The Abdus Salam
International Centre for Theoretical Physics

United Nations
Educational, Scientific and
Cultural Organization

IAEA
International Atomic Energy Agency

2065-19

Advanced Training Course on FPGA Design and VHDL for Hardware Simulation and Synthesis

26 October - 20 November, 2009

FPGA Design & VHDL
Fundamentals of FPGAs

Nizar Abdallah

ACTEL Corp. 2061 Stierlin Court Mountain View
CA 94043-4655
U.S.A.

FPGA Design & VHDL

Nizar Abdallah
nizar@ieee.org

October 2009

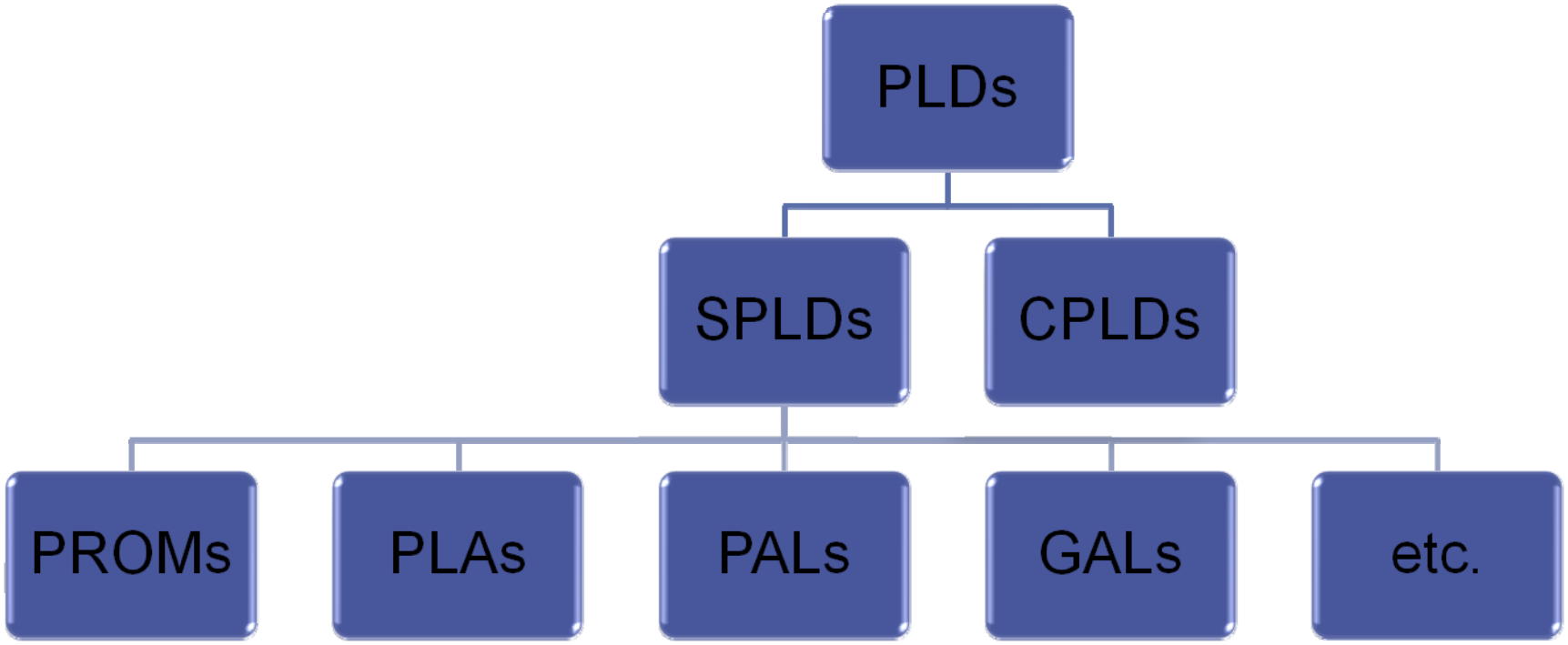
Fundamentals of FPGAs

Agenda

- FPGA Fundamentals
- Mixed-Signal FPGAs
- Actel Fusion Architecture
- FPGA Design Considerations
- Trends
- Choosing an FPGA
- Development Tools
- Summary

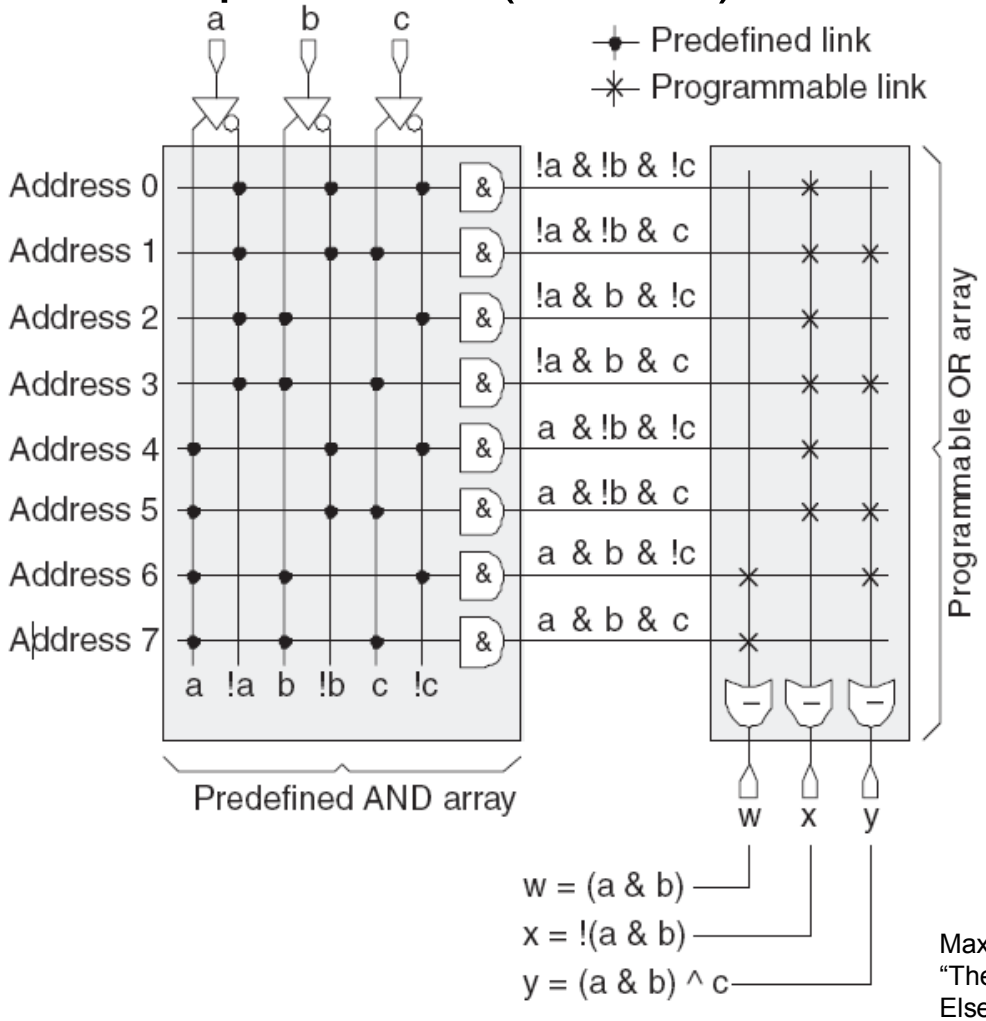
FPGA Fundamentals: A Brief History

- Before FPGAs



FPGA Fundamentals: A Brief History

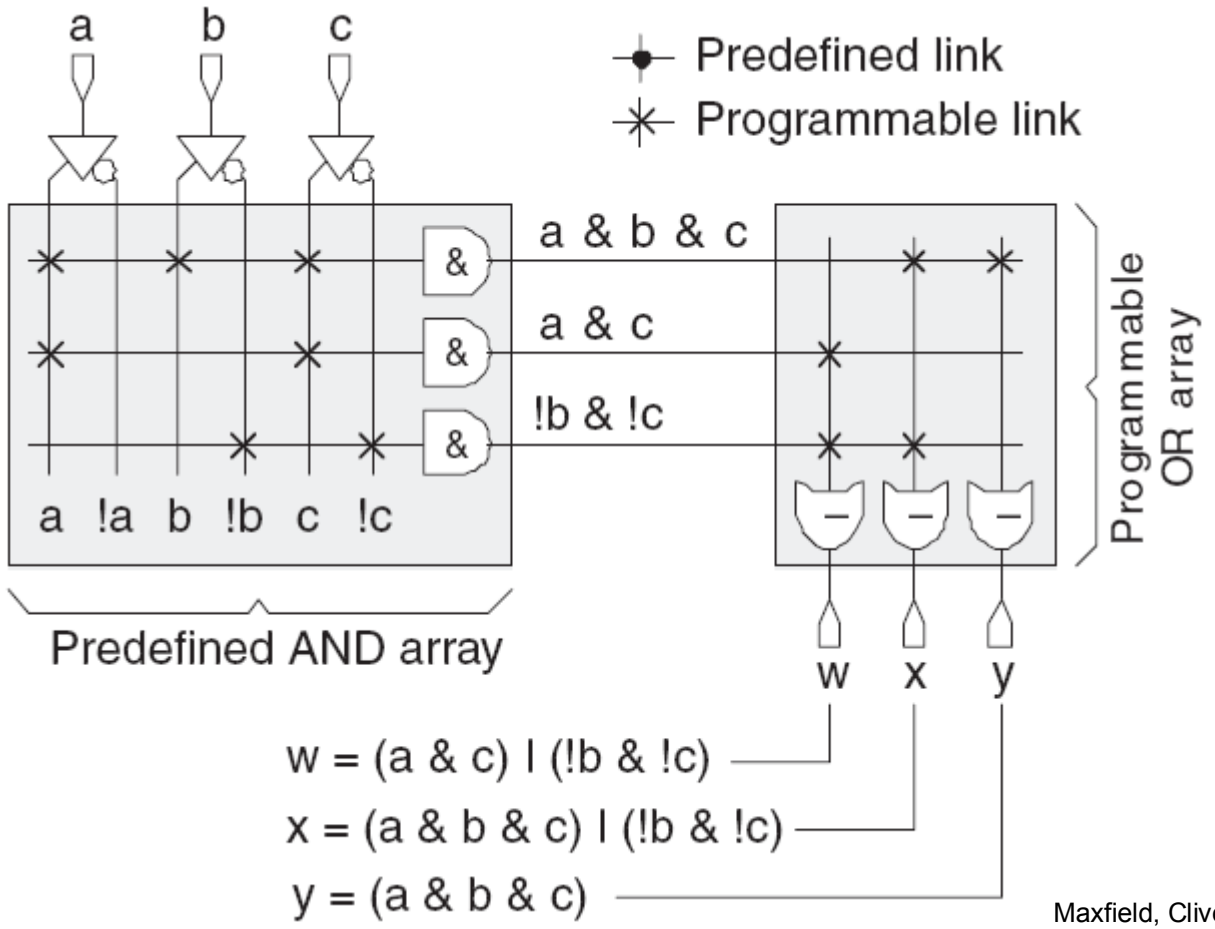
- Before FPGAs: Simple PLDs (SPLDs)–PROMs



Maxfield, Clive
"The Design Warrior's Guide to FPGAs"
Elsevier

FPGA Fundamentals: A Brief History

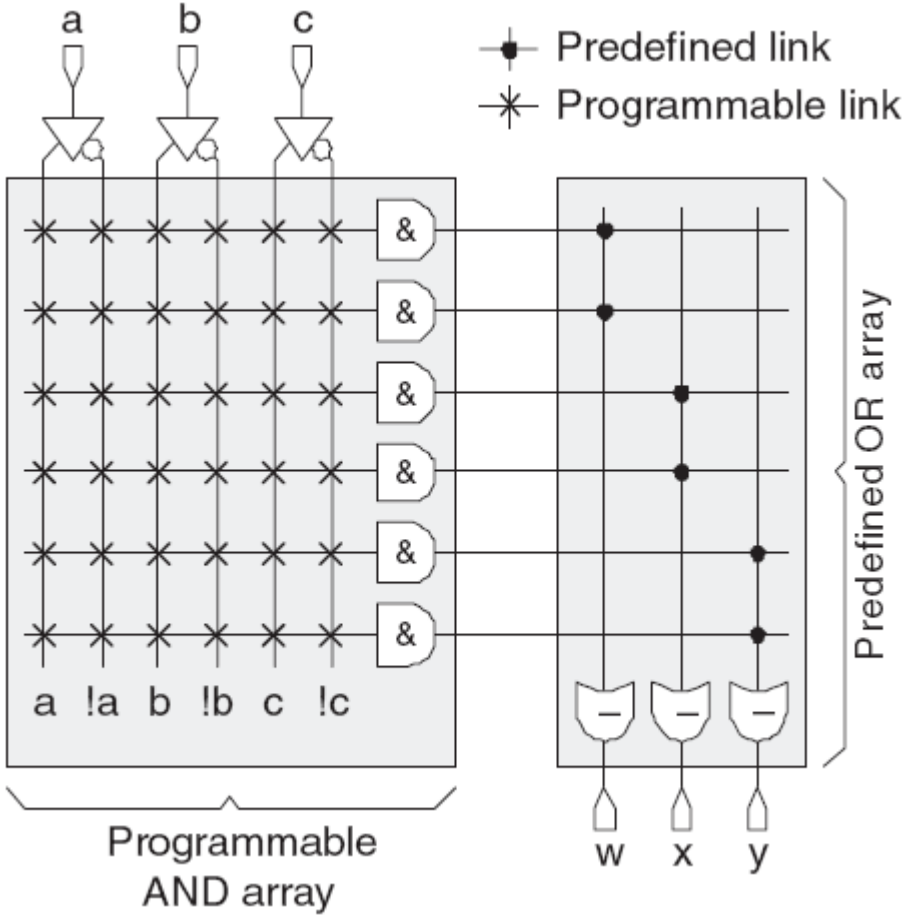
- Before FPGAs: Simple PLDs (SPLDs)–PLAs



Maxfield, Clive
"The Design Warrior's Guide to FPGAs"
Elsevier

FPGA Fundamentals: A Brief History

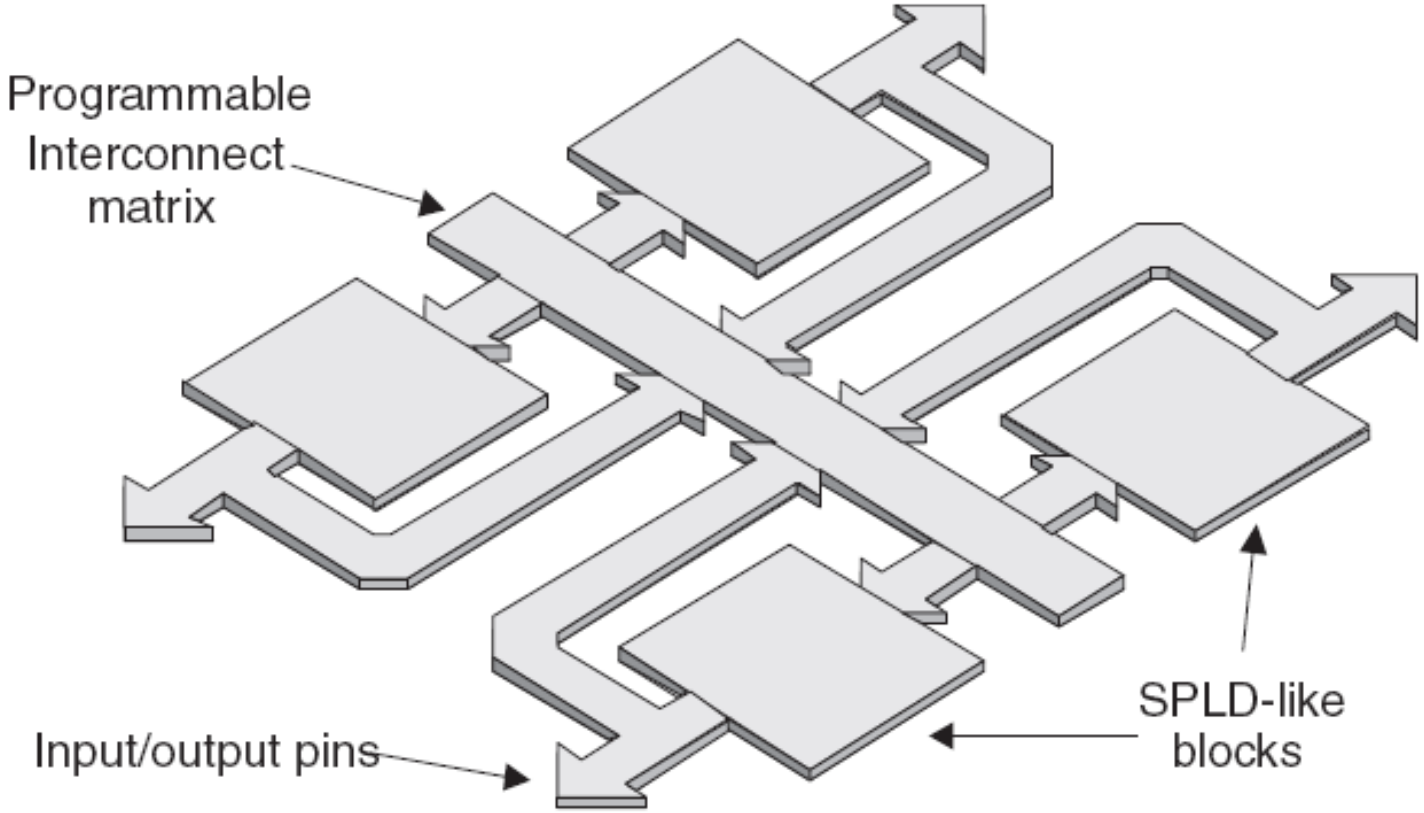
- Before FPGAs: Simple PLDs (SPLDs)–PALs



Maxfield, Clive
"The Design Warrior's Guide to FPGAs"
Elsevier

FPGA Fundamentals: A Brief History

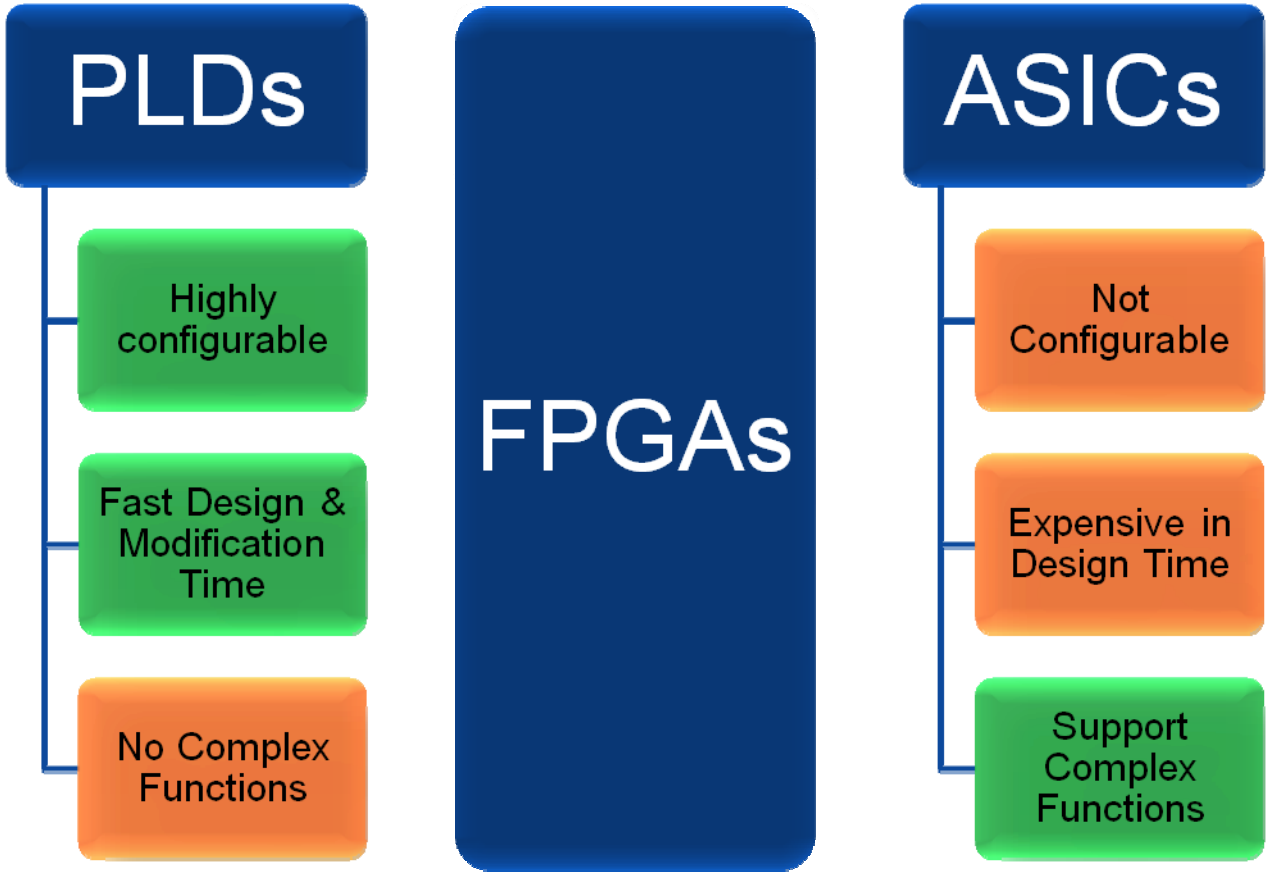
- Before FPGAs: Complex PLDs (CPLDs)



Maxfield, Clive
"The Design Warrior's Guide to FPGAs"
Elsevier

FPGA Fundamentals: A Brief History

- The GAP



FPGA Fundamentals: Definition

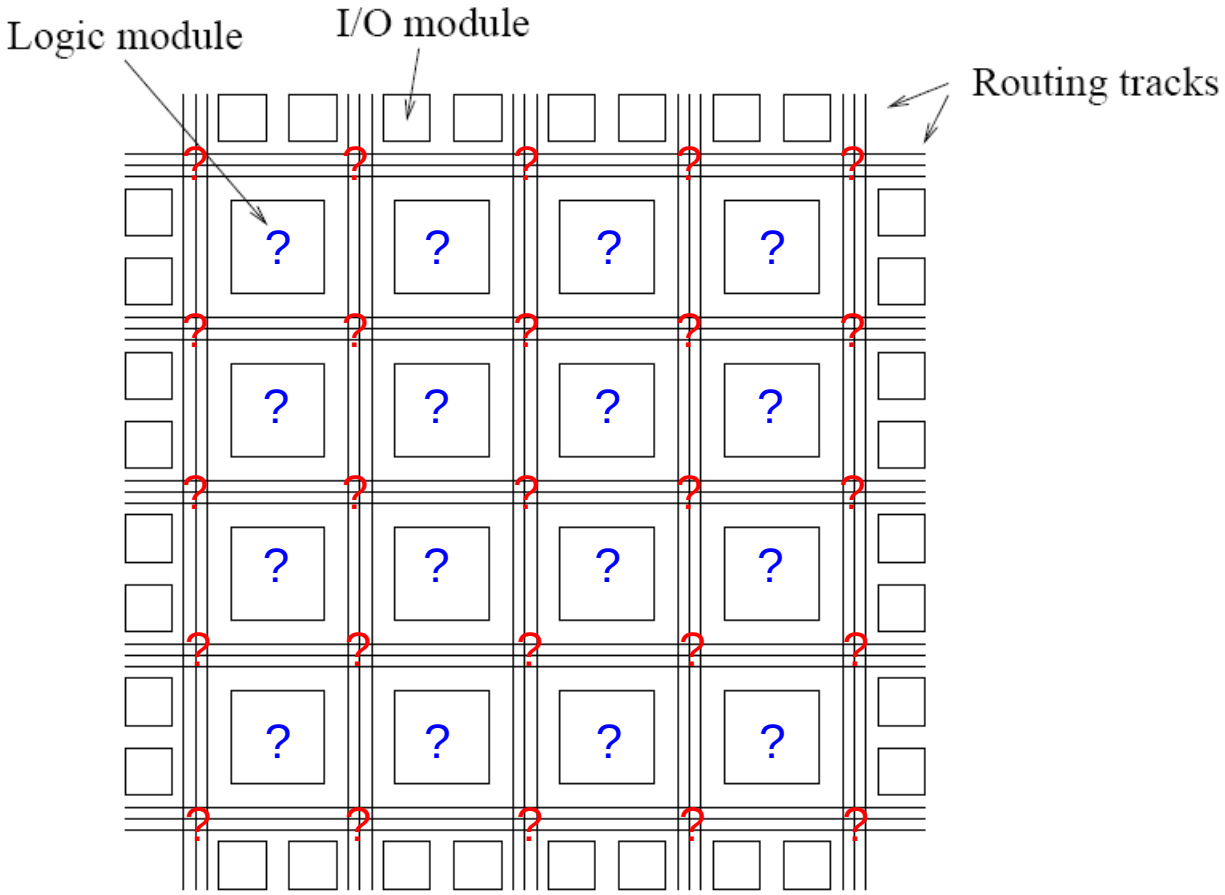
Field Programmable Gate Array

A large number of logic gates in an IC array that can be connected (configured) electrically

- **The Four Components of FPGAs**
  - **The Configuration Element**
  - **The Logic Module**
  - **The Memory**
  - **Control Circuits/Special Features**

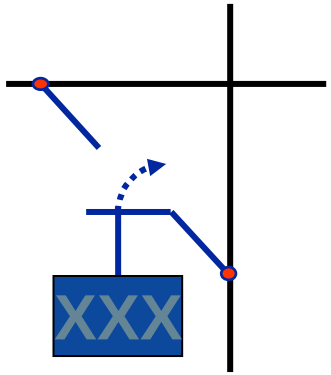
FPGA Fundamentals: Basic Architecture

- Generic FPGA Architecture

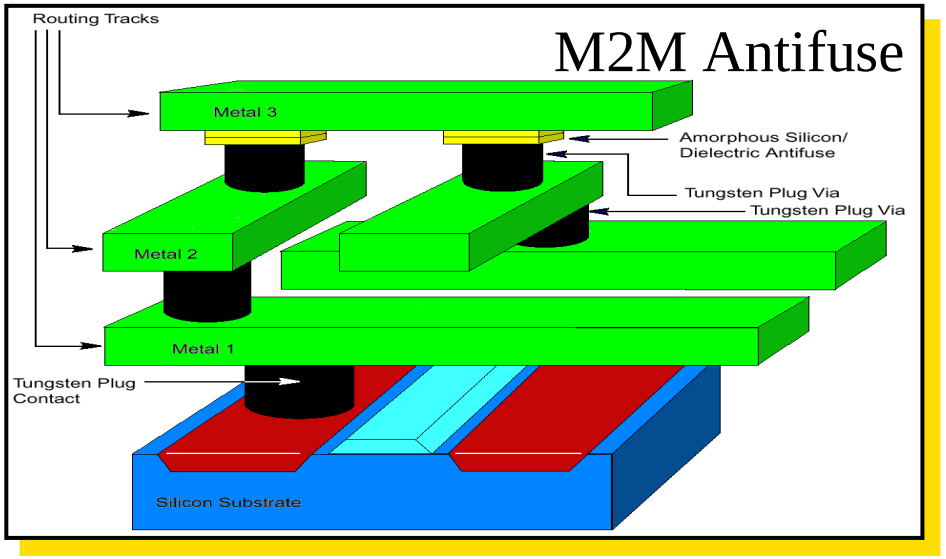


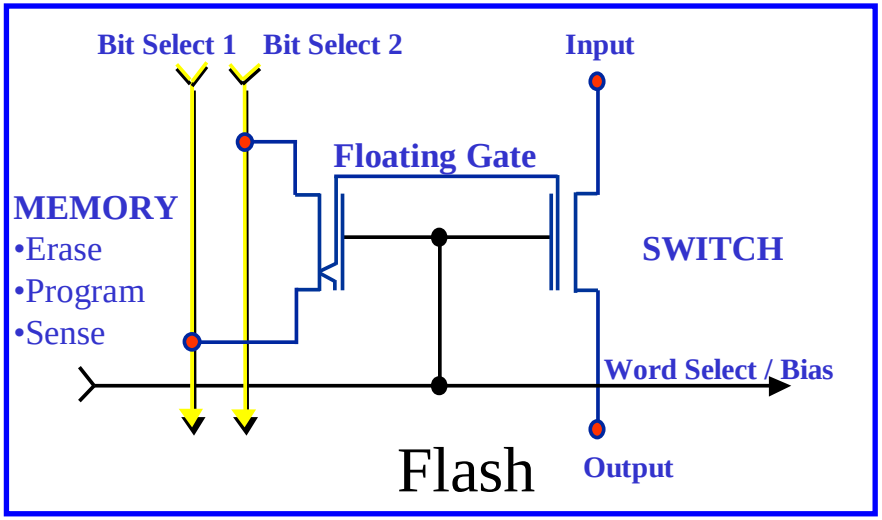
FPGA Fundamentals: Routing Technologies

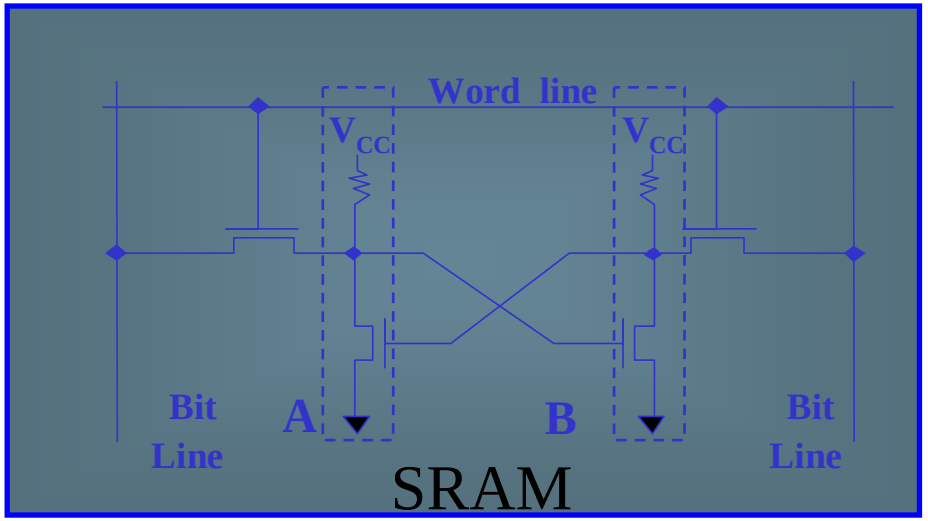
- The Interconnect Switch



FPGA Fundamentals: Routing Technologies

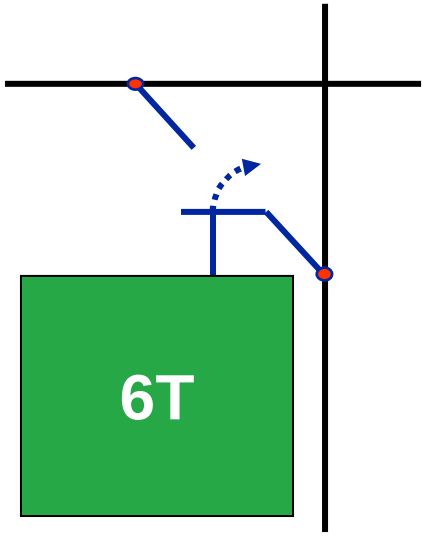






FPGA Fundamentals: Routing Technologies

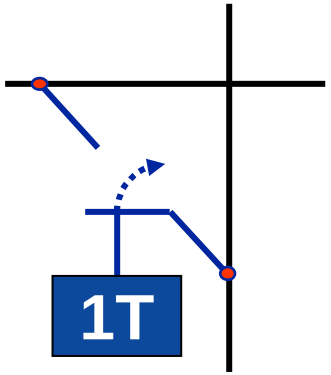
SRAM



Reprogrammable

Large Switch
expensive wires
Low Logic Utilization
typ 60%

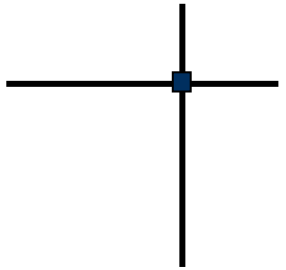
Flash



Best of Both Worlds
Reprogrammable & Nonvolatile

Small Switch
cheap wires
High Logic Utilization
typ >85%

Anti-fuse

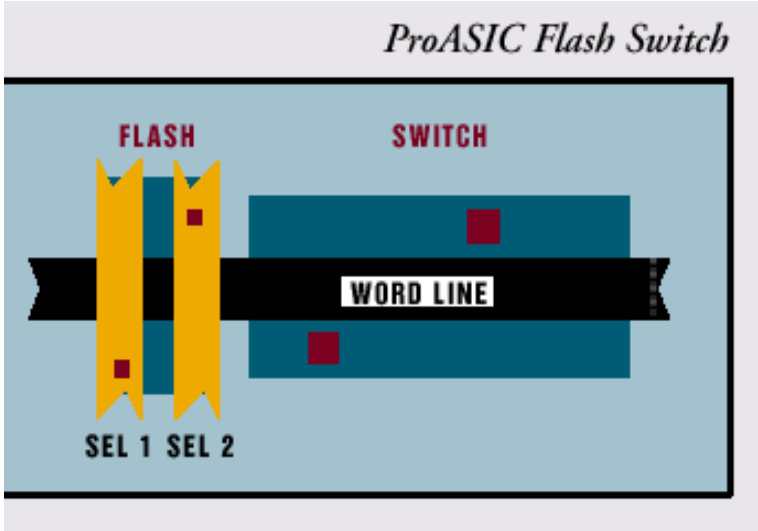


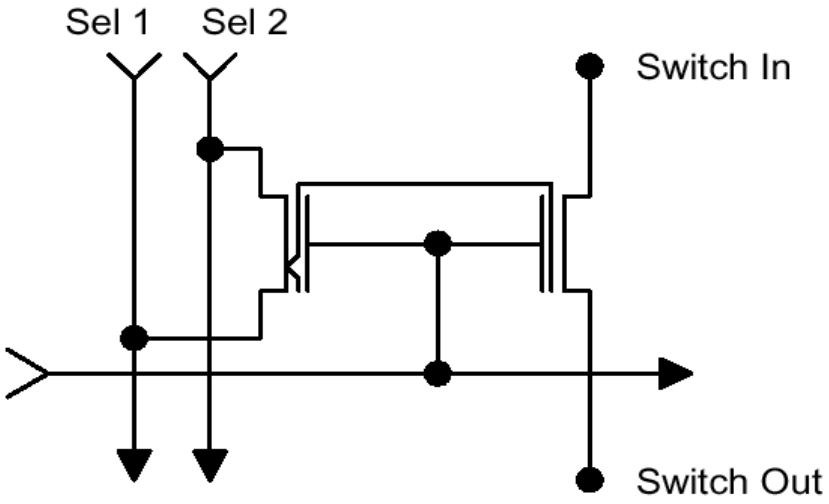
Nonvolatile

Smallest Switch
cheapest wires
Highest Logic Utilization
typ >90%

FPGA Fundamentals: Routing Technologies

ProASIC, ProASIC^{Plus}, ProASIC3 Routing Switch





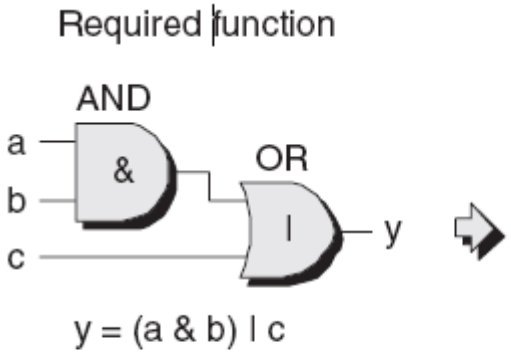
FPGA Fundamentals: Routing Technologies

- Programming Technologies

Feature	SRAM	Anti-fuse	Flash
Technology node	State of the art	Behind by 1-2 generations	Behind by 1-2 generations
Reprogrammable	Yes	No	Yes
Preserves configuration when off	No	Yes	Yes
Requires external configuration file	Yes	No	No
Instantly on	No	Yes	Yes
IP security	Acceptable	Excellent	Excellent
Power consumption	Medium	Low	Low

