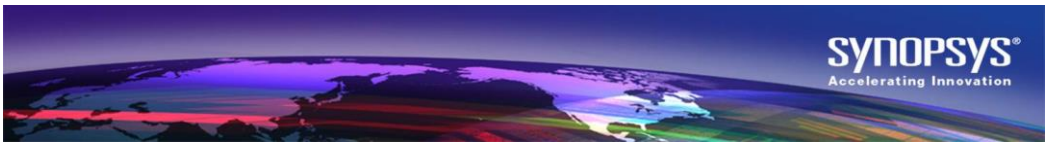


Smart Sensor Systems

The Next Multiprocessor Challenge

MPSoC2013, Otsu, Japan

Dr. Yankin Tanurhan
VP Engineering, Solutions Group
Synopsys, Inc.

16/7/2013

Agenda

Introduction

Smart Sensor Systems

Components

Performance

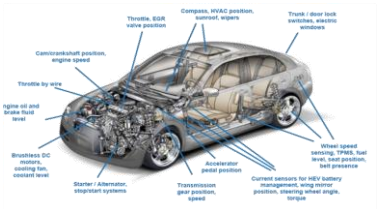
Conclusions

Introduction

Analog sensors meet digital and software

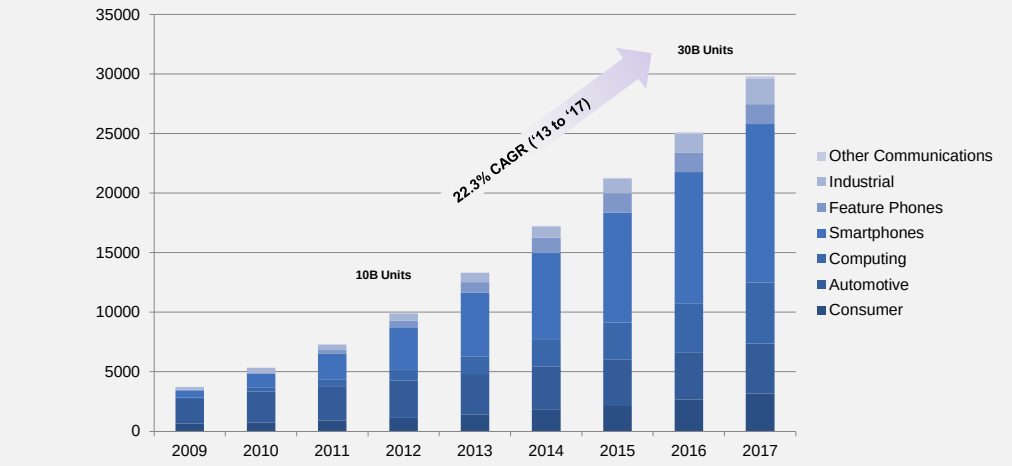
Introduction

- Smart Sensor: adding a (digital) CPU core to an analog sensor
- Digital processing
  - Digital filtering
  - Calibration, linearization
  - Digital interfaces
  - Sensor fusion



Significant Growth in Sensors

Sensor Units to Grow to 30 Billion Units in 2017



Source: Semico Research, 2013

SYNOPSYS Accelerating Innovation

Introduction

Topics addressed in this presentation

- Smart sensor systems
  - Analog sensor becomes intelligent sensor
  - Multitude of sensors (analog and digital) combined for more precise data
  - Sensor fusion host off-loading
- Components that are part of a smart sensor system
- Performance requirements
  - Area, power, footprint

