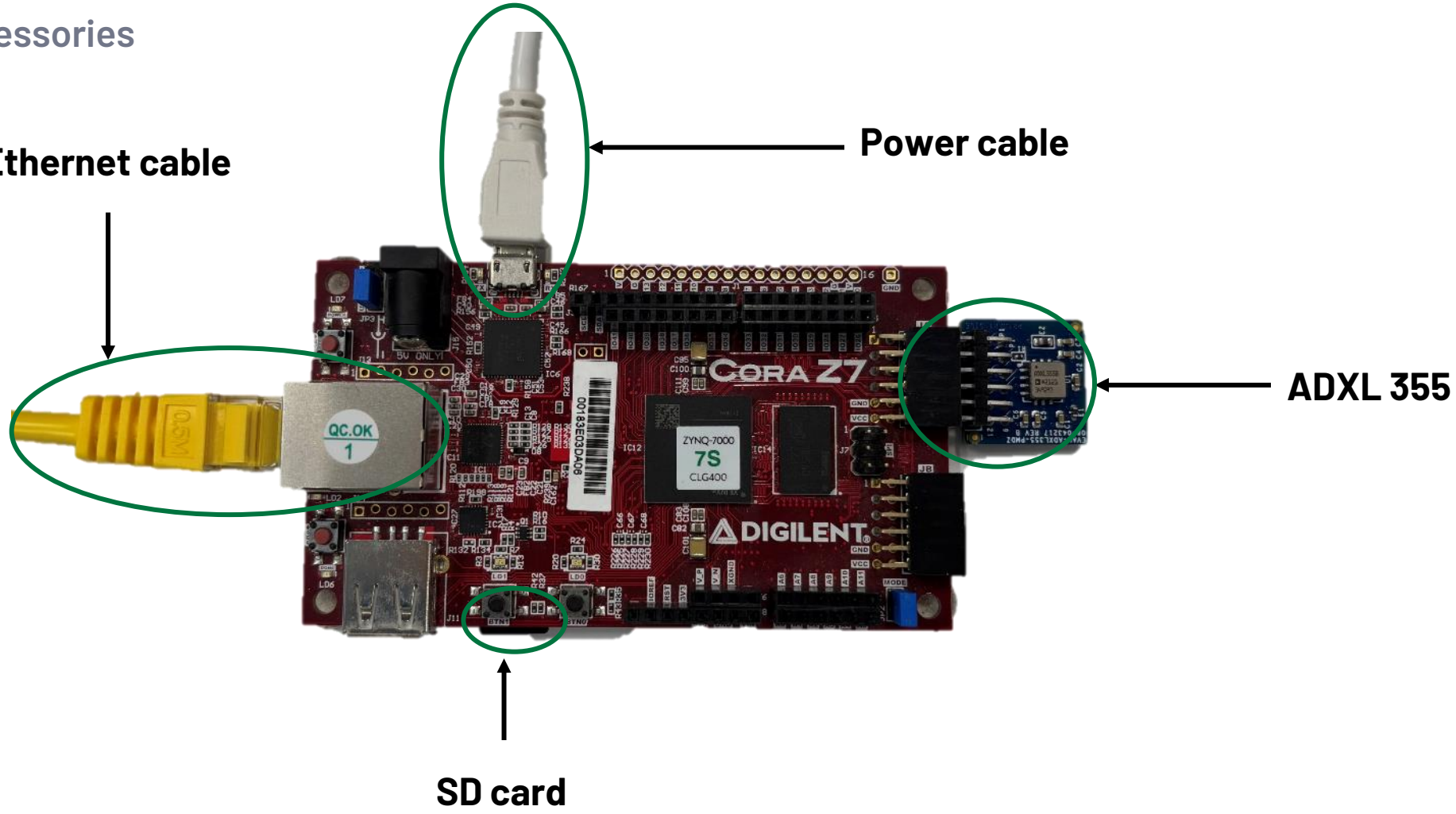
```
$ sudo bash build-docker.sh
```

Password: *analog*.

Congratulations! You started your first Kuiper 2 custom build!

