

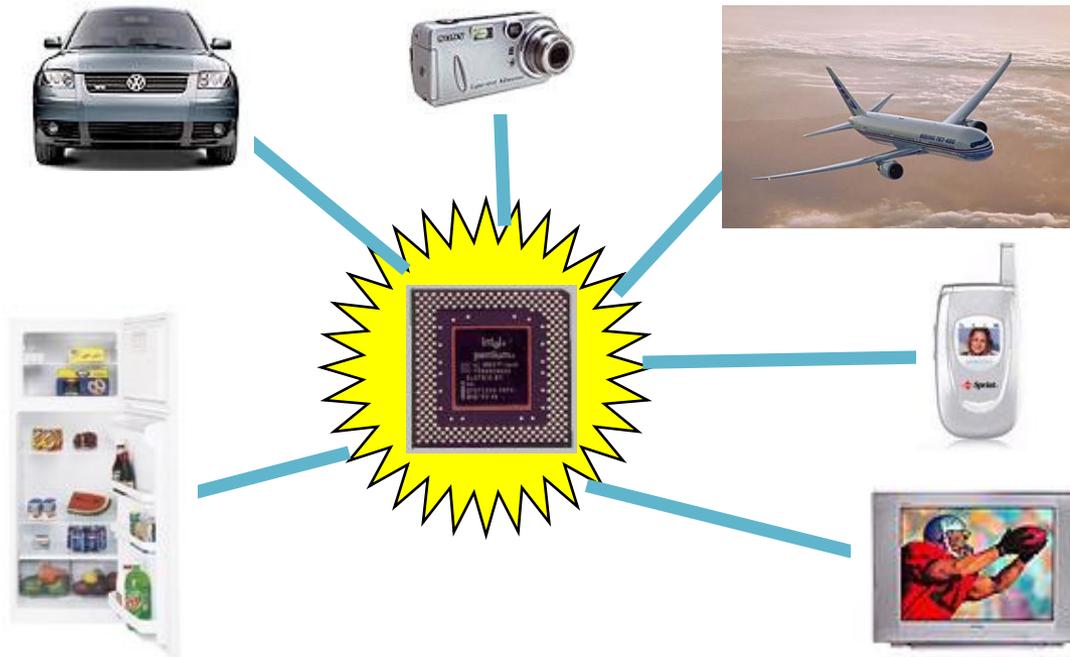
Push-Button Creation of an Optimized
Application Specific OS

SynthOS

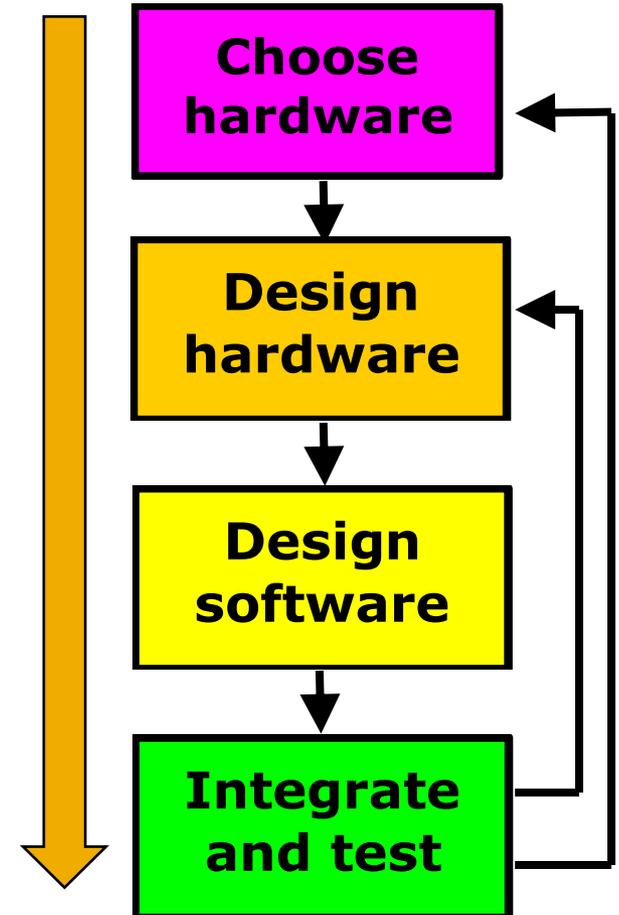
*Zeidman Technologies has created a
fundamentally new way to develop embedded
software and hardware for the Internet of Things.*