SYNOPSYS Accelerating Innovation

Smart Sensor Systems

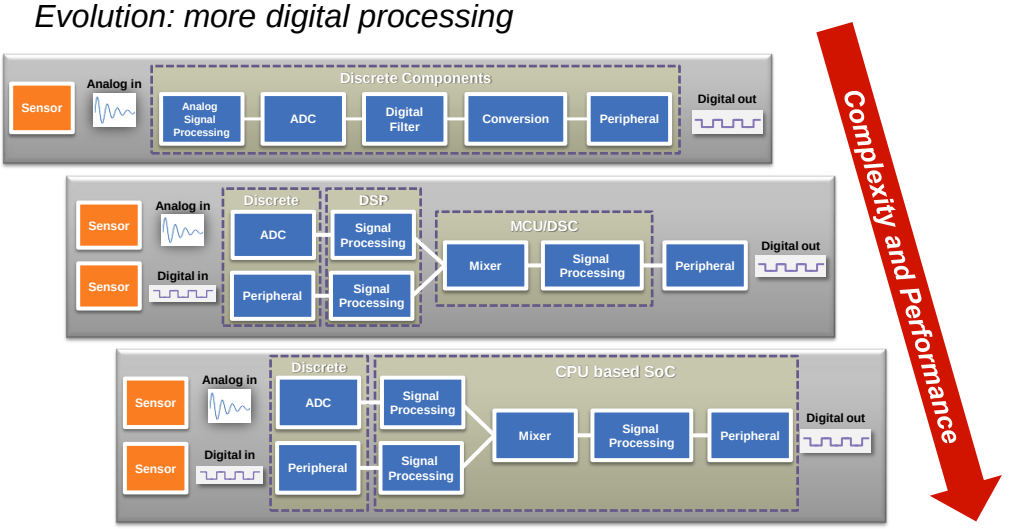
Digital processing

Scalable solution

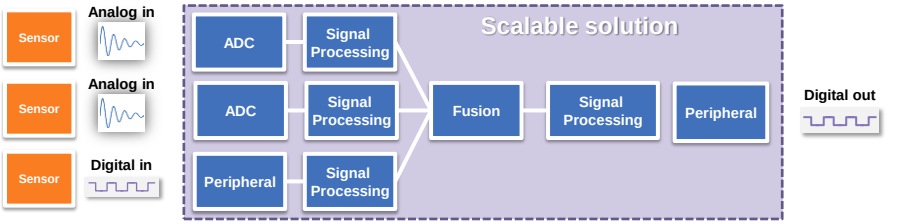
Sensor system partitioning

Smart Sensor System

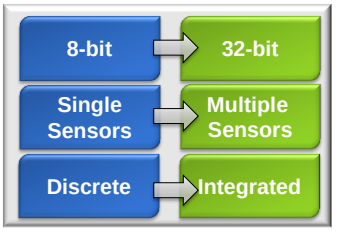
Evolution: more digital processing



Sensor Systems Getting More Complex

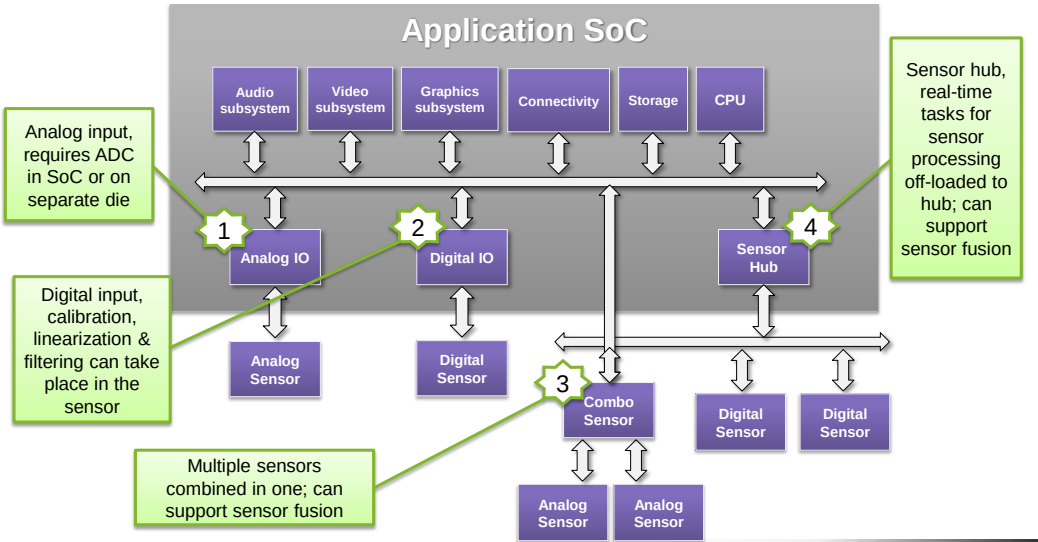


- Increased processing drives higher performance processor
- Sensor fusion with multiple analog & digital sensors requires higher level of integration
- Transitioning from discrete ICs to integrated sensor control within larger, more complex SoC