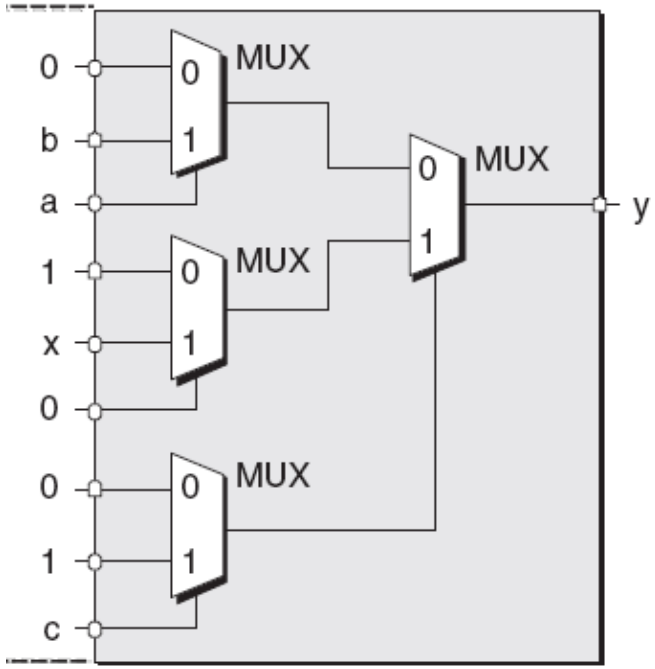
FPGA Fundamentals: Logic Elements

- LUT based vs. MUX based



Truth table

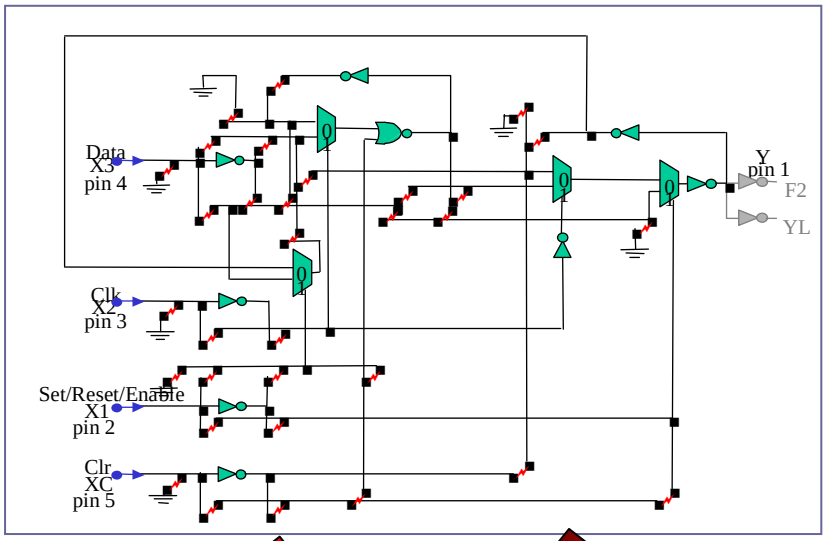
a	b	c	y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1



Maxfield, Clive
"The Design Warrior's Guide to FPGAs"
Elsevier

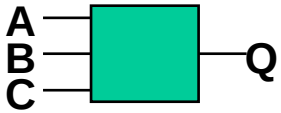
FPGA Fundamentals: Logic Elements

- Actel's Flash MUX based Logic Module



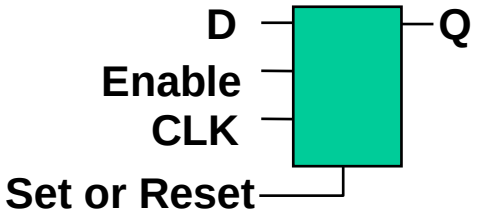
Combinatorial

Any 3 Input Function (3 input LUT Equivalent)

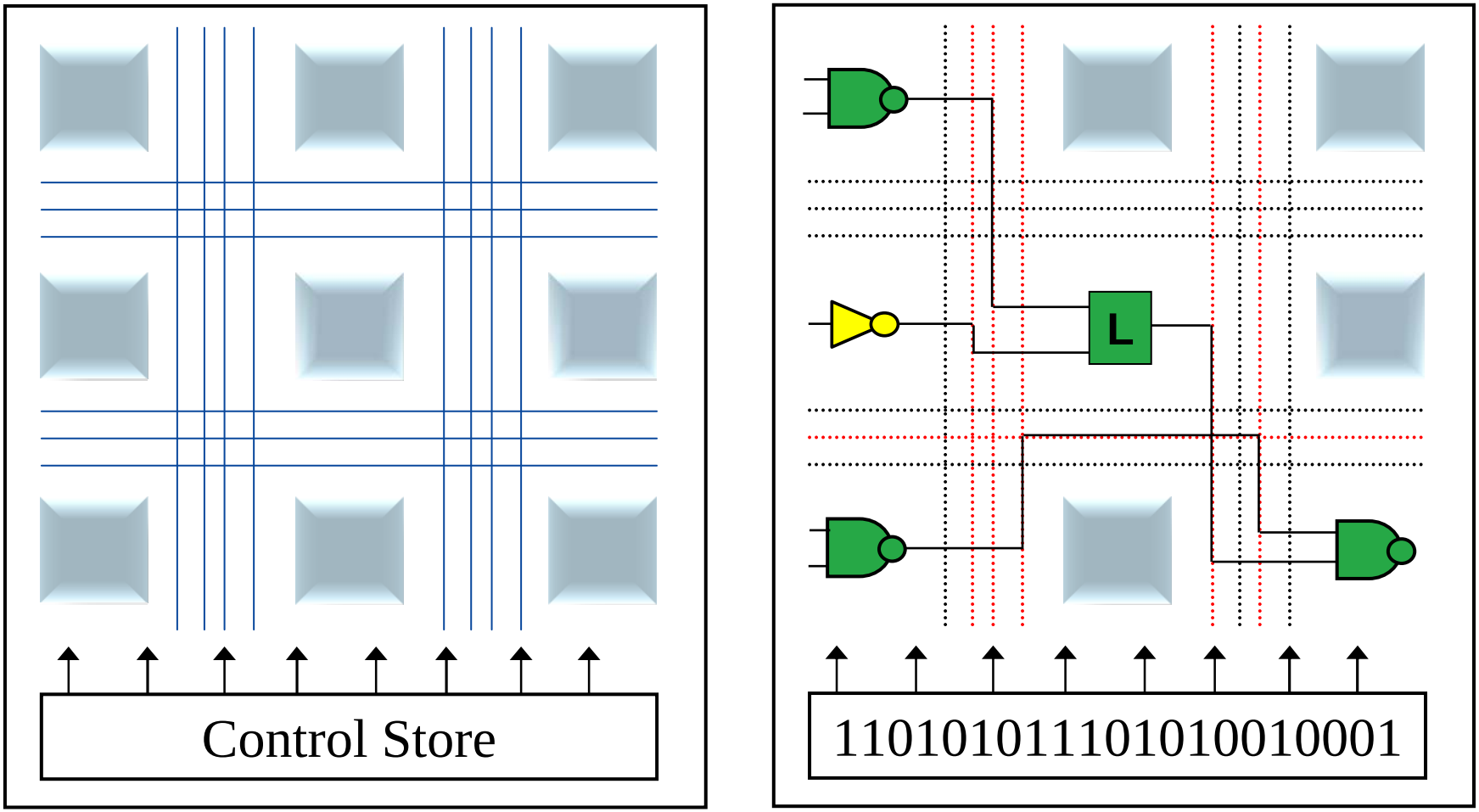


Sequential

D Flip-Flop With Enable and Set or Reset

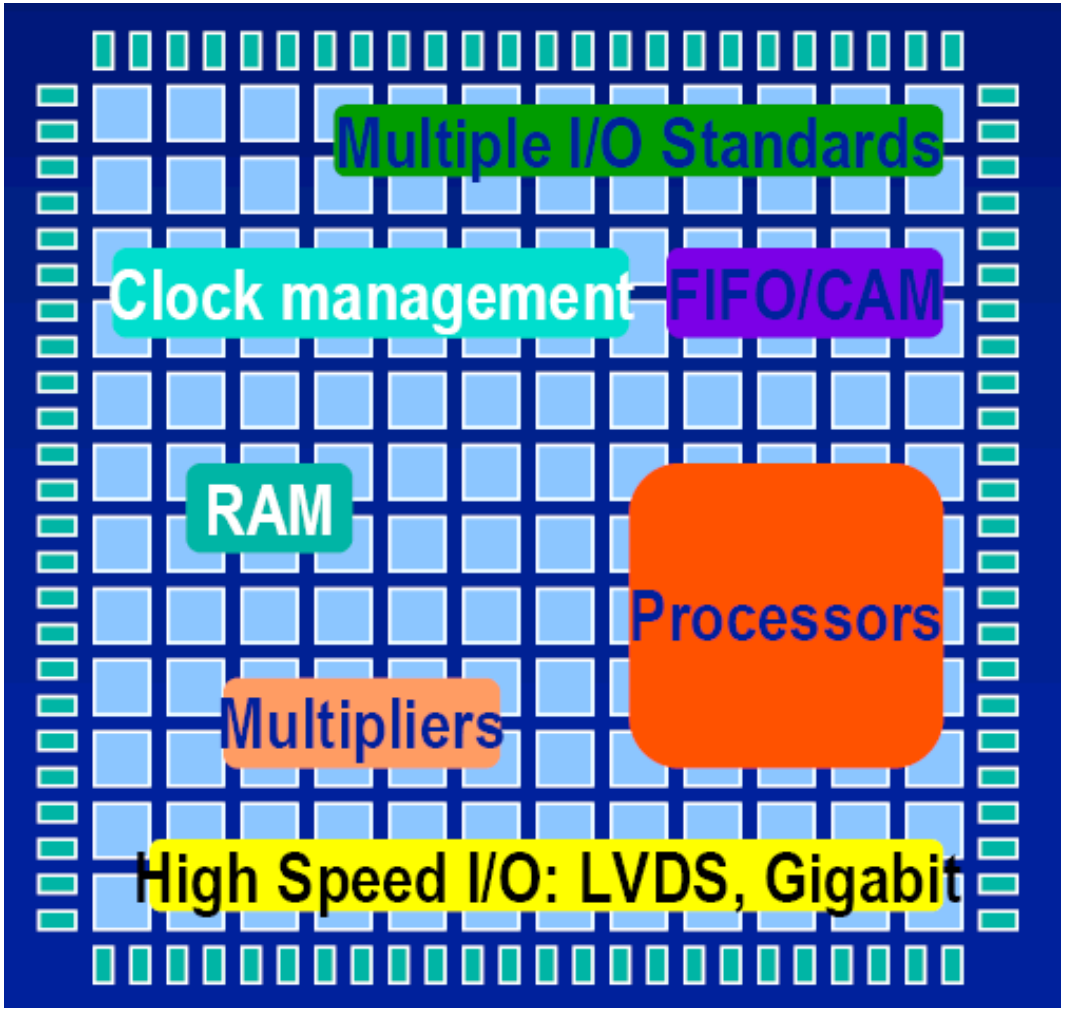


FPGA Fundamentals: An Example



FPGA Fundamentals: Complex Architecture

- A Plethora of IPs...

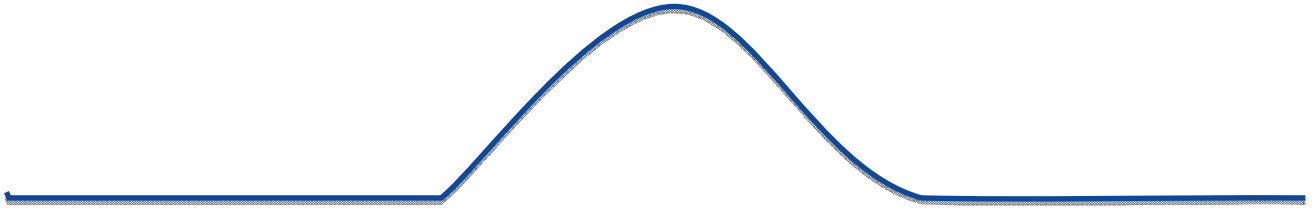


Fundamentals of Mixed-Signal FPGAs

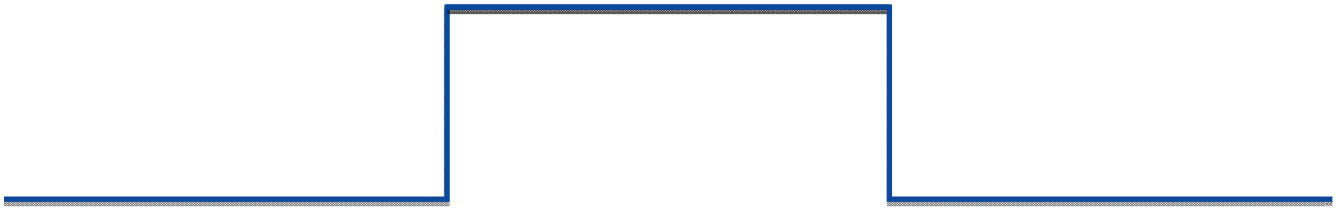
Actel Fusion® FPGA

Mixed-Signal FPGAs: Analog vs. Digital

- Analog Signal

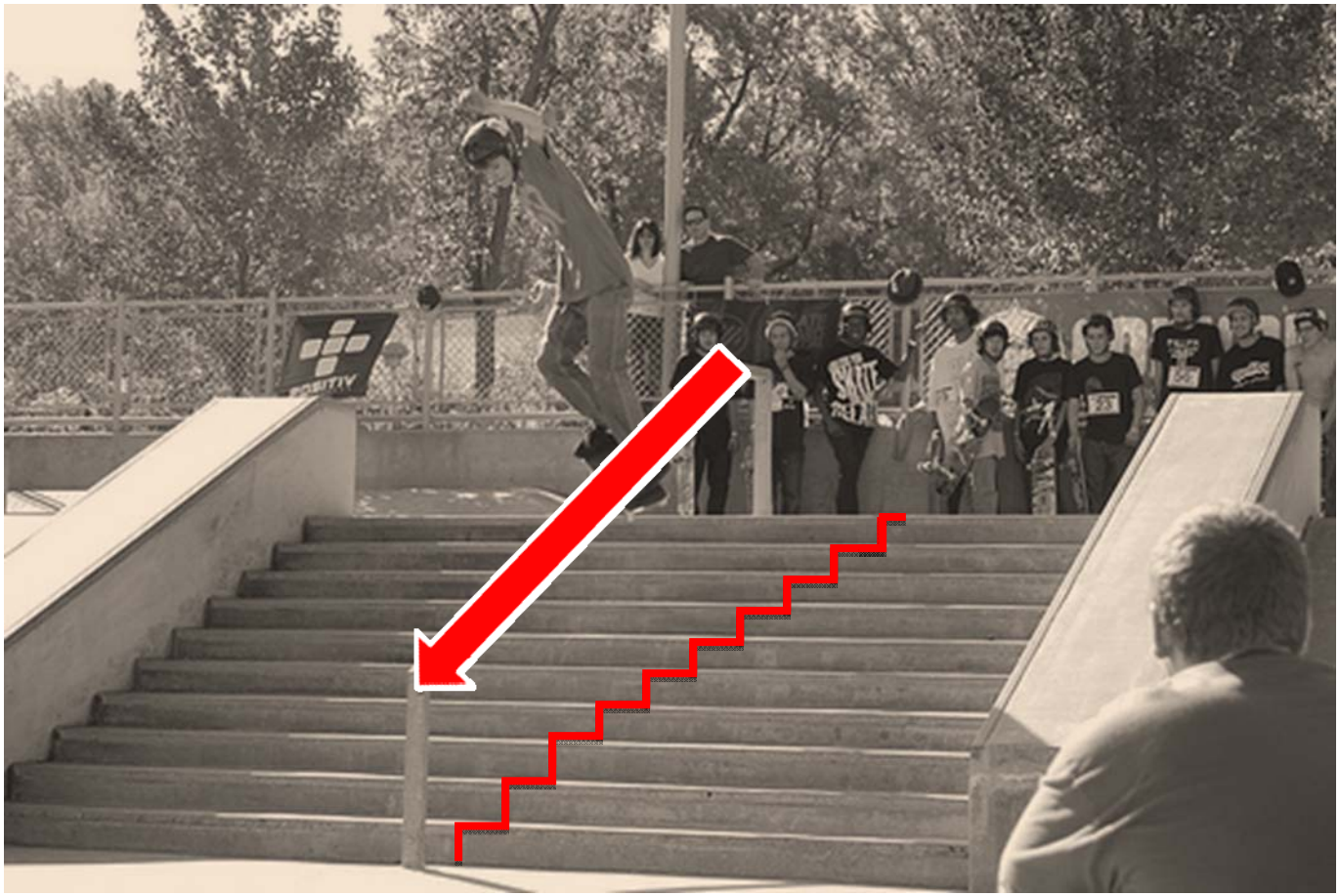


- Digital Signal



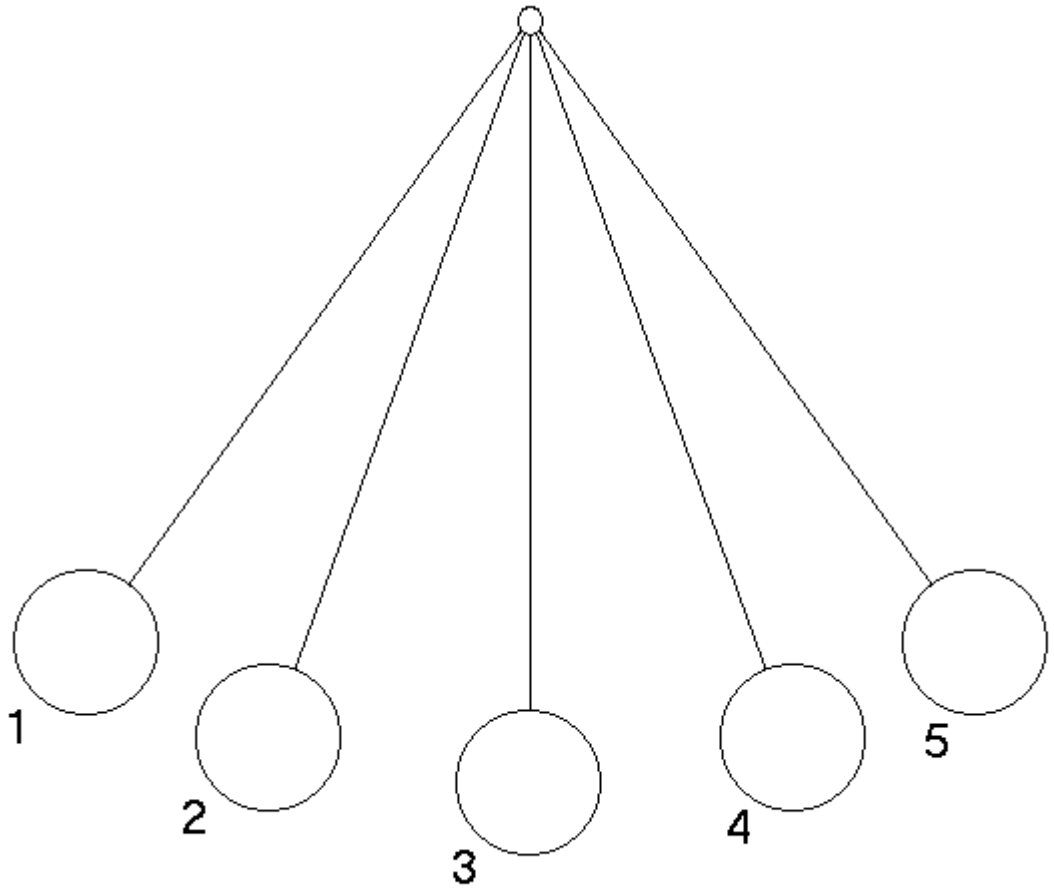
Mixed-Signal FPGAs: Analog vs. Digital

- Quanta



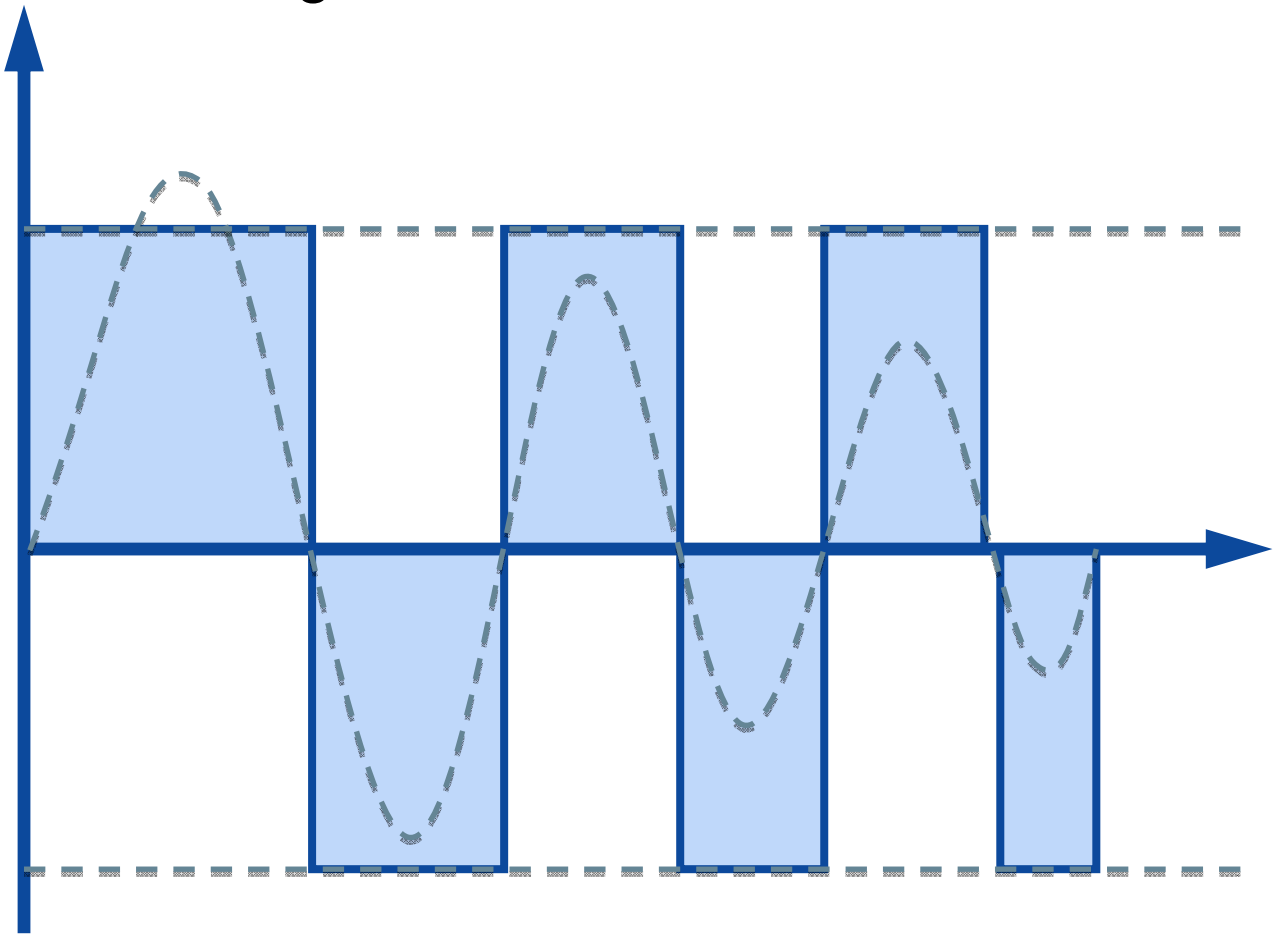
Mixed-Signal FPGAs: Analog vs. Digital

- Pendulum Tracking



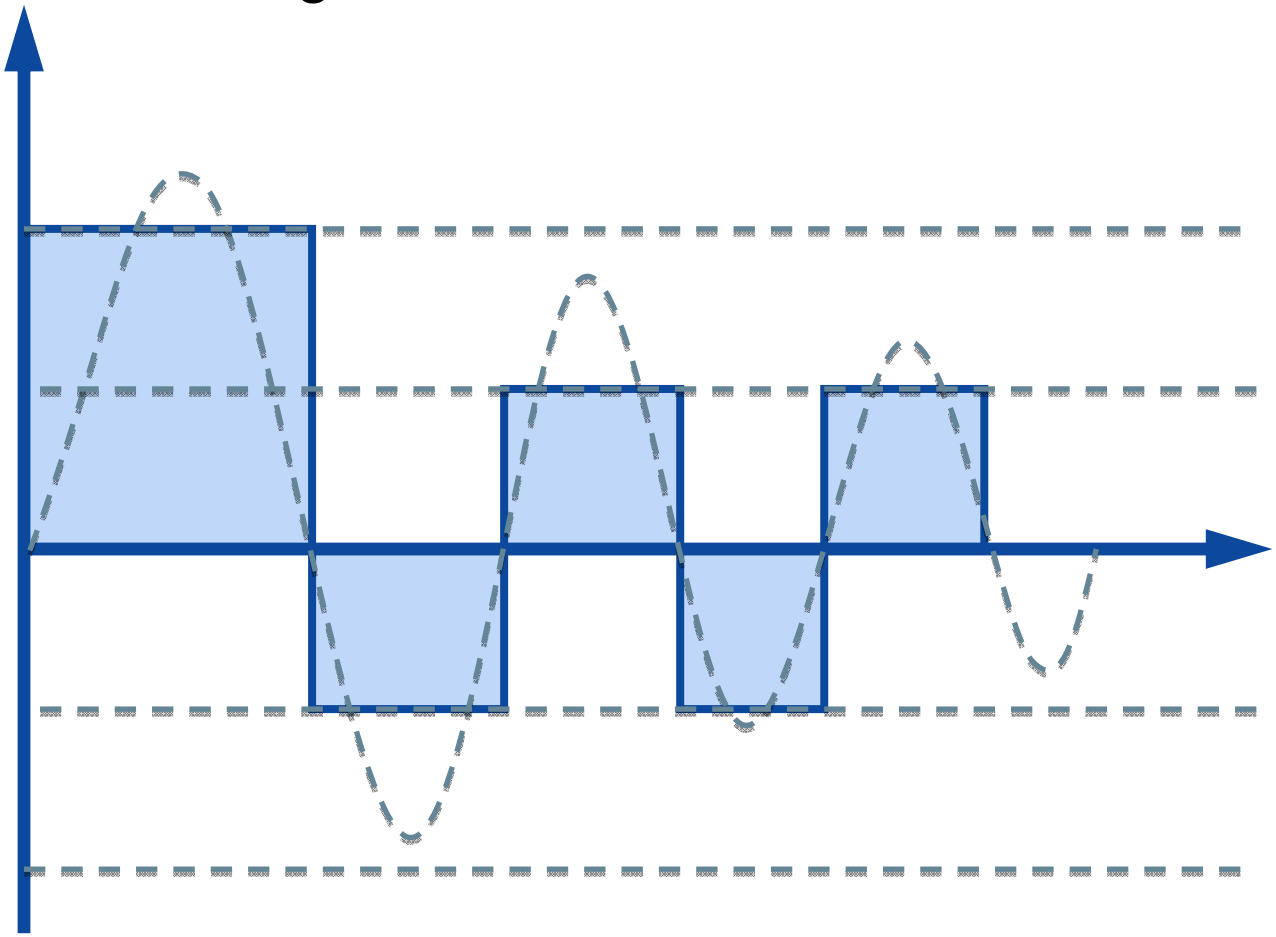
Mixed-Signal FPGAs: Analog vs. Digital

- Pendulum Tracking-2 Quanta-

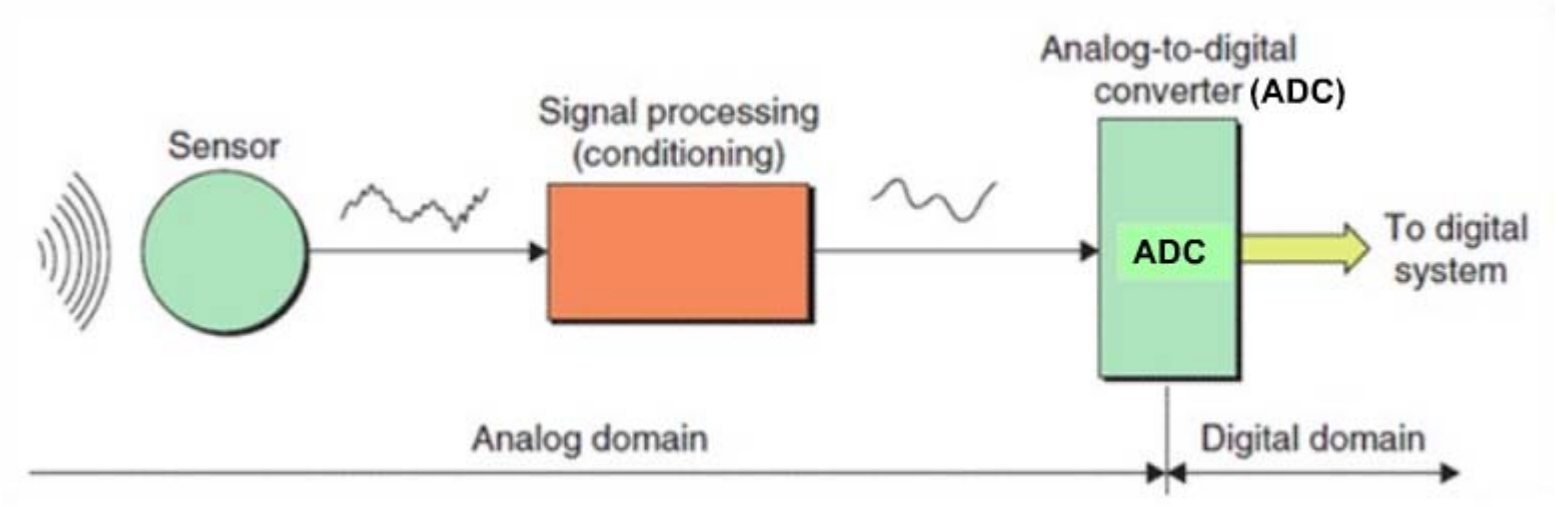


Mixed-Signal FPGAs: Analog vs. Digital

- Pendulum Tracking-5 Quanta-

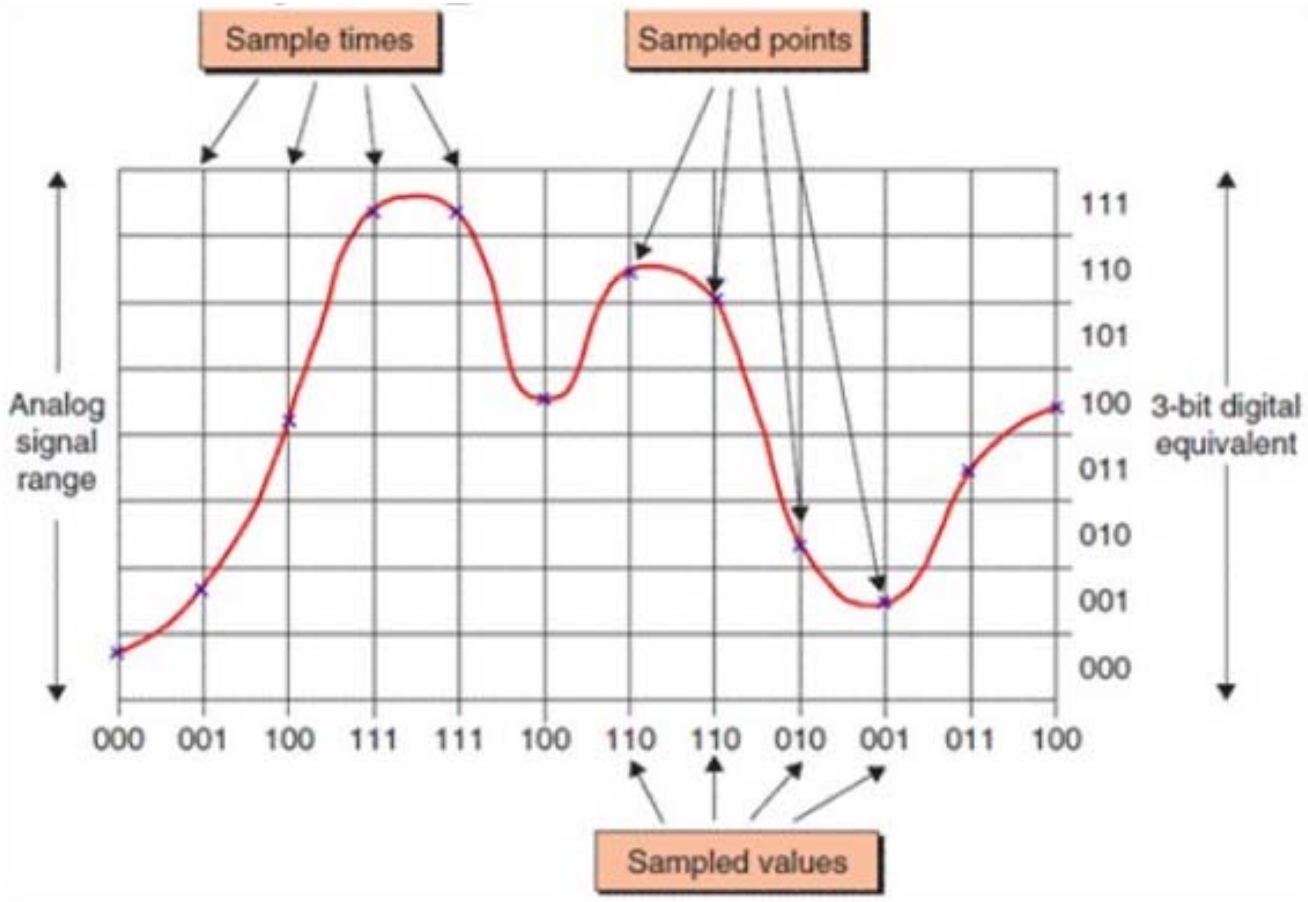


Mixed-Signal FPGAs: Analog-to-Digital (A/D)



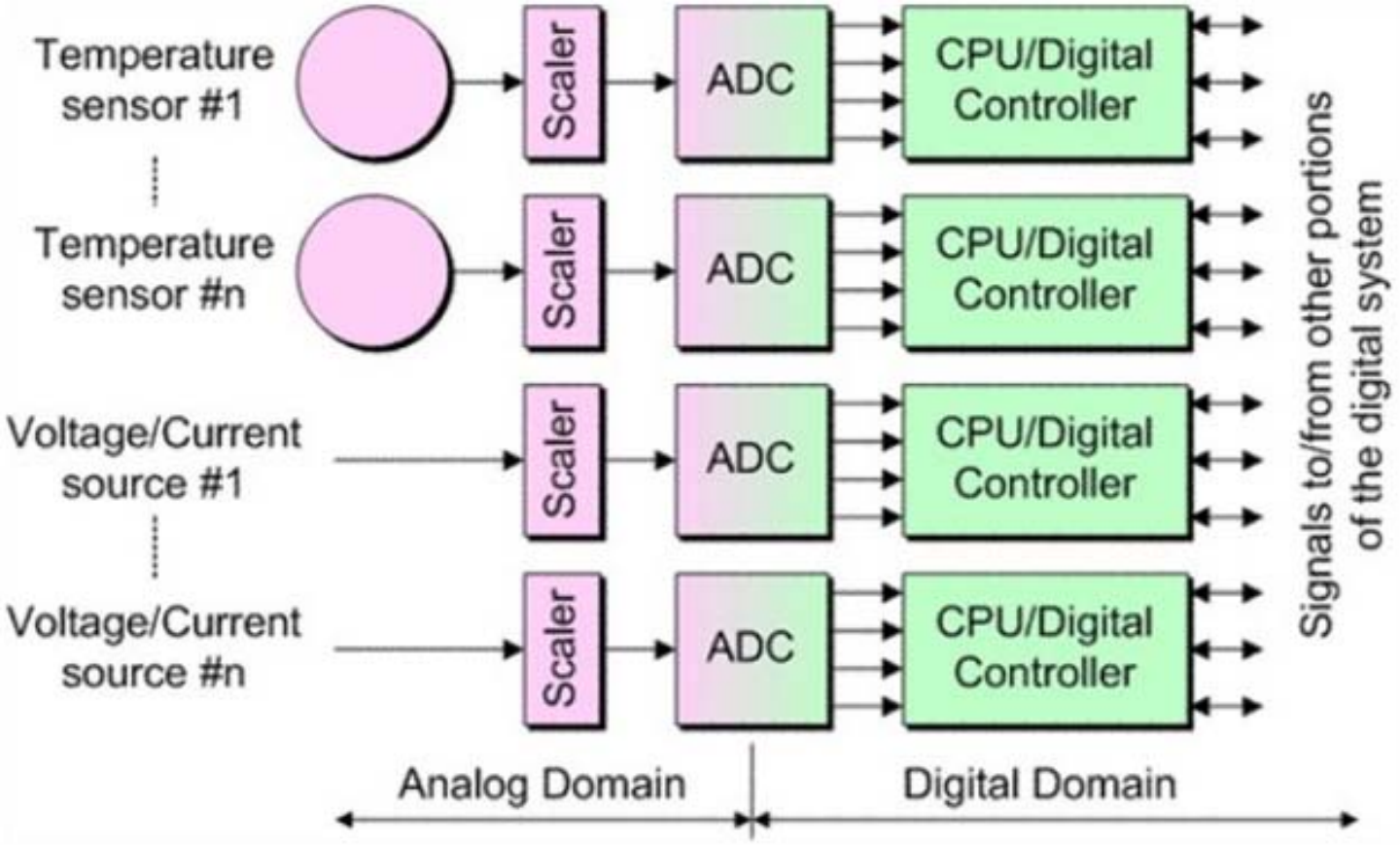
Mixed-Signal FPGAs: Analog-to-Digital (A/D)

