Tutorial: How to Deploy a Program on an Electronic Board (e.g., KV260)

🧰 Prerequisites – Software to Install

- Vitis
- Vivado
- PowerShell (comes with Windows)
- PuTTY
- balenaEtcher

---

1. Create the Hardware Design in Vivado

1. Open **Vivado** and create a new project.
2. This project defines the **hardware configuration** of your board (e.g., FPGA logic).
3. Run the following steps in order:
    - **Run Linter**
    - **Run Synthesis**
    - **Run Implementation**
    - **Generate Bitstream**
4. Then, export the hardware:
    - Go to File > Export > Export Hardware
    - Check **Include bitstream**
    - This generates a .xsa file needed for the next step in Vitis.

---

2. Create the Embedded Application in Vitis

1. Launch **Vitis** and create a new **platform project**.
    - For the hardware design, select the .xsa file generated in Vivado.
    - Choose Hardware Design (XSA) for Implementation.
2. Then, create a new **Application Project**.
    - This is where you'll write your C code (e.g., hello_world.c).
    - Once ready, **build** the project to generate a .elf file (your program).
    - Fix any compilation errors if they appear.

---

3. Install Ubuntu on the SD Card

1. Download the correct version of Ubuntu for ARM (⚠️ not AMD/Intel):
   👉 Ubuntu for ARM boards (Kria/embedded)
2. Use **balenaEtcher** to flash Ubuntu onto the SD card:
    - Select the .img or .iso file you downloaded.
    - Insert the SD card into your PC.
    - Click **Flash** to install Ubuntu.

---

4. Connect to the KV260 Board

1. Connect the following to your KV260:
    - Power cable

- USB Serial cable
- Ethernet cable (to your PC or router)
- Insert the SD card with Ubuntu

2. On your PC, open **Device Manager** and find the **COM port** associated with the board.
3. Open **PuTTY**, select:
   - **Connection Type**: Serial
   - **Serial line**: COMxx (your detected port)
   - **Speed**: 115200
   - Click **Open**
4. When the terminal opens:
   - If prompted, press Enter to auto-select U-Boot mode.
   - Ubuntu should start booting.
   - Default login credentials:
     - **Username**: ubuntu
     - **Password**: ubuntu
   - You'll be asked to change the password. Choose a secure one and **don't lose it**!

---

5. Set a Static IP Address

Option 1 – On your PC (Windows)

- Go to:
  - Control Panel > Network and Internet > Network and Sharing Center > Change adapter settings
  - Right-click on Ethernet > Properties > IPv4 settings
  - Set a static IP, e.g.:
    - IP: 192.168.1.1
    - Subnet mask: 255.255.255.0

Option 2 – On the board (Ubuntu terminal via PuTTY)

1. First, check your current IP:

```
ip a
```

2. If no valid IP is set, edit the netplan config:

```
sudo nano /etc/netplan/01-netcfg.yaml
```

3. Example configuration:

```
network:
 version: 2
 renderer: networkd
 ethernets:
  eth0:
   addresses: [192.168.1.2/24]
   gateway4: 192.168.1.1
   nameservers:
    addresses: [8.8.8.8, 8.8.4.4]
```

4. Save and exit:
    - Ctrl+O, Enter, then Ctrl+X

5. Apply the changes:

```
sudo netplan apply
```

6. Verify:

```
ip a
ping 192.168.1.1  # Ping your PC to confirm the connection
```

---

6. Transfer and Run the Program on the Board

1. Open **PowerShell** on your PC and navigate to the directory containing the .elf file:

```
cd "C:\path\to\your\project"
```

2. Use scp to send the .elf file to the board:

```
scp hello_world.elf ubuntu@192.168.1.2:~
```

Replace:

- hello_world.elf with your actual filename
- 192.168.1.2 with the board's IP address

3. Log in to the board via SSH:

```
ssh ubuntu@192.168.1.2
```

4. Make the file executable:

```
chmod +x hello_world.elf
```

5. Run the program:

```
./hello_world.elf
```

Conclusion

You've now successfully:

- Created a hardware configuration using Vivado
- Built and compiled a C program with Vitis
- Flashed Ubuntu on the board
- Configured the network
- Deployed and executed your program on the embedded system