Firmware/Software

Industrial IoT

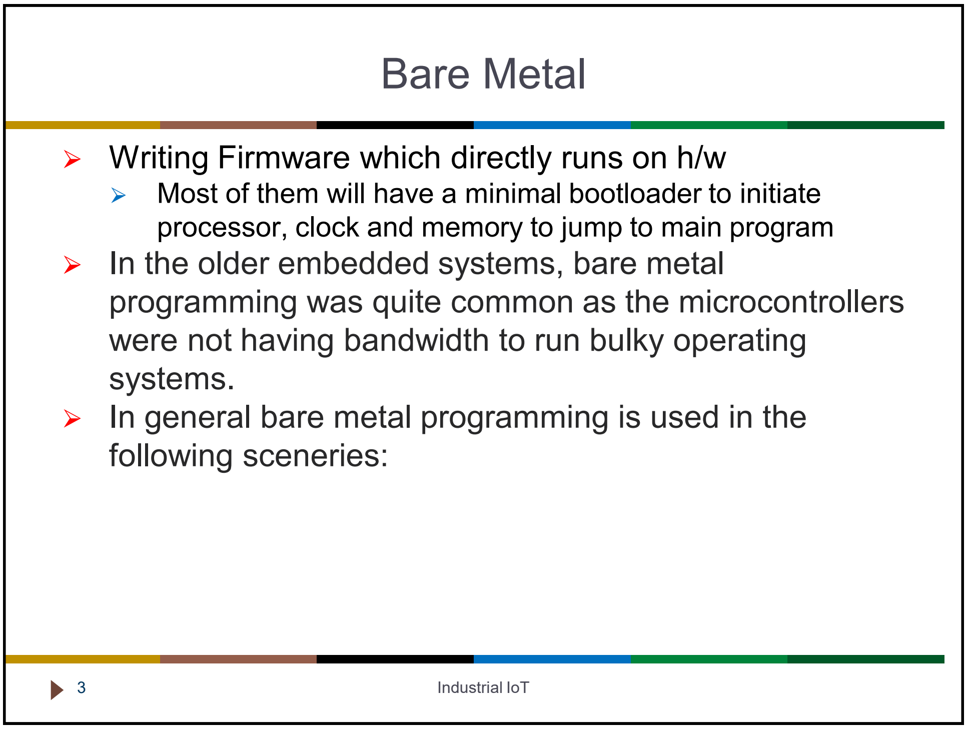
Christos Sotiriou

1

Contents

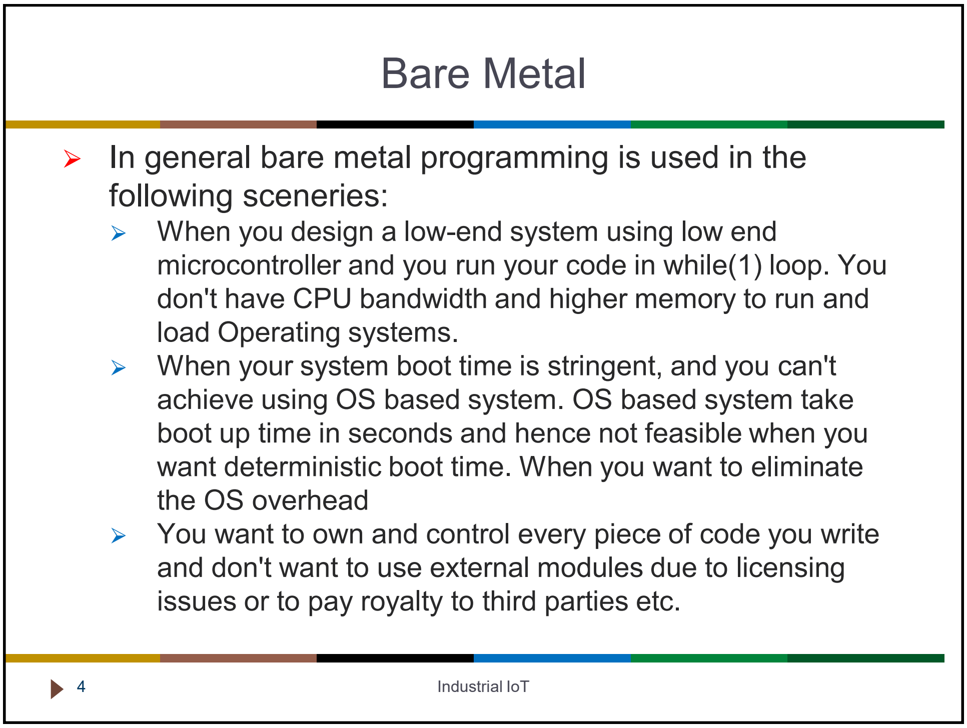
- ➢ **Bare Metal**
- ➢ Embedded OS
- ➢ Creating Custom Embedded Linux (Yocto)
- ➢ High Level Languages for Embedded Systems

