



Introduction

Building Extensions of Embedded Systems

Rafal Kapela

June 26, 2016

Outline



1 What is ES?

2 What is not ES?

3 Where you can find ES?

Outline



1 What is ES?

2 What is not ES?

3 Where you can find ES?

Introduction to ES's

What is an embedded system?



- An embedded system is a combination of the computer hardware and software accomplished with additional mechanical or other parts designed to perform a specific function.
- Embedded software is an almost every electronic device in the use today. There is a software hidden away inside our watches, cellular phones even washing machines!
- A well-designed embedded system *conceals* the existence of the processor and the software.
- Each embedded system is usually unique, and the hardware is highly specialized to the application domain.

Introduction to ES's

What is an embedded system?



- An embedded system is a combination of the computer hardware and software accomplished with additional mechanical or other parts designed to perform a specific function.
- Embedded software is an almost every electronic device in the use today. There is a software hidden away inside our watches, cellular phones even washing machines!
- A well-designed embedded system *conceals* the existence of the processor and the software.
- Each embedded system is usually unique, and the hardware is highly specialized to the application domain.

Introduction to ES's

What is an embedded system?



- An embedded system is a combination of the computer hardware and software accomplished with additional mechanical or other parts designed to perform a specific function.
- Embedded software is an almost every electronic device in the use today. There is a software hidden away inside our watches, cellular phones even washing machines!
- A well-designed embedded system *conceals* the existence of the processor and the software.
- Each embedded system is usually unique, and the hardware is highly specialized to the application domain.

Introduction to ES's

What is an embedded system?



- An embedded system is a combination of the computer hardware and software accomplished with additional mechanical or other parts designed to perform a specific function.
- Embedded software is an almost every electronic device in the use today. There is a software hidden away inside our watches, cellular phones even washing machines!
- A well-designed embedded system *conceals* the existence of the processor and the software.
- Each embedded system is usually unique, and the hardware is highly specialized to the application domain.

Introduction to ES's

ES vs RTS (Real-time system)



- A real time system is specified in terms of its ability to make certain calculations or decisions in a timely manner to face the deadline.
- A missed deadline is just as bad as a wrong answer which is very crucial.
- A real-time system must be diligently designed which must guarantee reliable operation of the hardware and the software under all possible conditions.

Introduction to ES's

ES vs RTS (Real-time system)



- A real time system is specified in terms of its ability to make certain calculations or decisions in a timely manner to face the deadline.
- A missed deadline is just as bad as a wrong answer which is very crucial.
- A real-time system must be diligently designed which must guarantee reliable operation of the hardware and the software under all possible conditions.

Introduction to ES's

ES vs RTS (Real-time system)



- A real time system is specified in terms of its ability to make certain calculations or decisions in a timely manner to face the deadline.
- A missed deadline is just as bad as a wrong answer which is very crucial.
- A real-time system must be diligently designed which must guarantee reliable operation of the hardware and the software under all possible conditions.

Introduction to ES's

Hard RTS vs soft RTS



- A hard real-time system guarantees that critical tasks be completed on time, the goal requires that all the delays in the system need to be bounded. The current task is of high priority and there shall be no preemption.
- A less restrictive type of real-time system is a soft real-time system, where a critical real time-task gets priority over other tasks, and retains the priority until it completes it's task.

Introduction to ES's

Hard RTS vs soft RTS



- A hard real-time system guarantees that critical tasks be completed on time, the goal requires that all the delays in the system need to be bounded. The current task is of high priority and there shall be no preemption.
- A less restrictive type of real-time system is a soft real-time system, where a critical real time-task gets priority over other tasks, and retains the priority until it completes it's task.

Outline



1 What is ES?

2 What is not ES?

3 Where you can find ES?

Introduction to ES's

Differences between ES and CS?



- small (pocket size);
- low power;
- usually application specific;
- low cost

- big (desktop size);
- not power friendly;
- general purpose;
- high cost

Introduction to ES's

Differences between ES and CS?



- small (pocket size);
- low power;
- usually application specific;
- low cost

- big (desktop size);
- not power friendly;
- general purpose;
- high cost

Introduction to ES's

Differences between ES and CS?



- small (pocket size);
- low power;
- usually application specific;
- low cost

- big (desktop size);
- not power friendly;
- general purpose;
- high cost

Introduction to ES's

Differences between ES and CS?



- small (pocket size);
- low power;
- usually application specific;
- low cost

- big (desktop size);
- not power friendly;
- general purpose;
- high cost

Introduction to ES's

Why not PLCs?



- PLCs are not that smart. The intelligence of smart devices resides in embedded systems.
- PLCs are not that autonomous – power independence of ES is its one of the essential features.
- PLCs are not that compact.
- PLCs are not that flexible – application areas are much broader.
- PLCs are not that cheap.

Introduction to ES's

Why not PLCs?



- PLCs are not that smart. The intelligence of smart devices resides in embedded systems.
- PLCs are not that autonomous – power independence of ES is its one of the essential features.
- PLCs are not that compact.
- PLCs are not that flexible – application areas are much broader.
- PLCs are not that cheap.

Introduction to ES's

Why not PLCs?



- PLCs are not that smart. The intelligence of smart devices resides in embedded systems.
- PLCs are not that autonomous – power independence of ES is its one of the essential features.
- PLCs are not that compact.
- PLCs are not that flexible – application areas are much broader.
- PLCs are not that cheap.

Introduction to ES's

Why not PLCs?