- Sampling and Quantization



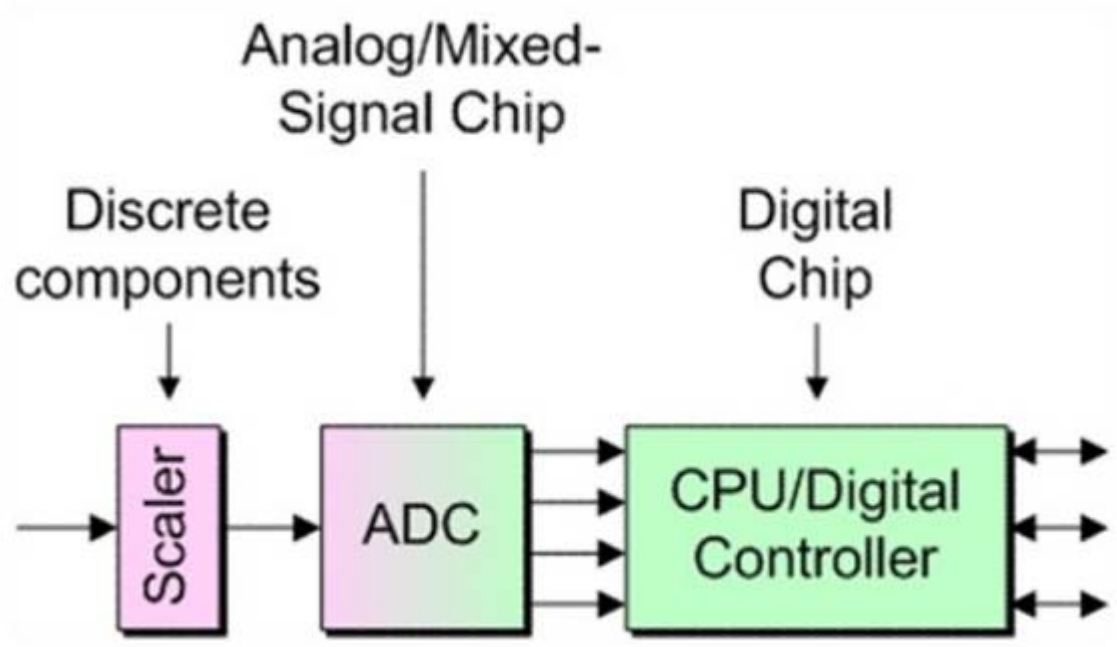
Mixed-Signal FPGAs: Analog-to-Digital (A/D)

- System Monitoring



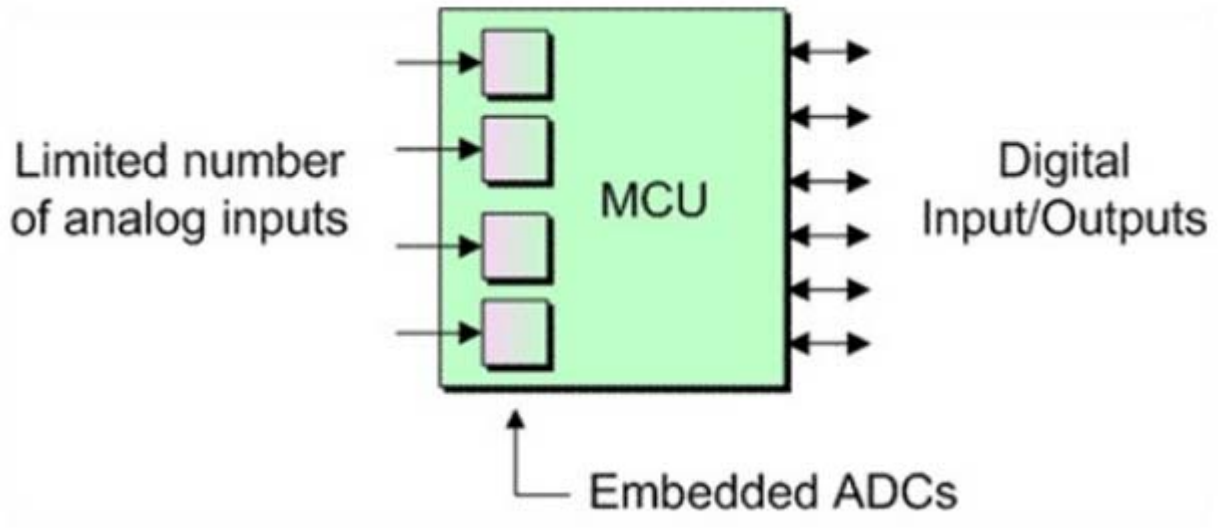
Mixed-Signal FPGAs: Analog-to-Digital (A/D)

- Separate Analog and Digital

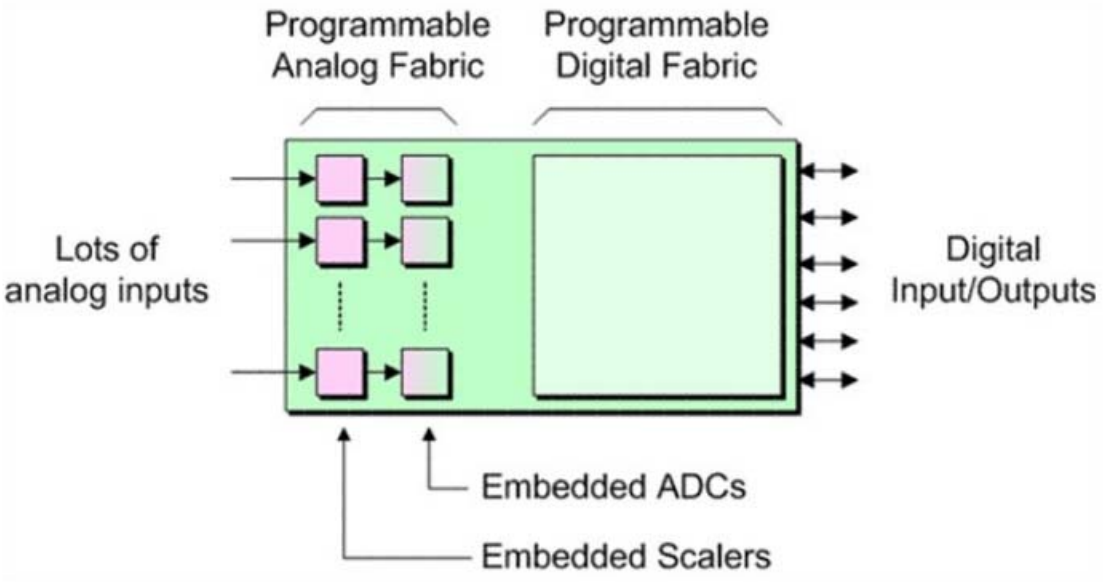


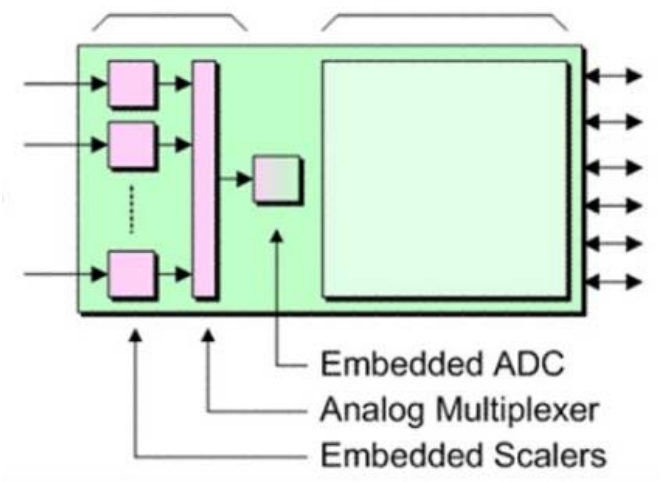
Mixed-Signal FPGAs: Analog-to-Digital (A/D)

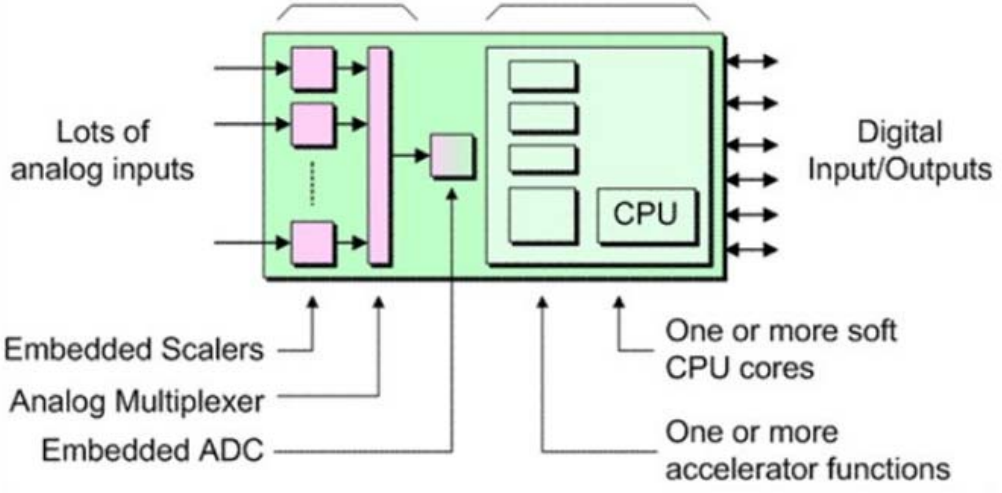
- Microcontrollers



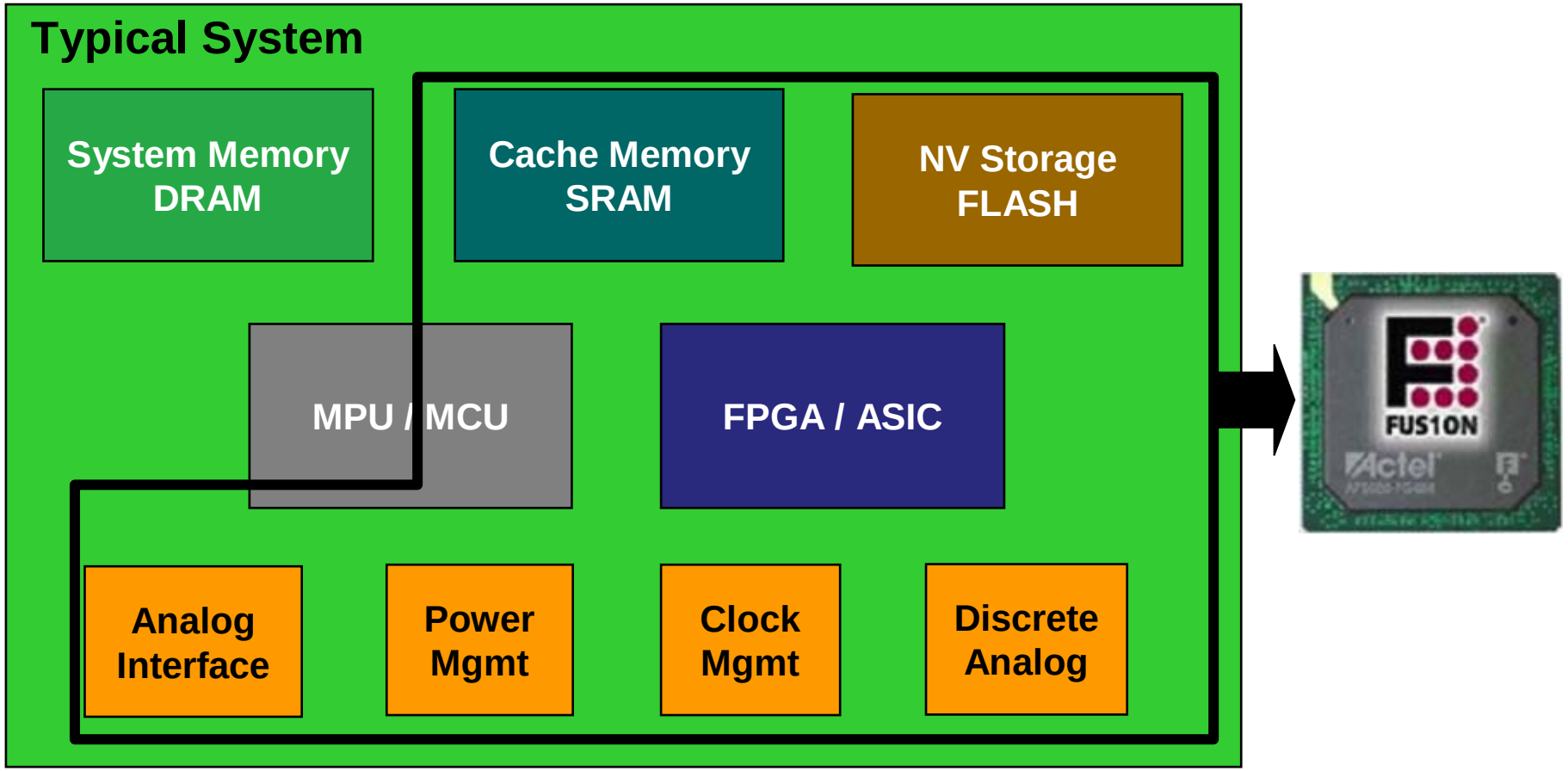
Mixed-Signal FPGAs: All in one





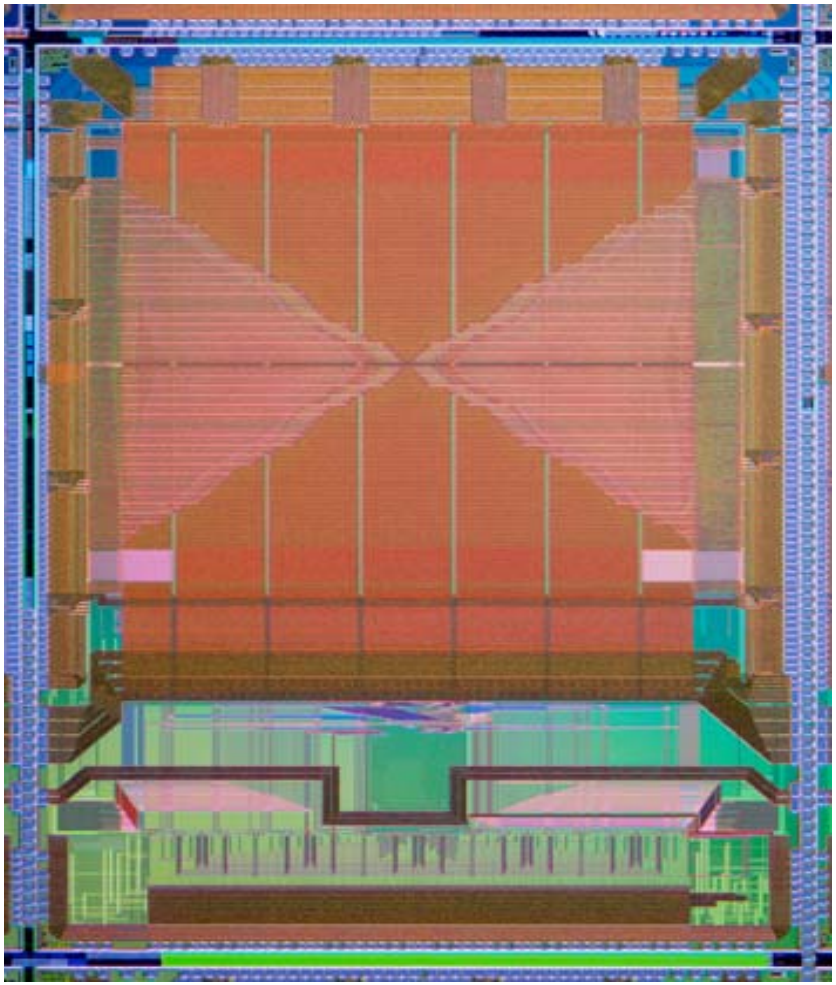


Actel's Fusion Mixed-Signal FPGA



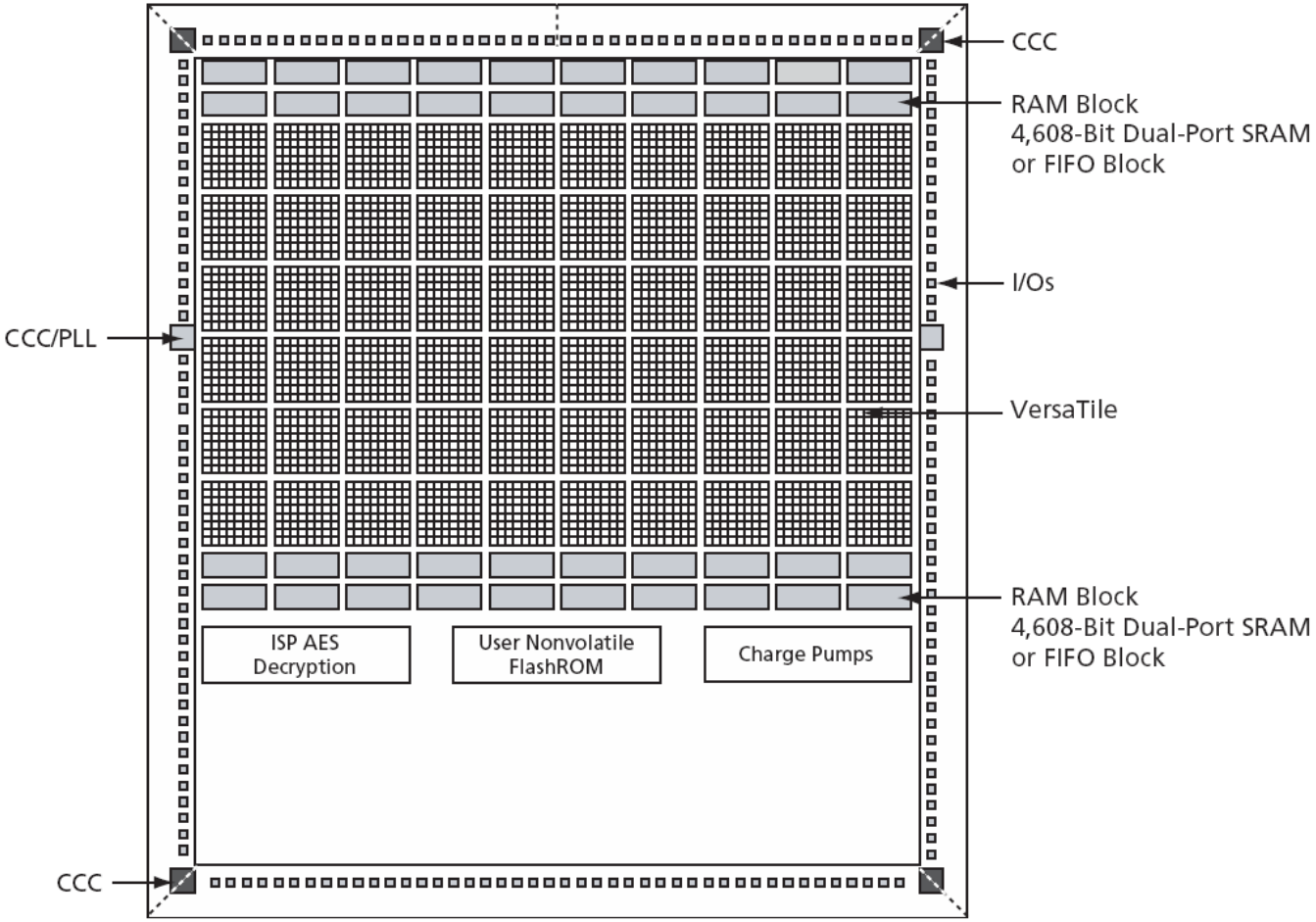
Actel's Fusion Mixed-Signal FPGA

- Flash based
- Up to 30 analog inputs
- ADC (12 bits, 600 ksps)
- Up to 1.5M system gates
- Advanced I/O support
- SRAM / FIFO blocks
- Embedded Flash Memory
- … and much more…



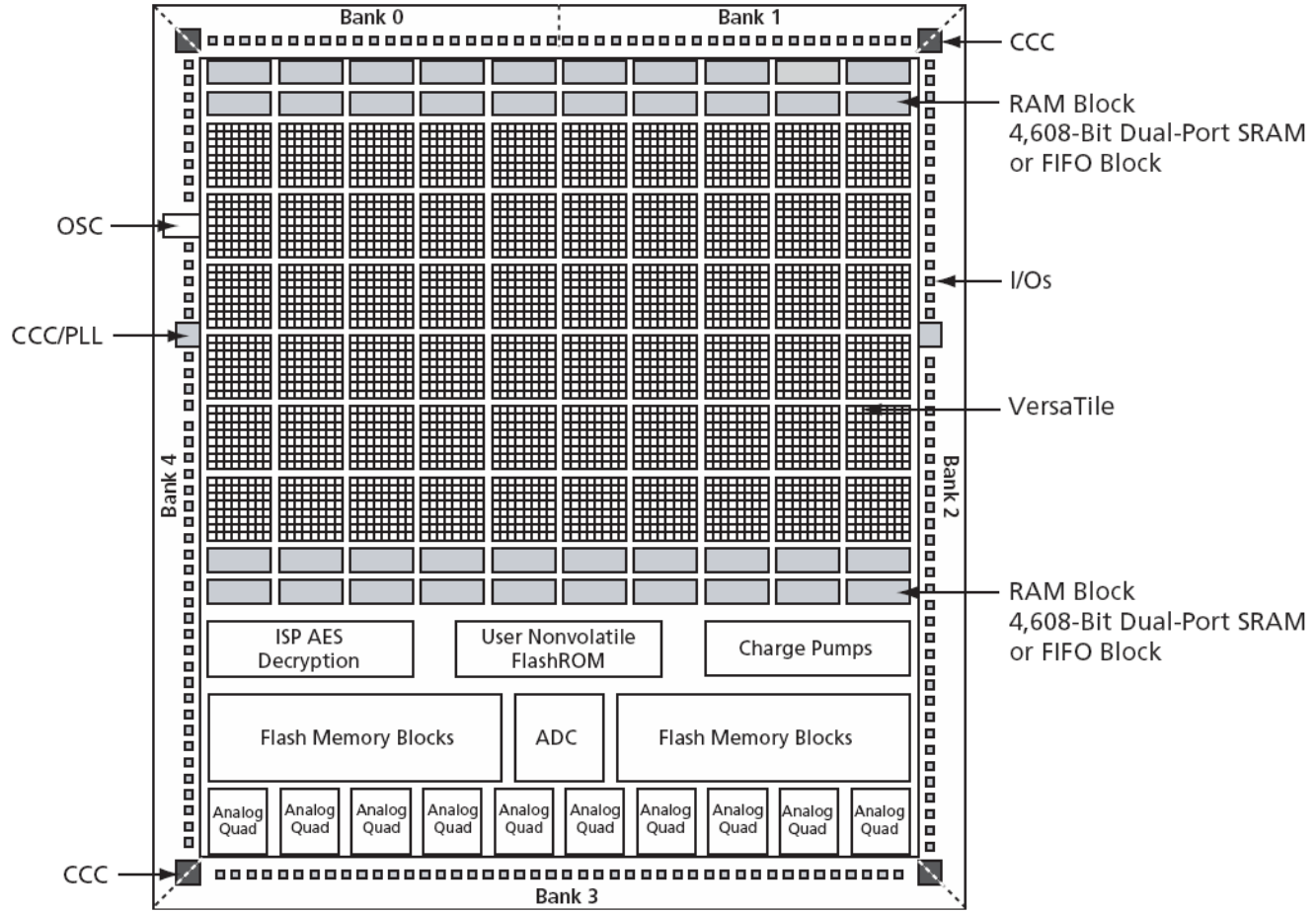
Fusion Architecture

- Fusion starts with ProASIC3…



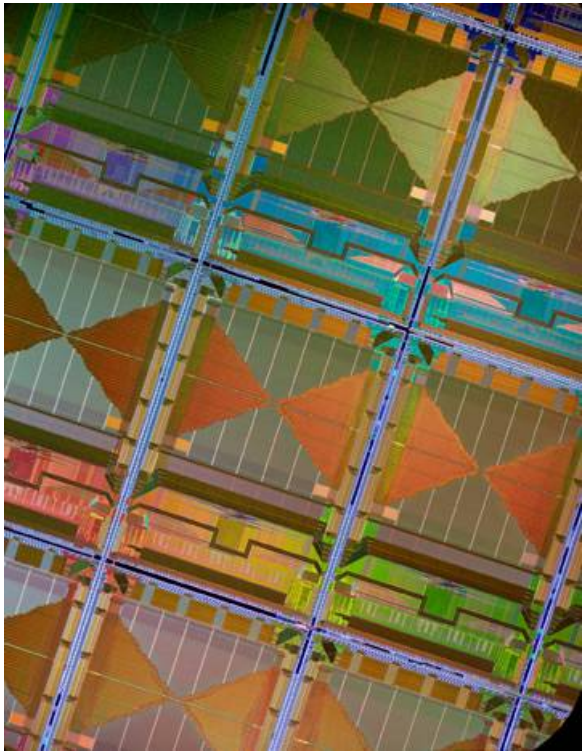
Fusion Architecture

- And adds Flash Memory Blocks and Analog Peripherals…



Fusion PSC: Target Applications

- Power and Temperature Management
  - Power sequencing with tracking control
  - Smart battery charging
  - Voltage, current, temperature monitors and alarms
  - Fan and heat-element control and monitoring
  - Intelligent Platform Management Interface (IPMI)
- Motor and Motion Control
  - Motor control – stepper, 3-phase and solenoid control
  - Anti-lock brakes
- System initialization and configuration
  - Context save and restore
  - Context switching
  - System boot codes
- Storage
  - Program code storage
  - EEPROM emulation
  - Data acquisition and logging
- Low Power and Clocking
  - Control for sleep mode and wake-up
  - Live at power-up clock generation, conditioning, and distribution

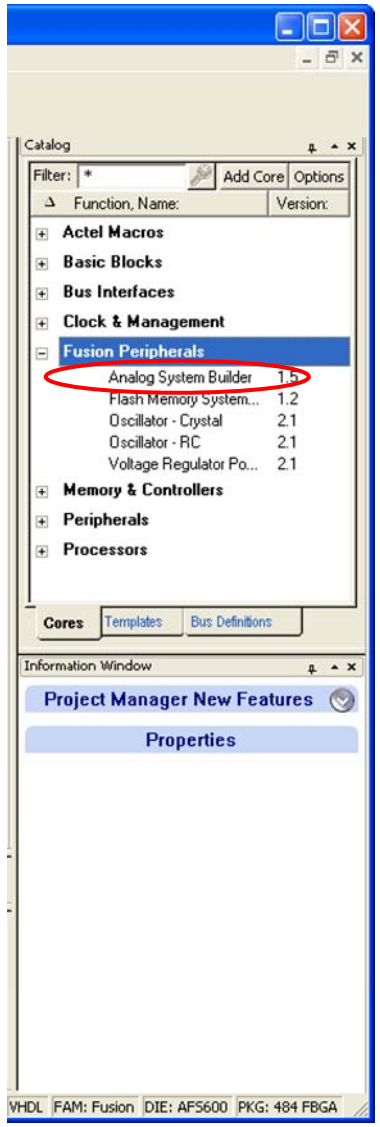


Fusion Family

	Part #	AFS090	AFS250	AFS600	AFS1500
	ARM enabled	-	-	M7AFS600	-
	Cortex M1 enabled	-	M1AFS250	M1AFS600	M1AFS1500
General	System Gates	90K	250K	600K	1,500K
	Tiles (D-FF)	2,304	6,144	13,824	38,400
	Secure (AES) ISP	Yes	Yes	Yes	Yes
	PLLs	1	1	2	2
	Globals	18	18	18	18
Memory	RAM blocks (512x9)	6	8	24	60
	Total RAM	27 Kbits	36 Kbits	108 Kbits	270 Kbits
	FlashROM bits	1Kbits	1Kbits	1Kbits	1Kbits
	Flash Memory Blocks	1	1	2	4
	Total Flash Memory	2 Mbits	2 Mbits	4 Mbits	8 Mbits
Analog	Analog Quads	5	6	10	10
	Analog Inputs	15	18	30	30
	Output Gate Drivers	5	6	10	10
I/O	I/O Types	Analog / LVDS / Std+	Analog / LVDS / Std+	Analog / LVDS / Pro	Analog / LVDS / Pro
	I/O Banks (+ JTAG)	4	4	5	5
	Max Digital I/O	75	114	172	252
	Analog I/O	20	24	40	40
I/O: Single / Double ended (analog)	QN108	37/9 (16)			
	QN180	60/16 (20)	65/15 (24)		
	PQ208		93/26 (24)	95/46 (40)	
	FG256	75/22 (20)	114/37 (24)	119/58 (40)	119/58 (40)
	FG484			172/86 (40)	228/86 (40)
	FG676				252/126 (40)

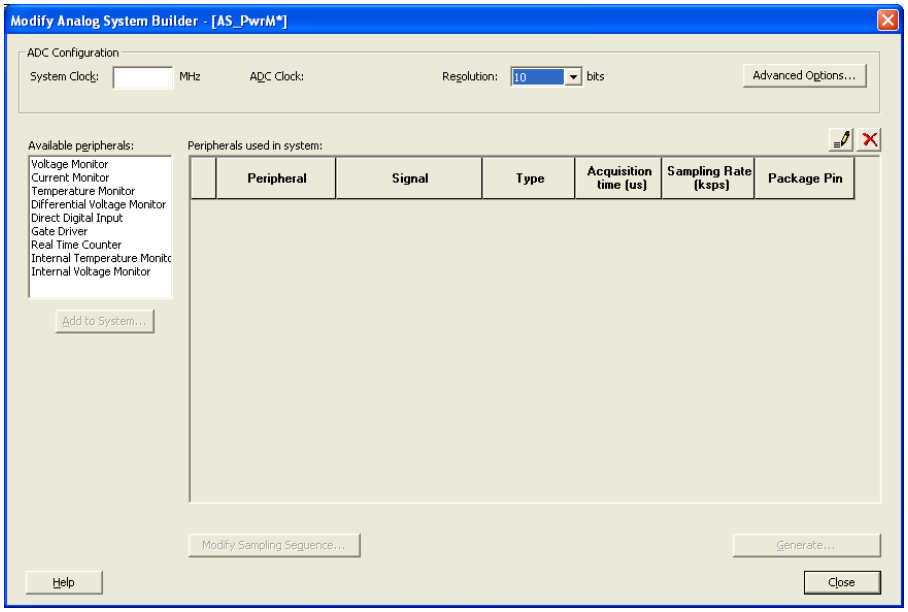
Analog System Builder

- Creates Complete Analog System Including
  - AB Hard-Macro
  - Analog System Soft-IPs (RTL)
  - Analog System data storage RAMs
  - Memory files for simulation
  - Configuration file for import into NVM System

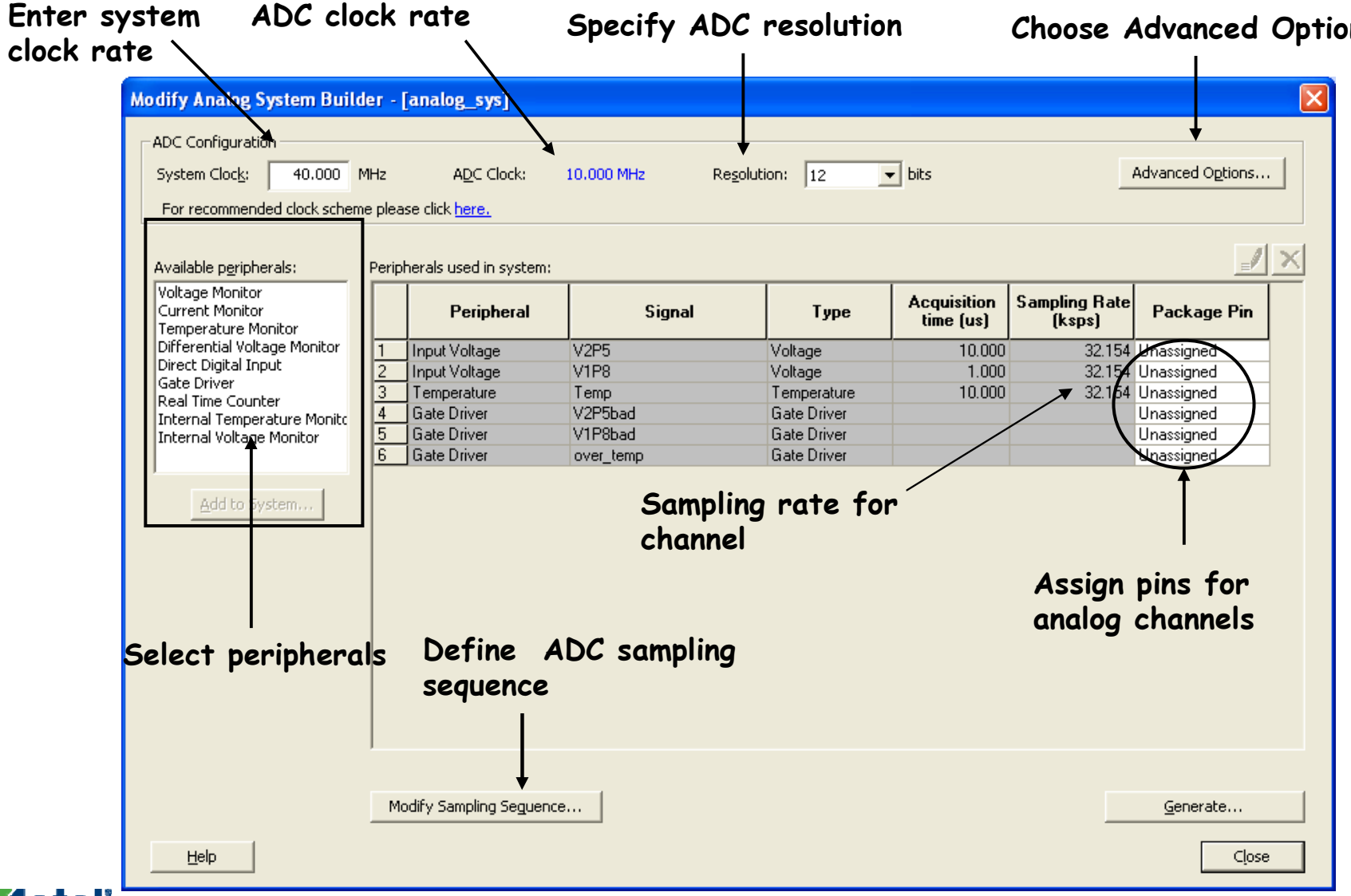


Analog System Builder: Supported Peripherals

- Voltage Monitor
- Current Monitor
- Differential Voltage Monitor
- Temperature Monitor
- Direct Digital Input
- Output Gate Driver
- Internal Temperature Monitor
- Internal Voltage Monitor
- RTC (Real Time Counter)