The Problem: Embedded System Development



Initial hardware choices determine
functionality and performance



SynthOS

The Problem: Embedded System Development

- Software is limited by hardware functionality
- Increasingly difficult design problem
- Requires specialized RTOS knowledge
- RTOS requires more hardware resources
- Difficult to optimize and debug
- Cannot test software until hardware is done

The Problem: The Internet Of Things

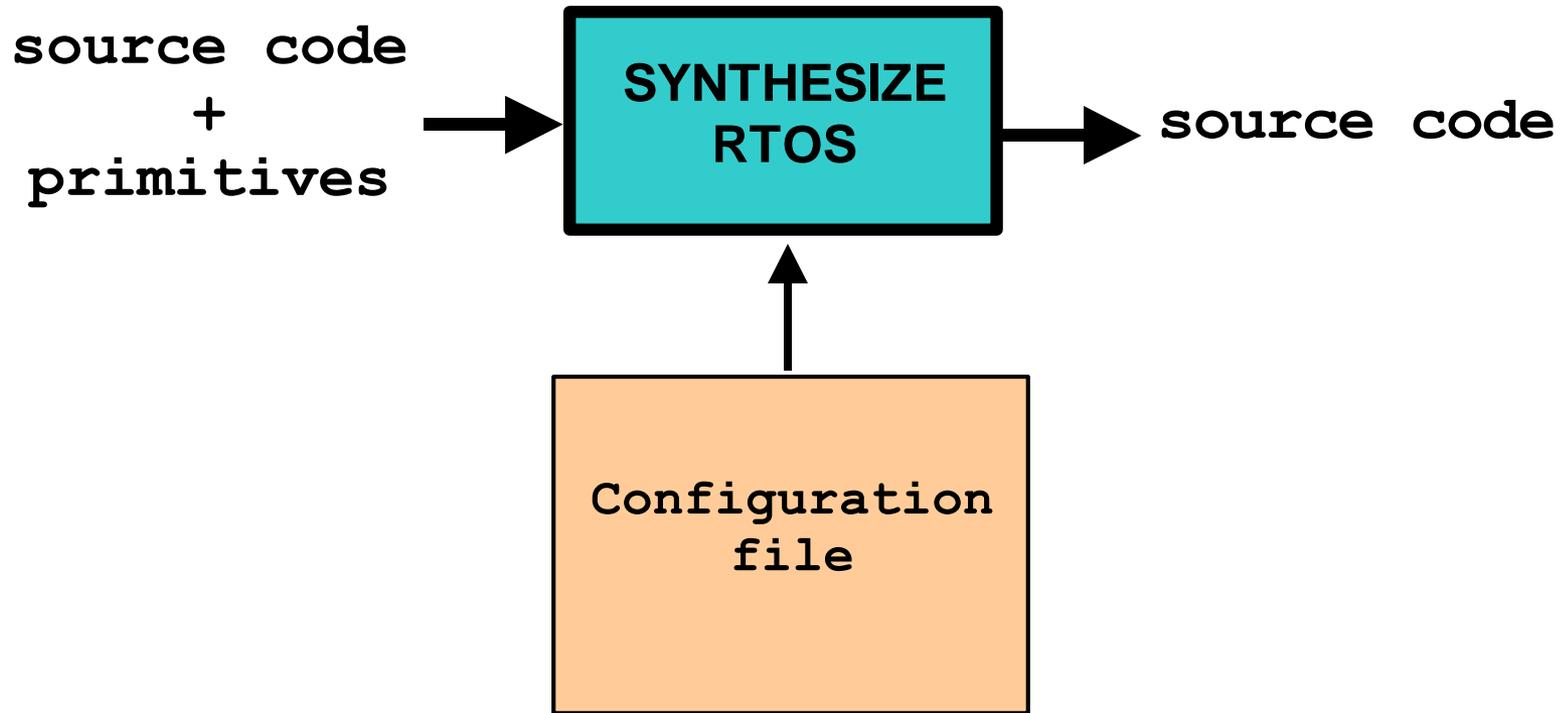
- Processing power within system limitations
- Power management
- Cost effectiveness
- Quality and reliability
- Security

What is SynthOS?