Hands-On Activities

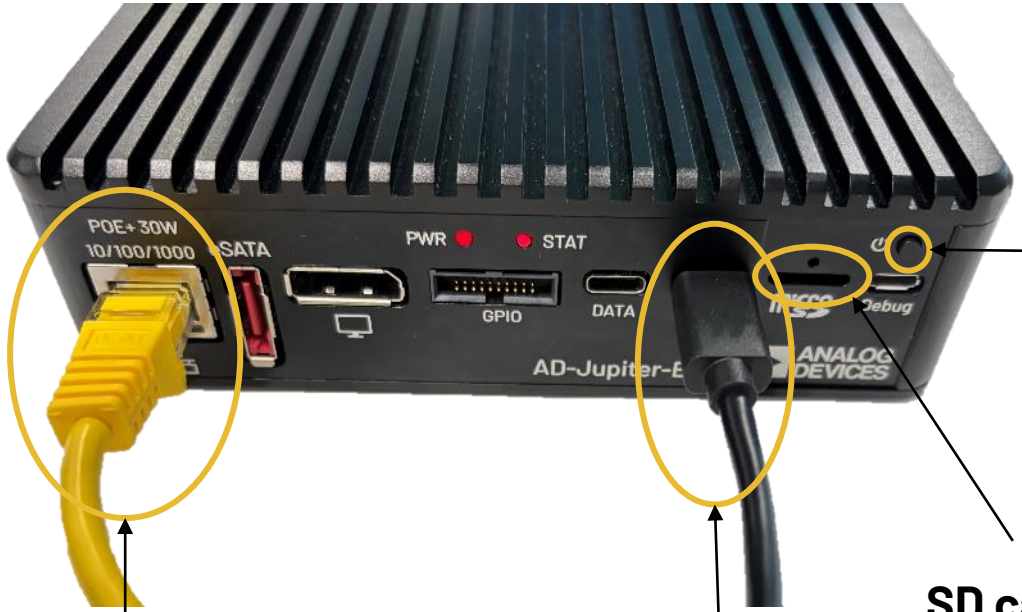
CoraZ7s and accessories



Hands-On Activities

Jupiter SDR and accessories

Loopback cable + 2 antennas



Power button

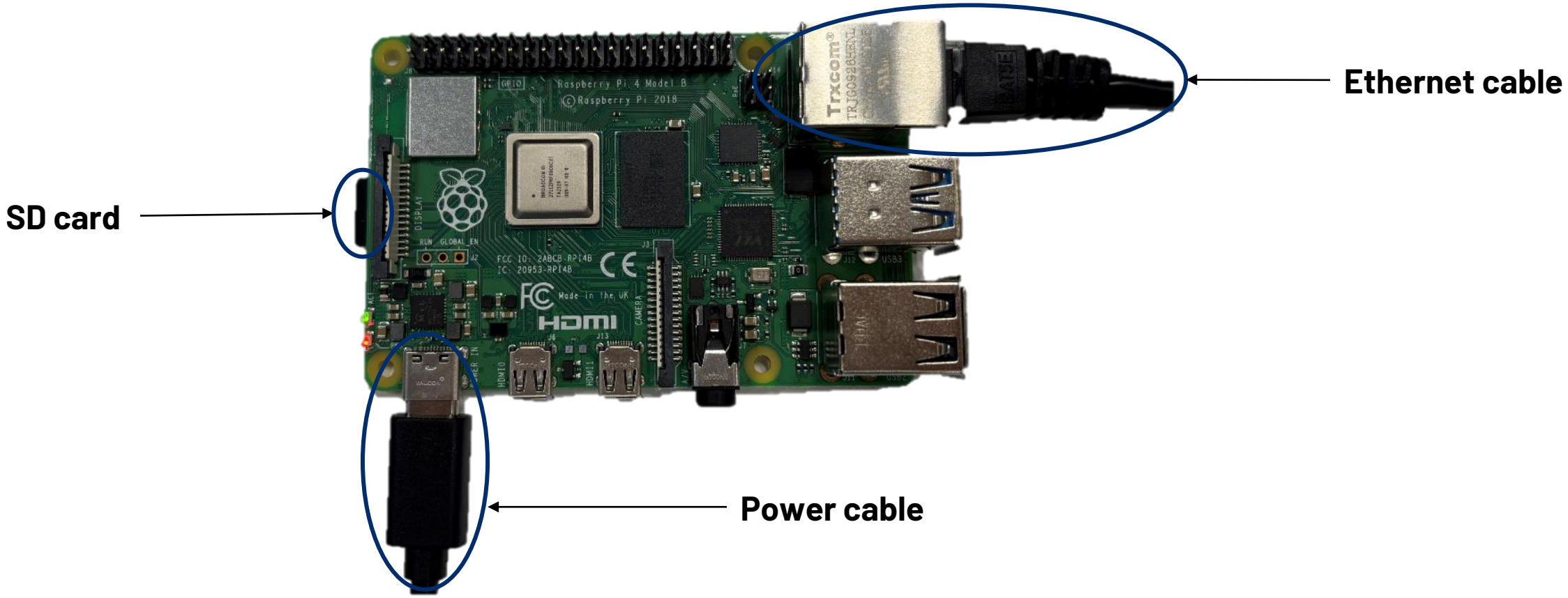
SD card

Ethernet cable

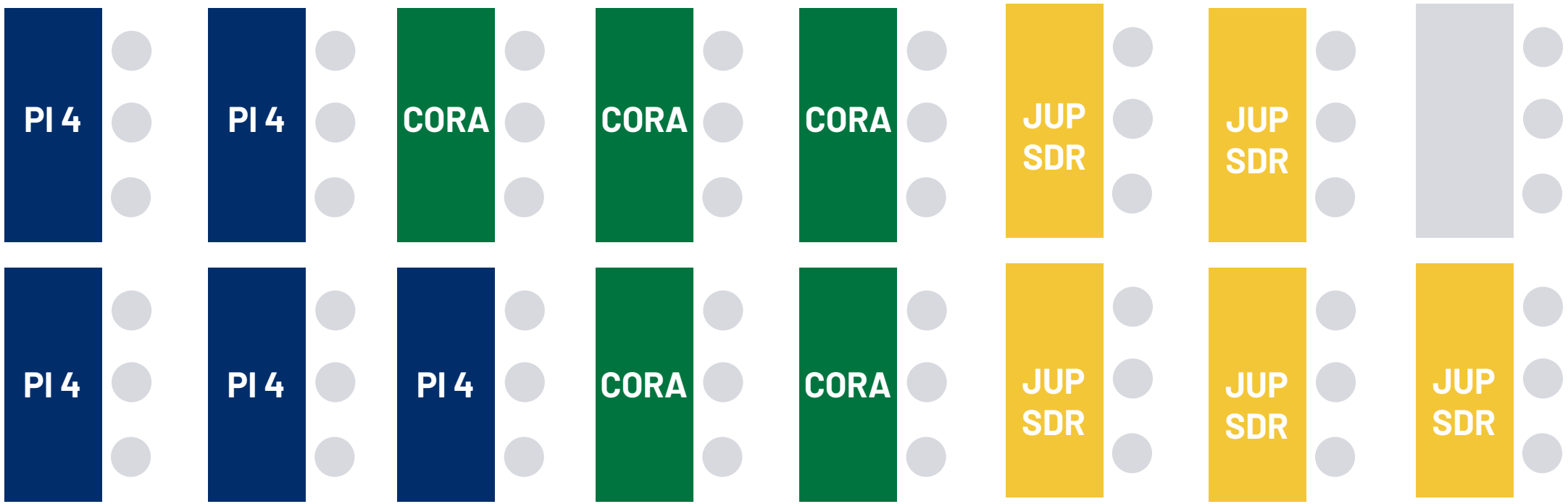
Power cable

Hands-On Activities

Raspberry Pi 4 and accessories



Hands-On Activities



Hands-On Activities

Networking and SSH

Wi-Fi Public IP
10.87.54.161

Raspberry Pi 5 Workstation

Local Link IP:
169.254.67.65

Cora Z7s

ftc25-cora@169.254.187.204

Jupiter SDR

ftc25-jupiter@169.254.107.214

Raspberry Pi 4

ftc25-rpi4@169.254.187.80

Hands-On Activities

Exercises

CoraZ7s

- Integrate and control hardware devices within Linux
- Use sensors and outputs to control interactive applications
- Apply scripting and automation to manage device behavior

Jupiter SDR

- Loopback testing with GNU Radio – transmit/receive complex sinewaves and visualize signals
- Wireless experimentation using antennas with hands-on RF parameter tuning

Raspberry Pi 4

- Read temperature sensors locally using IIO libraries and visualize data
- Access remote sensors over the network and compare RPi5 vs RPi4 temperatures in real-time