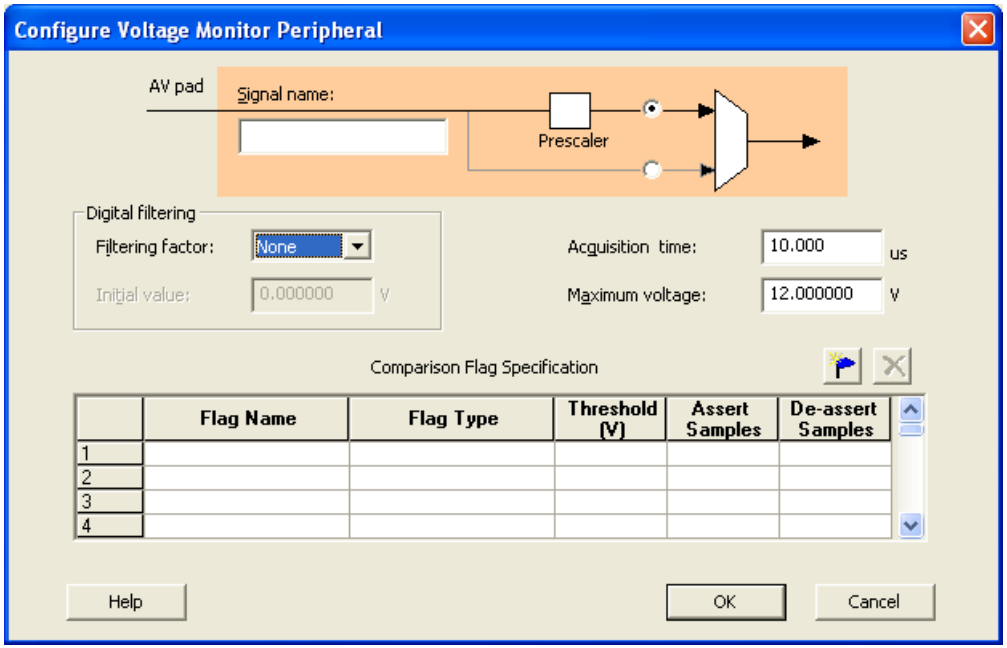


Analog System Builder GUI



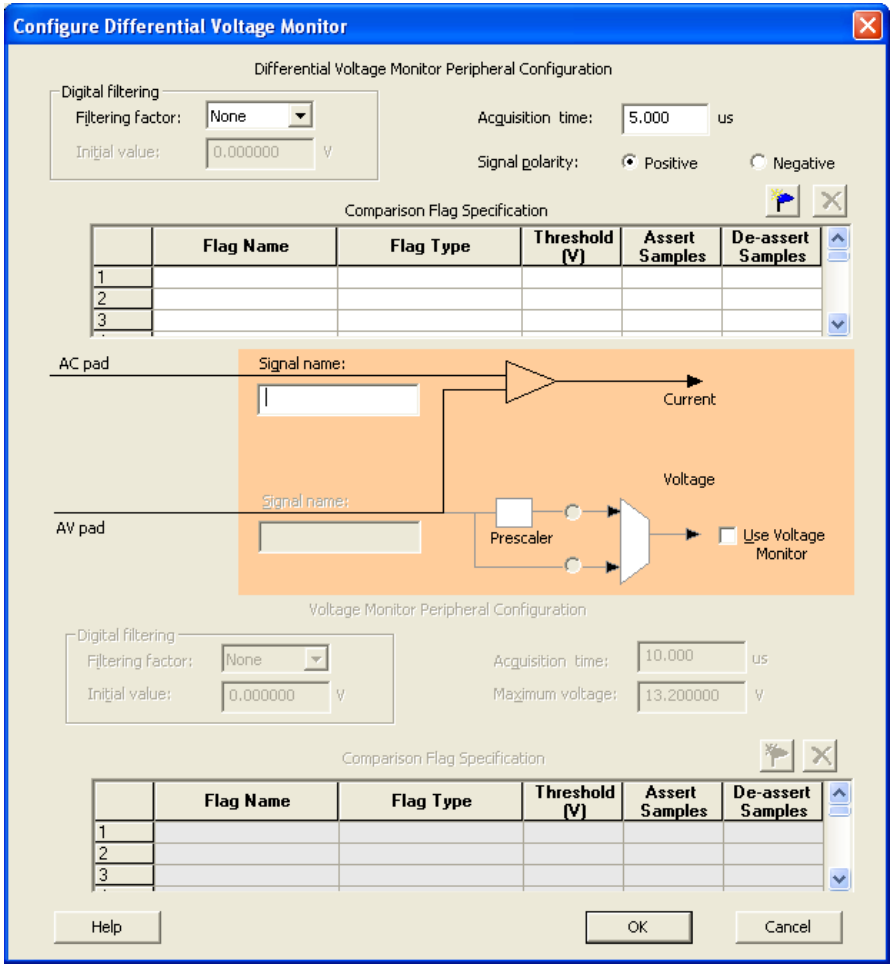
Analog System Builder GUI

- To Configure a Voltage Monitor



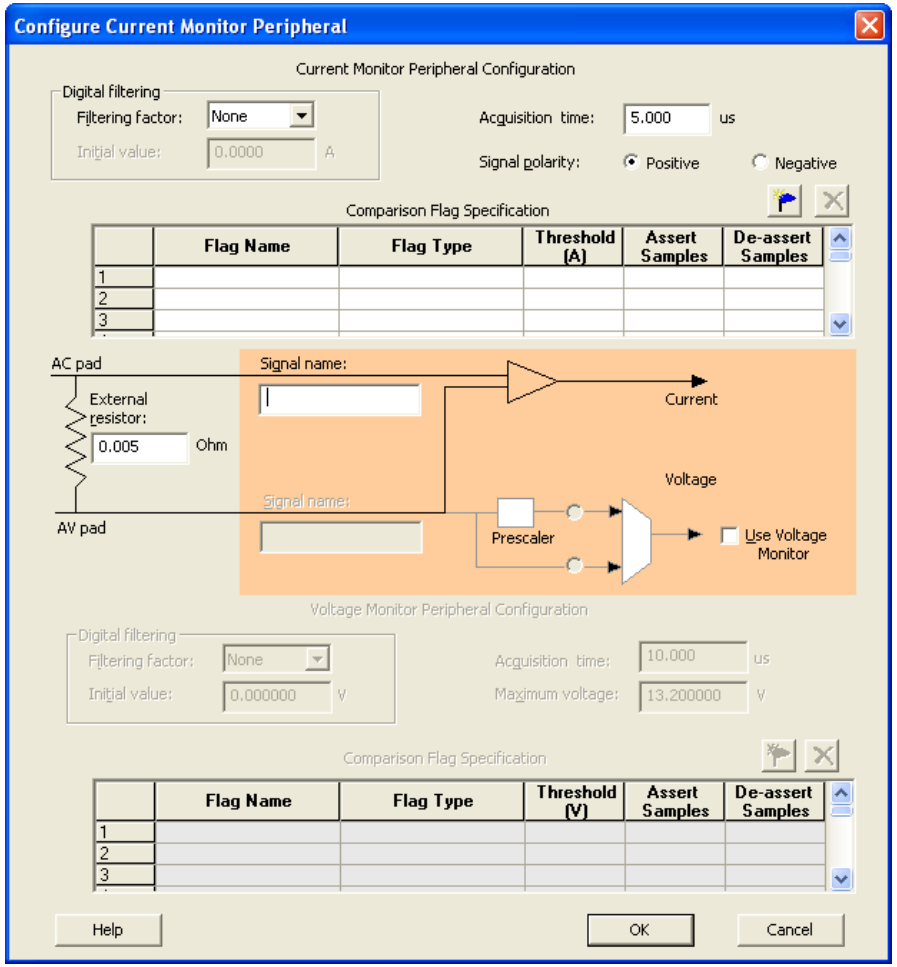
Analog System Builder GUI

- To Configure Differential Voltage Monitor



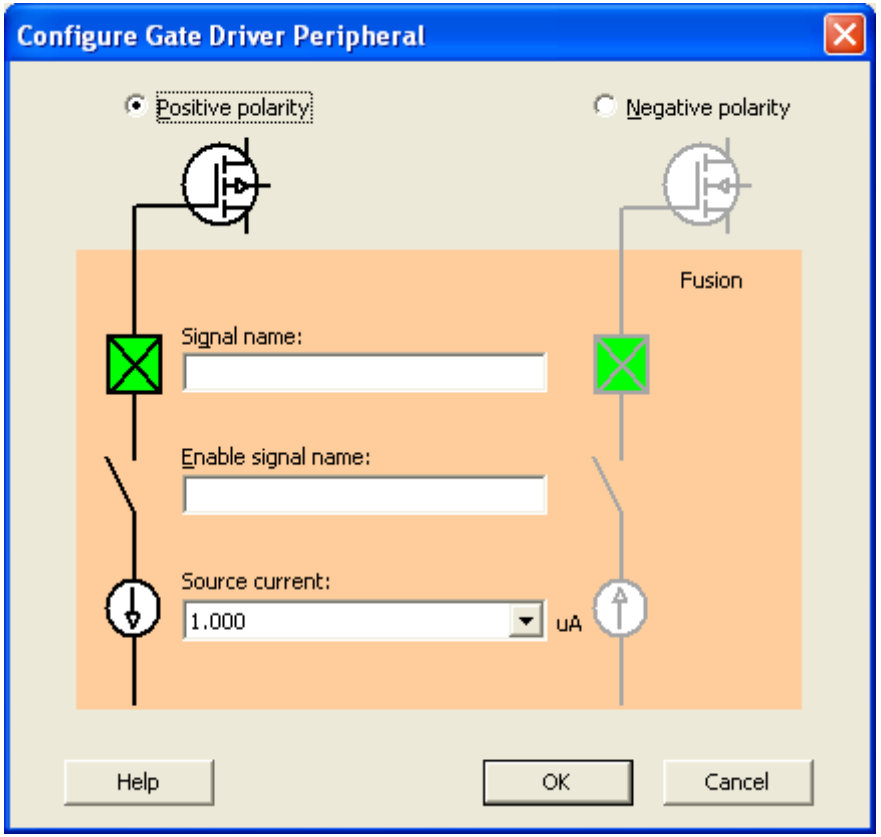
Analog System Builder GUI

- To Configure Current Monitor



Analog System Builder GUI

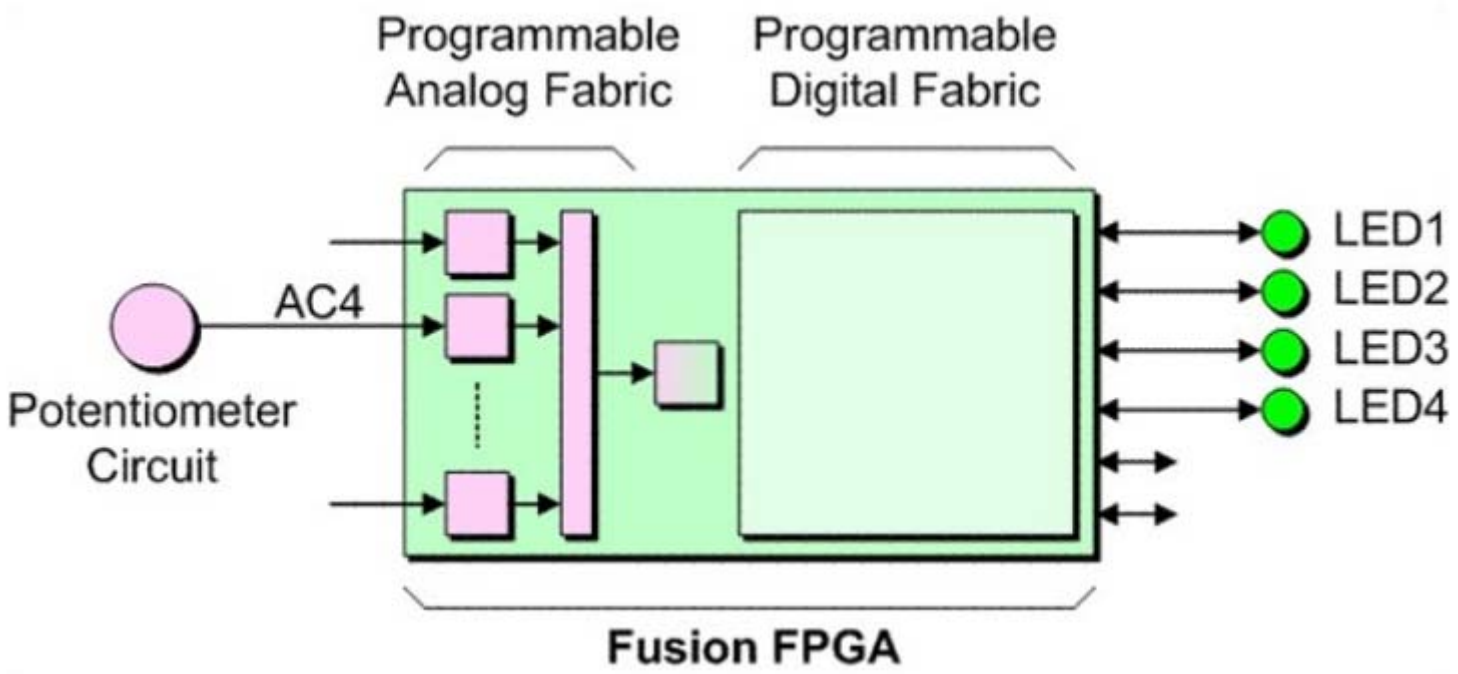
- To Configure Gate Driver Outputs



A Simple Example

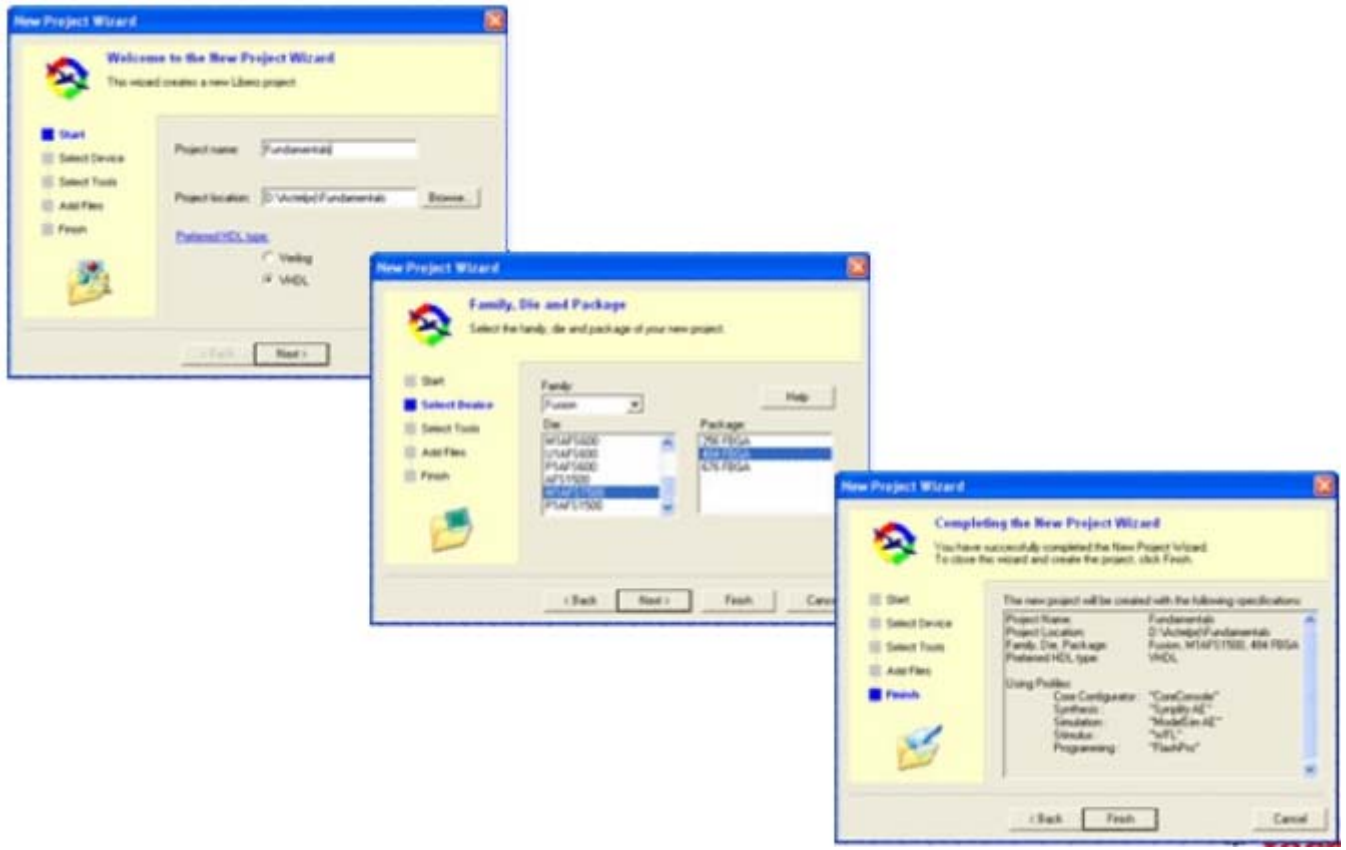
A Simple Example

- Displaying Flags for Different Voltage Thresholds



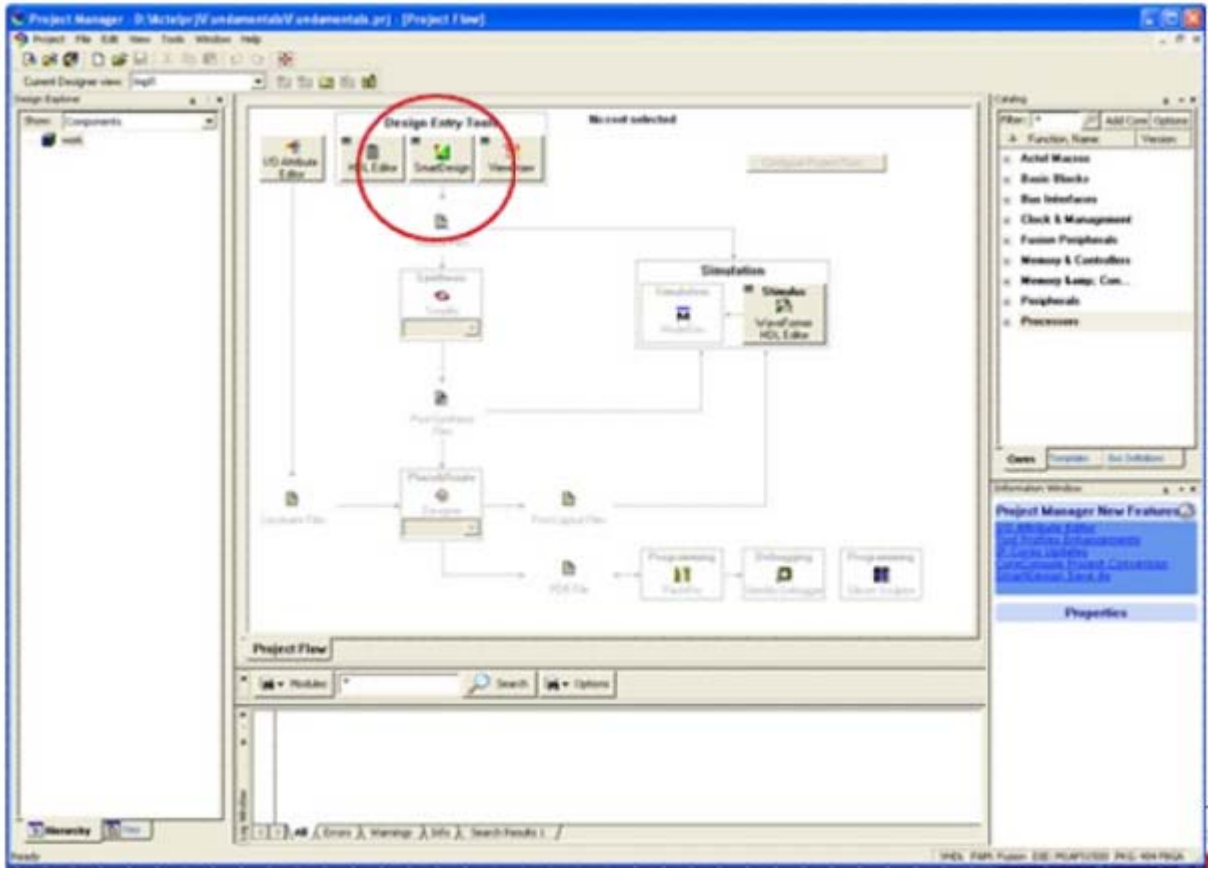
A Simple Example

- Creating a Libero Project



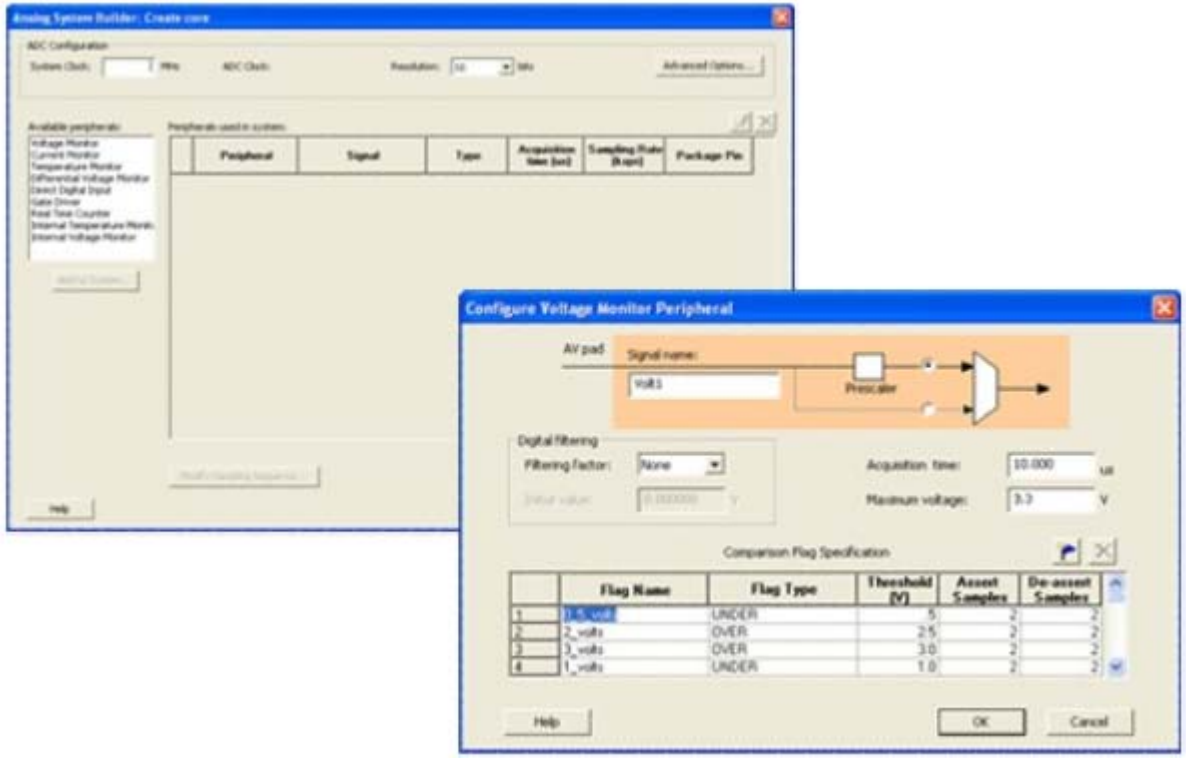
A Simple Example

- Use SmartDesign to Connect Various Analog & Digital Blocks on a Virtual Canvas



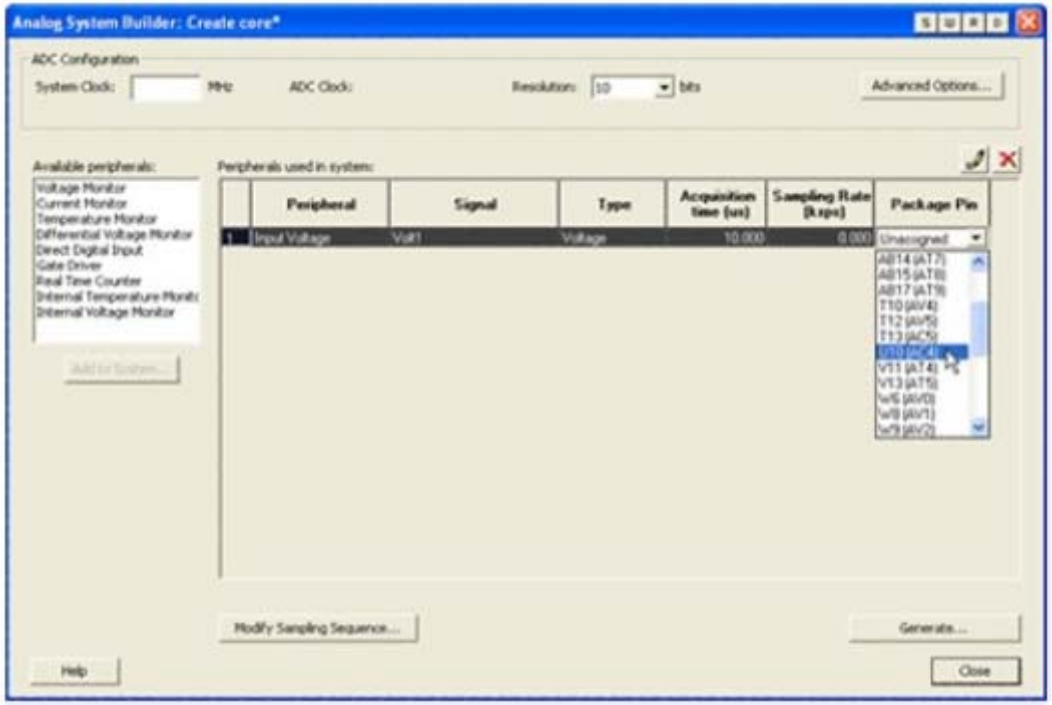
A Simple Example

- Within SmartDesign, use the Analog System Builder to Create the Voltage Monitor – Max from Potentiometer=3.3V



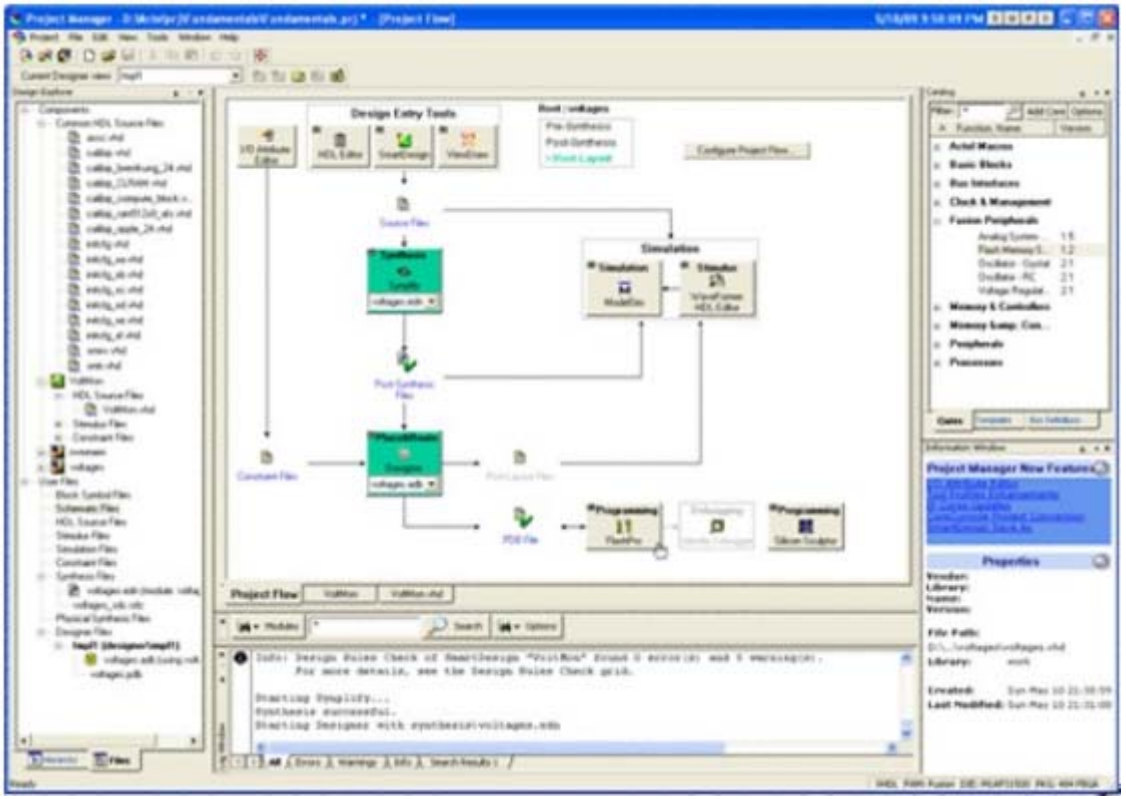
A Simple Example

- Assigning to a Voltage Channel (ex. AC4)



A Simple Example

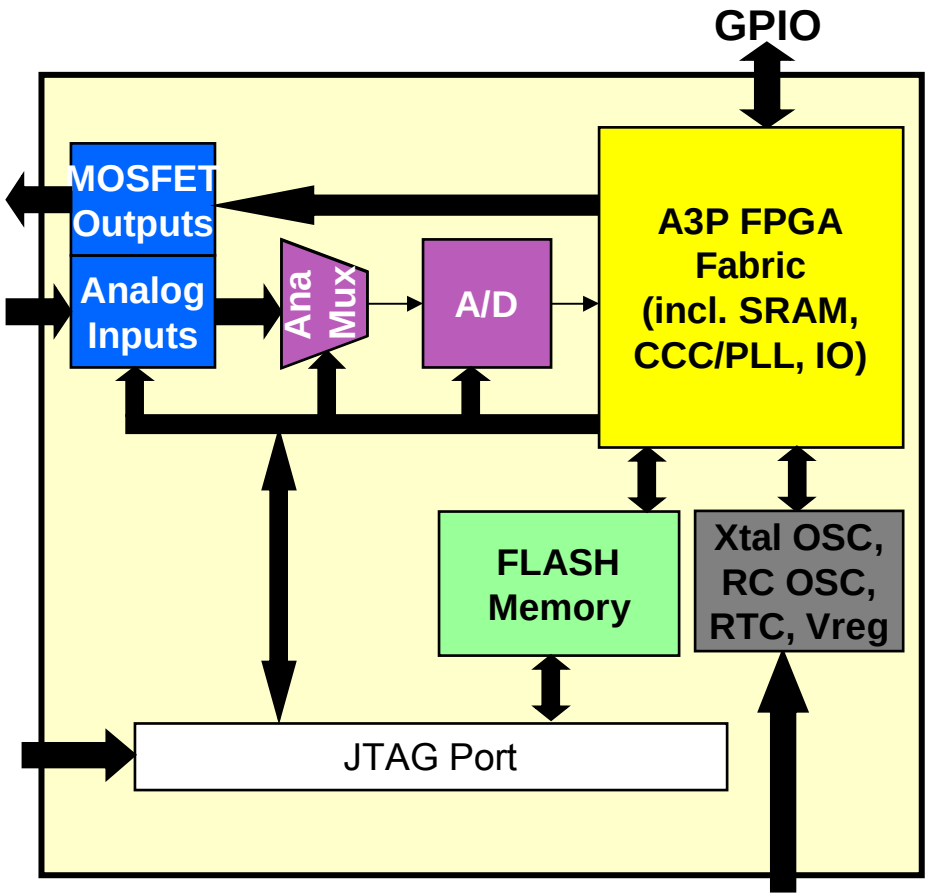
- Synthesis -> P&R -> Programming



Fusion Peripherals

- Clocking
  - RC and Crystal Oscillators
  - Clock Conditioning Circuitry
  - No-Glitch MUX
- Embedded Memory
- Analog Block

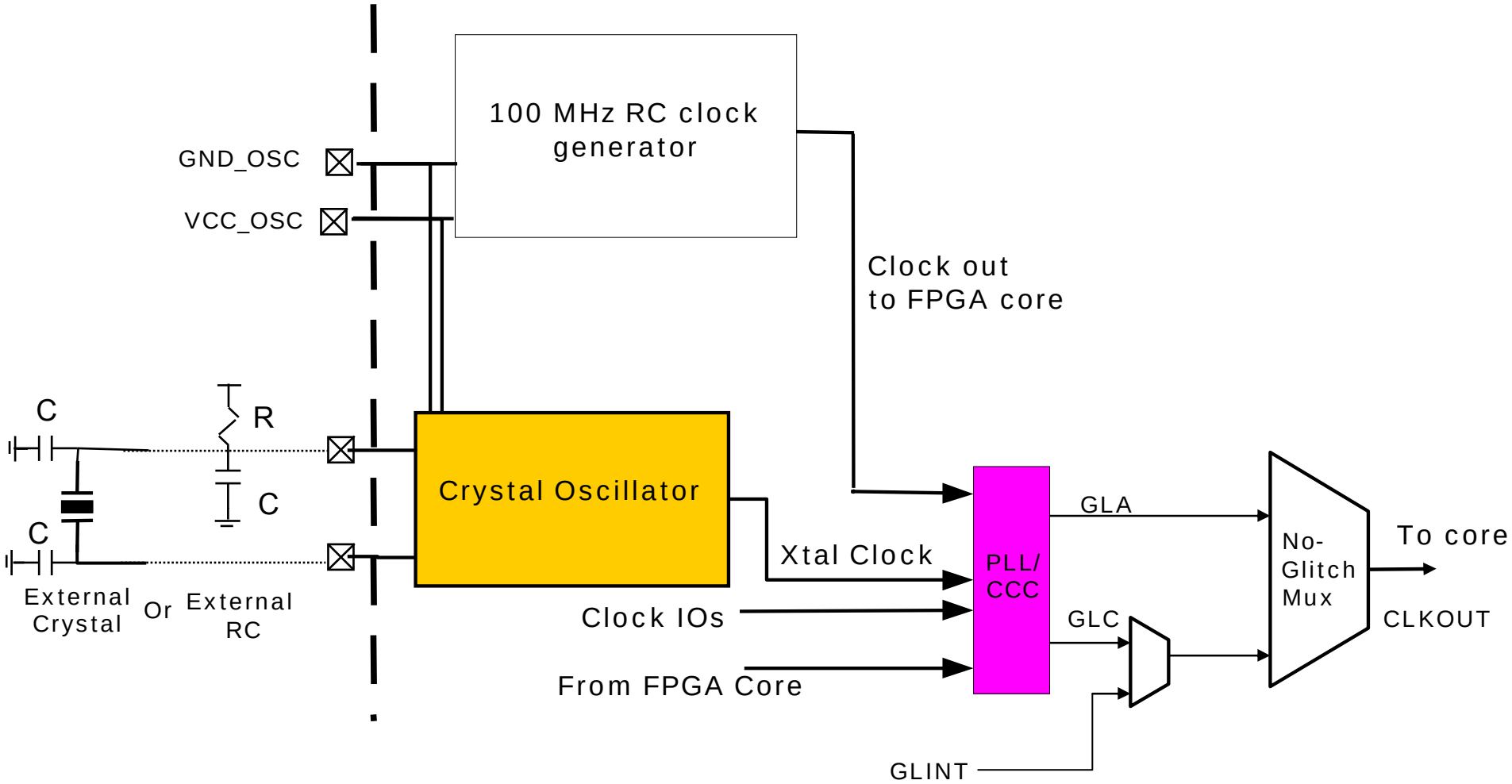
Fusion Clock Resources



- Fusion Provides Multiple Clocking Resources:
  - On chip clock sources:
    - RC oscillator @ 100MHz
    - Crystal Oscillator Circuit
    - Six CCC Blocks / PLLs (1 or 2)
  - No-Glitch MUX
  - Real Time Counter (RTC)
- Use Models
  - Internal 100MHz RC oscillator
  - Crystal OSC circuit
    - 32 KHz – 20 MHz
  - CCC/PLLs can multiply, divide, and phase shift clock signals for user applications
    - Sources include: crystal Osc, RC Osc, or external clock
  - RTC enables low power sleep mode

Fusion Clock Resources

System Block Diagram



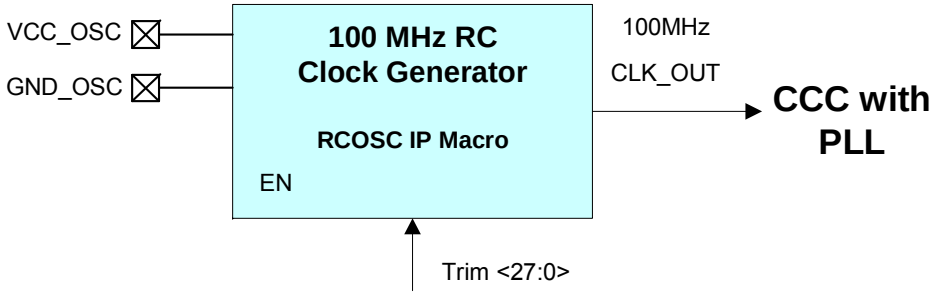
RC & Crystal Oscillators

Fusion RC Oscillator

Features

- Summary
  - Clock frequency: 100 MHz
    - +/-1%: 0 to 70 C
  - Duty cycle 40% – 60%
  - Requires no external components
  - Can be used to drive either a PLL or Output Pad
  - Provides an integrated, accurate on-chip clock source
- Additional Features
  - Factory trim capability for high precision

Block Diagram

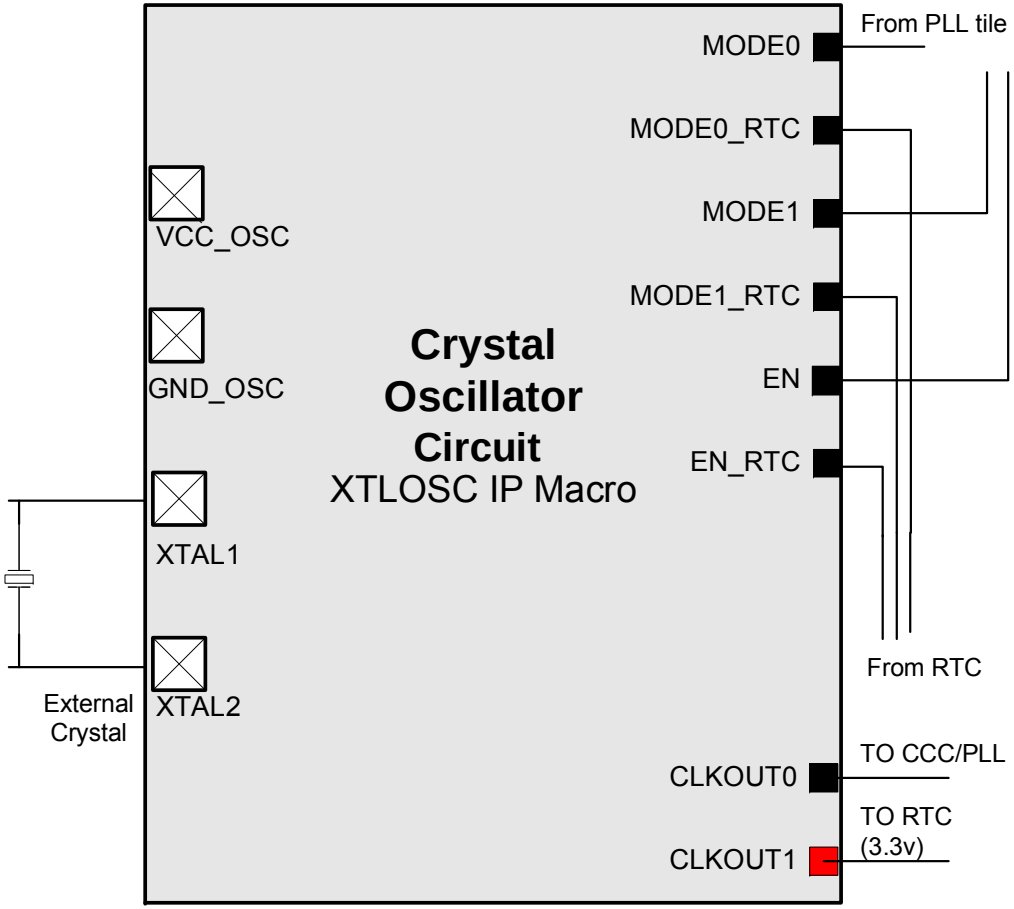


Note*: Crystal oscillator and RC oscillator physically share the same VCC and GND power pads.

Fusion Crystal Oscillator

Features

- High-precision Clock Source
  - Up to 100ppm (0.01%) precision
- Supports Both On-chip and Off-chip Resources
  - Source for PLL/CCC
- Modes
  - External Crystal:
    - High gain: 2 – 20 MHz
    - Med gain: 0.2 – 2 MHz
    - Low gain: 32 – 200 KHz
  - RC Network: 32kHz – 4MHz
- Specifications
  - Maximum output jitter – 50pS RMS
    - 10 MHz crystal (0.05%)
  - Duty cycle: 40% - 60%



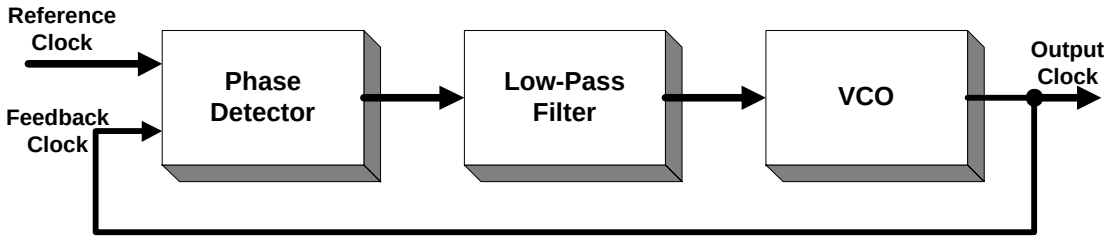
Note: Crystal oscillator and RC oscillator physically shared the same VCC and GND power pads.