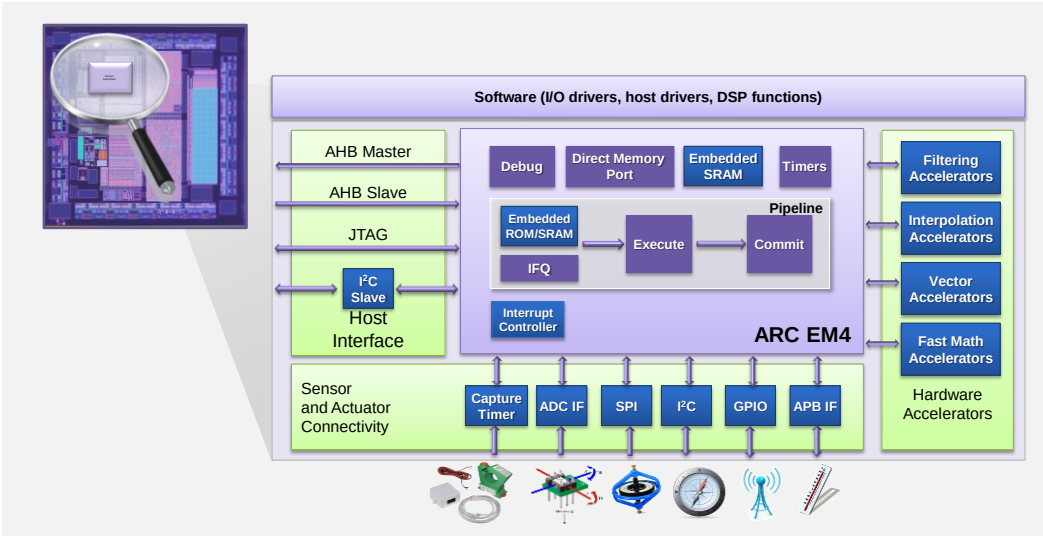


Smart Sensor System

Multiple ways to get sensor data to the application processor



Integrated Sensor IP Subsystem



Components

Closely coupled memories

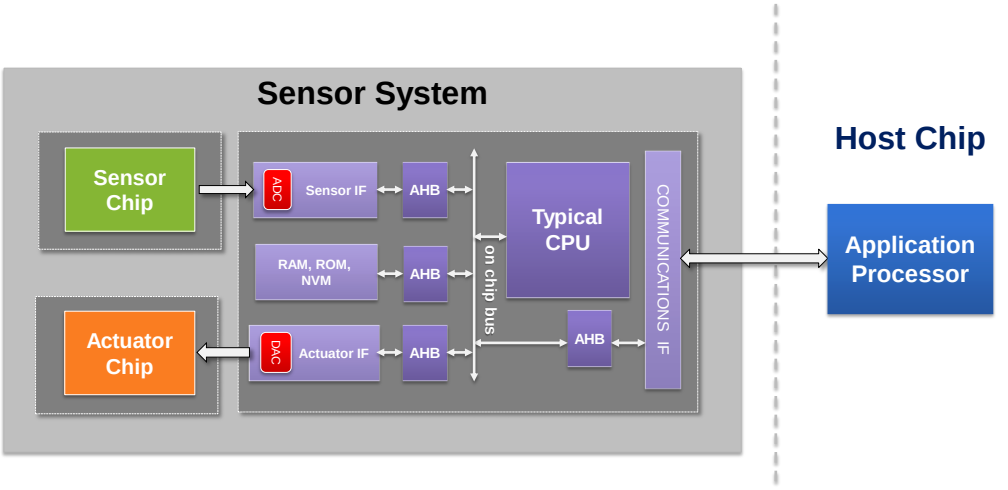
Tightly coupled IO functions

DSP extensions/Hardware accelerators

Software view

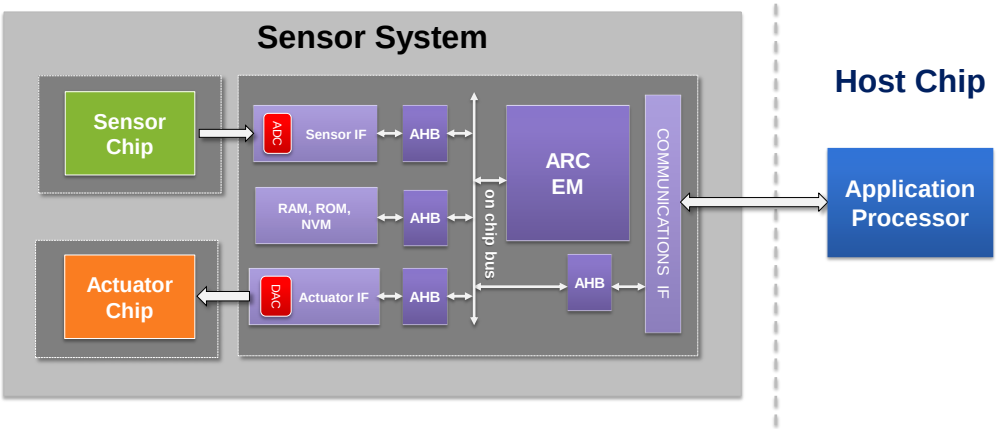
Typical CPU based Sensor System

bus-based design



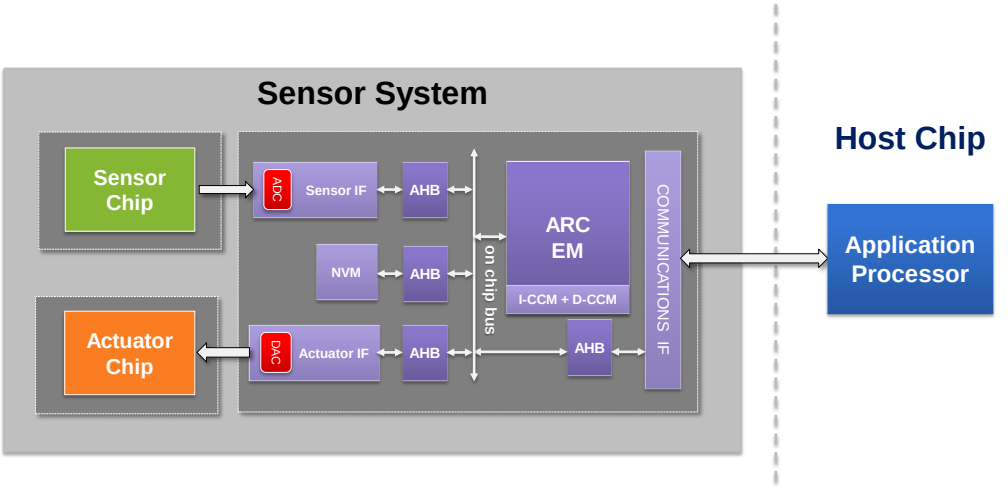
Typical CPU based Sensor System

bus-based design using ARC EM processor



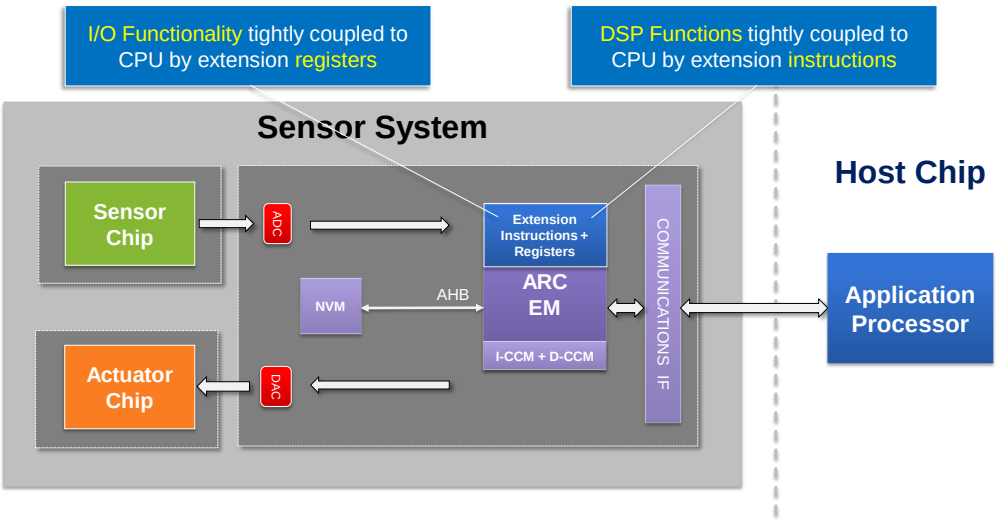
Typical CPU based Sensor System