Hands-On Activities

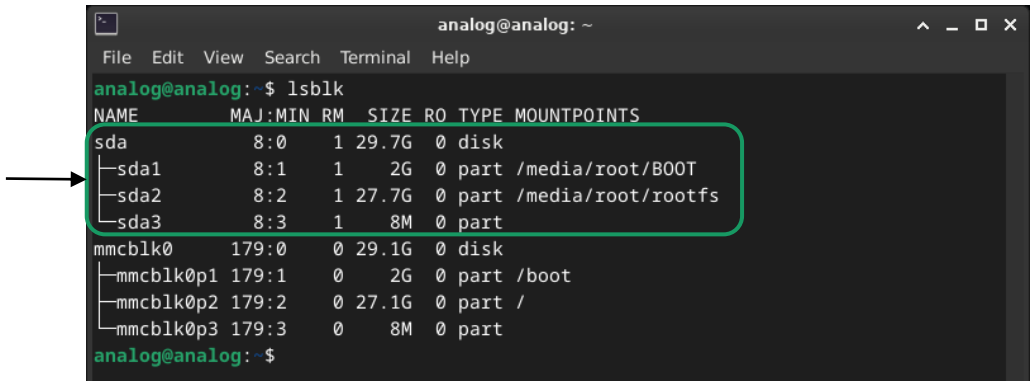
Prepare the SD card to boot

Steps

1. Connect the adapter via USB to the RPi5 workstation.
2. Insert the SD card from the carrier board (CoraZ7S, RPi 4, Jupiter SDR) into the adapter.
3. Check connected storage devices (typically /dev/sda)

```
$ lsblk
```

Identify the SD card (verify its size and mountpoint).



4. List available boot configurations:

```
$ sudo configure-setup.sh -b /media/root/BOOT --help
```

5. Prepare the boot files:

For Jupiter SDR:

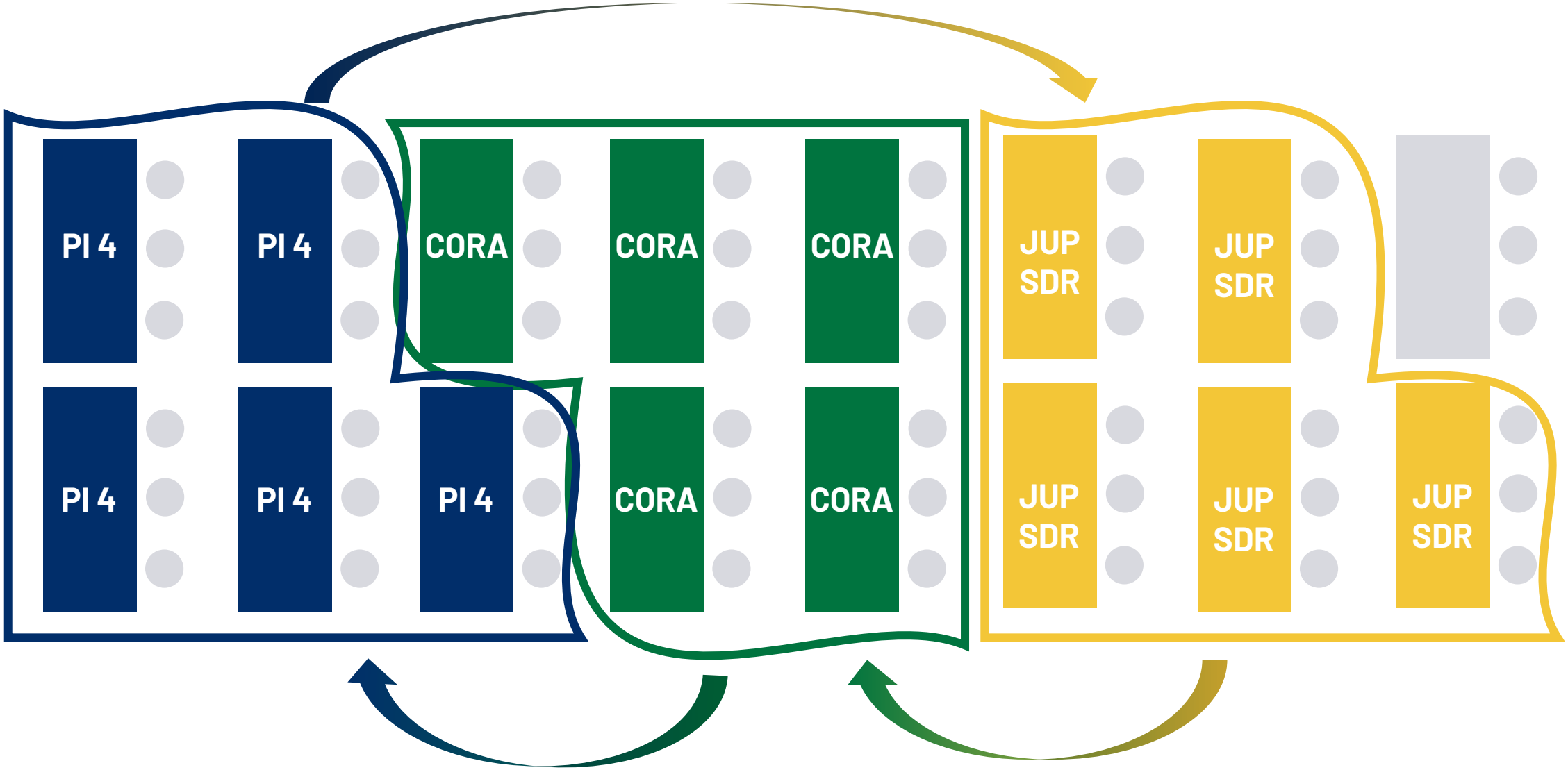
```
$ sudo configure-setup.sh -b /media/root/BOOT
jupiter_sdr jupiter_sdr
```