Fusion
Clock Conditioning Circuitry (CCC)

- All Devices Have 6 Clock Conditioning Circuitry (CCC) Blocks, BUT …
  - … Some CCC Blocks do not contain PLLs
    - AFS090 / AFS250 – 1 PLL
    - AFS 600 / AFS1500 – 2 PLLs
  - … Non-PLL Functionality Still Available
    - Divider and Delay Elements
    - Global Access from I/O or Internal Signal
- 3 Global MUX Blocks
  - Steer Signals from Global Pads and FPGA Fabric into Global Networks

Fusion PLL

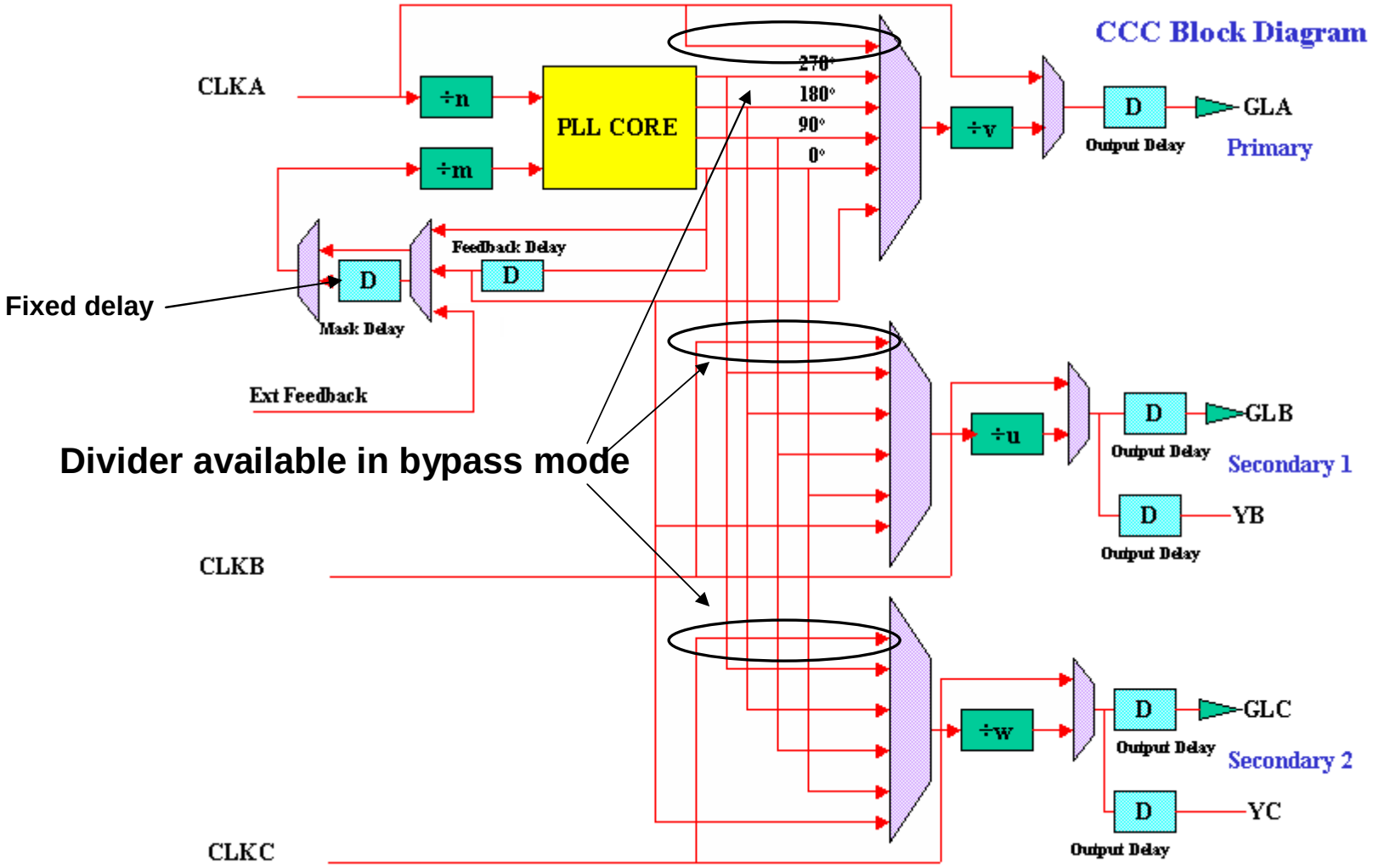
- Functions
  - Clock Phase Adjustment
  - Clock Delay Minimization
  - Clock Frequency Synthesis
  - Allows Access From Global Pads To Global Network and PLL
  - Allows Access From PLL To FPGA Core
- Input Sources:
  - Single Ended I/O, Differential I/O, FPGA Core, RC Osc, XTAL Osc



Fusion PLL
Features

- Delay Blocks (6 Programmable and 1 Fixed)
  - Programmable Delay/Advance up to 5.56 ns in 160 ps increments for Clock Skew Minimization
- 5 Frequency Divider Blocks
  - Provide Frequency Multiplication/Division
- Clock Phase Adjustment
  - 0°, 90°, 180°, and 270°
- Dynamic Shift Register
  - Provides Dynamic Reconfiguration Capability

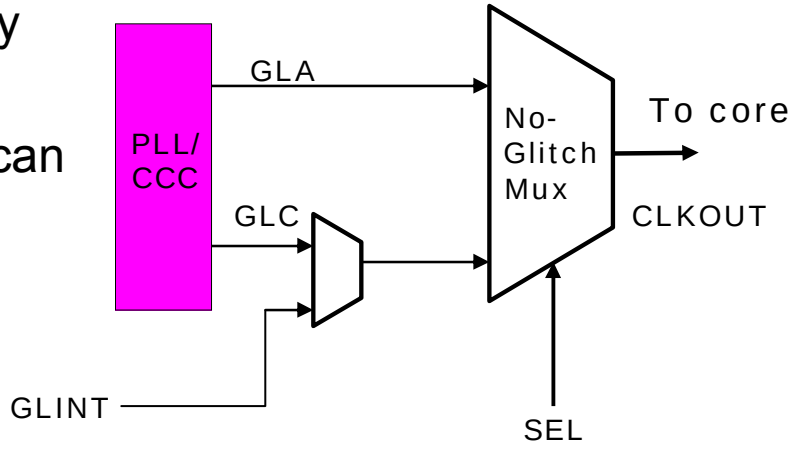
Fusion PLL
Block Diagram



No-Glitch MUX (NGMUX)

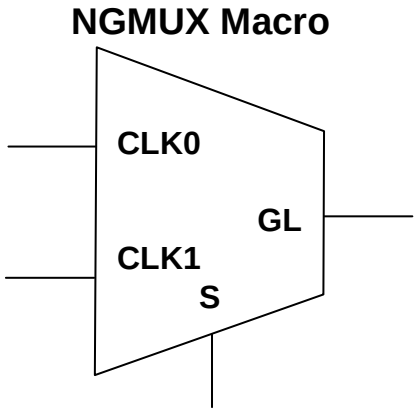
Fusion No-Glitch MUX (NGMUX)

- Provides Special Switching Sequence Between Two Asynchronous Clock Domains
  - User-configurable to select between 2 of 3 possible clock sources – GLA, GLC, or other internal signal
  - Time-out circuitry included in case one of the clocks stops or runs at very low frequency
- Advantage
  - Eliminates narrow pulses/glitches which can cause clocking errors
    - Especially critical in hi-rel apps
- Uses
  - Clock domain control
  - Power reduction by transitioning to lower frequency



Fusion No-Glitch MUX Usage

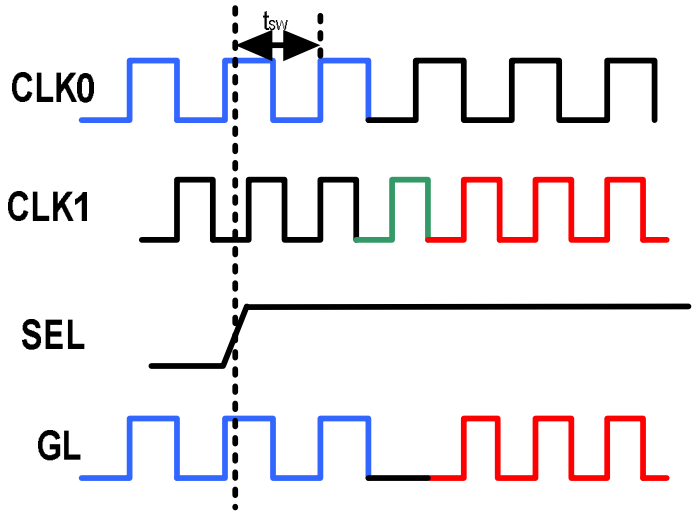
- NGMUX is Implemented as a 2:1 Mux in the Software
  - Instantiate NGMUX in VHDL or Verilog Description



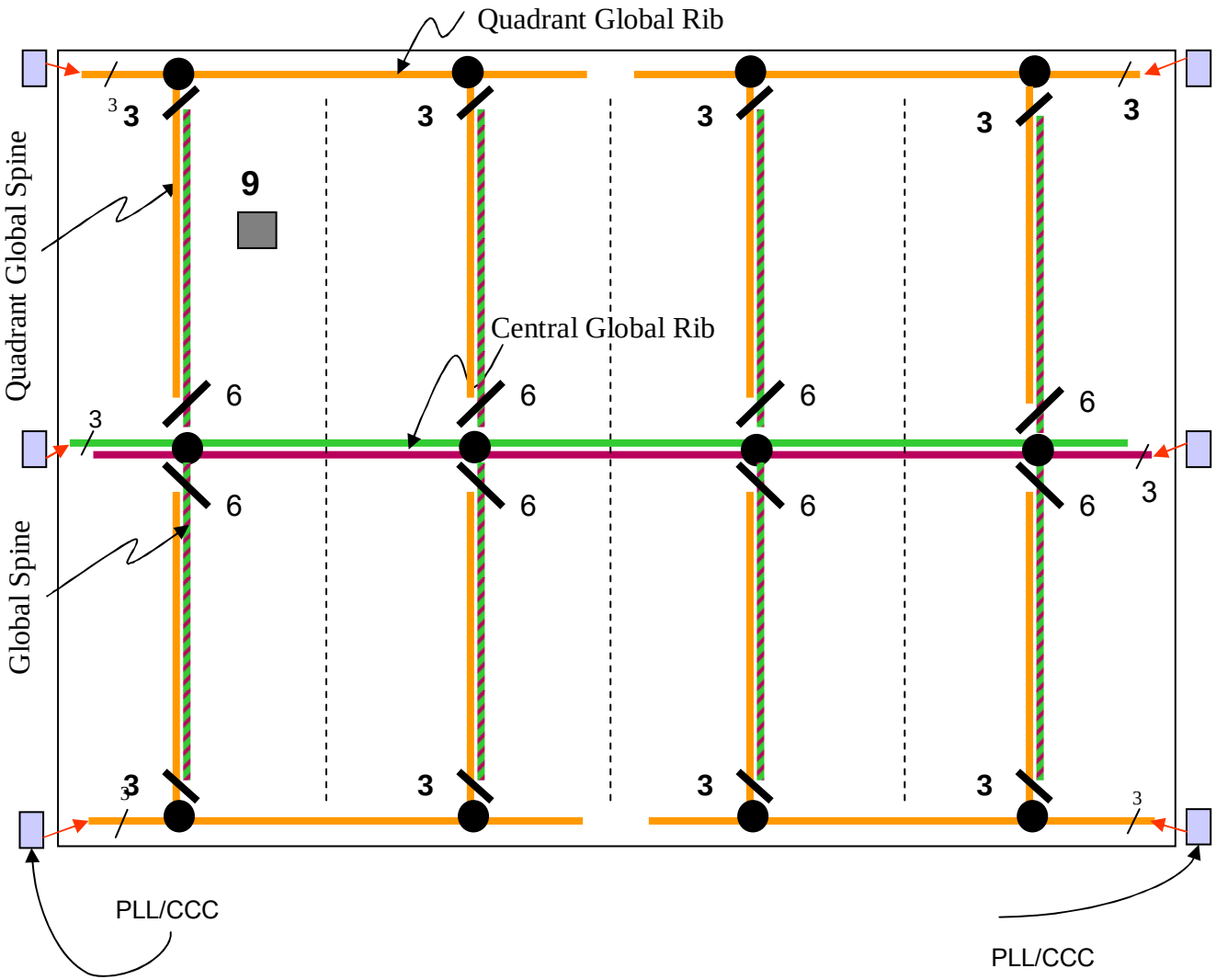
Signal Name	Direction	Function
CLK0	Input	Clock Input
CLK1	Input	Clock Input
S	Input	Mux Select 0 -> 1 CLK1 1 -> 0 CLK0
CLKOUT	Output	Clock Output

Fusion No-Glitch MUX
Operation

- Switching from CLK0 to CLK1:
  - GL will drive one last complete CLK0 positive pulse (i.e. one rising edge followed by one falling edge).
  - From that point GL stays low until the second rising edge of CLK1 occurs.
  - At the second CLK1 rising edge, GL will continuously deliver CLK1 signals.



Fusion: Global Distribution Network



- Left and Right CCCs Provide 6 Chip-wide Global Networks (Access from I/Os in Middle of Left and Right Sides)
- 12 Quadrant Global Networks (3 per Quadrant – Access from I/Os in 4 Corners)
- Each VersaTile Has Access to 9 Global Resources
- Access from PLLs and Internal Signals

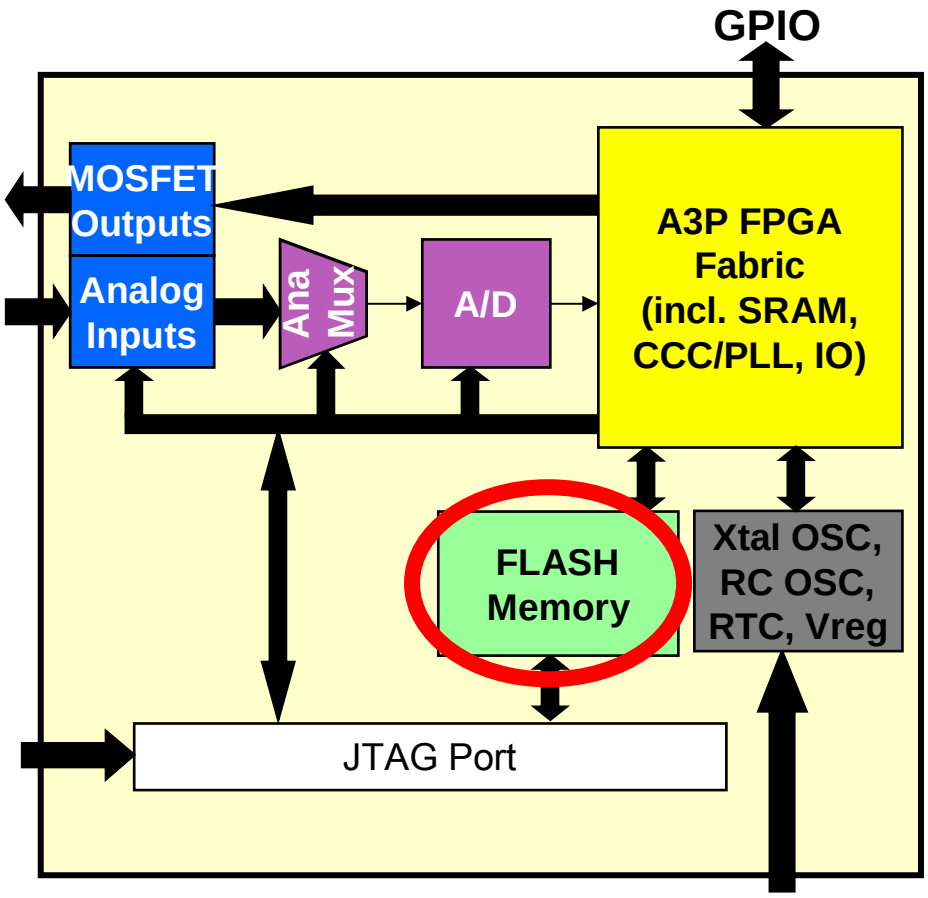
Fusion Peripherals

- Clocking
- Embedded Memory
  - Flash Memory Block
  - FlashROM
  - SRAM and FIFO
- Analog Block

Fusion

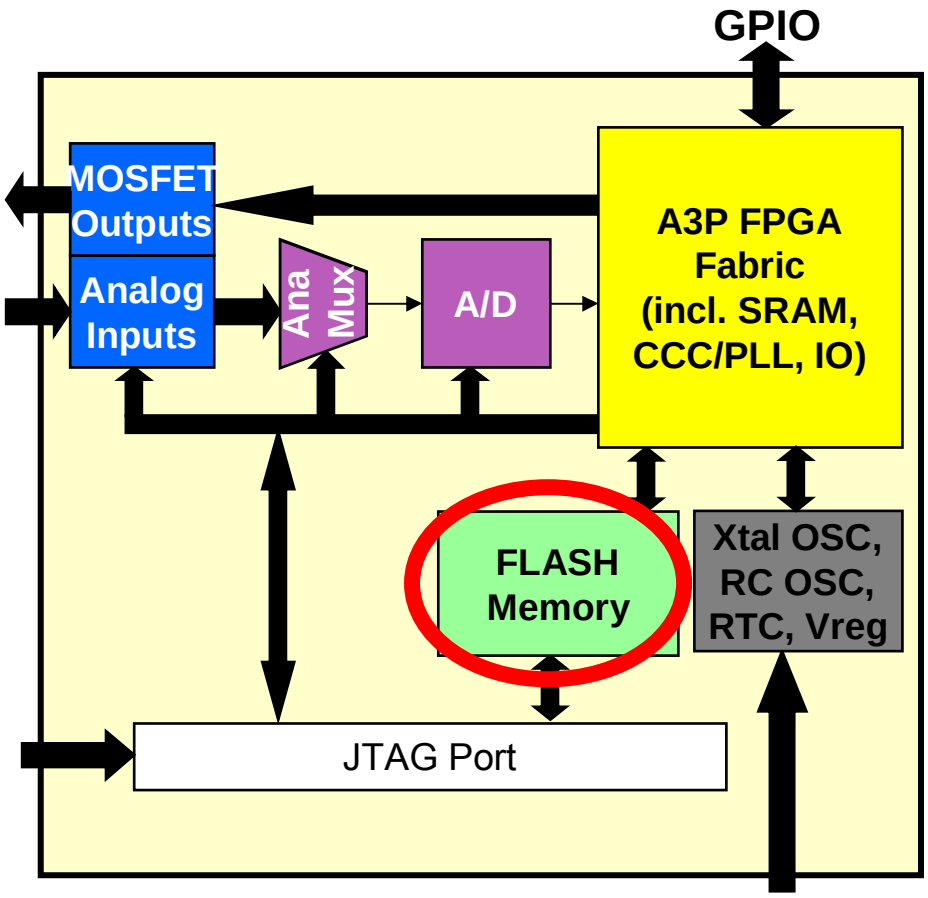
Only FPGA with Flash Memory

Fusion Flash Features



- Flash Memory 2 Mb Density
  - 1 – 4 blocks/device
  - Each 2 Mb array has independent controller
  - Independent JTAG access
- Flexible Operation
  - x8, x16, and/or x32 FPGA
  - Each supports multiple partitions
  - Small page size (1kb)
  - Can be accessed by either on-chip or off chip resources
- High Performance
  - 60 ns random access
  - Pipelined 10 ns access of sequential memory addresses

Fusion Flash Features (cont.)



- Flash Memory Level:
  - FPGA access
  - Password security
  - JTAG access for programming
- Page Level:
  - JTAG read / write protection
  - Program/erase
  - Partition on page boundaries
- Block Level Error Detect:
  - Single error correct
  - Double error detect

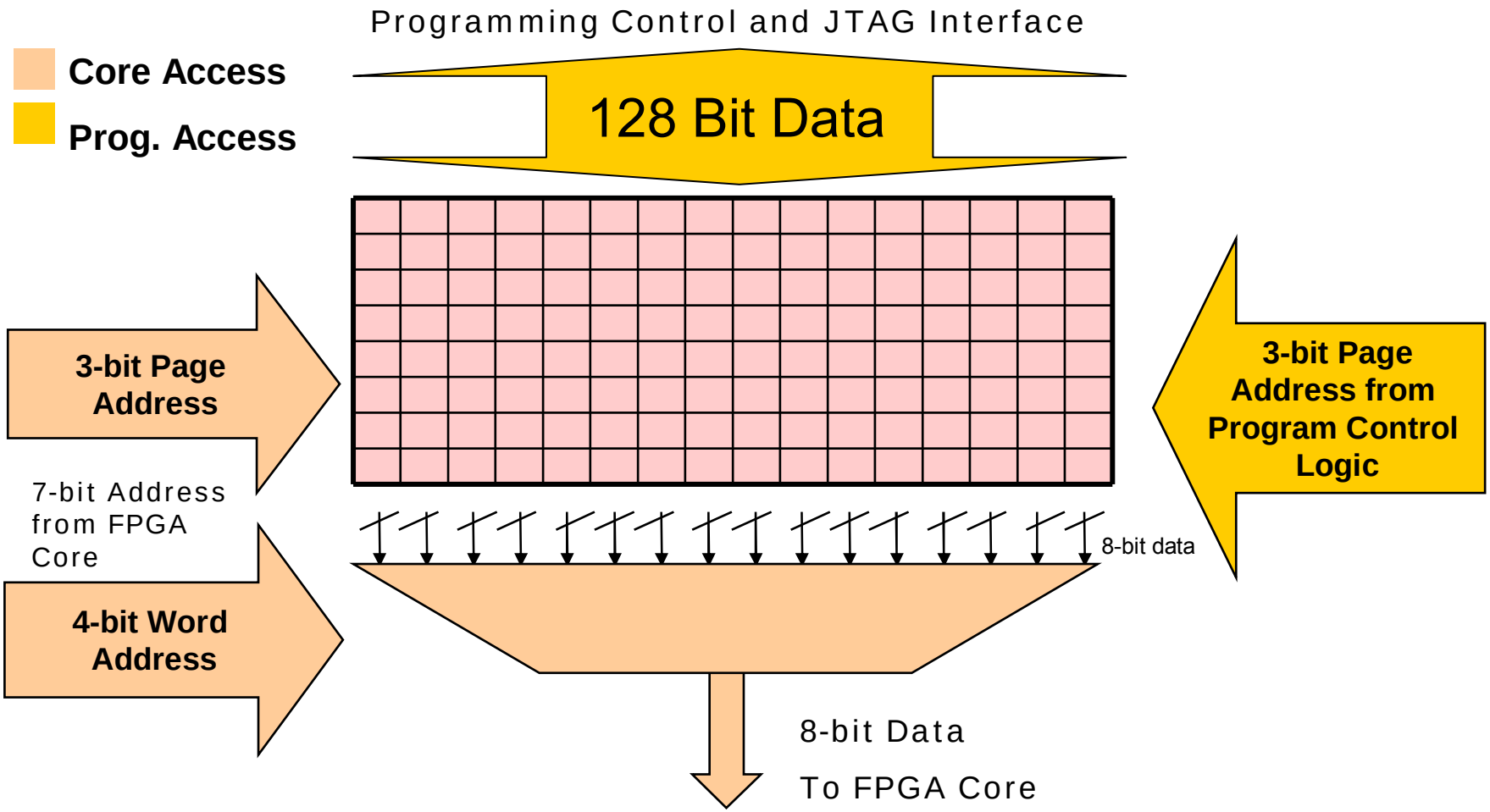
Fusion Peripherals

- Clocking
- Embedded Memory
  - Flash Memory Block
  - FlashROM
  - SRAM and FIFO
- Analog Block

FlashROM (FROM) Memory

- 8 pages of 128 bits (8x128)
  - Same as ProASIC3\E FROM
- FPGA Core and FlashROM Memory Can Be Programmed Separately
  - Allows Changing FROM without Erasing Core
  - Core Powered Down during FROM Programming
- Example Applications
  - IP Addressing
  - User/System Preference Storage
  - Device Serialization
  - Inventory Control
  - Subscription Models (Set-top Boxes)
  - Secure Key Storage
  - Presets
  - Date Stamping
  - Version Management

FlashROM
Logical View



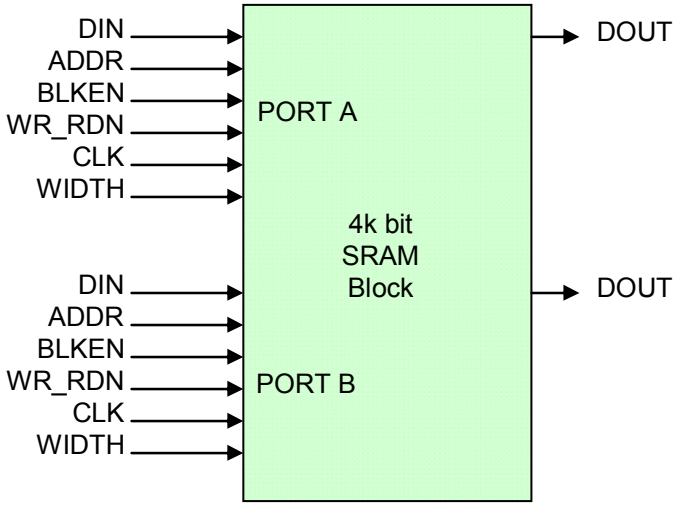
Every 128-bit Page Can Be Reprogrammed Independently

Fusion Peripherals

- Clocking
- Embedded Memory
  - Flash Memory Block
  - FlashROM
  - SRAM and FIFO
- Analog Block

Dual-Port SRAM Blocks

- Multiple 4K bit Embedded Memory Blocks:
  - 2 Write/Read ports OR independent 2-port
  - Synchronous operation up to 250 MHz
  - Fully programmable
    - Scalable aspect ratio from 256x18 to 4Kx1
    - Cascadable - wide and deep
    - SmartGen tool automates memory generation
- FIFO Capability
  - Decoder, FIFO control and flag logic built into RAM block
  - Programmable FIFO depth and flag threshold



Fusion SRAM Implementation

- True Dual-port RAM:
  - Variable Aspect Ratios – 4096x1, 2048x2, 1024x4 or 512x9
    - Independent Read and Write Port Widths
  - Dual-port Options – Both Read, Both Write, One Read & One Write; Same Clock Frequency or Two Different Clock Frequencies
  - Pass-through of Write Data or Hold Old Data on Output
- Two-port RAM:
  - Variable Aspect Ratios – 512x9 or 256x18
    - Independent Read and Write Widths
  - Dedicated Read and Write Ports
- Both Macros Have
  - Synchronous Write
  - Synchronous Read – Pipelined or Non-Pipelined
  - Asynchronous Output Reset

Fusion RAM
FIFO

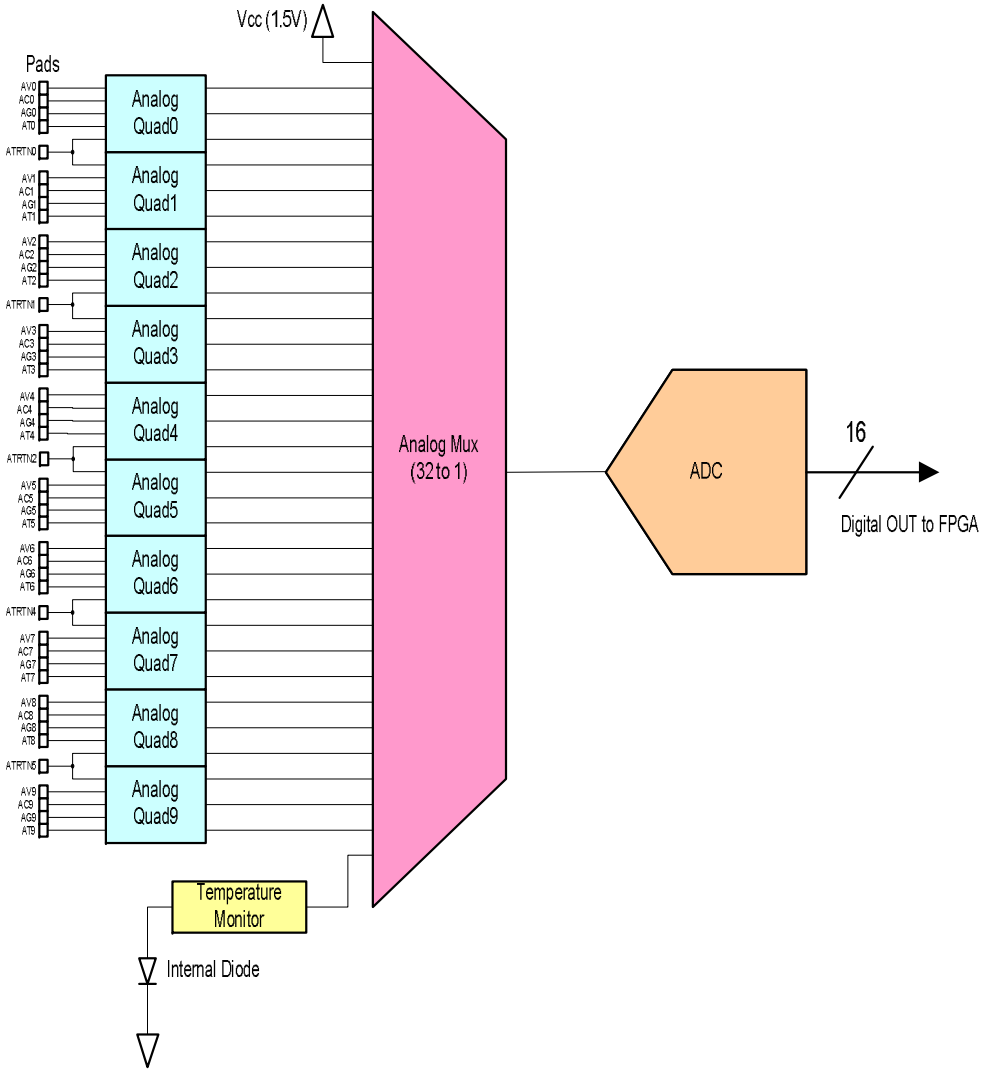
- Fusion Has One FIFO Macro:
  - Variable Aspect Ratios – 4096x1, 2048x2, 1024x4, 512x9, or 256X18
    - Independent Read and Write Port Widths
  - Four FIFO Flags – Empty, Full, Almost-empty, Almost-full
    - FIFO Empty/Full Flags Synchronized to Read Clock and Write Clock, Respectively
    - Programmable Threshold Values of 'Almost' Flags
  - Asynchronous Reset
  - Active-low Block Enable
  - Active-low Write Enable and Active-high Read Enable
  - FSTOP and ESTOP – FIFO Counters Can Count after FIFO Is Full or Empty
    - Allows Writing to FIFO Once and Repeatedly Reading Same Contents without Rewriting Contents

Fusion Peripherals

- Clocking
- Embedded Memory
  - Flash Memory Block
  - FlashROM
  - SRAM and FIFO
- Analog Block

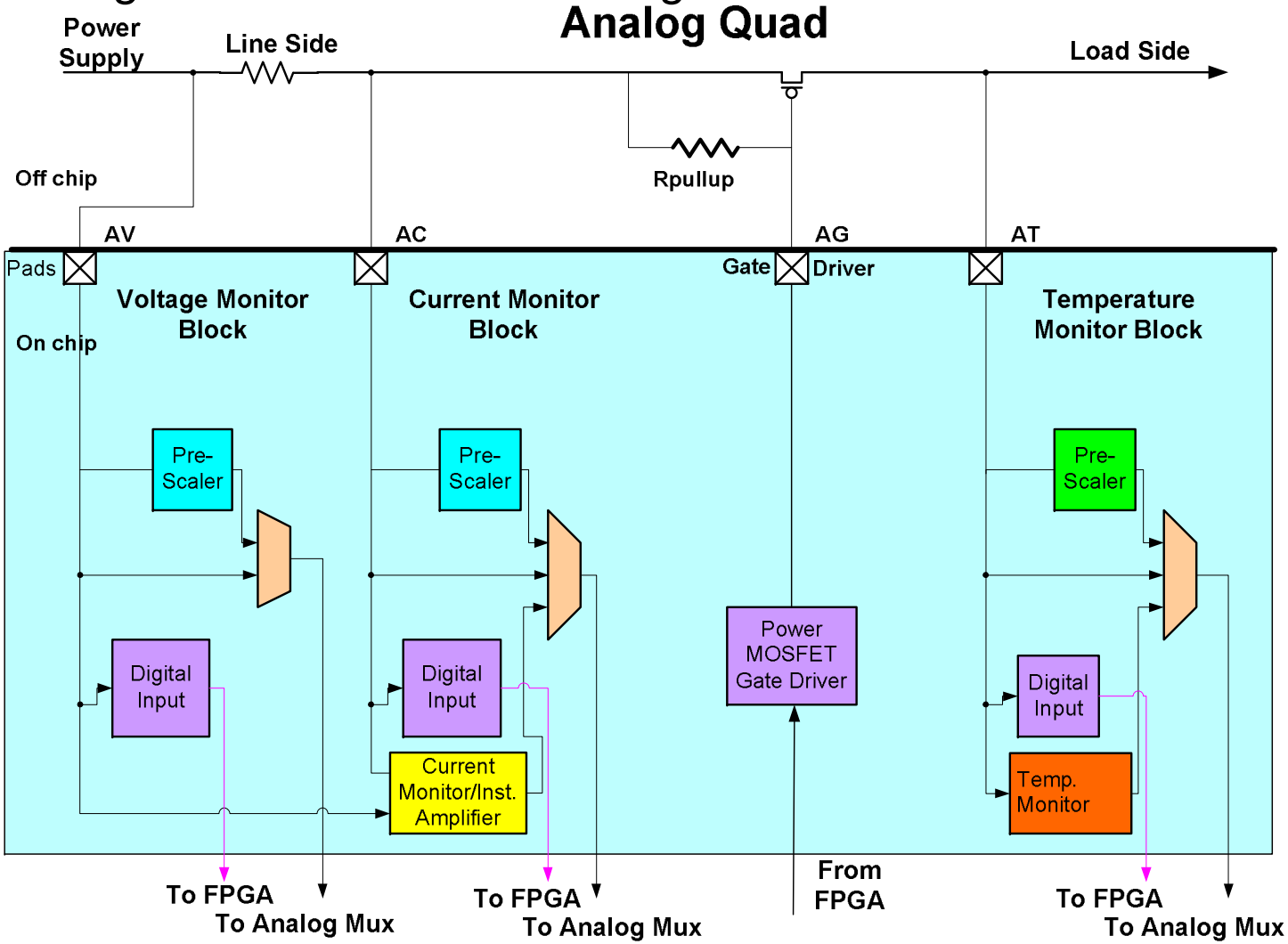
Fusion Analog Block

- 1.5V Voltage regulator
- -3.3V Voltage converter (for internal use only)
- Bandgap voltage reference
- Power System Monitor (Generates Flash and ADC Reference)
- Real Time Counter System (RTC)
- Analog Quad (up to 10 Analog Quads)
  - Voltage monitor block
  - Current Monitor block
  - Temperature monitor block
  - Gate control/driver block
- Analog MUX
- ADC
  - Selectable 8/10/12 bit resolution
  - 32 input channels
  - Up to 600K samples per second



Analog Quad (AQ)

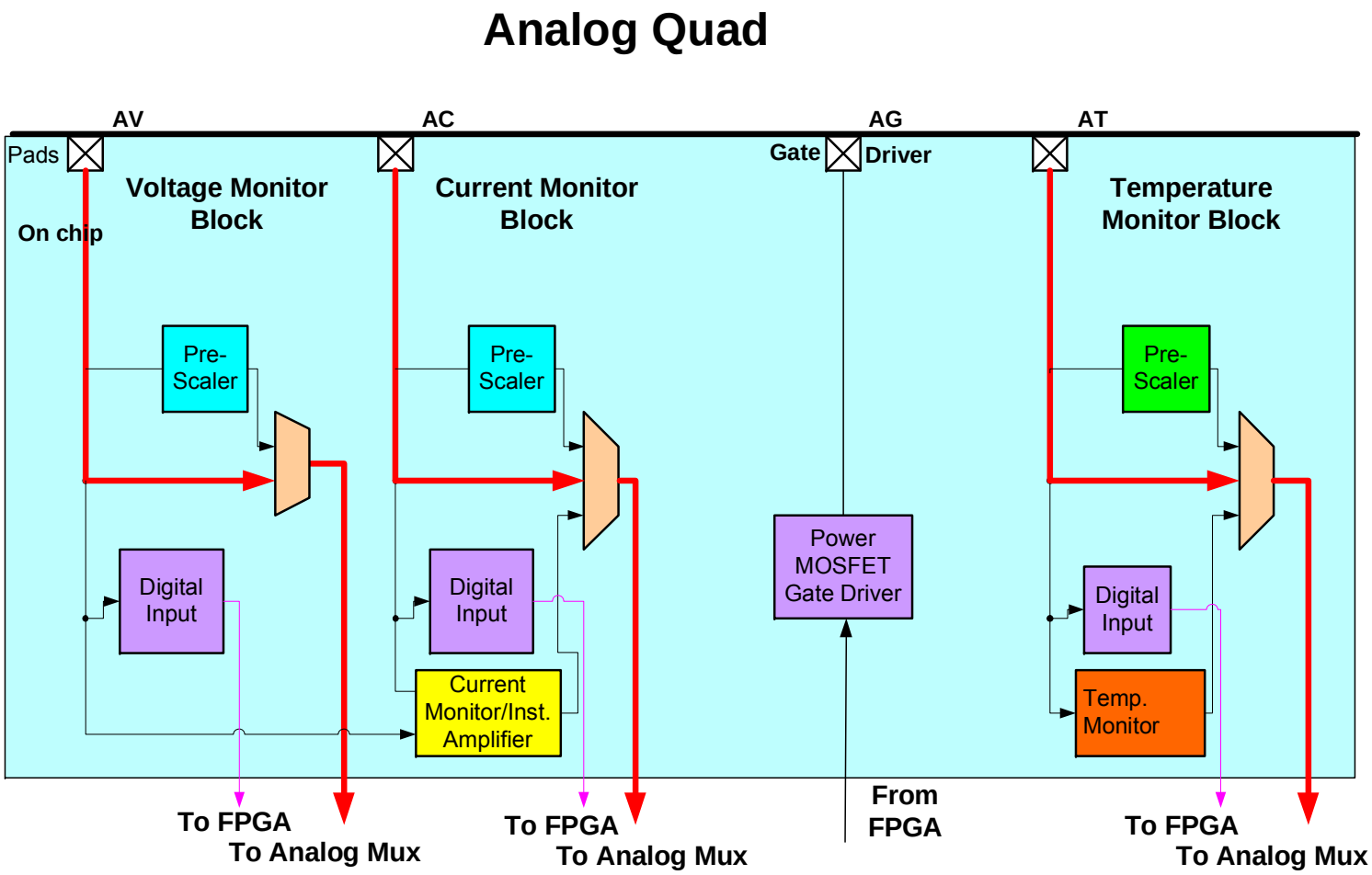
- Analog Quad is the Basic Analog I/O Structure