ASOS

- It is **NOT** an RTOS
- It is a **tool** that creates an RTOS (ASOS)
- *The new engine behind IoT software development for smaller, faster, and more efficient embedded systems.*
One that requires **no RTOS knowledge**
- One that is **optimized** for your project
- One that **secures**
- It is **NOT** an RTOS

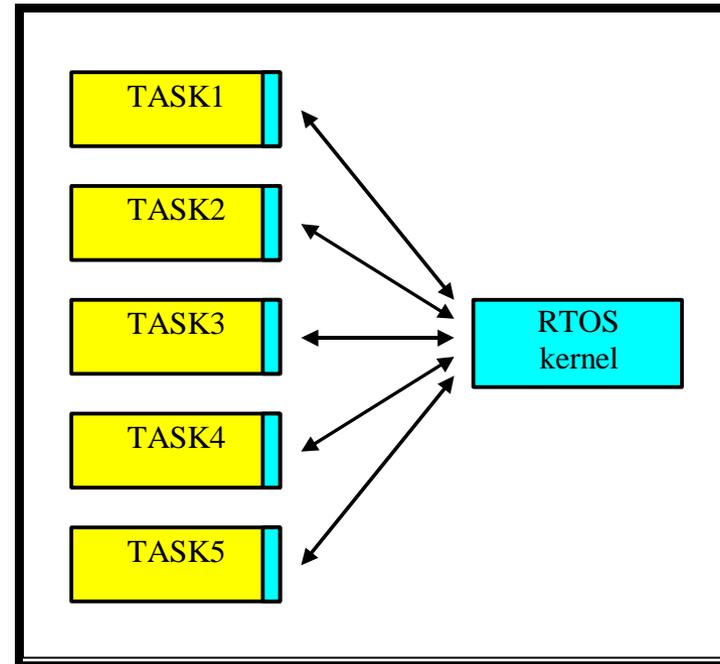
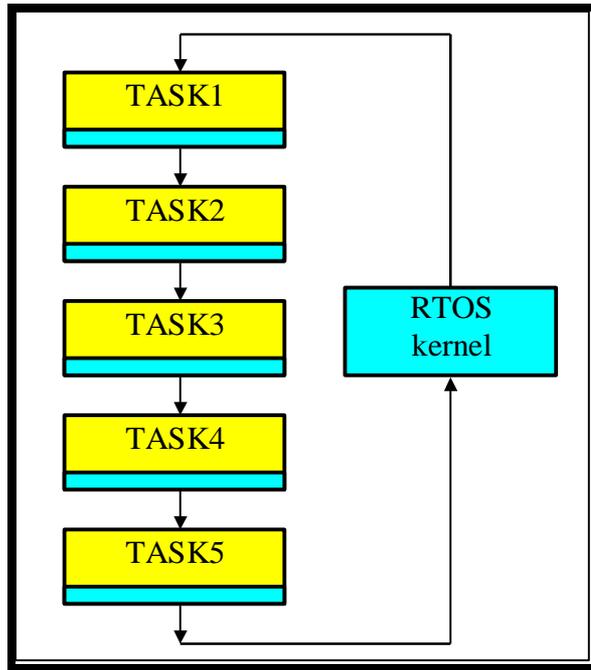
SynthOS



SynthOS Advantages

- Time to Market
- Security
- Cost of goods sold
- Development costs
- Power consumption
- Portability
- Development and Debugging
- Licensing

Custom RTOS (ASOS)



KEY:  Task management code
 User code

Fast Time to Market and Low Development Costs

- Inexpensive software tool
- Small learning curve
 - 19-page users guide
 - 5 primitives
- Push a button, get an optimized OS
- Standard ANSI C output
- Use your current C development environment
- No RTOS purchase
- No RTOS royalty

Low Cost of Goods Sold and Low Power Consumption

- Supports low cost microcontrollers and microprocessors that off-the-shelf RTOSes don't support
- Requires smaller memory than an off-the-shelf RTOS
- Simpler system design results in less expensive parts

Strong Security

- Every synthesized system is custom
- Hackers can't get a system to find vulnerabilities
- Vulnerabilities in one synthesized system are not found in other synthesized systems
- Can't add malware after synthesis