For CoraZ7s:

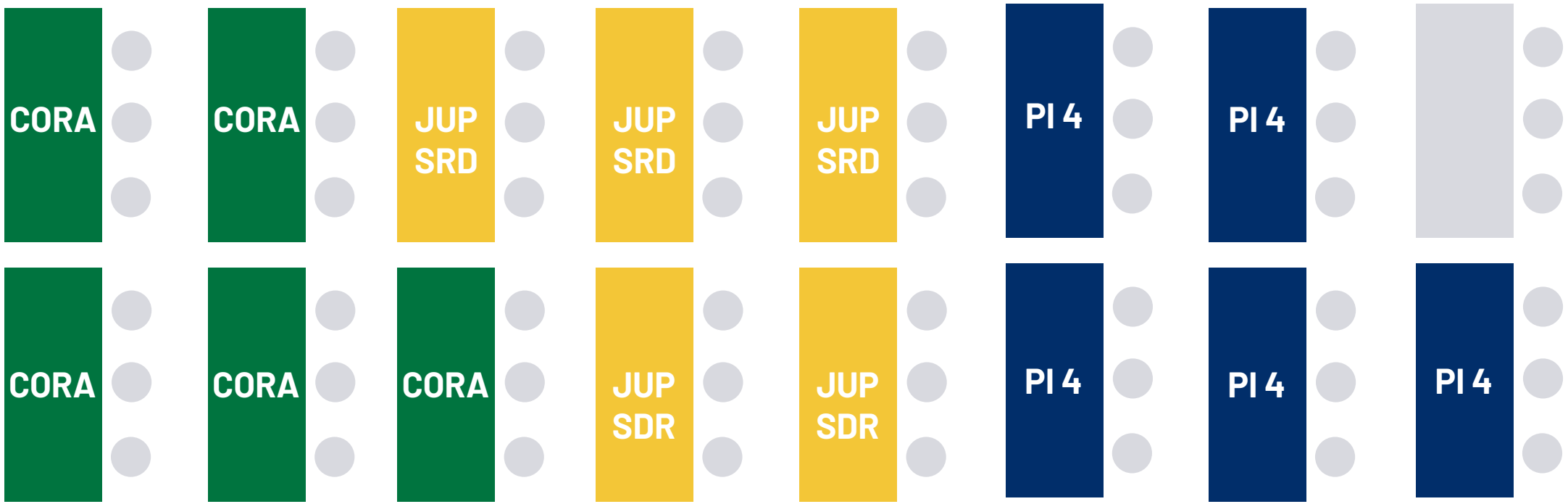
```
$ sudo configure-setup.sh -b /media/root/BOOT adxl355
coraz7s
```

No need for configuration on Raspberry Pi 4.

Hands-On Activities



Hands-On Activities



Hands-On Activities

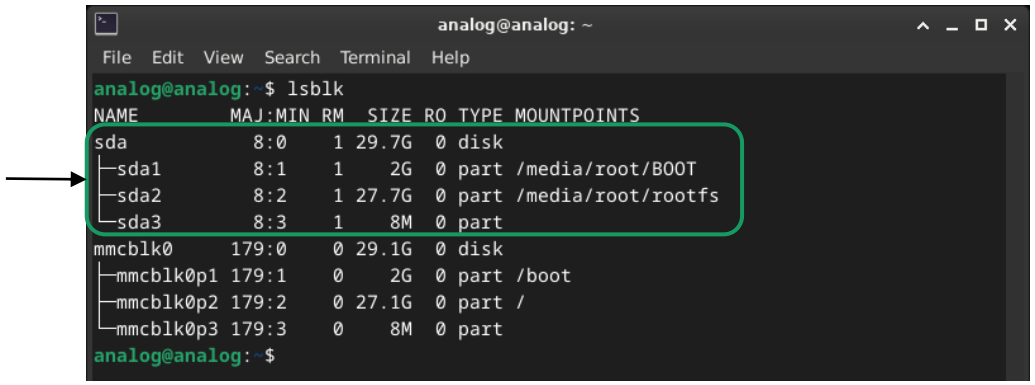
Prepare the SD card to boot

Steps

1. Connect the adapter via USB to the RPi5 workstation.
2. Insert the SD card from the carrier board (CoraZ7S, RPi 4, Jupiter SDR) into the adapter.
3. Check connected storage devices (typically /dev/sda)

```
$ lsblk
```

Identify the SD card (verify its size and mountpoint).



