Embedded Linux laboratory

May 8, 2026

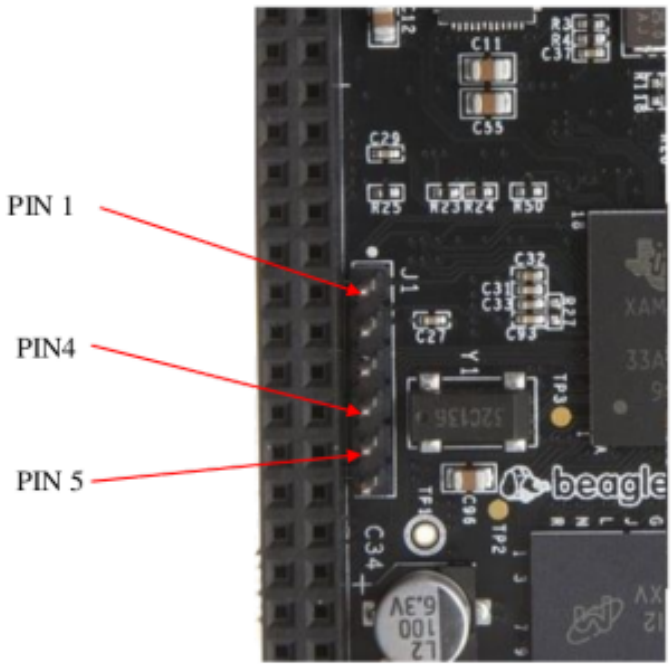
Embedded Linux Laboratory: reference books

- “Embedded Linux System with the Yocto Project”
- “Using Yocto Project with BeagleBone Black”

Embedded Linux Laboratory

Function	FTDI Cable	BeagleBone
Ground	Black GND wire	Pin 1 J1 Header GND
TX→RX	Orange transmit wire	Pin 4 J1 Header RXD
RX←TX	Yellow receive wire	Pin 5 J1 Header TXD

Figure: BBB Serial Port (J1 header)



Embedded Linux Laboratory

1. **System Compilation:**
   - Start from the `yoctobbb` Docker container
   - Configure the Yocto project properly
   - Initiate the build process using BitBake
   - Check results under `deploy` directory
2. **System Deployment:**
   - Copy and install the bootloader, Linux kernel, and filesystem to the SD card
   - Boot the BeagleBone Black board with the custom image
   - Resize rootfs to maximum with `parted` and `resize2fs`
3. **Cross-Compilation Test:**
   - Cross-compile a sample application within the Docker container
   - Verify the target architecture using `file` command on the produced ELF binary

Embedded Linux Laboratory

4. **Example with devtool and bitbake-layers:**
   - Use devtool to configure your source code
   - `https://github.com/ltenze/helloworld.git`
5. **Adding a Custom Package to the Filesystem:**
   - Compile a new package (e.g., `pacman`) using BitBake
   - Integrate the package into the root filesystem using `devtool`
6. **Kernel Configuration and Compilation:**
   - Modify the Linux kernel configuration (e.g., via `menuconfig`)
   - Recompile the kernel with the new settings
   - Verify the changes in the updated kernel image
   - `https://wiki.koansoftware.com/index.php/Using_devtool_to_modify_recipes_in_Yocto`