

Push-Button Creation of an Optimized
Application Specific OS

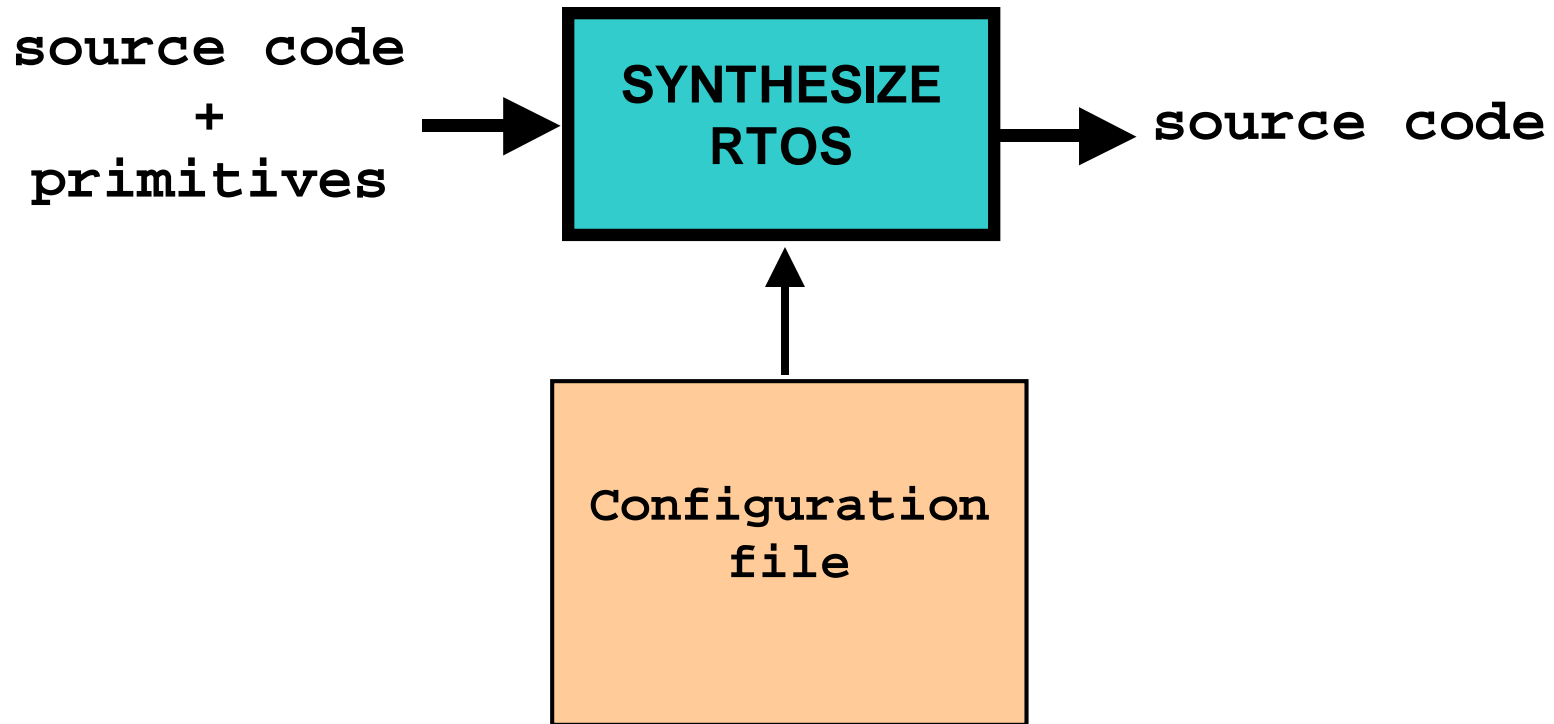
SynthOS



What is SynthOS?

- It is **NOT** an RTOS
- It is a **tool** that creates an RTOS (ASOS)
- One that requires **no RTOS knowledge**
- One that is **optimized** for your project
- One that is **secure**
- It is **NOT** an RTOS