4. List available boot configurations:

```
$ sudo configure-setup.sh -b /media/root/BOOT --help
```

5. Prepare the boot files:

For Jupiter SDR:

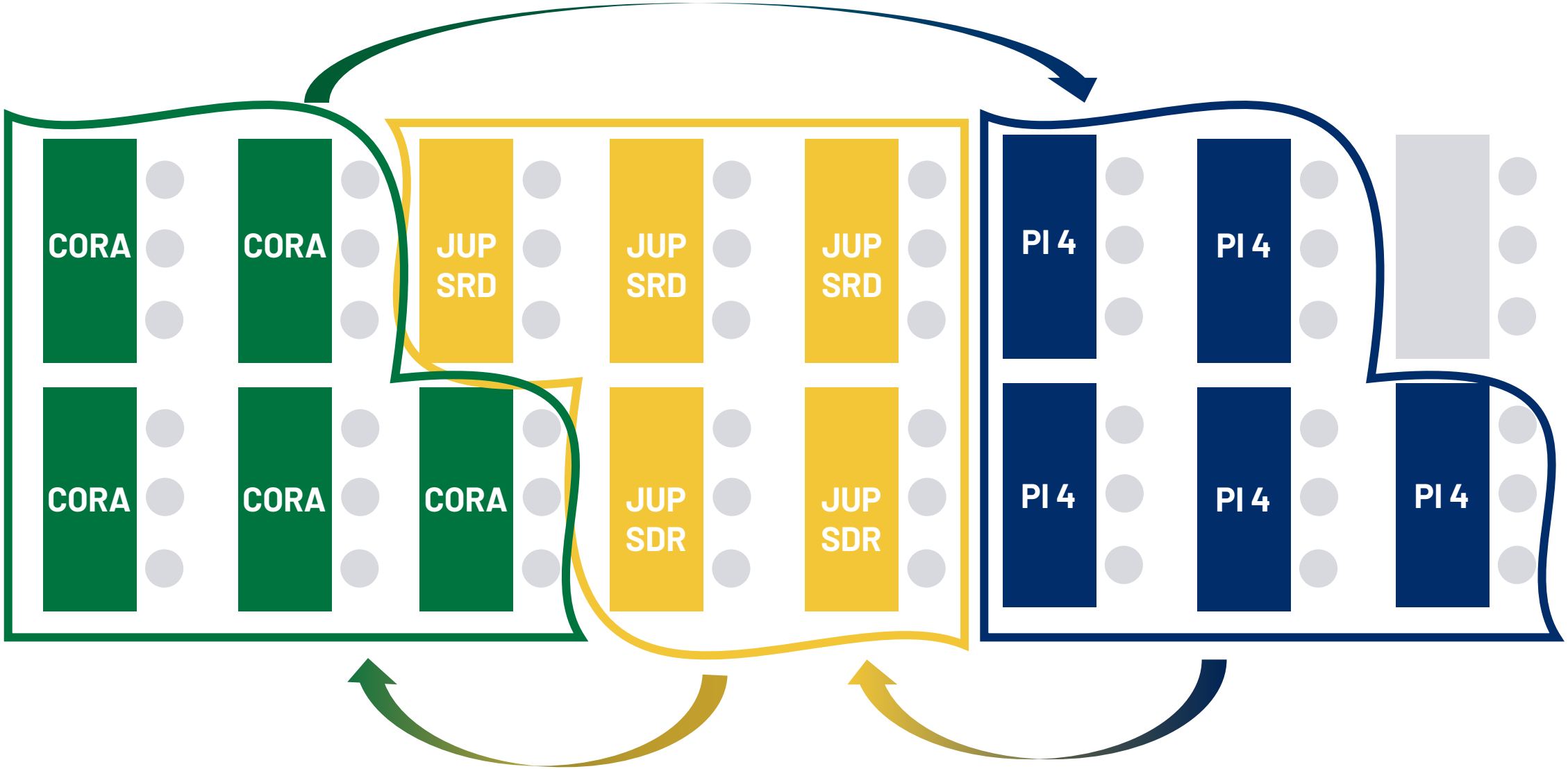
```
$ sudo configure-setup.sh -b /media/root/BOOT
jupiter_sdr jupiter_sdr
```