2



Bare Metal

- Writing Firmware which directly runs on h/w
  - Most of them will have a minimal bootloader to initiate processor, clock and memory to jump to main program
- In the older embedded systems, bare metal programming was quite common as the microcontrollers were not having bandwidth to run bulky operating systems.
- In general bare metal programming is used in the following sceneries:



Bare Metal

- In general bare metal programming is used in the following sceneries:
  - When you design a low-end system using low end microcontroller and you run your code in while(1) loop. You don't have CPU bandwidth and higher memory to run and load Operating systems.
  - When your system boot time is stringent, and you can't achieve using OS based system. OS based system take boot up time in seconds and hence not feasible when you want deterministic boot time. When you want to eliminate the OS overhead
  - You want to own and control every piece of code you write and don't want to use external modules due to licensing issues or to pay royalty to third parties etc.

Bare Metal

- But there are some drawbacks also.
  - You don't get the luxury of having OS which abstracts you from underlying hardware, you don't have to worry about writing device drivers, scheduler, memory management, threads management etc. All you to do is to focus on your application development.
  - You just have to call API's and no need to worry about layering your code,
  - No need to worry about fixing driver or kernel level issues as those modules are already in good shape..

Contents

- Bare Metal
- **Embedded OS**
- Creating Cutom Embedded Linux (Yocto)
- High Level Languages for Embedded Systems

Embedded OS

- A specialized operating system designed to perform a specific task for a device that is not a computer.
- The embedded OS also makes the device's hardware accessible to the software that is running on top of the OS.
- Most of them are **Open Source** and contributed by hardware design companies or companies who use their benefits.
  - They have open source licenses:
    - Apache 2.0
    - MIT
    - …

Embedded OS

- Most known Embedded OS:

freeRTOS

Zephyr™

Contiki

The Open Source OS for the Internet of Things

TinyOS

Embedded OS

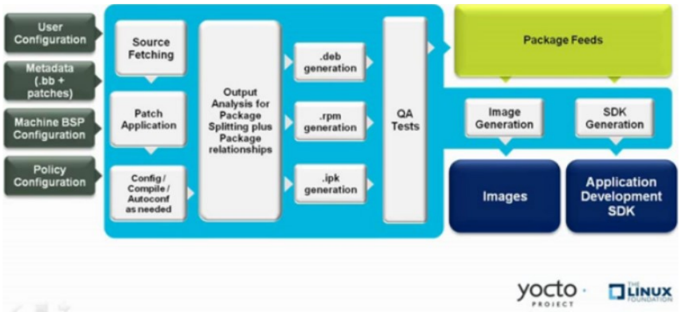
- ➢ Most of them support WSN operations:
  - ➢ Bluetooth® Low Energy, Wi-Fi*, 802.15.4
- ➢ Minimized Linux kernel
- ➢ Simple schedulers to support a few threads
- ➢ Drivers specific for IoT applications