Fusion Analog Quad

- Features
  - Includes 4 analog interface pins
    - AV pin – Input: direct and prescaler voltage monitor
    - AC pin – Input: direct and prescaler voltage monitor, current monitor
    - AT pin – Input: direct and prescaler voltage monitor, temp monitor
    - AG pin – Output Power FET gate control or high voltage, high drive output
  - Input voltage range for AV and AC pads: - 10.5V to 0 V or 0V to 12V +/- 10%
  - Input voltage range for AT pad: 0V to 16V +/- 10%
  - AV, AC, AT pads can be used as low speed digital inputs (Tr, Tf > 20nS)
  - Modular building block used on all Fusion family members
    - AFS090: 5 Quads
    - AFS250: 6 Quads
    - AFS600, AFS1500: 10 Quads

Analog Quad: Direct Input

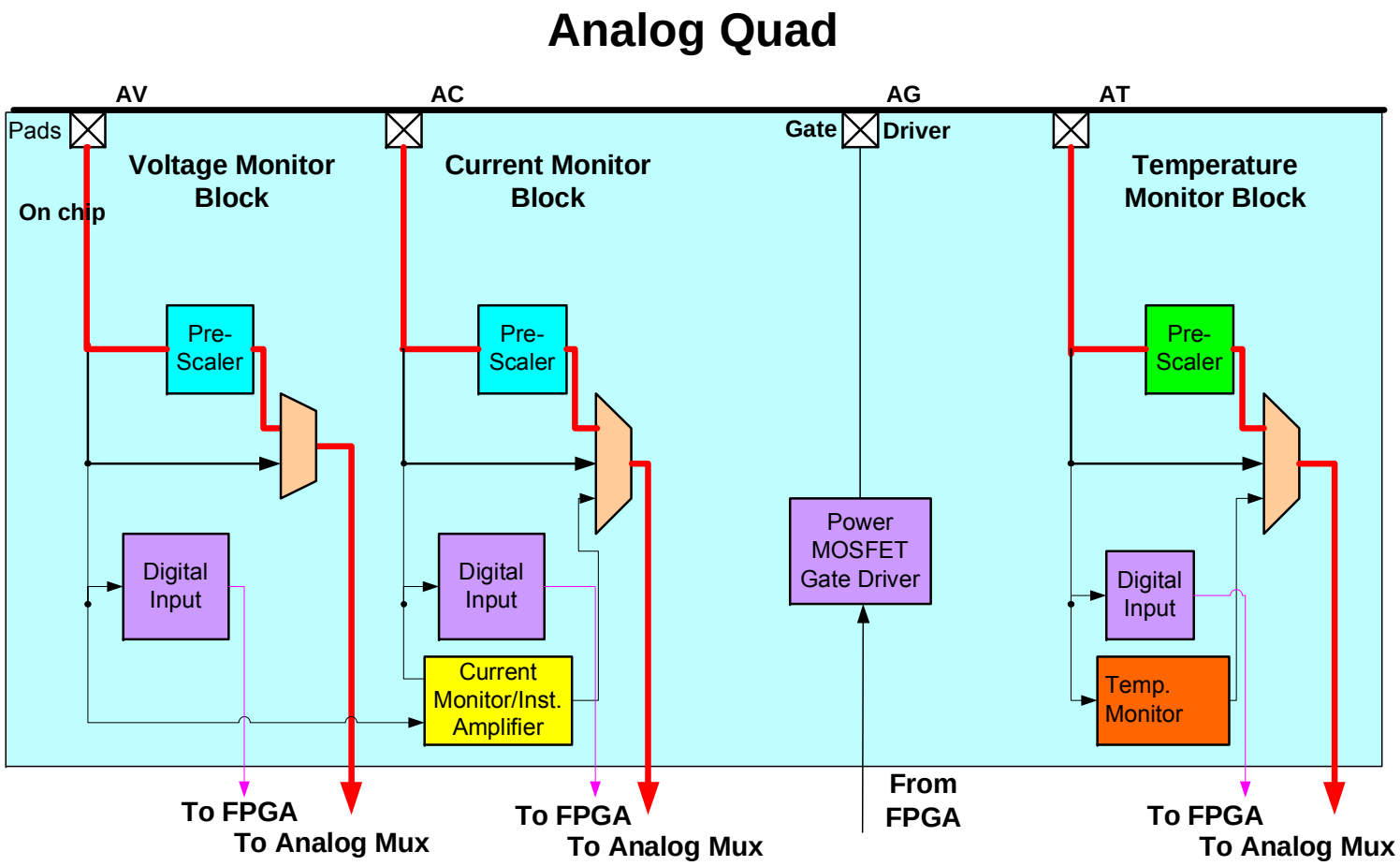


Direct Input used when Maximum Input Voltage is between 2V and V_{REF}

Analog Quad: Direct Input

- AV, AC, and AT pads can be Routed Directly to the Analog Mux by Configuring the Analog Quad
- Features
  - Best accuracy, lowest offset
  - No Buffers or Amplifiers between Input pin and ADC
  - No DC input current; capacitive load only (~20pF)
  - Resistive connection to ADC: ~4k ohm
  - One Input range: 0V to ADC reference level
  - Full scale level for internal reference is 2.56V
    - Decimal value of 8 most significant bits is input level in ‘centiVolts’ (i.e., ADC count of 207 = 2.07V at input)
  - Equivalent function to micro-controller style ADC
- Limitations
  - Positive input only
  - Single range
  - Needs low impedance input source for full bandwidth
  - Input capacitance varies greatly as input selection MUX is changed and Sample/Hold cycles

Analog Quad: Pre-scaler Input



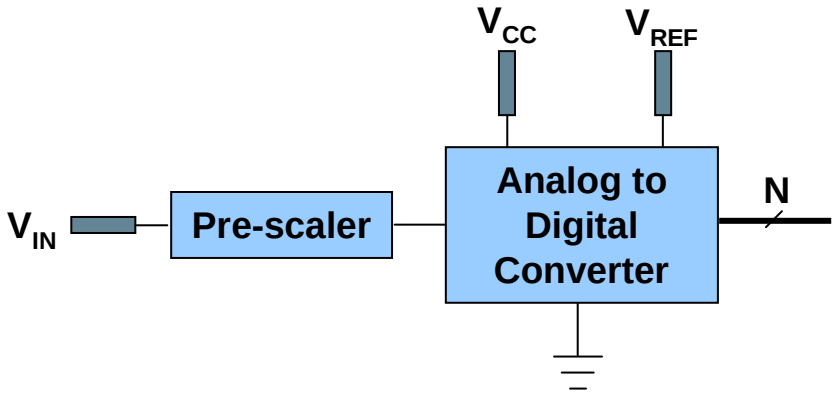
Pre-scaler Input used when Input Voltage is not between 2V and V_{REF}

Analog Quad: Pre-scaler Input

- Features
  - Input voltage ranges for AV and AC pads
    - Pre-Scaler input from -10.5V to 0V or 0V to 12V
      - Full scale ranges for ADC: ±16V, ±8V, ± 4V, ± 2V, ± 1V, ± 0.5V, ± 0.25V and ±0.125V
  - Input voltage ranges for AT pads
    - Pre-Scaler input from 0V to 16V
      - Full scale ranges for ADC: +16V and +4V
  - All ranges indicate input level in mV with a simple left or right shift of the binary value
    - On 4V range with 12 bit ADC setting, LSB = 1mV at input pin
  - Constant input impedance; ~1M ohm, 5pF
- Limitations
  - Pre-scale circuits add offset and gain error
  - Dynamic range changes are discouraged (will cause transient measurement errors on all channels)

Pre-Scaling

- Overdriving the ADC Input Can Result in:
  - Erroneous Conversions
  - Input Latch Up
    - Permanent Damage To Device Is Possible
- To Avoid Overdriving, The Input Can Be Pre-scaled To Ensure Maximum Voltage To ADC Does Not Exceed VREF
  - “Scale Down” Voltages That Are Greater Than VREF
  - “Scale Up” Voltages That Are Less Than VREF
  - Invert If Input Voltage Is Negative



Pre-Scaling (Cont.)

- Pre-scaler Value (Gain) is Chosen to Ensure (Max V_{IN} * G) ≤ V_{REF}
- Gain May Be Chosen Such That the LSB of the Converter Output is a Convenient Number
  - Example:
    For V_{REF} = 2.56V and V_{IN} = 6V, G = 0.4267 Ensures Input to ADC ≤ V_{REF}
    But . . .
    LSB of ADC equals the following:
    8 bit: 23.44 mV
    10 bit: 5.86 mV
    12 bit: 1.47 mV
  - Using A Different Gain Value Can Result In Simpler Calculations
- Limitations
  - Pre-scale Circuits Add Offset and Gain Error

Analog Quad: Pre-scaling Factors

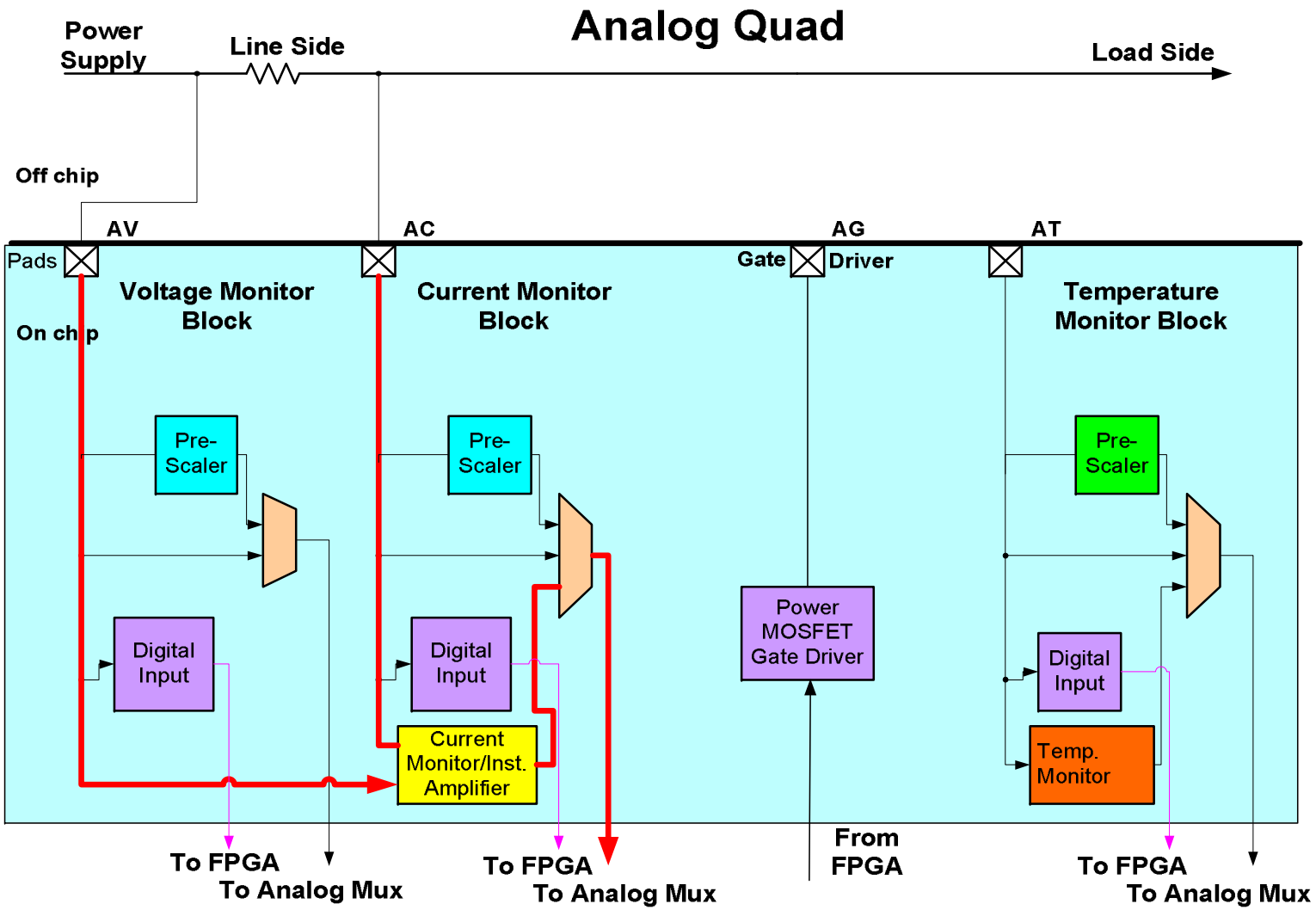
V_{IN}	Scaling Factor (G)	Full Scale Voltage (VREF / G)	LSB 8-bit conversion (mV)	LSB 10-bit conversion (mV)	LSB 12-bit conversion (mV)	Range Name
$12V \geq V_{IN} > 8V$	0.15625	16.368 V	64	16	4	16V
$8V \geq V_{IN} > 4V$	0.3125	8.184 V	32	8	2	8V
$4V \geq V_{IN} > 2V$	0.625	4.092 V	16	4	1	4V
$2V \geq V_{IN} > 1V$	1.25	2.046 V	8	2	0.5	2V
$1V \geq V_{IN} > 0.5V$	2.5	1.023 V	4	1	0.25	1V
$0.5V \geq V_{IN} > 0.25V$	5.0	0.5115 V	2	0.5	0.125	0.5V
$0.25V \geq V_{IN} > 0.125V$	10.0	0.25575 V	1	0.25	0.0625	0.25V
$V_{IN} \leq 0.125V$	20.0	0.127875 V	0.5	0.125	0.03125	0.125V

- Fusion Scaling Factors Give Convenient LSB Values For All ADC Resolutions
  - All Ranges Indicate Input Level in mV with Simple Left or Right Shift of Binary Value
- Maximum Allowable Input Voltage:
  - -10.5V or +12V for AV and AC pads
  - +16V for AT pad
- Pre-scaling Offset and Gain Error
  - Gain error
    - Positive DC inputs: 1% typ
    - Negative DC inputs: 2% typ
  - Offset error
    - 2 ± 0.2% of range

Analog Quad: Pre-scaler Input Examples

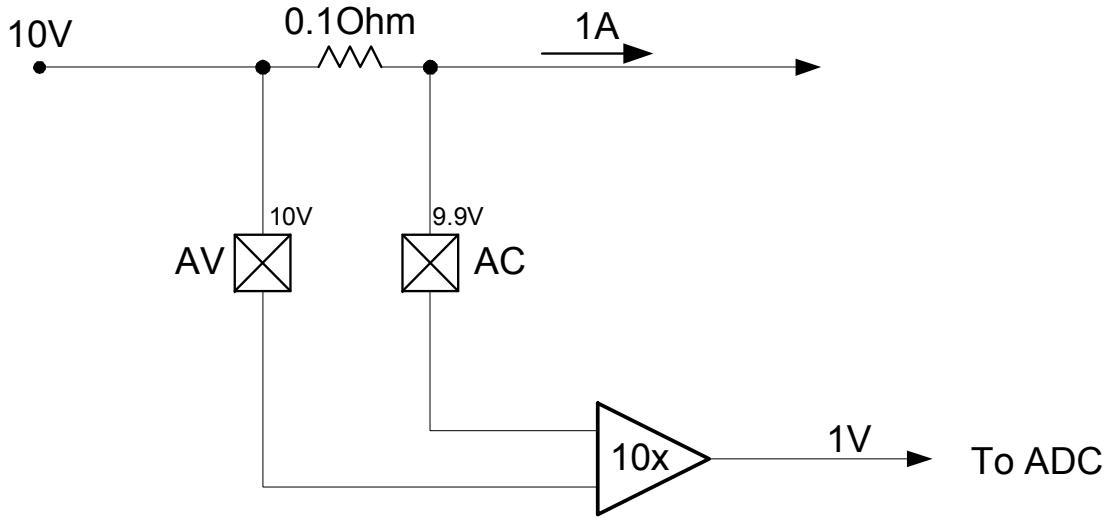
- *Example 1*
  - 8V Is Applied to AV Pad
    - Scaling Factor Is Set to 0.3125
    - Output of Pre-scaler (Input to ADC) is (8 * 0.3125) = 2.5V
    - For 12-bit Resolution, ADC output = 2^{12} * (2.5 / 2.56) = 4000
      - LSB = (2.56 / 0.3125) / 2^{12} = 2 mV
- *Example 2*
  - 12V Is Applied to AV Pad
    - Scaling Factor Is Set to 0.15626
    - Output of Pre-scaler (Input to ADC) is (12 * 0.15625) = 1.875V
    - For 12-bit Resolution, ADC output = 2^{12} * (1.875 / 2.56) = 3000
      - LSB = (2.56 / 0.15626) / 2^{12} = 4 mV
- *Example 3*
  - –0.2V Is Applied to AV Pad
    - Scaling Factor Is Set to 10
    - Output of Pre-scaler (Input to ADC) is (0.2 * 10) = 2.0V
      - Input signal is inverted to make ADC input positive
    - For 12-bit Resolution, ADC output = 2^{12} * (2.0 / 2.56) = 3200
      - LSB = (2.56 / 10) / 2^{12} = 0.0625 mV

Analog Quad: Current Monitor



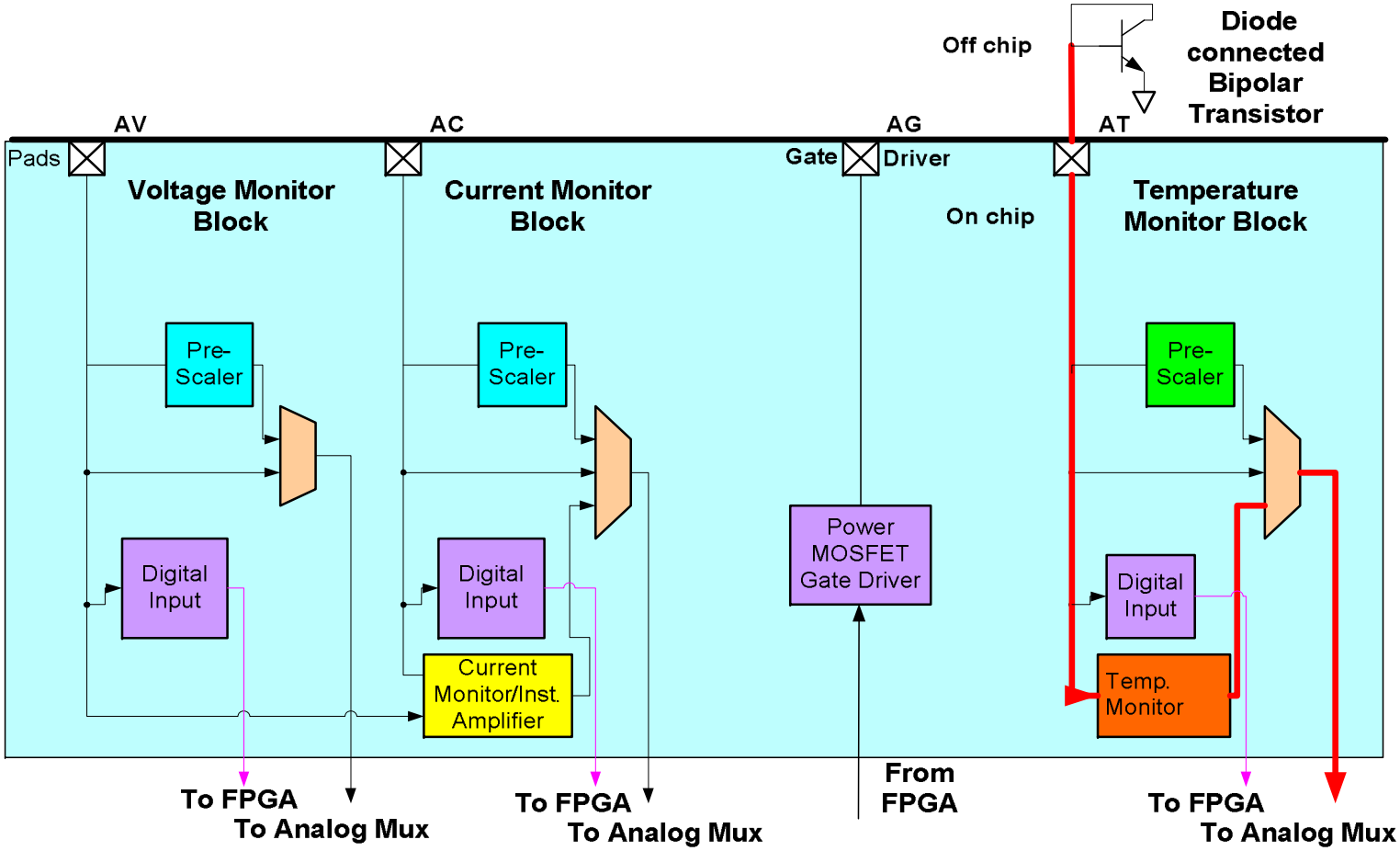
Analog Quad: Current Monitor

- Can sense Current by Measuring Voltage Drop Across an External Resistor



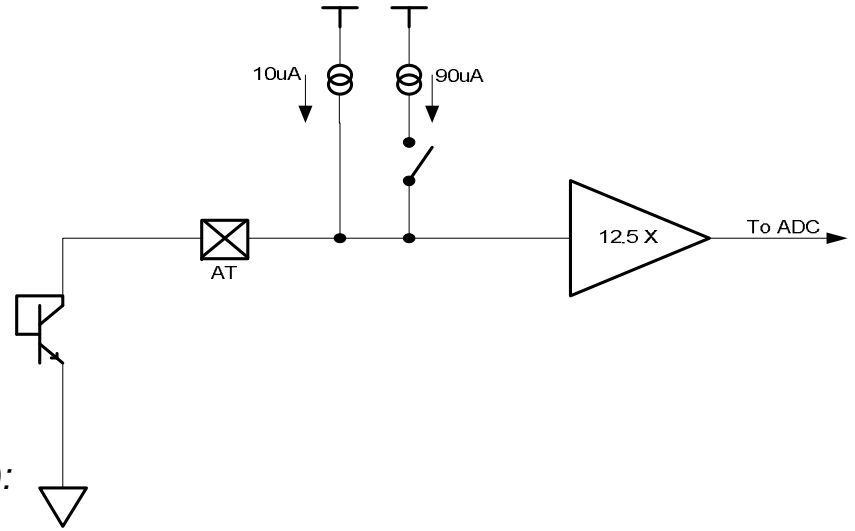
- If the Polarity bit is set to '0' (positive):
  - Current monitor output will be = 10 * (AV-AC)
- If the Polarity bit is set to '1' (negative):
  - Current monitor output will be = 10 * (mag(AV)-mag(AC))
    - Note: ADC input should be positive only

Analog Quad: Temperature Monitor



Analog Quad: Temperature Monitor

- Can Monitor Temperature of External Transistor Connected as Diode
  - Basic accuracy of 5 degrees C
  - Output count of ADC reads directly as absolute temperature (K)

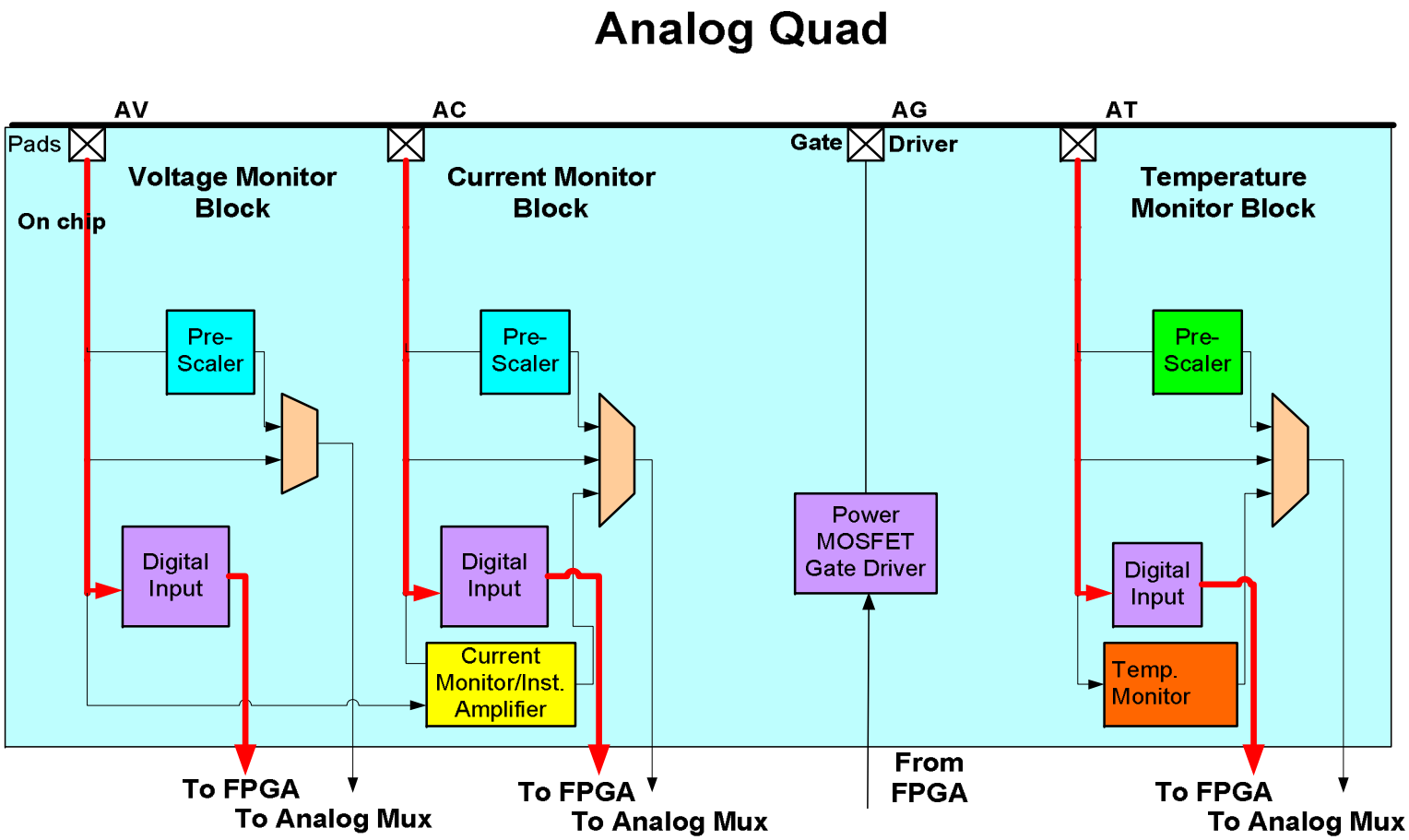


- *Example: @ room temperature (25°C = 298.15°K):*
  - *Diode equation –*
    *V = (KT/q) * ln(I1/I2)*

    Where I1/I2 is 10 and K/q is 86.258x10^{-6}

    V = 59.2mV
- *After gain of 12.5 voltage to ADC will be = 59.2mV * 12.5 = 740mV*

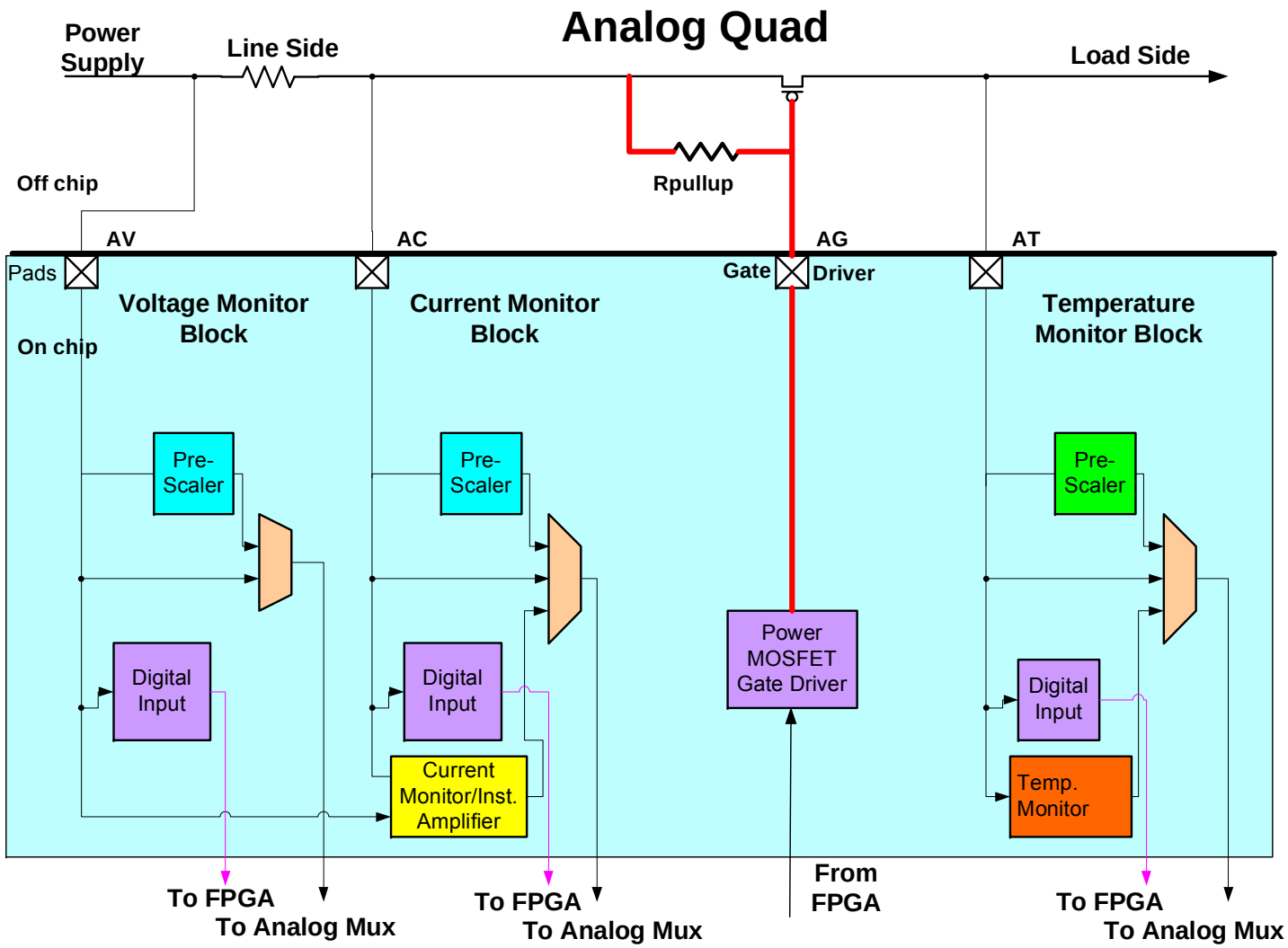
Analog Quad: Direct Digital Input



Analog Quad: Direct Digital Input

- AV, AC, and AT pads can be used as low speed Digital Inputs (LVTTL)
  - Operating speed – 10MHz (Max)
  - (Tr, Tf > 20nS)
- Delay – 10nS (Typ)
- The Digital Buffers can be Disabled if not used

Analog Quad: Gate Driver

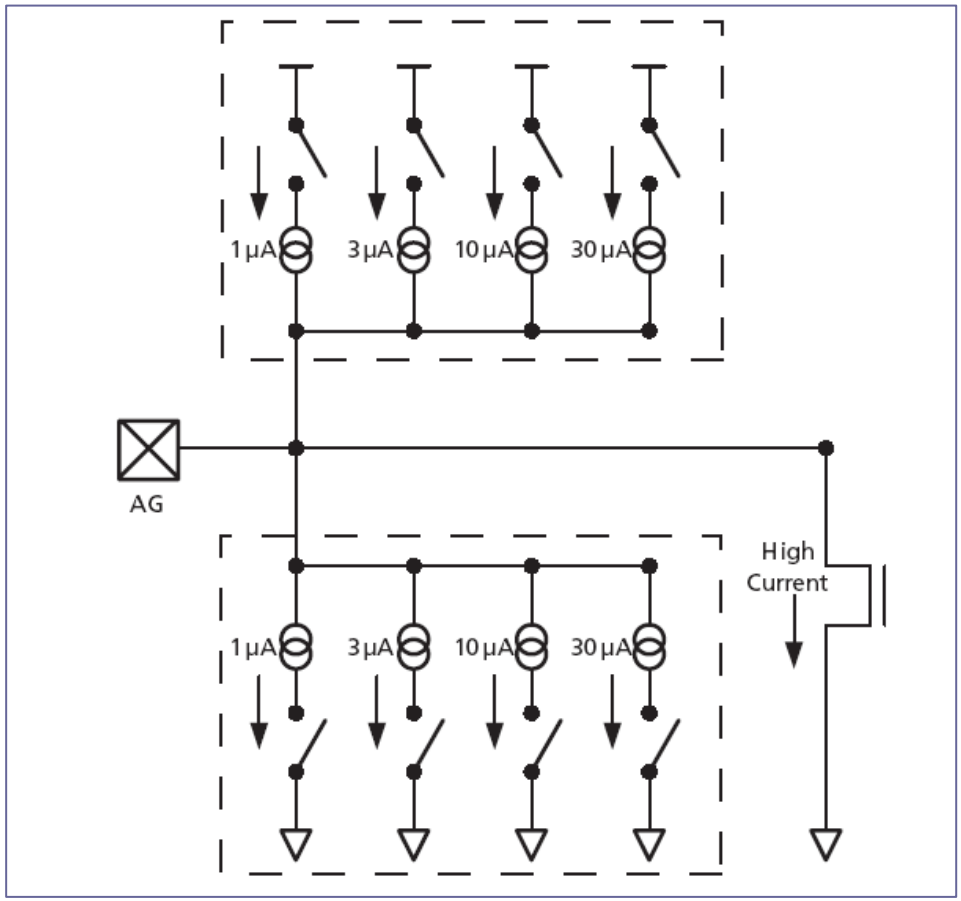


Analog Quad: Gate Driver

- Features
  - Controls turn on of external Power FET by pulling the gate towards ground
  - FET Vgs limited to Ig * Rpullup
  - Slew rate of load controlled by Ig/Cgd = dV/dt
  - Works for positive supplies (with P-FET) and negative supplies (with N-FET)
  - High drive mode can sink/source 25mA
- Limitations
  - Requires external pullup resistor
  - Open drain style output. Does not output a voltage level

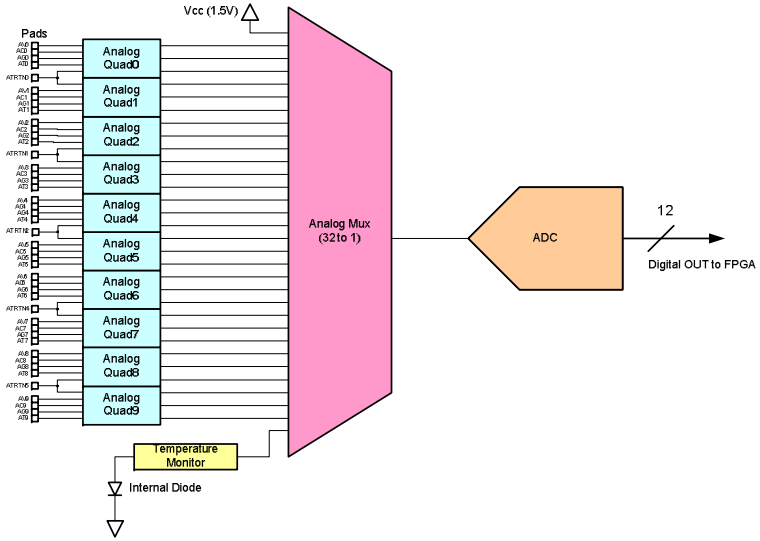
Analog Quad: Gate Driver

- Gate Driver has two Modes:
  - High current drive - 25mA @ 1V
  - Low current drive – choose between 1, 3, 10 and 30uA



Fusion Analog MUX

- Input to the Fusion ADC is a 32:1 Analog MUX
- 30 Input Channels are User Definable
  - Connected to Analog Quad AV. AC and AT inputs
- Two Channels are Hardwired Internally
  - Channel 0 is wired to the FPGA's 1.5 V supply
    - Fusion device can monitor its own power supply
  - Channel 31 connects to an internal temperature diode
    - Monitor temperature of the Fusion device
- Analog Block Input **`CHNUMBER[4:0]`** Selects Channel for Conversion



Fusion ADC

- 8/10/12 bit Selectable Resolution
- 32 Input Channels
- Up to 600K samples/sec
- TUE (Total Unadjusted Error)
  - +/- 2 LSB in 8 bit mode
  - +/- 4 LSB in 10 bit mode
  - +/- 6 LSB in 12 bit mode
- Clock
  - ADC interface clock max frequency – 100MHz
  - Selectable clock divider (divide by 4 to 1024) to generate ADC internal clock
  - ADC internal clock frequency range – 1 – 10MHz
- Sample and Hold
  - Selectable sample time 2 – 257 * ADC internal clock period
- Self Calibration
  - Automatic full calibration on powerup
  - Optional incremental calibration after each conversion
- Status Signals:
  - Conversion in progress
  - Sampling in progress
  - Calibration in progress
  - Data valid
- Power Requirements
  - 3.3V for Analog, 1.5V for Digital, and reference voltage (2.56V to 3.3V)

Analog Block Summary

- Up to 10 Analog Quads per Device
  - Pre-scaler input range:
    - -10.5 V to 0V or 0V to 12V for AV and AC inputs
    - 0 to 16V for AT input
  - Voltage Monitor
  - Current Monitor
  - Temperature Monitor
  - Gate Driver – For controlling Power MOSFETs
- 32 Input Channel ADC with Selectable Resolution (8/10/12 bit)
- Selectable Internal/External Voltage Reference
- 1.5V Voltage Regulator
- Real Time Counter (RTC)

Fusion: I/O Overview

- Fusion I/Os Are Organized in Banks Supporting Multiple Standards
  - 1.5 V, 1.8 V, 2.5 V and 3.3 V
- Common I/O Features
  - Programmable Slew Rate
  - Programmable Drive Strength
  - Weak Pull-up / Pull-down
- I/Os Power Up in Known State
  - No special power up sequencing is required

Fusion: I/O Functions

- Regular I/Os
  - Input, Output, Tristate and Bidirectional Buffers
- Registered I/Os
  - Built-in Input, Output and Output-Enable Registers
    - Each Register Equivalent to 1-tile Core Flip-flop
- DDR I/Os
  - Built-in Input and Output DDR Registers

Fusion I/O Bank Types

- Hot Swap Bank
  - Support for Single-ended I/O Standards
    - LVTTL, LVCMOS
  - Hot Swappable
  - 3 Drive Strengths, Weak Pull-up / Pull-down Circuits
  - DDR Transmit / Receive
- LVDS Bank
  - Support for Single-ended and Differential I/O Standards
    - Single-ended
      - LVTTL, LVCMOS
      - PCI, PCI-X
    - Differential
      - High-Speed 700Mb/s LVDS with External Resistors
      - LVPECL I/O
  - 2 Programmable Slew Rates, 6 Drive Strengths, Weak Pull-up / Pull-down Circuits
  - DDR Transmit / Receive
  - No Hot-swap Capability

Fusion I/O Bank Types (cont.)