Portability

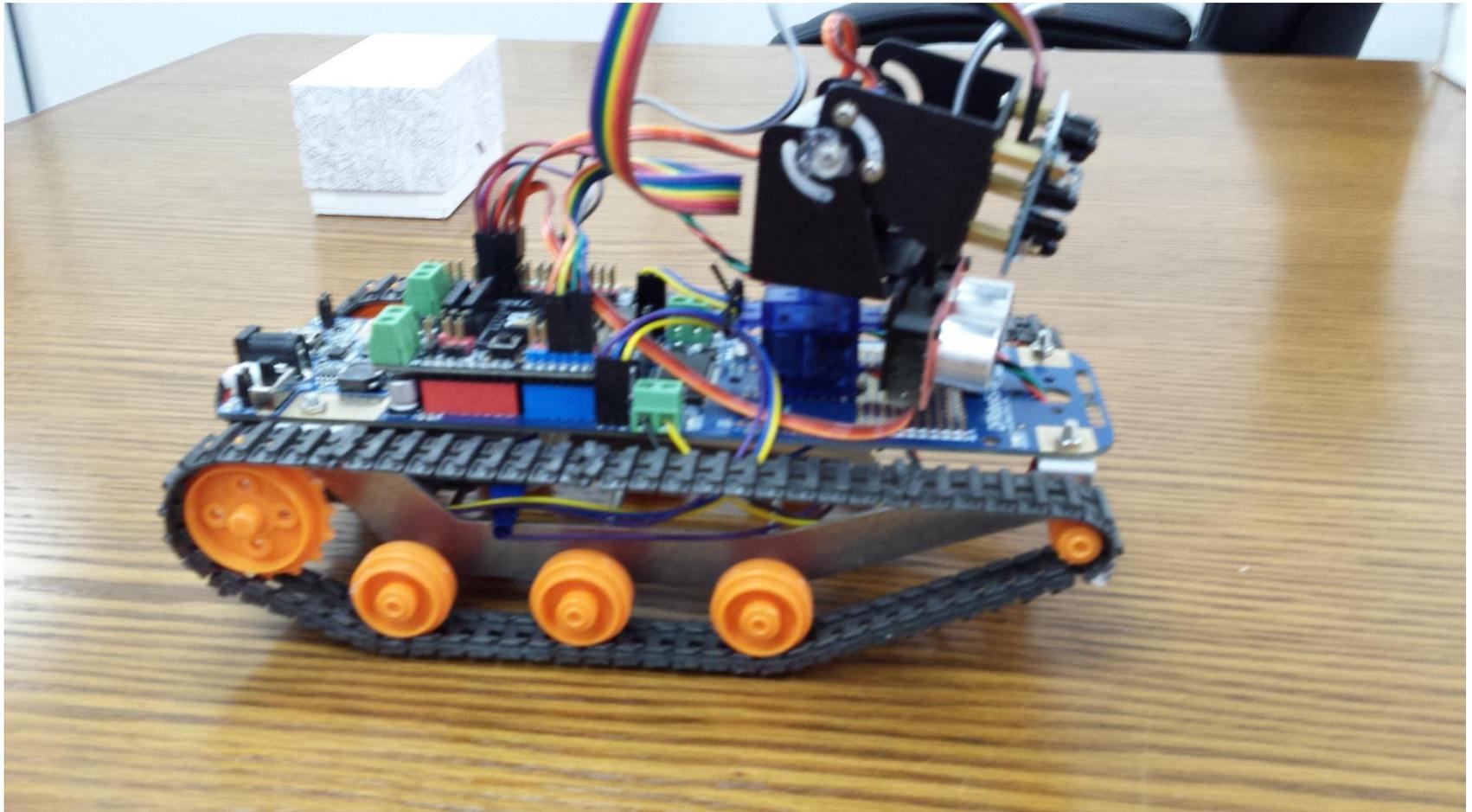
- Works on any processor with a C compiler (i.e., any processor)
- Supports every new processor when it is available
- No need to develop new systems for new processors
- No need to find new RTOS to take advantage of new processors

Case Study:

Heterogeneous Multiprocessing

- Xilinx Vertex-II Pro FPGA
- Xilinx MicroBlaze 32-bit soft processor
- PowerPC 32-bit hard processor
- **MicroBlaze kernel size < 0.9 Kbytes**
- **PowerPC kernel size ~ 2.3 Kbytes**
- **Development time ~ 3 days**

Arduino Robot Project



Contact Info

Jacob Harel

VP Business Development

Jacob@zeidman.biz

www.zeidman.biz

Try it out completely free at

www.SynthOSonline.com