Session - 43

Eclipse for BeagleBone Black

Setting up **Eclipse for BeagleBone Black (BBB)** development involves configuring Eclipse to **cross-compile** code on your host PC and then **deploy** and **debug** it on the BBB. Here's a full step-by-step guide:

⚙️ Eclipse Setup for BeagleBone Black (BBB)

✅ 1. Prerequisites (on Host PC)

Install the following on your **Linux host PC** (Ubuntu recommended):

sudo apt update

sudo apt install eclipse build-essential g++ gdb-arm-none-eabi cmake

Also install the **GNU ARM cross-toolchain**:

sudo apt install gcc-arm-linux-gnueabihf g++-arm-linux-gnueabihf

✅ 2. Download Eclipse IDE for C/C++ Developers

- Download from: https://www.eclipse.org/downloads/
- Install the **"Eclipse IDE for C/C++ Developers"**

✅ 3. Create a New C/C++ Project

1. Open Eclipse.
2. Go to File → New → C Project

3. Choose Hello World ANSI C Project or Empty Project.
4. **Toolchains:** Choose **Cross GCC**.
5. Name your project (e.g., bbb_project), then click **Finish**.

✅ 4. Configure Cross-Compiler

1. Right-click your project → **Properties**
2. Go to C/C++ Build → Settings
3. Under **Tool Settings**, set:
    - **Cross GCC Compiler:** arm-linux-gnueabihf-gcc
    - **Cross G++ Compiler:** arm-linux-gnueabihf-g++
4. Under C/C++ Build → Toolchain, set prefix:

arm-linux-gnueabihf-

and path to toolchain:

/usr/bin

✅ 5. Add Remote BBB Connection (via SSH)

1. Go to Window → Preferences → C/C++ → Remote Systems
2. Add a new **Remote Connection**
    - Type: **SSH Only**
    - Host: <BBB IP address>
    - User: debian or root
3. Test the connection and save.

✅ 6. Remote Build & Deploy (optional)

You can either:

- **Build locally** (cross-compile) and **manually copy** the binary via SCP:

scp ./bbb_project debian@<BBB_IP>:/home/debian/

- Or set up **Remote Tools** or **Remote System Explorer** to automatically deploy.

✅ 7. Debugging (optional)

To enable **remote debugging**:

1. Install gdbserver on BBB:

sudo apt install gdbserver

2. Run on BBB:

gdbserver :2345 ./your_program

3. On host PC, in Eclipse:
    - Create a **Remote Debug Configuration**
    - Set connection to BBB IP, port 2345

🧪 Quick Test Code

```
#include <stdio.h>


int main() {
   printf("Hello from BeagleBone Black!\n");
   return 0;
}
```

Build, transfer to BBB, then run:

./bbb_project

🧠 Summary

Step	Task
1	Install Eclipse & cross-compiler
2	Create project & set toolchain
3	Build C/C++ code for ARM
4	Transfer binary to BBB
5	Run or debug on target