For CoraZ7s:

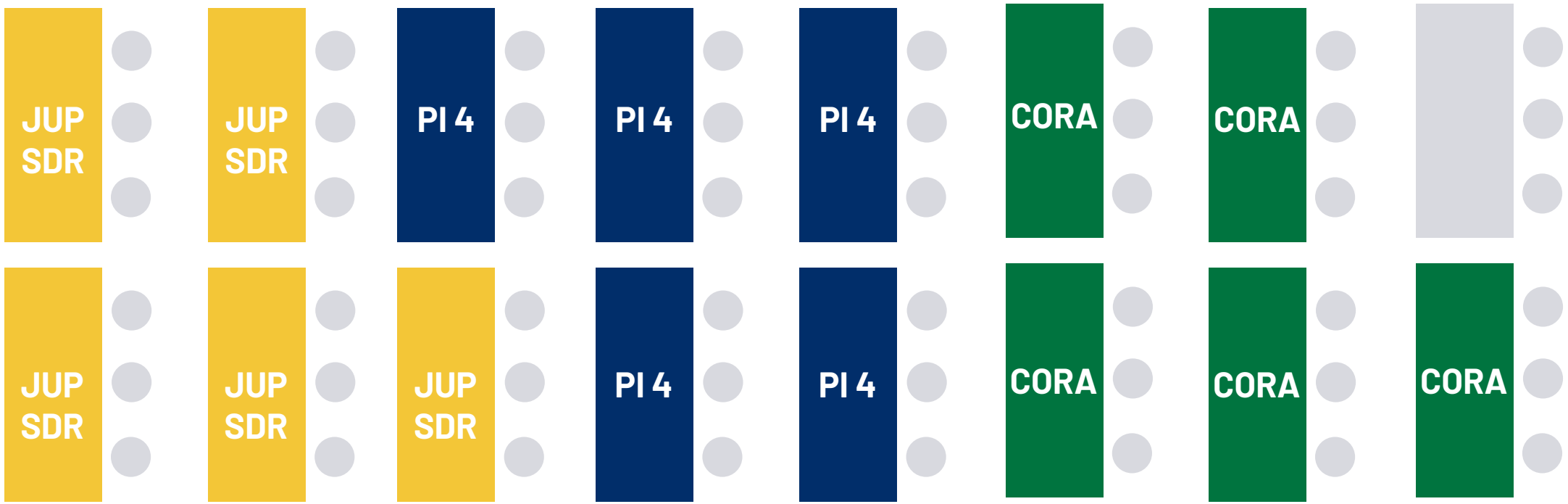
```
$ sudo configure-setup.sh -b /media/root/BOOT adxl355
coraz7s
```

No need for configuration on Raspberry Pi 4.

Hands-On Activities



Hands-On Activities



Hands-On Activities

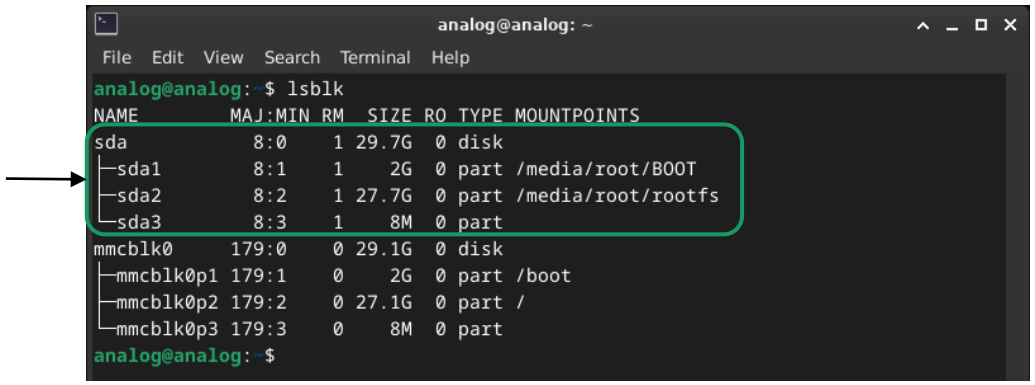
Prepare the SD card to boot

Steps

1. Connect the adapter via USB to the RPi5 workstation.
2. Insert the SD card from the carrier board (CoraZ7S, RPi 4, Jupiter SDR) into the adapter.
3. Check connected storage devices (typically /dev/sda)

```
$ lsblk
```

Identify the SD card (verify its size and mountpoint).