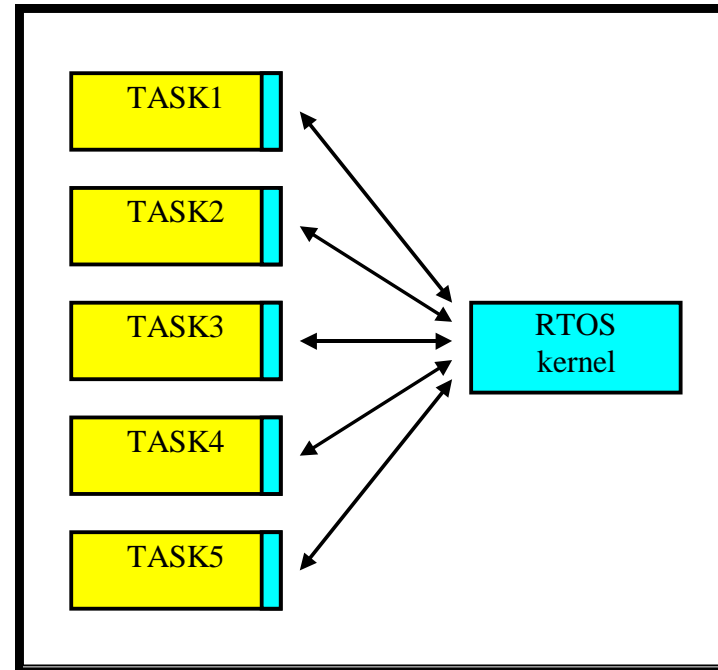
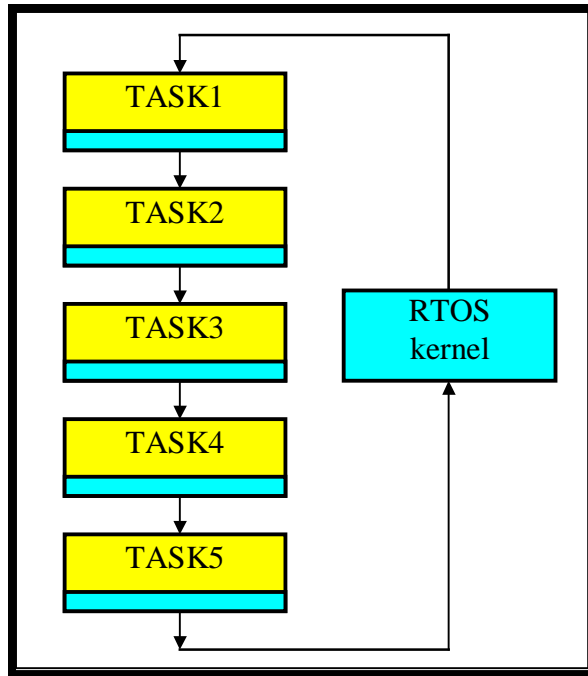
SynthOS





SynthOS Advantages

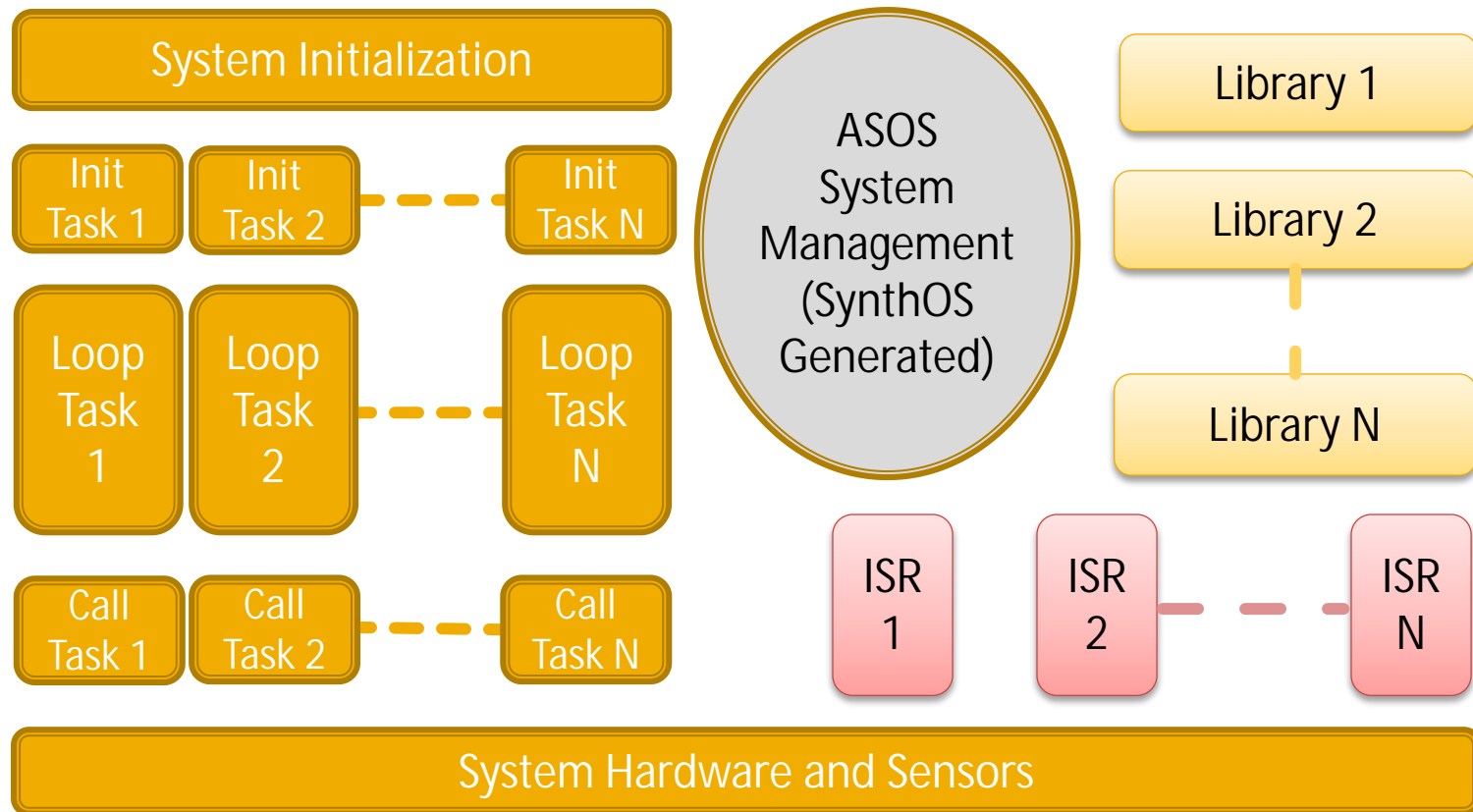
- | Time to Market
- | Security
- | Cost of goods sold
- | Development costs
- | Power consumption
- | Portability
- | Development and debugging
- | Licensing