Contents

- ➢ Bare Metal
- ➢ Embedded OS
- ➢ **Creating Custom Embedded Linux (Yocto)**
- ➢ High Level Languages for Embedded Systems

Creating Custom Embedded Linux

- In case you want to make your custom Embedded OS
  - Yocto
  - Petalinux (for Xilinx SoC's)



Yocto Project (1)

- The Yocto Project is an open source collaboration project that helps developers create custom Linux-based systems that are designed for embedded products regardless of the product's hardware architecture
- Yocto Project provides a flexible toolset and a development environment that allows embedded device developers across the world to collaborate through shared technologies, software stacks, configurations, and best practices used to create these tailored Linux images.

Contents

- Bare Metal
- Embedded OS
- Creating Custom Embedded Linux (Yocto)
- **High Level Languages for Embedded Systems**

Python in Embedded Systems

- Python is the most used Programming Language in CS and easy to start using it
- Mostly used today from hobbyists and people who are not used in C/C++
  - MicroPython
  - Raspberry Pi
- In the future, it is said to be the most used language also for embedded systems

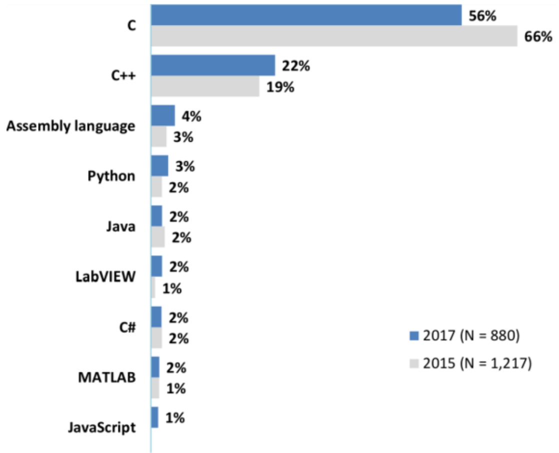
Python vs C

Python	C
Quick Development	Lower Development
shallow learning curve	Has to
Interpreter	Compiler
Runtime	Lower Runtime Execution

JavaScript in Embedded Systems

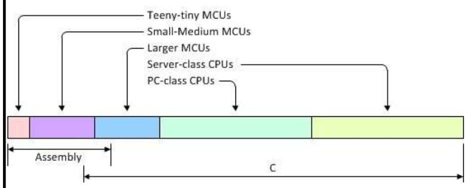
- Mostly used by Web-Developers when writing f/w for embedded devices
- Mostly used for Gateways, where a web interface is more needed

Most Used Languages in Embedded Systems

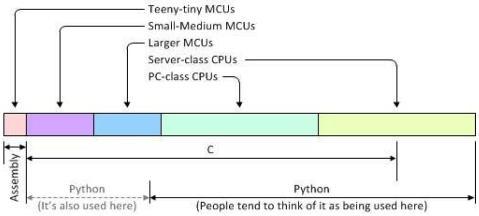


Source: https://www.cnx-software.com/2017/08/15/aspencore-2017-embedded-markets-study-programming-languages-operating-systems-mcu-vendors-and-more//

Where are languages more used



1990



2016

Source: https://www.embedded.com/python-is-better-than-c-or-is-it-the-other-way-round/

References

- https://www.quora.com/What-is-bare-metal-programming-in-Embedded-systemsEmbedded OS
- https://docs.zephyrproject.org/latest
- https://www.cnx-software.com/2017/08/15/aspencore-2017-embedded-markets-study-programming-languages-operating-systems-mcu-vendors-and-more/
- **https://www.embedded.com/python-is-better-than-c-or-is-it-the-other-way-round/**
- https://micropython.org/
- https://www.youtube.com/watch?v=jR5E5Kz9A-k&t=427s&ab_channel=NordicSemiconductor
- https://www.youtube.com/watch?v=3EjHXL3PQ04&ab_channel=WendelineGabrielle