optimization: use closely-coupled memories



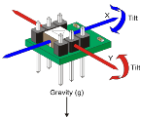
Optimized CPU based Sensor System

ARC EM Processor with IO and DSP extensions



IO functionality

Overview of sensor interfaces



Function	Range	Interface
Motion	Accelerometers	Analog/PWM/SPI/I²C
	Gyroscopes	Analog/SPI/I²C
Magnetic and Hall	Angle	Analog/SPI/I²C
	Rotational speed	Analog[current/voltage]
	Hall effect	Analog[current]/SPC/PWM/SENT
Proximity	Proximity	I²C/SPI
Optical	Ambient	Analog[current]/I²C/PWM
	Distance	Analog[voltage]
	CCD image	Serial Digital + GPIO
	CMOS image	Parallel Digital + GPIO/SPI/I²C
Pressure	Compensated	Analog[voltage]
	Uncompensated	Analog[voltage]
	Integrated	Analog[voltage]
Temperature	Analog output	Analog[voltage]/Switch[on/off/interrupt]
	Digital output	I²C/SMBus/SPI/1-wire/PWM/switch[on/off]/QSPI/SST
	Silicon	Analog[resistance]

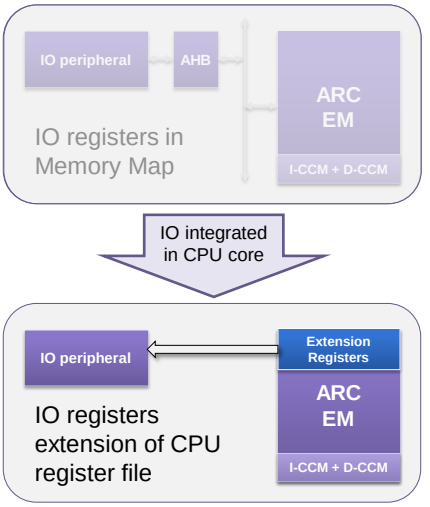
Sensor IF
ADC if
I2C
SPI
PWM
…

Host IF
I2C
SPI
UART
(on-chip) bus

Source: Silica, Sparkfun

IO functionality

Tightly integrated IO



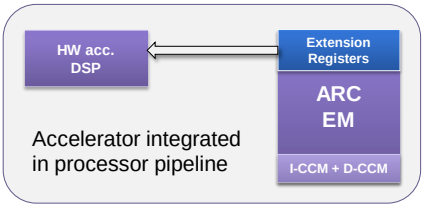
- "Tightly integrated"
  - Start from existing bus based IO peripheral
  - Eliminate interface to on-chip bus
  - Replace *load/store* instructions to IO peripheral by *register move* instructions