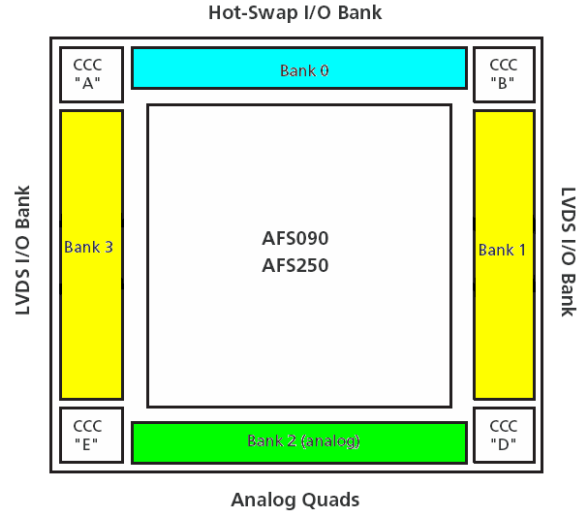
- Pro I/O Bank
  - LVDS Bank I/O Standards Support PLUS …
  - … Voltage-Referenced I/O Standards
    - HSTL1
    - SSTL2/3
    - GTL+
  - 2 Programmable Slew Rates, 6 Drive Strengths, Weak Pull-up / Pull-down Circuits
  - DDR Transmit / Receive
  - Hot-Swappable
- Analog I/O Bank
  - Analog Quad I/O Structure
  - Can be used for low speed digital input signals

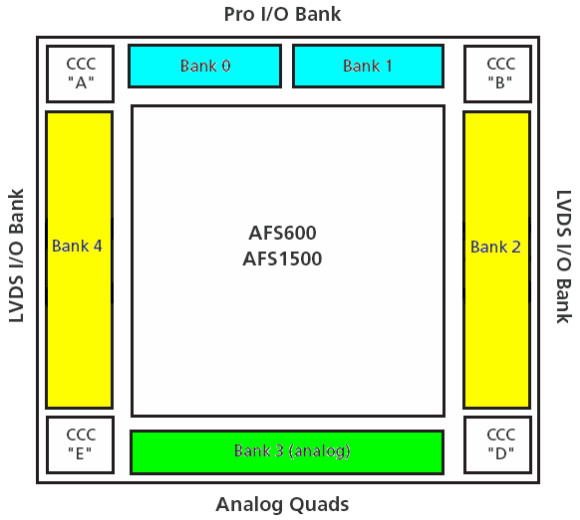
Fusion I/O: I/O Bank Type Summary

I/O Bank	Single-Ended I/O Standard	Differential I/O Standard	Voltage-Referenced	Hot-Swap
Hot-Swap	LVTTL/LVCMOS 3.3 V, LVCMOS 2.5 V/1.8 V/ 1.5 V, LVCMOS2.5/5.0 V	–	–	Yes
LVDS	LVTTL/LVCMOS 3.3 V, LVCMOS 2.5 V/1.8 V/ 1.5 V, LVCMOS2.5/5.0 V, 3.3 V PCI/3.3 V PCI-X	LVPECL and LVDS	–	–
Pro I/O	LVTTL/LVCMOS 3.3 V, LVCMOS 2.5 V/1.8 V/ 1.5 V, LVCMOS2.5/5.0 V, 3.3 V PCI/3.3 V PCI-X	LVPECL and LVDS	GTL+ 2.5 V / 3.3 V, GTL 2.5 V / 3.3 V, HSTL Class 1 and 2, SSTL2 Class 1 and 2, SSTL3 Class 1 and 2	Yes

Fusion: I/O Banks per Device

- AFS090 / AFS250
  - Four I/O Banks:
    - 1 Hot-swap Bank - North side of chip
    - 2 LVDS Banks – East, West sides
    - 1 Analog Bank – South side
- AFS600 / AFS1500
  - Five I/O Banks:
    - 2 Pro I/O Banks - North side of chip
    - 2 LVDS Banks – East, West sides
    - 1 Analog Bank – South side





LVDS / Pro I/O Banks: Output Drive and Slew Rate

- LVDS Bank

I/O Standards	OUT_DRIVE (mA)							
	2	4	6	8	12	16	Slew	
LVTTL/LVCMOS 3.3 V	✓	✓	✓	✓	✓	✓	High	Low
LVCMOS 2.5 V	✓	✓	✓	✓	✓	–	High	Low
LVCMOS 1.8 V	✓	✓	✓	✓	–	–	High	Low
LVCMOS 1.5 V	✓	✓	–	–	–	–	High	Low

- Pro I/O Bank

I/O Standards	OUT_DRIVE (mA)							Slew	
	2	4	6	8	12	16	24		
LVTTL/LVCMOS 3.3 V	✓	✓	✓	✓	✓	✓	✓	High	Low
LVCMOS 2.5 V	✓	✓	✓	✓	✓	✓	✓	High	Low
LVCMOS 2.5 V/5.0 V	✓	✓	✓	✓	✓	✓	✓	High	Low
LVCMOS 1.8 V	✓	✓	✓	✓	✓	✓	–	High	Low
LVCMOS 1.5 V	✓	✓	✓	✓	✓	–	–	High	Low

Fusion: I/O Banks and User I/O Counts

	Part #	AFS090	AFS250	AFS600	AFS1500
I/O	I/O Types	Analog / LVDS / Std+	Analog / LVDS / Std+	Analog / LVDS / Pro	Analog / LVDS / Pro
	I/O Banks (+ JTAG)	4	4	5	5
	Max Digital I/O	73	114	172	278
	Analog I/O	20	24	40	40
I/O: digital / analog	QN108	36/14			
	QN180	48/20	62/24		
	PQ208		93/24	95/40	
	FG256	73/20	114/24	119/40	119/40
	FG484			172/40	228/40
	FG676				278/40

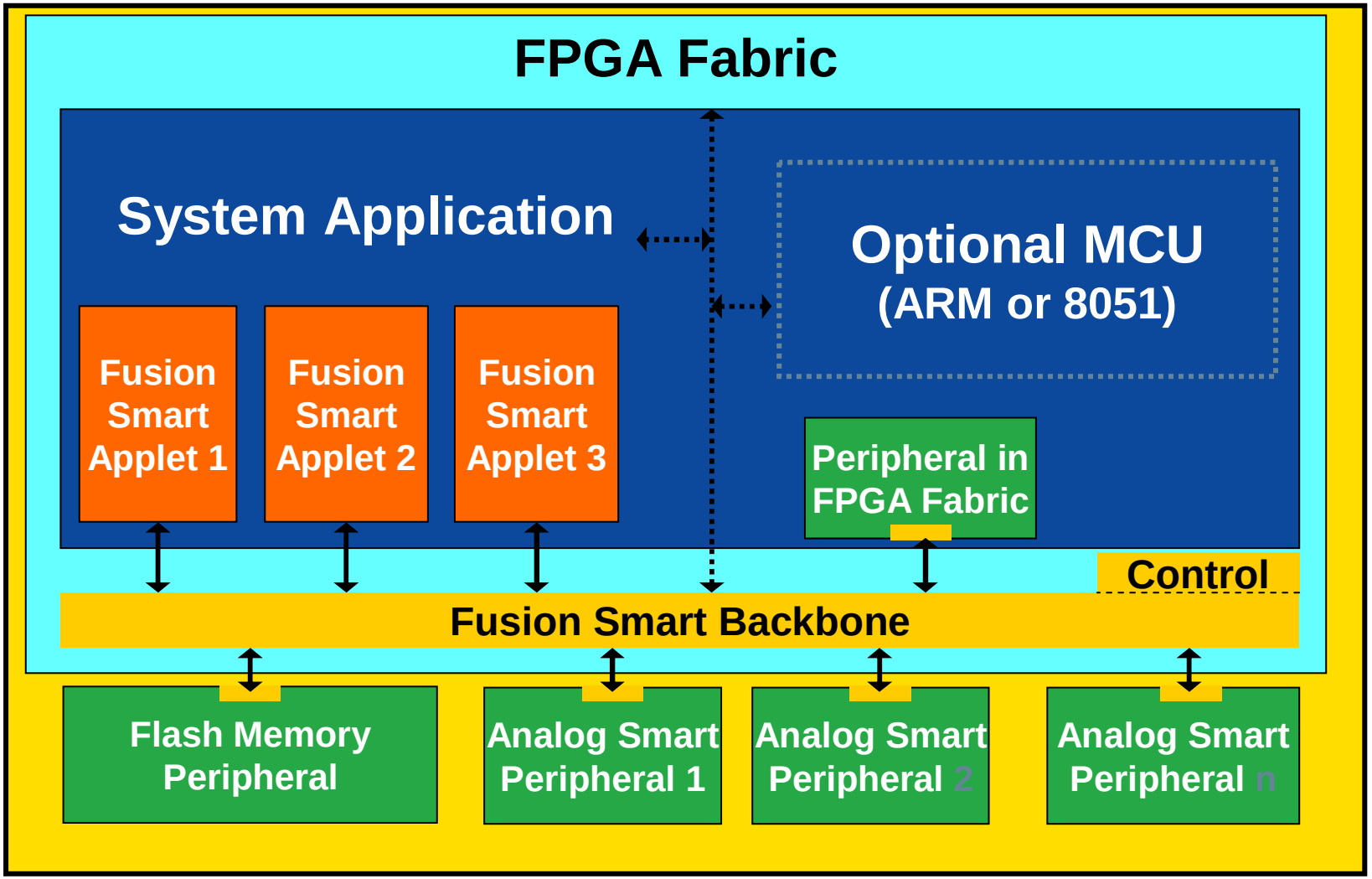
Agenda

- Fusion Overview
- Fusion Architecture
- Fusion Design Flow
- Development Support

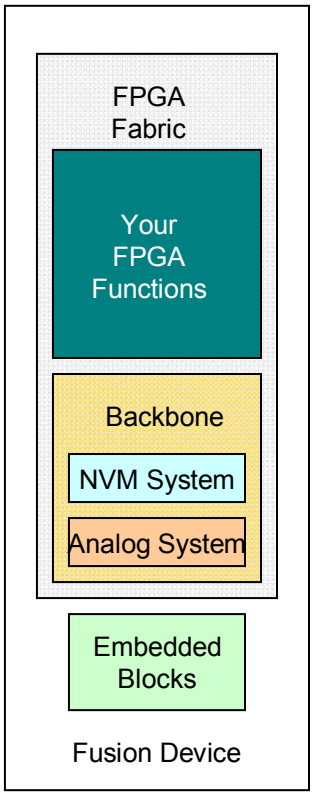
Fusion Design Flow

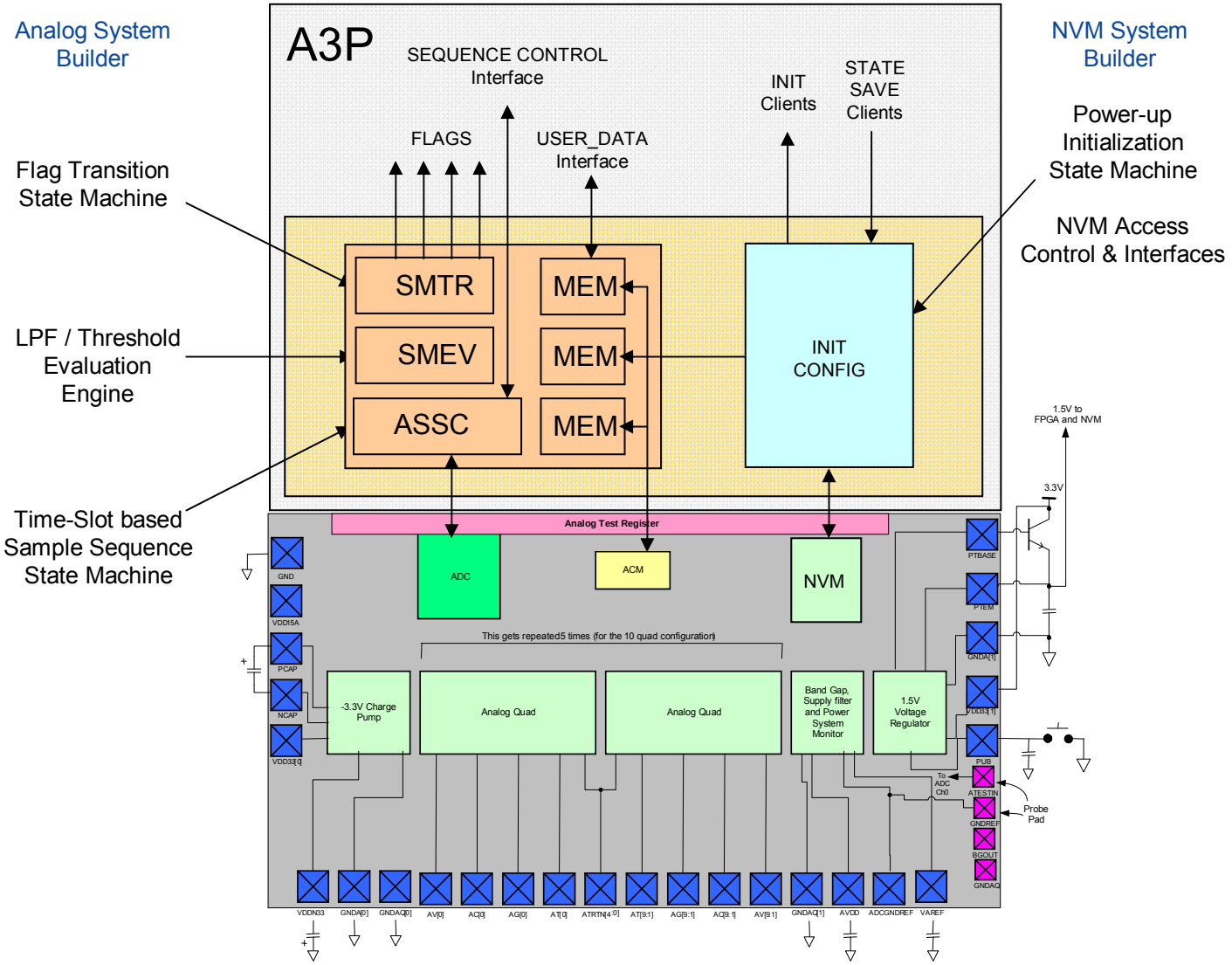
- Fusion Smart Backbone
- Libero Flow
- SmartGen
- Simulation
- Programming and Security

Actel Fusion Silicon: Physical View



The Fusion Backbone

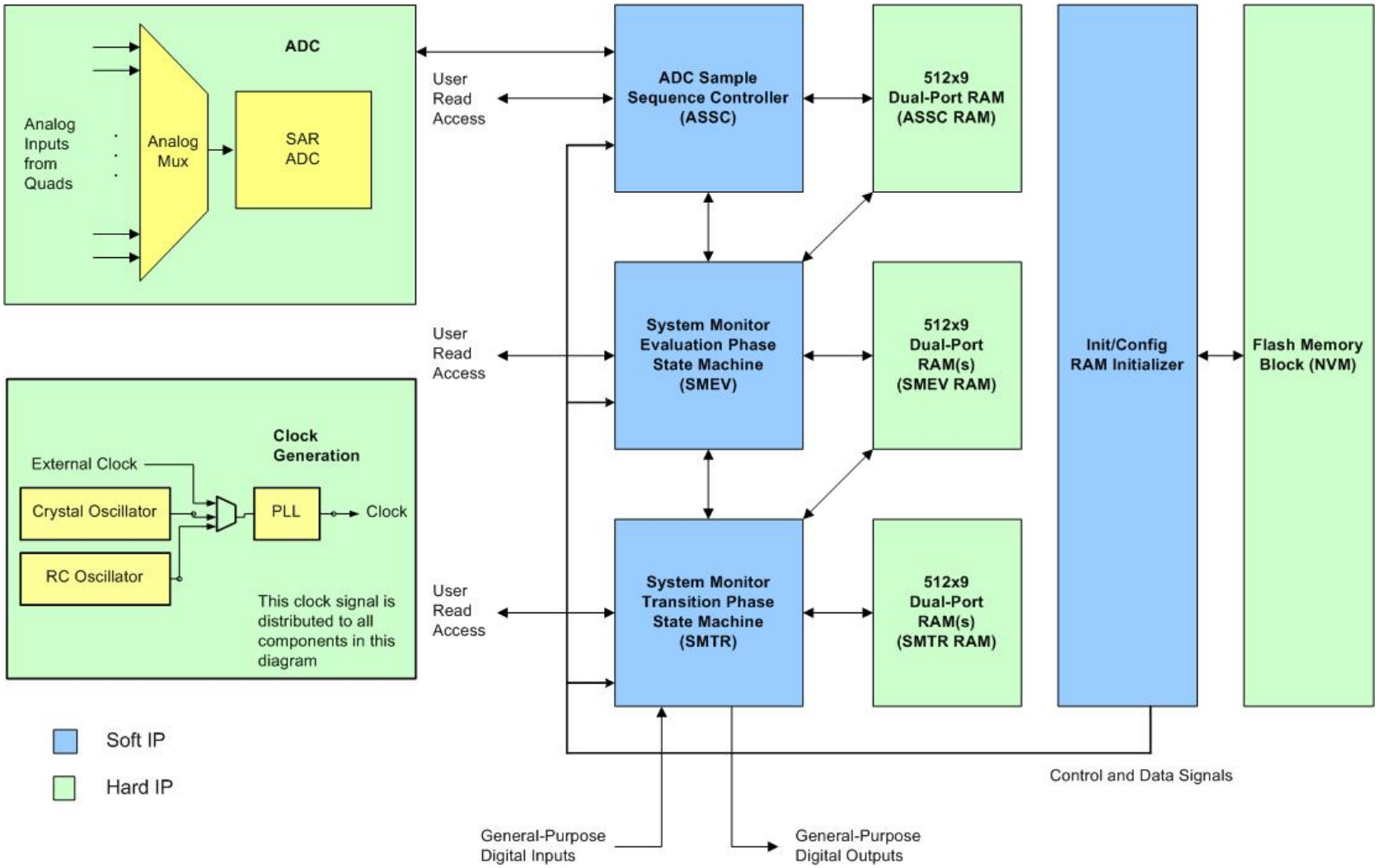




System Overview

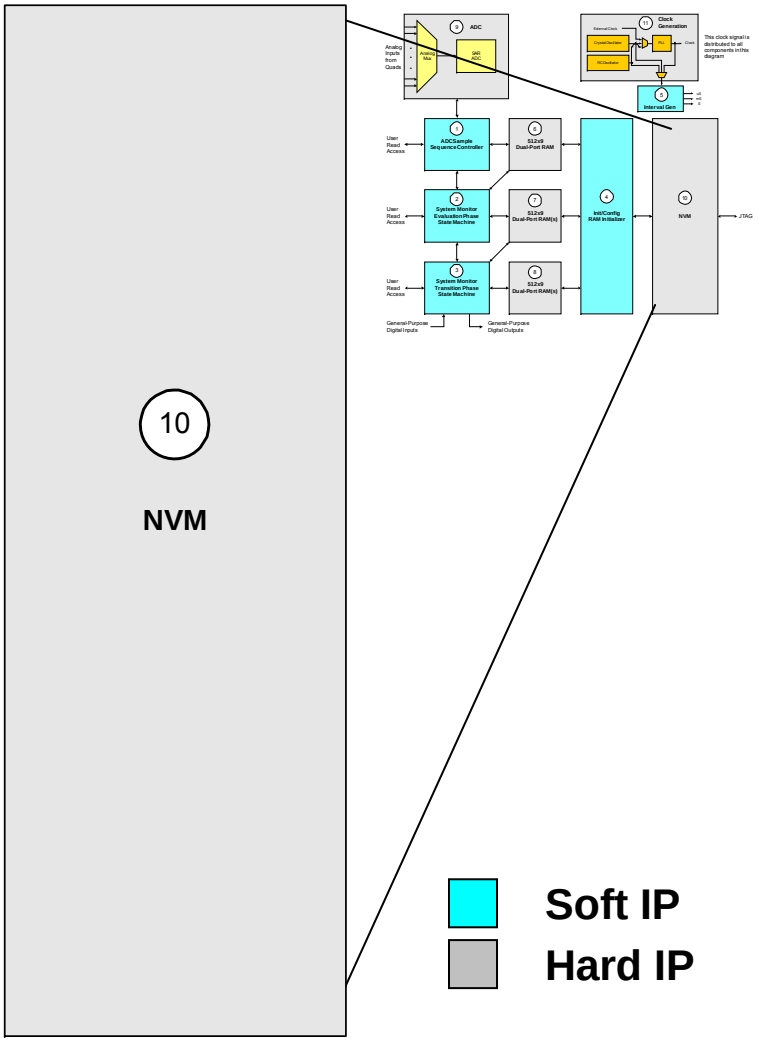
- Soft IP Role in System
  - “Glue” That Binds System Components Together
  - Main Control for Analog System
    - Sampling
    - Digital low-pass filtering
    - Threshold comparisons
    - State filtering
    - GPI/GPO
  - Makes microcontroller dependency unnecessary (self-sustained)
- Soft IP Configuration
  - All configuration within the Analog I/F Soft IP is controlled via setting top-level Generics/Parameters and propagating these values through hierarchy
    - Allow user read access to RAMs, RAM address space size, enable DLPF, control Current/Temperature monitoring functions, Declare number of GPI/GPO signals, etc.

System Overview
Hard/Soft IP Blocks



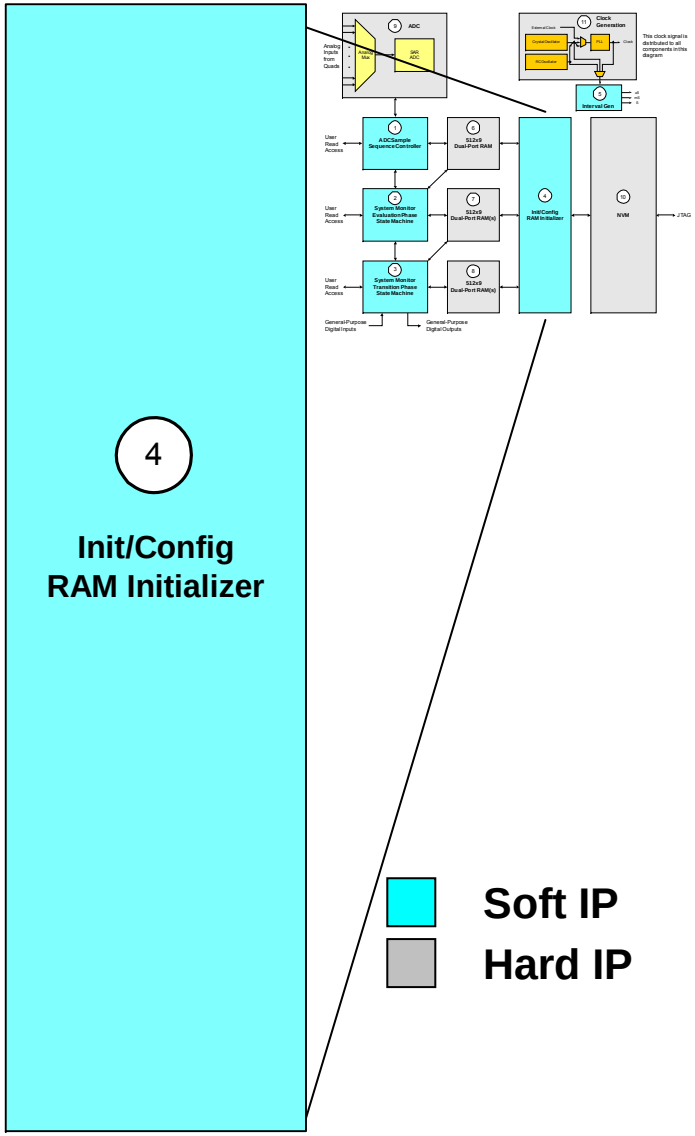
System Overview
NVM

- Non-Volatile Memory (NVM)
  - System configuration storage
    - Including Analog system configuration parameters
  - User-define storage
    - On-chip/off-chip microcontroller(s)
    - User data storage
    - etc.



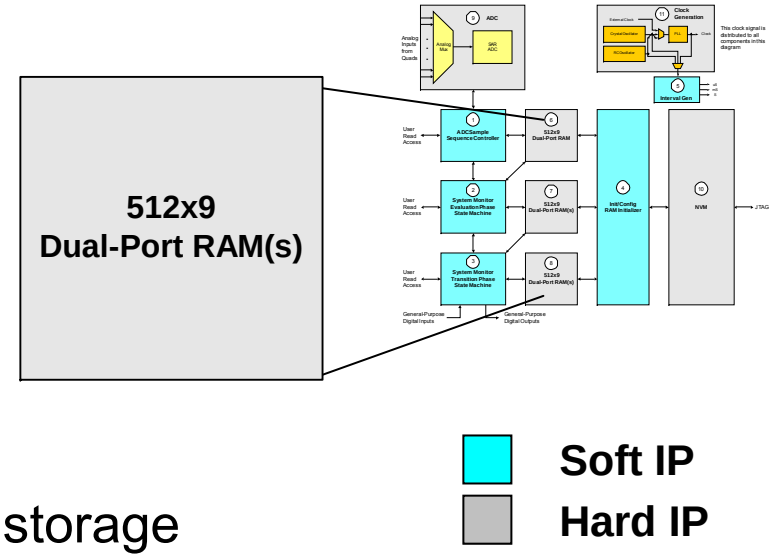
System Overview
Init/Config Block

- Init/Config Soft IP Block
  - Reads configuration information from NVM
  - Initializes (writes) all “clients”
    - Analog Quads, analog I/F soft IP, user IP
  - When init/config is done, the analog I/F soft IP commences operation



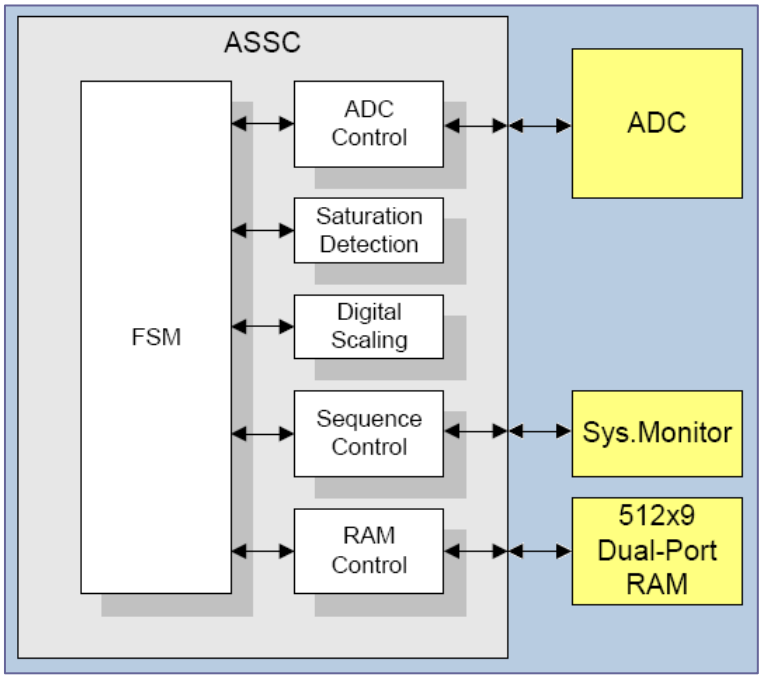
System Overview
Dual-Port RAM Blocks

- 512x9 Dual-port RAM Blocks
  - Heart of Analog I/F system
    - Initialized by Init/Config
  - ASSC block Configuration
    - Parameter storage
  - ADC sample storage for each analog channel
  - Digitally low-pass filtered ADC sample storage
  - Application sequence storage for SMEV and SMTR blocks
  - Threshold comparison results for SMEV block



System Overview
ASSC Soft IP Block

- ADC Sample Sequence Controller (ASSC)
  - Controls ADC
    - Calibration
    - Power-down
    - Sampling
    - 8/10/12-bit Resolution
  - Detects Saturation of Channels
  - Digital post-scaling of ADC Samples
  - Controls Current Monitor and Temperature Monitor Strobes
  - RAM Stores Sequence Information for ASSC



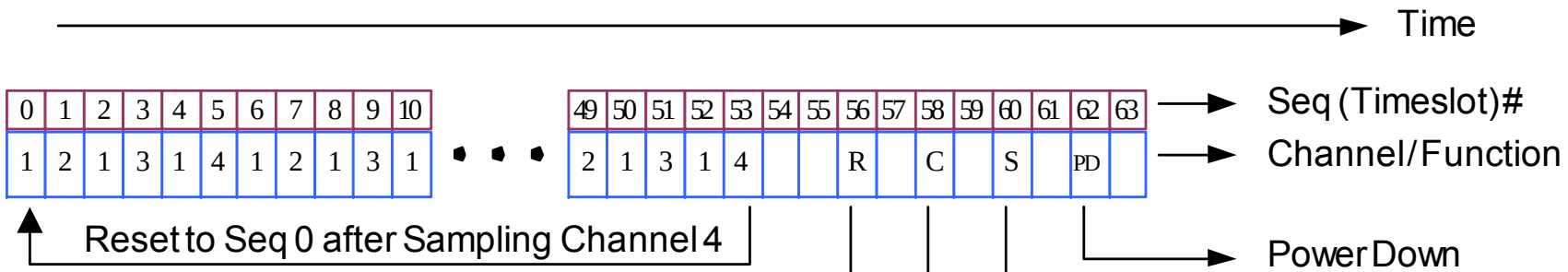
Sampling Sequence Specification

- Why do we need to Sequence the Samples???
  - 32 analog signals; 1 ADC
- TDM Sequencing
- 64 Time Slots
- More time slots for a Channel; more ADC Mindshare; Higher Throughput
- Each channel Added to Analog System Gets Added to Sequencer Automatically
- Sequence can be Optionally Modified for More/Less Samples
- Out of Sequence Jumps for Special Sampling During Runtime
  - One time jump; return to regular operating sequence
  - No automatic sequence, manual jump requests only
- Main Operating Sequence
  - Automatically computed based on target rate
  - Manual specification for customized applications

Main Feature Description

ASSC Time Slots

- Programmable Sequencer

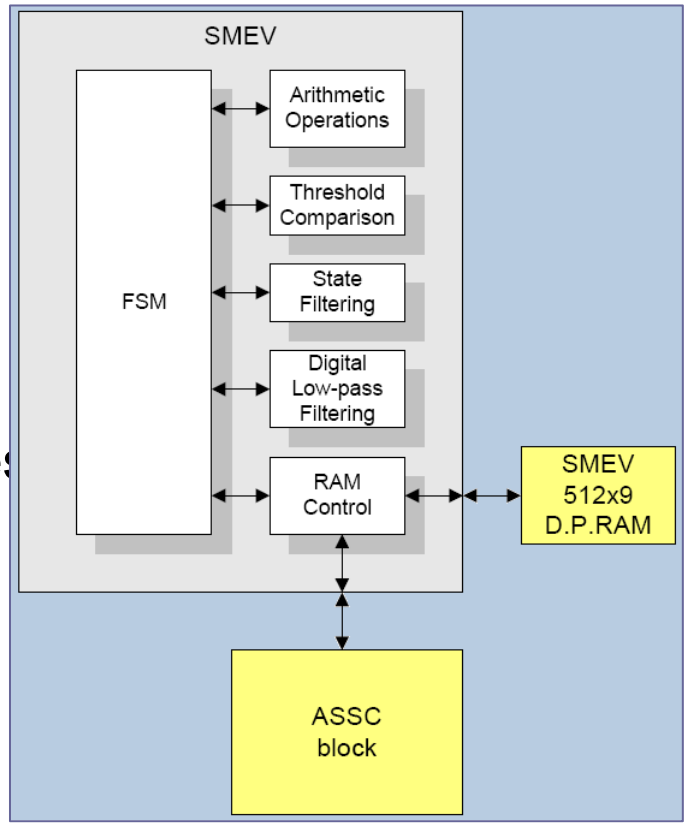


- 64 Possible Sample slots
- Variable sample rates
- Power down
- Timed sampling
- Automatic or external triggering
- External controlled ‘jump to Seq #’

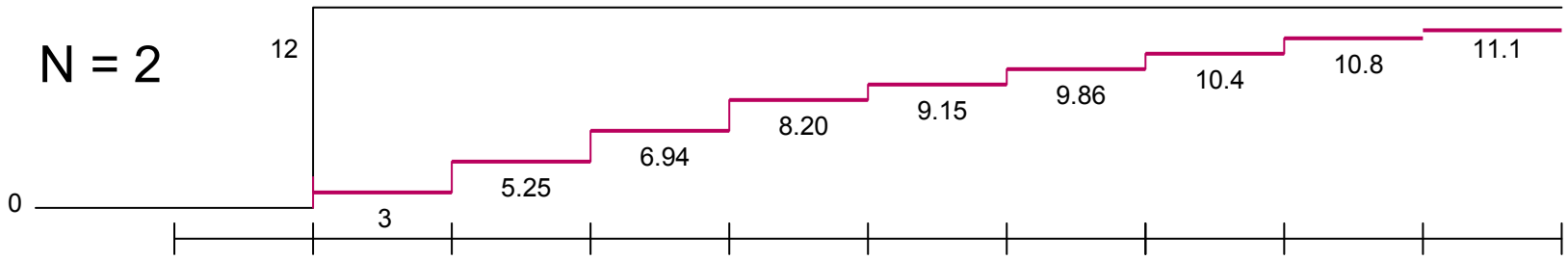
System Overview

SMEV Soft IP Block

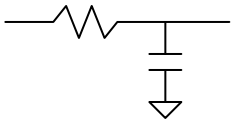
- System Monitor Evaluation Phase State Machine
  - Compares ADC Samples with user-defined Thresholds
  - Simple 12-bit, Unsigned Arithmetic Operations
    - A[11:0], B[11:0]
  - Digital Low-Pass Filtering of ADC Samples
    - Average ADC sampled data
    - Requires only current ADC sample and previous averaged data
  - State Filtering (Digital Correlation)
    - Look for 0->1 or 1->0 Transitions
  - RAM Stores Application Sequences for SMEV
    - Sequences controlled by Op-codes



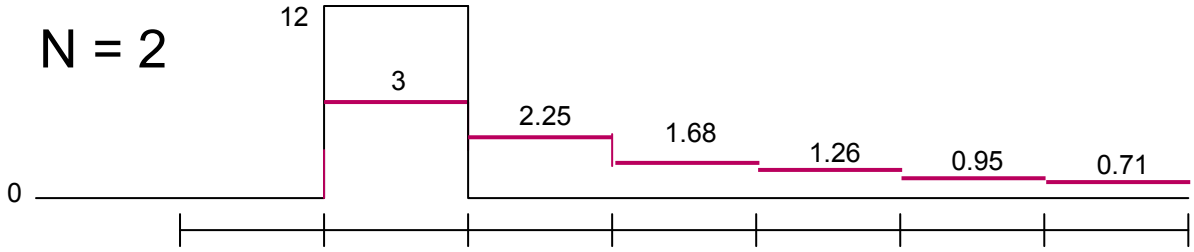
Digital Low Pass Filtering



Step Response



T = 4 Samples



Full charge/Discharge
At 5*T = 20 samples

Impulse Response

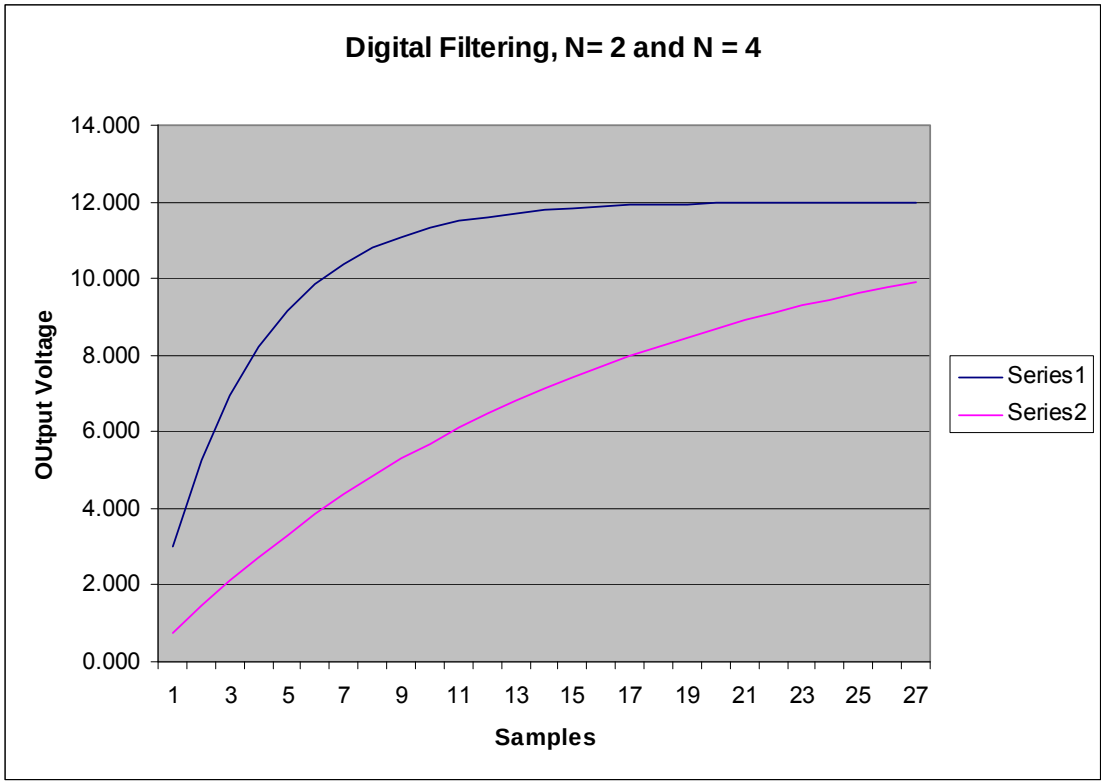
Equation: $X_n <= X_{n-1} – (X_{n-1}/2^N) + (I_n/2^N)$; where N = LPF factor

Digital Low Pass Filtering (cont.)

