Embedded Linux

```
Module Code: ELEE1119

Module Name: Advanced Computer Engineering

Credits: 30

Module Leader: Seb Blair BEng(H) PGCAP MIET MIHEEM FHEA
```

Download as a PDF

What is Embedded Linux?

- The system
  - Intuitively, an embedded Linux system simply denotes an embedded system running on the Linux kernel. Let us focus on the remaining two pieces.
- The Linux Kernel
  - Linus never shipped an “embedded version” of the Linux kernel. If you are an embedded developer, you may not require a tailored kernel for your system and might rely on an official release instead.
- The Distro
  - Umbrella term usually comprising software packages, services and a development framework on top of the OS itself

Why Embedded Linux?

- Hardware support
- Networking
- Modularity
- Commercial support

Hardware Support

- Linux runs on 32 and 64-bit ARM, x86, MIPS, and PowerPC architectures.
- Whereas a 32-bit processor is capable of storing 2^{32} values, a 64-bit processor can store more memory addresses.
- Processors below 32-bit aren’t capable of running Linux, ruling out traditional embedded systems.

Networking

Modularity

Commercial Support

RISC, ARM, x86

	RISC-V	ARM	x86
Origin	RISC-V International	Arm Ltd.	Intel and AMD
Instruction Set	RISC	RISC	CISC
Byte Order	Typically little-endian (user-configurable)	Typically bi-endian (user-configurable)	Little-endian
Applications	Embedded systems, IoT devices, custom solutions	Mobile devices, embedded systems, servers	Desktops, laptops, servers, workstations
Licensing Model	Open-source, royalty-free licensing	ARM licenses its designs to manufacturers	Intel and AMD produce their own chips
Ecosystem	Developing ecosystem, open-source initiatives	Large ecosystem, extensive third-party support	Large software and hardware ecosystem

Memory

The memory configuration of the embedded system, which includes the types of memory used (such as SRAM, DRAM, or non-volatile memory), plays a crucial role in determining the memory start address.

Based on ARM2+

System Archiecture:

System Architecture: The architecture of the system-on-a-chip (SoC) can influence where memory starts. Some systems have all their memory directly addressable by the main processor2.
Application Requirements: The specific needs of the application that the embedded system is designed to run can dictate the memory architecture, including the start address. For example, if an application requires fast access to certain data, it might be placed at the beginning of the memory space2.

Processor Design:

Processor Design: The design of the processor within the embedded device can also affect the memory start address. Processors may have a direct addressing mode that specifies a certain range of addresses for memory access.

For instance:

Arm... 0x02000 (BeagleBone Black)

Arm.... (RPI 5)

Arm..... (Rock C4+)

Communication note

The stop bit, as Chris notes, can be set to 1, 1.5 or 2 bits. If you're wondering how you can have 1.5 bits, the stop "bit" is really holding a signal state on the wire for a certain