HW support for DSP Functions

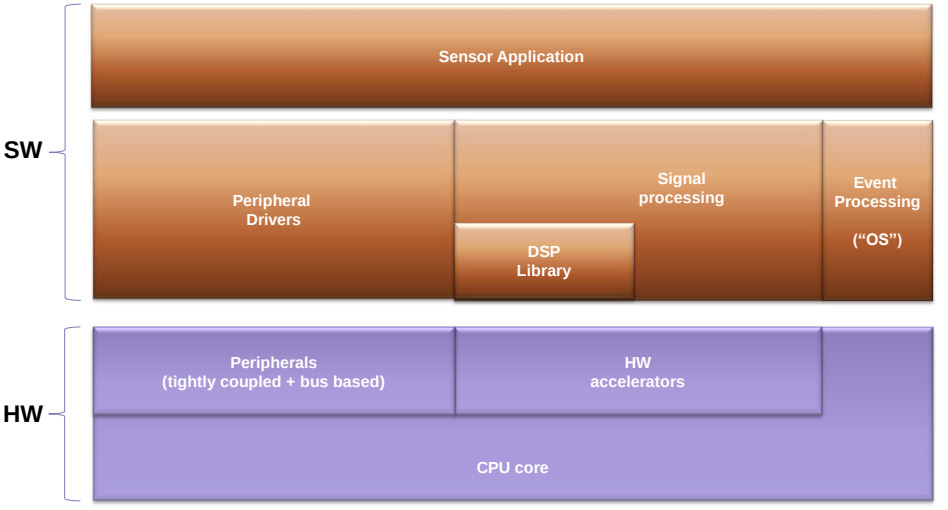
Accelerate application code

- Reduce memory footprint
  - HW accelerator instruction(s) replaces multiple SW statements or functions
- Reduces cycle count
  - more performance,
  - or less power
- Some silicon area increase
- Selection of HW accelerators tailored to (sensor) application

- DSP functions
  - Math
  - Complex Math
  - Filtering
    - FIR, IIR, correlation, …
  - Matrix/vector
  - Interpolation



Software



Performance

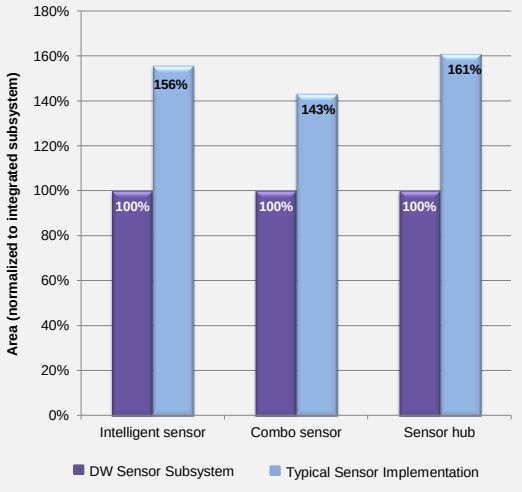
Area benefits

Cycle count and memory footprint benefits

System power benefits

Embedded Non-Volatile Memories

Significant Area & Power Benefits



- Typical Sensor Implementation
  - 40- 60% Larger
- Typical Sensor Implementation
  - Smaller & significantly more power efficient
- Synopsys' Integrated Sensor Subsystem
  - Smaller and significantly more power efficient

Functions Implemented in HW or SW for Area & Power Trade-offs

Software Only Functions
Cos, Sin Sqrt
Conjugate
DOT
Magnitude
MaqSquared
Multiply
Convolute
Correlate
LMS, FIR, IIR
Add, multiply, transpose
Linear, Bilinear

Hardware Functions
Multiply/Accumulate (MAC)
Accumulate/Multiply (ACM)
Sine/Cosine (SIN/COS)
Square Root (SQRT)
Absolute Value (ABS)
Add (ADD)
Subtract (SUB)
Multiply (MULT)
Negate (NEGATE)
Shift (SHIFT)
Scale (SCALE)

FIR Filtering
6x reduction in cycle count

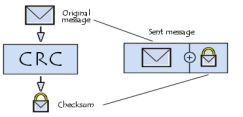
Square Root
14x reduction in cycle count