4. List available boot configurations:

```
$ sudo configure-setup.sh -b /media/root/BOOT --help
```

5. Prepare the boot files:

For Jupiter SDR:

```
$ sudo configure-setup.sh -b /media/root/BOOT
jupiter_sdr jupiter_sdr
```

For CoraZ7s:

```
$ sudo configure-setup.sh -b /media/root/BOOT adxl355
coraz7s
```

No need for configuration on Raspberry Pi 4.

Hands-On Activities

Raspberry Pi 5 Networking Exercise via MQTT

For this exercise you will communicate with other participants via a lightweight publish/subscribe protocol called MQTT.

Steps:

1. Open a terminal and run the following command to listen to a port where all participants will write messages:

```
$ mosquitto_sub -h test.mosquitto.org -t "ftc25-kuiper"
```

2. Open a new terminal and write the following command to send messages to the port everyone listens to:

```
$ mosquitto_pub -h test.mosquitto.org -t "ftc25-kuiper" -m "Hello from Pi $HOSTNAME"
```

Hands-On Activities

Boot your Custom Kuiper 2 Image on Raspberry Pi 5

Steps:

1. Power-off the carrier board in front of you (CoraZ7s, Raspberry Pi 4 or Jupiter SDR), take the SD card from it and connect it to the Raspberry Pi 5 via the adapter.

2. Open a terminal in *adi-kuiper-gen/kuiper-volume* folder, unarchive and copy the Kuiper image on the SD card. Use *lsblk* command to check the partition. As you saw earlier, the SD card from the adapter is mounted at SDA.

```
$ lsblk
$ sudo unzip image_2025-11-10-ADI-Kuiper-Linux-arm64.zip
$ sudo dd if=image_2025-11-10-ADI-Kuiper-Linux-arm64.img of=/dev/sda bs=1M status=progress conv=fsync
```

3. After the image is copied successfully, poweroff the Raspberry Pi 5 and disconnect it from the power cable, take out the SD card from it and replace it with the recently copied card, and then connect back the Pi board to power.

4. Look after the file you copied with the extra script and check what it contains.

Product Support and Acknowledgements

Contact information:

Larisa Radu – larisa.radu@analog.com

Alisa Roman – alisa.roman@analog.com

Stefan Raus – stefan.raus@analog.com

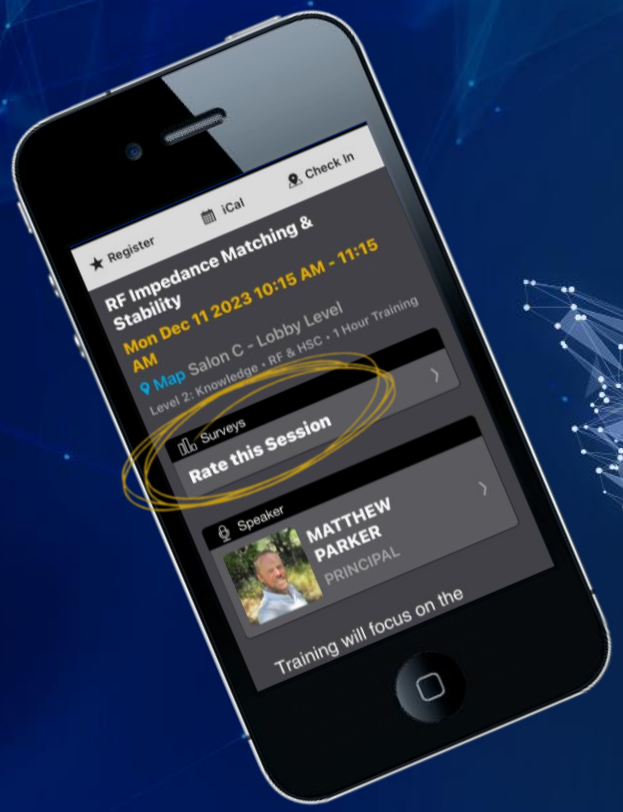
Dragos Bogdan – dragos.bogdan@analog.com

SDP Enablement - SDP_Enablement@analog.com

Questions and problems:

Linux Software Drivers – EngineerZone

Link to Kuiper images for this workshop:
https://github.com/analogdevicesinc/adi-kuiper-gen/actions/workflows/ftc25-build.yml



Thank You!

*Please Remember to **Rate** this Session in the Mobile App!*

AHEAD OF WHAT'S POSSIBLE

analog.com