- PLCs are not that smart. The intelligence of smart devices resides in embedded systems.
- PLCs are not that autonomous – power independence of ES is its one of the essential features.
- PLCs are not that compact.
- PLCs are not that flexible – application areas are much broader.
- PLCs are not that cheap.

Introduction to ES's

Why not PLCs?



- PLCs are not that smart. The intelligence of smart devices resides in embedded systems.
- PLCs are not that autonomous – power independence of ES is its one of the essential features.
- PLCs are not that compact.
- PLCs are not that flexible – application areas are much broader.
- PLCs are not that cheap.

Outline



1 What is ES?

2 What is not ES?

3 Where you can find ES?

Introduction to ES's

Generalization of application areas



Embedded systems cover such a broad range of products that generalization is difficult. Some exemplary categories include:

- Aerospace and Defense Electronics
- Automotive
- Broadcast and Entertainment
- Consumer and Internet Appliances
- Data Communication
- Digital Imaging
- Industrial Measurement and Control
- Telecommunications
- Mobile Data Infrastructures etc.

Introduction to ES's

The world of ES's – Digital Consumer Markets



Cable or xDSL modems, Home Gateways, Home Media Servers

Geos X86 system, you can install easily Linux (Debian 5 for example) on CF reader and manage 2 ADSL lines.

Technical features:

- AMD Geode LX800 500MHz processor (x86 instruction set)
- DDR400 RAM SO-DIMM (512MB)
- 2 x ADSL2+ Ports
- 2 x 10/100 Ethernet Ports
- Mini PCI socket
- Compact Flash socket
- RS-232 Serial
- 1 x GPIO switch
- Open Source BIOS (Coreboot)



http://www.anteor.com/adsl_bonding.htm

Introduction to ES's

The world of ES's – Digital Consumer Markets



Cable, Satellite and Terrestrial STBs, HDTVs

The *Nuvola NP-1* is the world's first streaming media player that supports 4K UltraHD movies.

Technical features:

- nVidia Tegra 4 (Quad Core Cortex-A15)
- 72 GPU's
- 2GB DDR RAM
- 16GB Internal Flash Storage
- external connections for USB storage devices
- Ethernet, 2x2 Wi-Fi, Bluetooth, and Infrared
- 4K UltraHD (3840x2160), HD (1920x1080), SD (720x480)
- MP4, MKV, AVI, MOV, OOG (H.264)
- BMP, GIF, JPG, PNG



<http://www.nuvola4k.com/>

Introduction to ES's

The world of ES's – Digital Consumer Markets



Digital Cameras, Printers, Scanners

COOLPIX S800c – the imaging power of a Nikon, the connectivity of an Android™ smart device.

Technical features:

- 1GHz processor
- 512MB RAM
- 4GB Internal Flash Storage
- Image Resolution: from 4608 × 3456 (15.9 MP, 4:3), to 640 × 480 (0.3 MP, 4:3).
- Movie Resolution: 1920×1080, 1280×720, 640×480
- Movie framerate: 30 fps
- USB 2.0 High Speed, WiFi
- Mini (Type-C) HDMI



<http://www.nikonusa.com/en/Nikon-Products/Product/Compact-Digital-Cameras/26356/COOLPIX-S800c.html>

Introduction to ES's

The world of ES's – Digital Consumer Markets



Mobile phones, Personal Digital Assistants (Palmtops)

The *Motorola Droid Maxx* delivers astonishingly long battery life, a big, colorful screen, and a durable, attractive design.

Technical features:

- Android OS, v4.2.2 (Jelly Bean), upgradable to v4.4 (KitKat)
- Qualcomm Snapdragon S4Pro; CPU: Dual-core 1.7 GHz Krait; GPU: Adreno 320;
- 32GB Internal Flash Storage; 2GB RAM
- Accelerometer, gyro, proximity, compass, barometer
- Java MIDP emulator
- Stand-by time Up to 600 h
- Touch focus, geo-tagging, face detection
- 1080p@30fps



<http://www.engadget.com/products/motorola/droid/maxx/specs/>



Introduction to ES's

The world of ES's – Digital Consumer Markets



Web-Tablets, Screen phones

Nexus 7 Powerful, portable and made for what matters to you.

Technical features:

- Android™ 4.3
- Qualcomm® Snapdragon™ S4 Pro 8064 Quad-Core, 1.5 GHz
- 2GB RAM; 16/32GB
- 7" LED Backlight WUXGA (1920x1200) Screen
- 10 finger multi-touch support
- Accelerometer, gyro, gyroscope, ambient light sensor
- Stereo Speakers
- 10 hours; 15Wh Li-polymer Battery



http://www.asus.com/Tablets_Mobile/Nexus_7_2013/

Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Introduction to ES's

Common design requirements of embedded systems



- Processing power – make sure your system has enough strength
- Memory – make sure your system can remember everything
- Development cost – make sure you can afford our system
- Number of units – make sure you can produce your system
- Expected life time – make sure your system will last
- Reliability – make sure your system is trustworthy



Resources

If you want to gain some knowledge by your own...



Wikipedia – Embedded system

http://en.wikipedia.org/wiki/Embedded_system



Embedded System Market – Global Industry Analysis

<http://www.prnewswire.com/>



Intelligent Systems Transforming the Embedded Industry

<http://www.idc.com/getdoc.jsp?containerId=prUS24138513>



Questions ?

Rafal Kapela

rafal.kapela@put.poznan.pl