CRC Check
25x reduction in cycle count

Fewer Cycles Equals Lower Power, Higher Performance

HW accelerator

Cyclic Redundancy Check



Host communication check
CRC polygon :$1+x^1+x^2+x^4+x^5+x^7+x^8+x^{10}+x^{11}+x^{12}+x^{16}+x^{22}+x^{23}+x^{26}+x^{32}$;

Metrics @ 180nm		Software Cycle count Optimized	CRC accelerator (hardware)
Footprint ROM	Bytes	1420	12
Cycles	#	76	3
Accelerator	Gates	0	475
Memory footprint	Gates	3140	25
Total area	Gates	3140	500

- ✓ Cycle count reduced by factor 25
- ✓ Area reduces (!) by 2640 ND2 equivalent gates (85%)
- ✓ CRC Code footprint reduces from 104 instructions to 3 instructions only

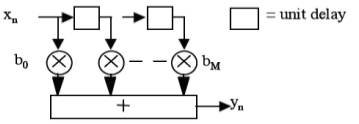
HW accelerator
Square root

Y = √ X

Metrics @ 180nm 48bit input 24bit output		Software Cycle count optimized	Hardware accelerator (cycle count opt)	Hardware accelerator (pipelined opt)	Hardware accelerator (area opt)
Footprint ROM	Bytes	272	4	4	4
Cycles[a]	#	353*n	n	6+n	24*n
Accelerator	Gates	0	3740	5490	1075
Memory footprint	Gates	600	10	10	10
Total area	Gates	600	3750[b]	5500	1085

- ✓ Cycle count reduced by factor 14
- – Area increases only 485 ND2 equivalent gates
- ✓ SQRT Code footprint reduces from 136 instructions to 1 instructions only

HW accelerator
Finite Impulse Response

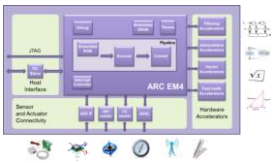


N-taps asymmetric FIR filter:
FIR[i] = (SUM k in [0..N): C[k] * X[i-k])

Metrics @ 180nm # taps =30		Software Cycle count Optimized	MAC accelerator (hardware)
Footprint ROM	Bytes	254	146
Cycles	#	2673	430
Accelerator	Gates	0	1520
Memory footprint	Gates	560	300
Area	Gates	560	1820

- ✓ Cycle count reduces by factor 6
- – Area increases only 1260 ND2 equivalent gates
- ✓ FIR Code footprint reduces from 86 instructions to 58 instructions only

Power Benefits
Sensor fusion application



- 9-D sensor fusion
  - Combo sensor based design
  - Design optimized using:
    - tightly coupled IO functions
    - addition of fusion specific HW accelerators

Metrics @ 90nm CPU + IO + Accelerators		Bus-based design	Optimized design
Cycles	[#]	100%	10%
Dynamic Power	[uW/MHz]	100%	107%
Standard cell Area	Gates	100%	121%

Overall power reduction of 9.3x

Conclusions

Components

Performance

measurements

Conclusions

Components for Smart Sensor System

- Identified Smart Sensor System variants
  - Intelligent sensor and Combo sensor (focus of presentation)
  - Sensor hub
- Addressed components that are part of a Smart Sensor System
  - Optimized CPU based solution
  - IO peripheral (tightly coupled) + driver SW
  - HW accelerators for DSP functions (tightly coupled) + DSP SW
- Mix of tightly coupled IO and bus-bused IO peripherals possible (esp. when area is not high priority)

Conclusions

Performance

- Performance improvements by
  - Tightly coupled IO peripherals
  - Tightly coupled HW accelerators
- Footprint
  - Silicon area, Software memory footprint (reducing ROM size)
- Power
  - Lower CPU core frequency
- Predictability
  - Direct access to tightly coupled IO and HW accelerators

Conclusions

Measurements

- All measurement and studies are based on Synopsys product portfolio
  - ARC EM core
  - Tool suite for customizing ARC EM core
    - development of tightly coupled EIA extensions/instructions
  - Complete SW development tool suite
  - EM Starter Kit development system

ARC® Synopsys


SYNOPSYS®
Accelerating Innovation