Using NuttX RTOS for Industrial IoT Solutions

Alan Carvalho de Assis, MSXi / Keenix
Agenda:

- Introduction to IIoT (or Industry 4.0)
- Introduction to NuttX RTOS
- Comparing Linux with NuttX
- Why to use NuttX for Industrial Applications?
- How to get involved?
The Industrial Revolutions ...

1st
Mechanization, water power, steam power

2nd
Mass production, assembly line, electricity

3rd
Computer and automation

4th
Cyber Physical Systems
The 4th Industrial Revolution

- Decentralized control;
- Driven by the Internet of Things (IoT);
- Manufacturing plant becomes smart;
- Since devices are connected to the Internet cyberattack is a real issue.
Industry 4.0 Principles:

- Interoperability;
- Information transparency;
- Service-Orientation;
- Decentralization;
- Virtualization;
- Modularity
Main components:
I.E.: Fiat Chrysler Automobiles’s Jeep Factory

- 700 robots mount almost everything in the car;
- Every single piece of the car is traceable;
- The factory system is integrated with supplier;
- This is the first FCA factory built following the Industry 4.0 concepts.
What an OS/RTOS needs to support IIoT:

- Network support: IPv4, IPv6, 6LoWPAN, MQTT, etc;
- Connectivity: WiFi, Bluetooth, RFID/NFC, etc;
- Industrial interfaces: RS485, CAN-Bus, EtherCAT, etc;
- Realtime support for processes and robot control;
What an OS/RTOS needs to support IIoT: (cont)

• Program Languages: C/C++, Java, Python, Lua, etc;
• AI library/tools (libdeep, MPlib, Torch, TensorFlow, etc);
• Image Recognition libraries (OpenCV, libccv, etc);
• Security: Crypto AES, DTLS, TLS, etc.
Here comes NuttX RTOS!
What is the NuttX RTOS?

• Created by Gregory E. Nutt;
• First public release: Feb 2007;
• Supports 8 to 32 bits uCs/uPs;
• Release under BSD License!
What is the NuttX RTOS? (cont)

• Small footprint;
• Very customizable;
• Inspiration from Linux/Unix:
  • VFS;
  • MTD;
  • PROCFS;
  • NuttShell;
  • Etc.
What is the NuttX RTOS? (cont)

- Supports:
  - ARM (ARM7, ARM9, ARM11, Cortex-Mx, Cortex-Rx, Cortex-Ax);
  - AVR
  - MIPS
  - LM32
  - RISC-V
  - X86
  - Xtensa, Z80, etc;
Companies using NuttX

- Sony;
- Samsung;
- Motorola Mobility;
- Haltian;
- 3DRobotics;
- Daruma;
- VergeAero;
- Many others.
NuttX’s key features:

• POSIX complaint;
• Fully preemptible;
• Virtual File System (VFS);
• Loadable kernel modules;
• Symmetric Multi-Processing (SMP);
• Realtime scheduling (FIFO, RR, SPORADIC);
• Tickless operation support (lower power consumption);
• Pseudo-terminals (PTY) and I/O redirection;
NuttX’s key features: (cont)

• Memory modes:
  • FLAT (no MMU),
  • PROTECTED (with MPU),
  • KERNEL (with MMU).
• High Performance/Zero Latency Interrupts (ARM);
• Native debug log support (INFO, WARN, ERR);
• Power Management;
• System log support.
File Systems support:

- FAT12/16/32;
- NFS;
- BINFS;
- SMARTFS;
- ROMFS;
- PROCFS;
- UnionFS;
- TMPFS;
Network Support:

- Multiple network interfaces support;
- Network routing support;
- Support to IPv4, IPv6, TCP, UDP, ICMP, etc;
- Unix socket;
- DNS name resolution;
- IEEE 802.11 (WiFi) FullMac;
- IEEE 802.15.4 (MAC, 6LoWPAN);
- SLIP (Serial), PPP (GSM Modem);
Graphical Support:

- Framebuffer support (similar to Linux FB);
- LCD Support (Parallel / Serial);
- NX Graphics server (similar to X server idea);
- NX Graphics lib (fonts, lines, rectangles, etc);
- NXWidgets: high level user interface;
- Foreign graphics supported (i.e. LittlevGL);
- VNC Server support.
Audio Support:

- Audio Tone Generator;
- Audio codecs: VS1053 SPI, CS43L22, WM8776, WM8904;
- NXPlayer audio player w/ HTTP download stream.
USB Host Classes supported:

- USB CDC/ACM Modem Class (i.e.: ttyACM0);
- USB Mass Storage Class (pen-driver);
- USB HID Keyboard Class;
- USB HID Mouse Class;
- USB HUB Support.
USB Devices Classes supported:

- USB CDC/ACM Modem Class (i.e.: ttyACM0);
- USB Mass Storage Class (pen-driver);
- USB RNDIS Ethernet-over-USB;
- USB Composite Device Support;
- USB Prolific PL2303 USB/Serial Emulation.
Microcontrollers with better support:

- STMicro STM32;
- Microchip SAM3/SAM4;
- Microchip SAMV7;
- NXP LPC17xx;
- NXP LPC43xx;
- NXP i.MX RT 10xx.
Comparing Linux vs NuttX:

- Linux:
  - High-end processors
  - MMU (except uClinux);
  - 32-bit (except ELKS);
  - Requires 8MB+ RAM;
  - Requires 2MB+ Flash;
  - Started for PC and was adapted for embedded;
  - No initial support for Realtime;
  - Power hungry.
Comparing Linux vs NuttX: (cont)

- **NuttX:**
  - Low/Mid-end microcontrollers/microprocessors
    - no MMU required;
    - 8/16/32-bit;
  - Requires 8KB+ RAM;
  - Requires 16KB+ Flash;
  - Started for 8-bit MCU;
  - Realtime from scratch;
  - Low power consumption.
Why to use NuttX for IIoT:

• Linux compatibility;
• Broad range of features;
• POSIX compliance (easy to port applications);
• Easy to move to high-end OS in the future;
• Flexibility to move among supported MCUs;
Linux applications ported to NuttX:

- FreeModBus;
- BACnet stack;
- LLVM libc++;
- SQLite;
- Many others;
Contact

• Alan Carvalho de Assis
• Email: acassis@gmail.com
• Blog: acassis.wordpress.com