Custom RTOS (ASOS)



KEY:  Task management code
 User code

System Architecture



Fast Time to Market

- | Small learning curve
- | 19-page users guide
- | 5 primitives
- | Push a button, get an optimized OS

Low Cost of Goods Sold

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

Low Development Costs

- Inexpensive software tool
- Reduced development time
- No RTOS purchase
- No RTOS royalty

Case Study

Flat Panel Display Firmware Development					
SW developer salary + benefits	\$150,000				
				Using SynthOS	
	Actual Schedule			Predicted Schedule	
Task	Man-weeks	Cost		Man-weeks	Cost
Writing firmware specification	6	\$17,308		4	\$11,538.46
Code cyclic executive	6	\$17,308		1	\$2,884.62
Code startup task	8	\$23,077		8	\$23,076.92
Simple interface driver tasks	6	\$17,308		5.4	\$15,576.92
Complex interface driver tasks	24	\$69,231		21.6	\$62,307.69
Simple video controller tasks	8	\$23,077		7.2	\$20,769.23
OSD controller tasks	8	\$23,077		7.2	\$20,769.23
Advanced video controller tasks	12	\$34,615		10.8	\$31,153.85
Internal tables for controlling video specs	2	\$5,769		2	\$5,769.23
NVRAM control task	2	\$5,769		1.8	\$5,192.31
Additional debug and integration	16.4	\$47,308		13.8	\$39,807.69
Code optimization	6	\$17,308		4	\$11,538.46
Documentation	4	\$11,538		3	\$8,653.85
TOTAL	108.4	\$312,692		89.8	\$259,038
Cost savings	\$53,654				
Time-to-Market savings (weeks)	9.3				

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

Low Power Consumption

- i Supports low power microcontrollers and microprocessors that off-the-shelf RTOSes don't support
- i Requires smaller memory than off-the-shelf RTOSes

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

Development and Debugging

- | Standard ANSI C output
- | Use your current C development environment (IDE)
- | Use your current C debugging tools
- | Command line integrates with other tools

Licensing

- | No license restrictions on synthesized code
- | All synthesized code is completely yours

Case Study: Multitasking Web Server

- | Altera NIOS 32-bit soft processor
- | Cyclone EP1C20 FPGA
- | Took the Altera web server code
- | Added functionality
- | Turned single task into multitask system
- | **Kernel size: ~ 3K bytes**

Case Study:

Heterogeneous Multiprocessing

- | Xilinx Virtex-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

Case Study: Lego Mindstorm Robot

- | Total RAM = 26K
- | Lego RTOS = 22K
- | BrickOS = 11K
- | **SynthOS = 2K**

Arduino Robot Project

- | Self directed robot
- | Arduino Uno (Atmel ATmega328p)
 - § 32K flash
 - § 2K RAM
 - § 16 MHz clock
- | Resources used

SynthOS

8.6K flash

0.5K RAM

FreeRTOS

12.8K flash

1.25K RAM

Arduino Robot Tasks

- | robot: Main *loop* task
- | left_motor: Left motor control *loop* task
- | right_motor: Right motor control *loop* task
- | drive_pan: *Call* task to manage scanning and movements of the ultrasound sensor
- | ultrasonic_measure: *Call* task to control the ultrasonic sensor distance calculation

Arduino Robot Project



Costs

- i Online
 - § Priority support, upgradesfree
\$2,000 / year
- i Single in-house license
 - § Priority support, upgrades\$20,000 / year
- i In-house site license
 - § Priority support, upgradesNegotiable

Contact Info

Jacob Harel
VP Product Management
jacob@zeidman.biz
www.zeidman.biz

Try it out completely free at
www.SynthOSonline.com