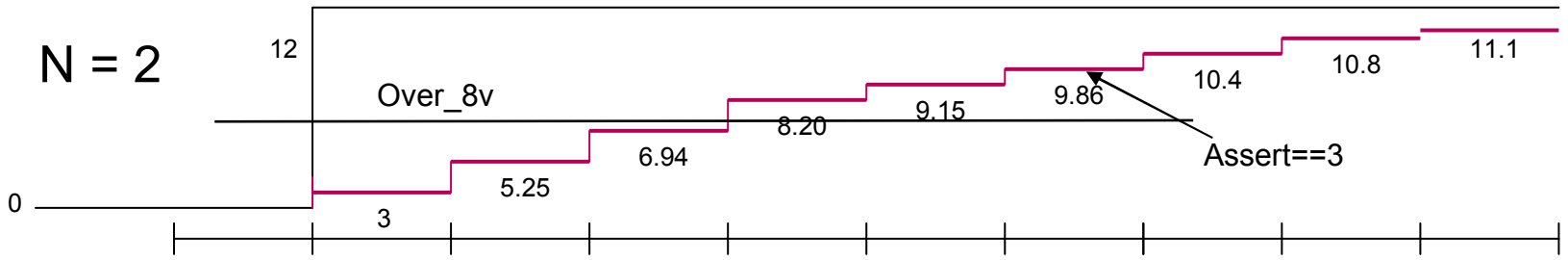
Equation: $X_n <= X_{n-1} – (X_{n-1}/2^N) + (I_n/2^N)$; where N = LPF factor

Digital Filtering - Step Response

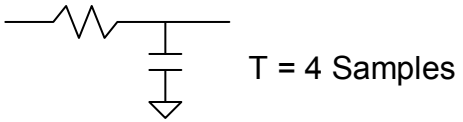
N =	2	4
In =	12	12
Initial Value =	0	0
Xo	3.000	0.750
X1	5.250	1.453
X2	6.938	2.112
X3	8.203	2.730
X4	9.152	3.310
X5	9.864	3.853
X6	10.398	4.362
X7	10.799	4.839
X8	11.099	5.287
X9	11.324	5.706
X10	11.493	6.100
X11	11.620	6.469
X12	11.715	6.814
X13	11.786	7.138
X14	11.840	7.442
X15	11.880	7.727



State Filtering



Step Response



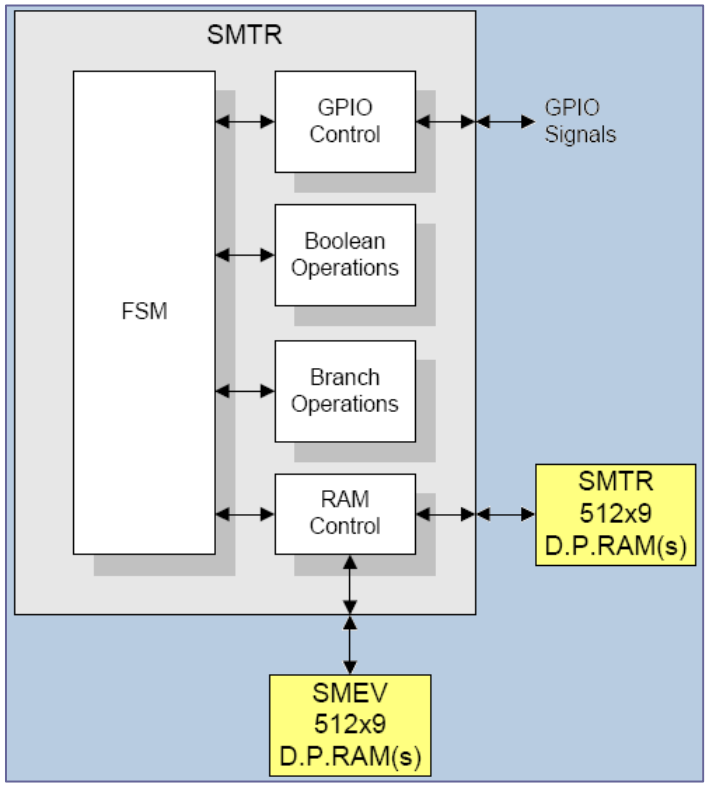
Full charge/Discharge
At 5*T = 20 samples

Impulse Response

Equation: $X_n <= X_{n-1} – (X_{n-1}/2^N) + (I_n/2^N)$; where N = LPF factor

System Overview
SMTR Soft IP Block

- System Monitor Transition Phase State Machine
  - User-defined Digital Inputs and Digital Outputs (flags)
  - Boolean Operations on Digital Inputs and Internal Temporary Registers
  - Branching Operations (if-then-else conditional jumps, looping)
  - “Microcontroller-like”
  - Reads and Processes SMEV Comparisons
  - RAM Stores Application Sequences for SMTR
    - Sequences Controlled by Op-codes



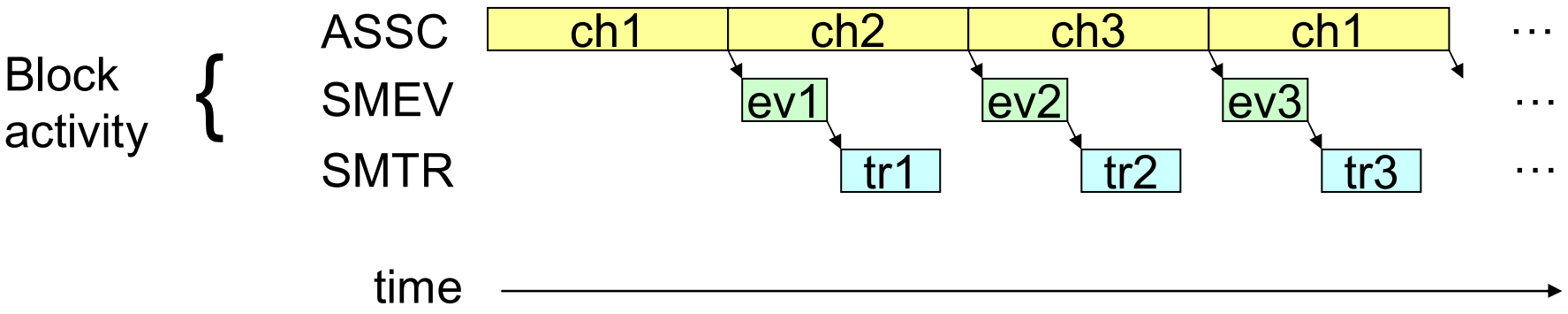
Data Processing Order

- Acquire Analog Signal
- Convert to Digital Data
- Average the Resulting Data to Smooth the Samples
- Compare the Result With the Threshold to Detect Overflow or Underflow
- Wait for Multiple Compare Results to Remove Glitches
- Raise the Over/Under Flow Flag

Main Feature Description

Analog I/F Operation Flow

- ASSC -> SMEV -> SMTR
  - Pipelined, TDM operation: ADC samples first, Evaluation of samples second, Transition decisions based on Evaluation third, repeat …



Soft IP Summary

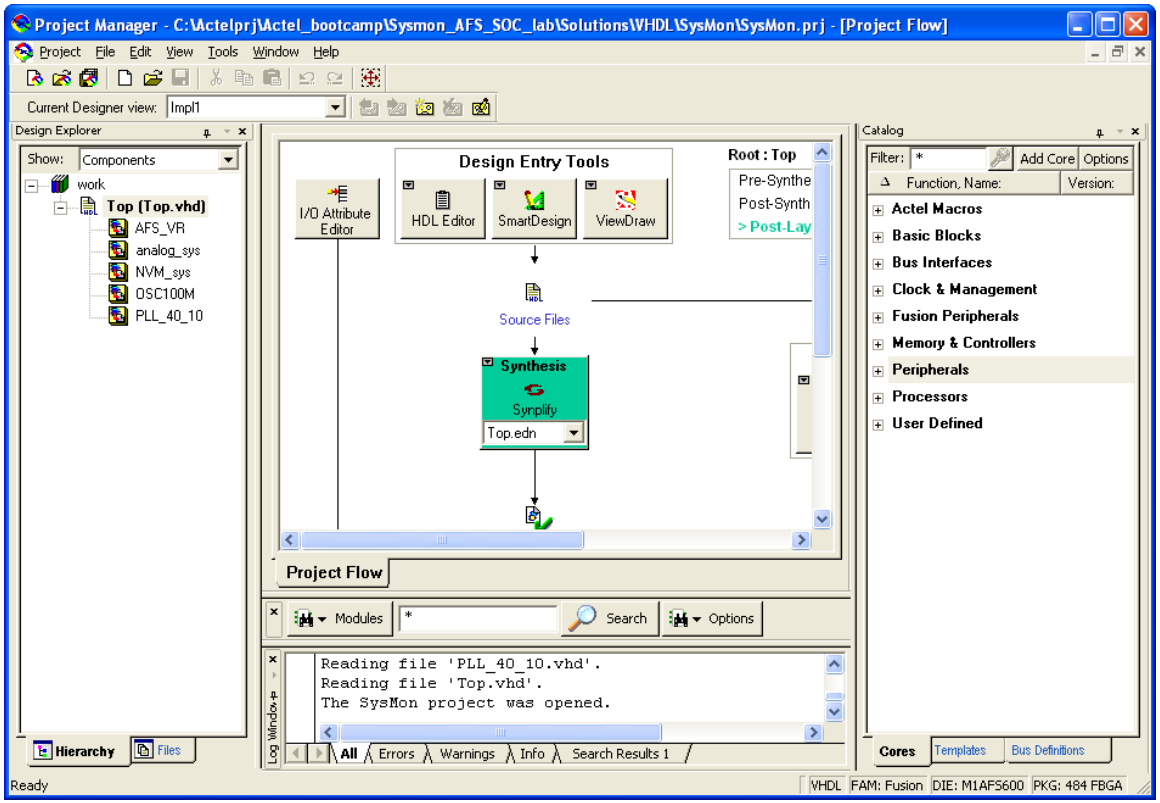
- Soft IP Main Control for Analog System
- Configurable via Setting Generics/Parameters at top-level
- Pipelined Operation
- Complex Sequencing, Branching, and Looping Operations Possible With Limited Hardware Resources
- Completely Self-sustained System (no Micro Required)

Fusion Design Flow

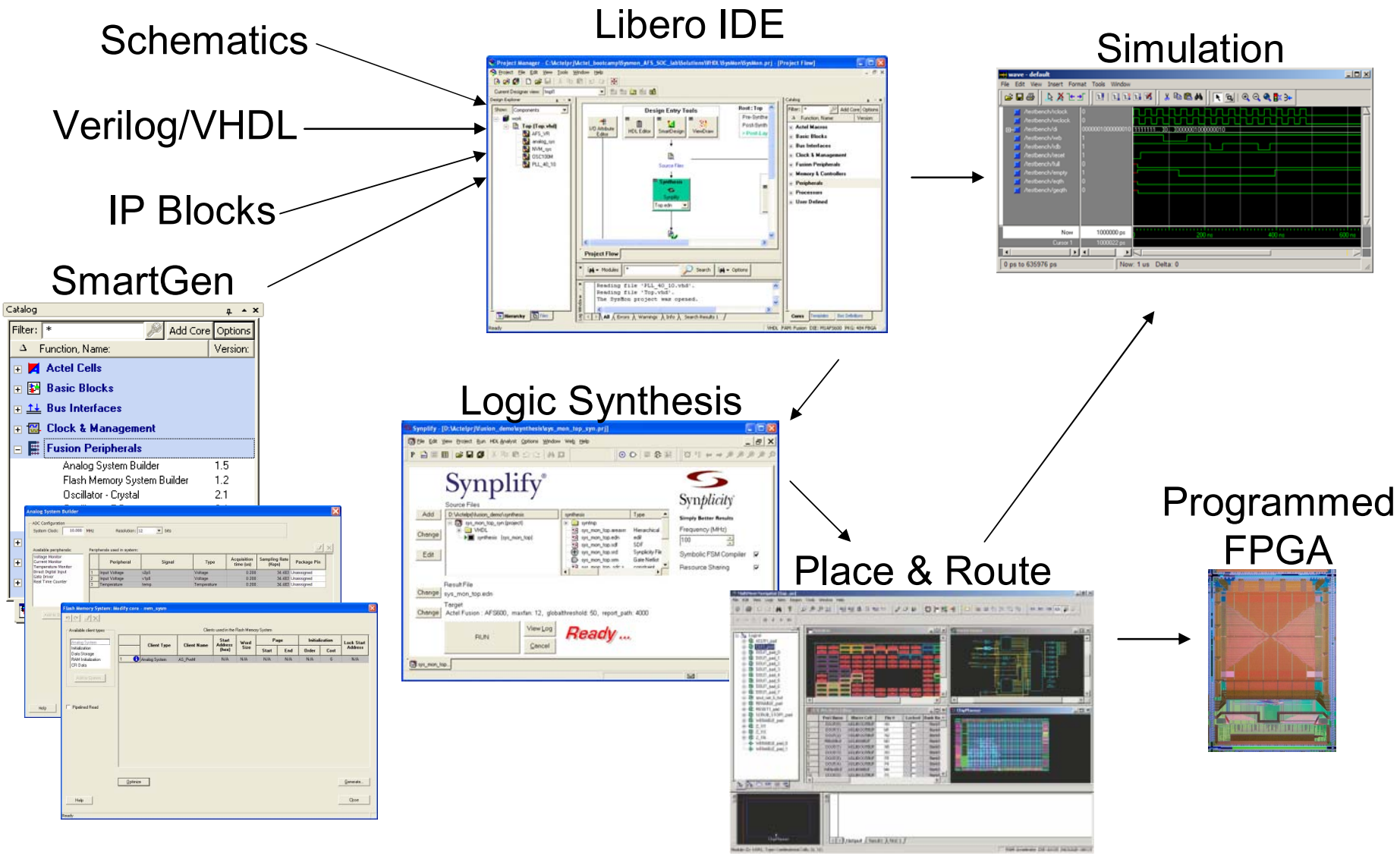
- Fusion Smart Backbone
- Libero Flow
- SmartGen
- Simulation
- Programming and Security

Libero IDE

- Integrated Development Environment
  - Schematic Editor
  - HDL entry tools
  - SmartGen Wizard
  - Actel IP Cores
  - Logic Synthesis
  - Logic Simulator
  - I/O Editor
  - Place & Route tools
  - Timing Analysis
  - Device Programming



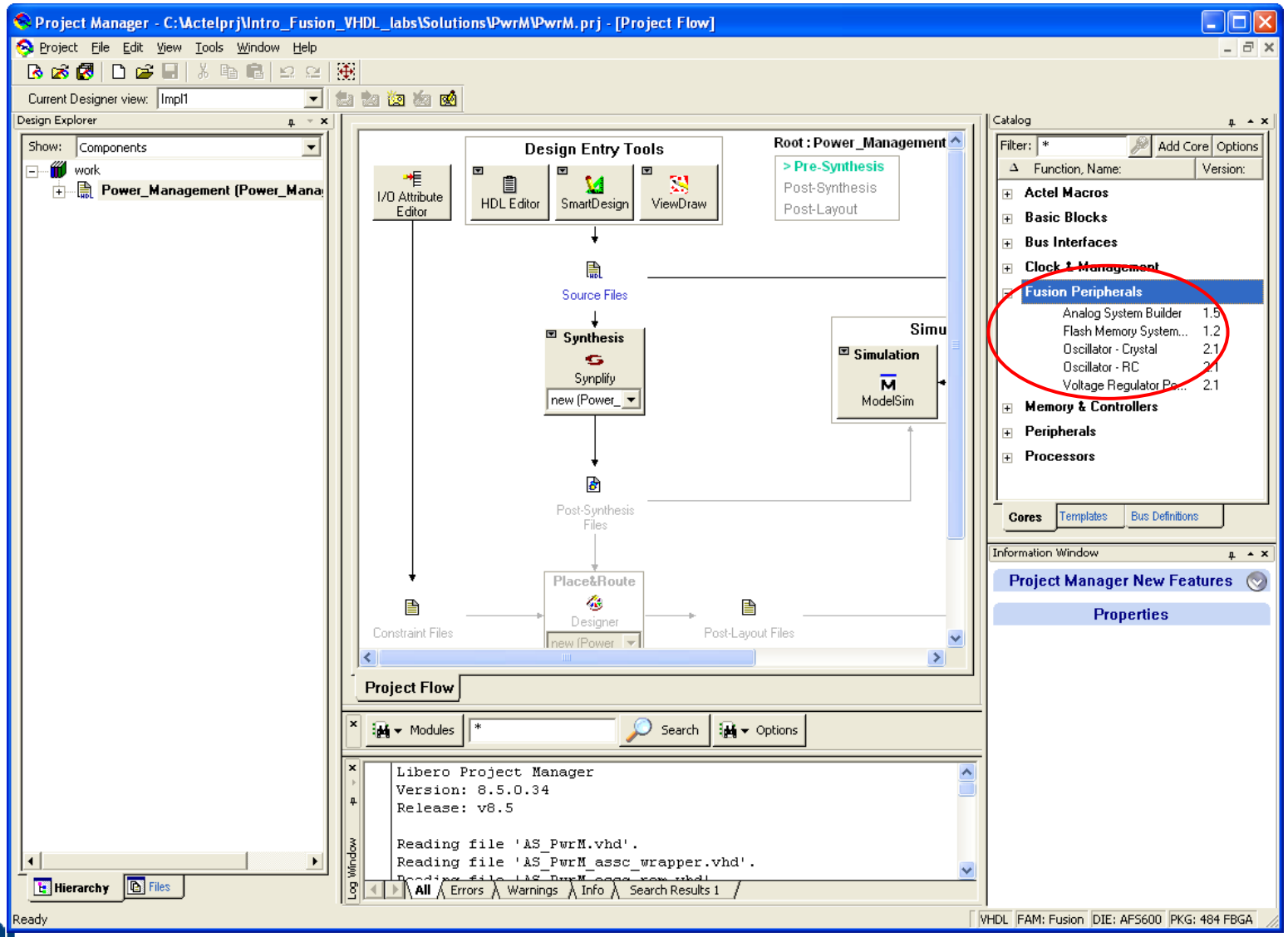
Familiar FPGA Design Flow



Fusion Design Flow

- Fusion Smart Backbone
- Libero Flow
- SmartGen
  - Flash Memory System Builder
  - Analog System Builder
  - Other SmartGen Cores
- Simulation
- Programming and Security

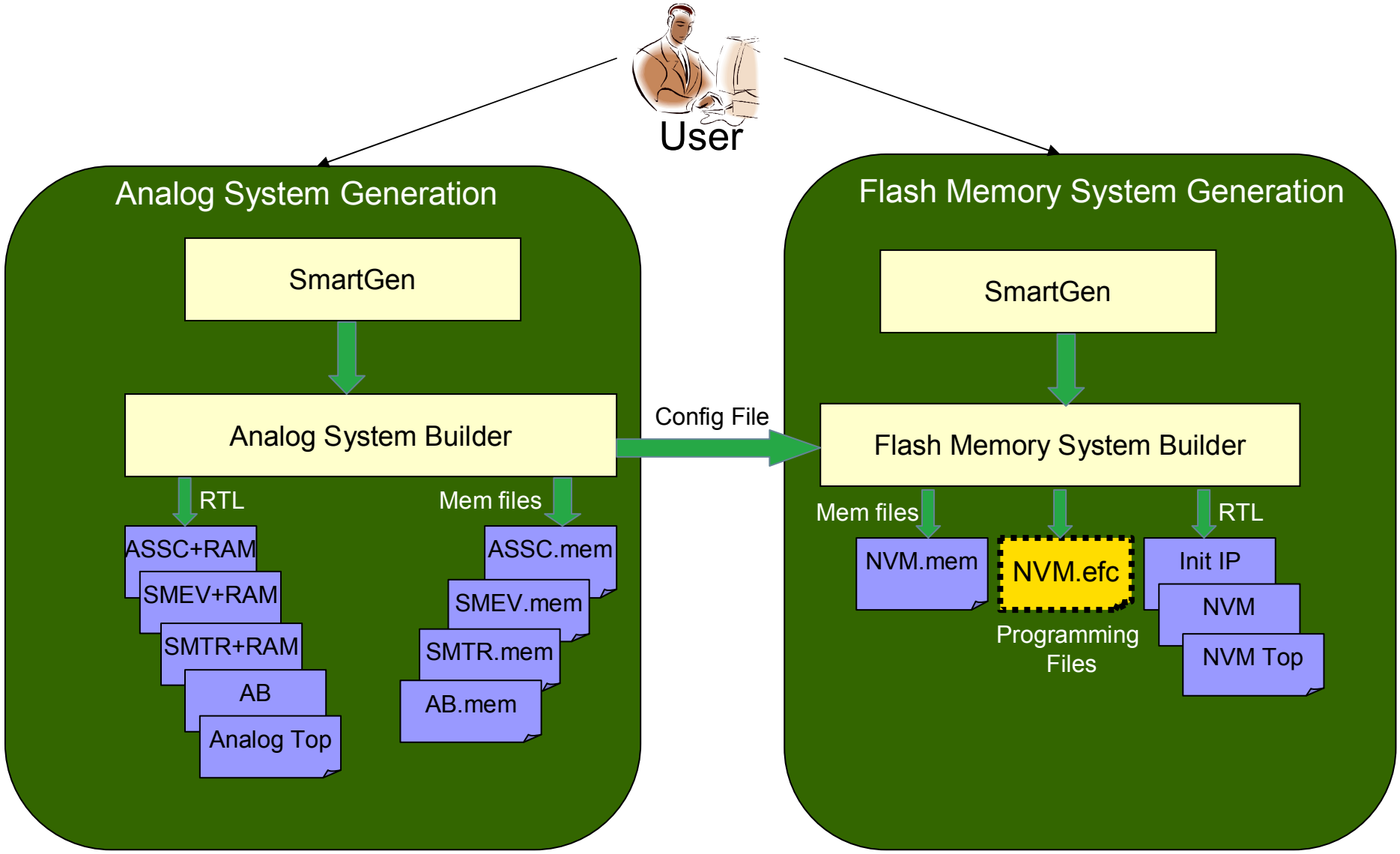
Libero with SmartGen



SmartGen Cores for Fusion

- New System Builders
  - Analog System Builder
  - Flash Memory System Builder
- New Silicon Cores
  - Divided and Delayed Clock
  - Crystal Oscillator
  - VRPSM
  - NGMUX
  - RC Oscillator
  - Dynamic CCC
- Enhanced Cores from ProASIC3/E
  - Static PLL
  - Delayed Clock
  - RAM
    - Initialization from Flash Memory

Analog & NVM System Generation

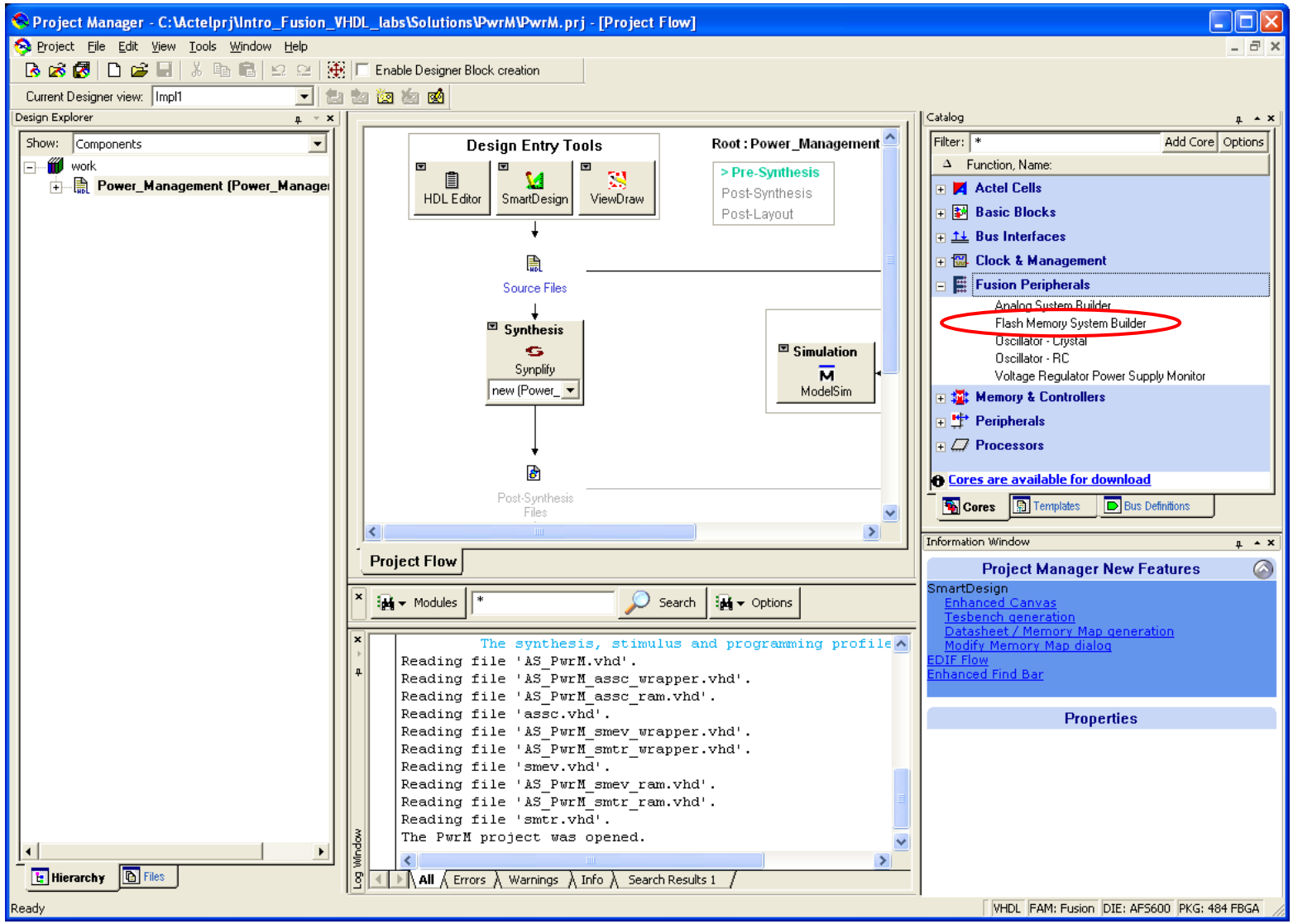


Fusion Design Flow

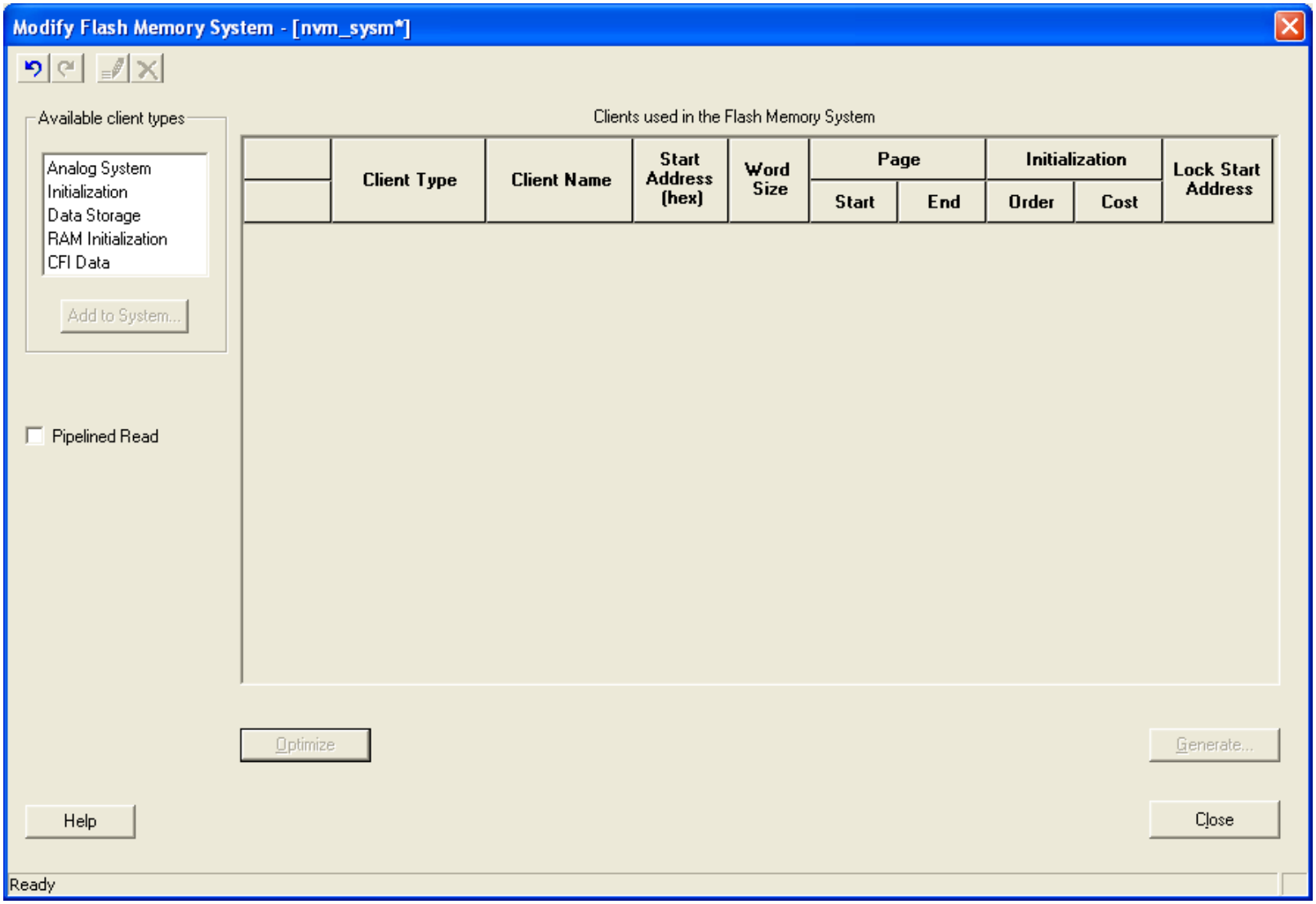
- Fusion Smart Backbone
- Libero Flow
- SmartGen
  - Flash Memory System Builder
  - Analog System Builder
  - Other SmartGen Cores
- Simulation
- Programming and Security

SmartGen

Flash Memory System Builder



Flash Memory System Builder



Flash Memory System Builder

Supported Clients

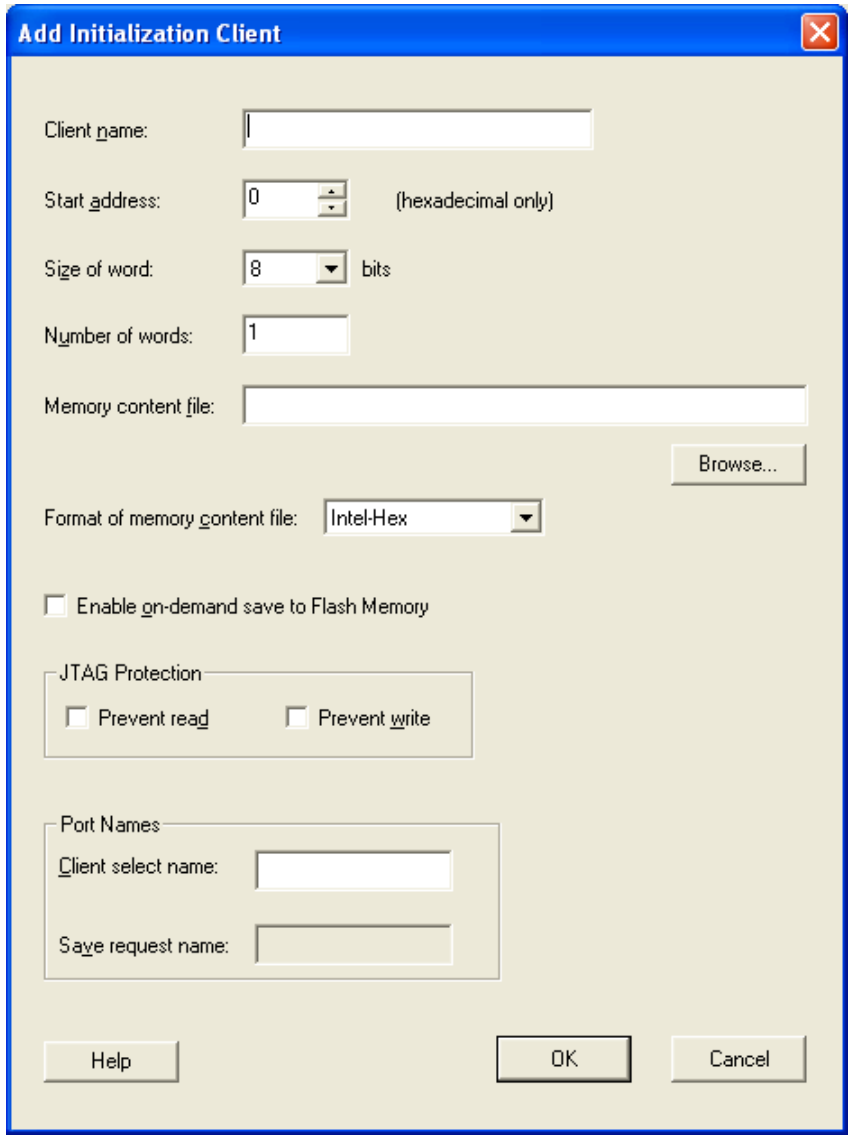
- Analog System
  - Initialize AB and Analog System soft IP RAM data
  - Uses configuration file generated by Analog System Builder
- Data Storage Clients
  - Use NVM as a hard-drive. Partition and access the NVM memory.
- Initialization Clients
  - User clients that required initialization and start-up
- RAM Initialization Client
  - Special type initialization client
  - Generated by SmartGen
  - Uses configuration file generated by SmartGen
- CFI Data Client
  - Used to store the query data for CoreCFI
    - The data is stored in a reserved page location. This client does not take up any of the 2048 pages in the Flash Memory
    - CoreCFI Provides an Industry-Standard External Interface to Actel Fusion™ Flash Memory

Flash Memory Builder Features

- Supports the Following Memory File Formats
  - Intel-Hex
  - Motorola-S
  - Actel-Hex
  - Actel-Binary
- Generates Map Files for Programming the NVM
- Clients are Page Aligned
- Clients are Initialized Sequentially
- Start Addresses for Partitions
  - Automatically managed
  - Manually specified and locked for applications requiring fixed addresses
- Automatic Conflict Resolution for Overlapping Client Partitions

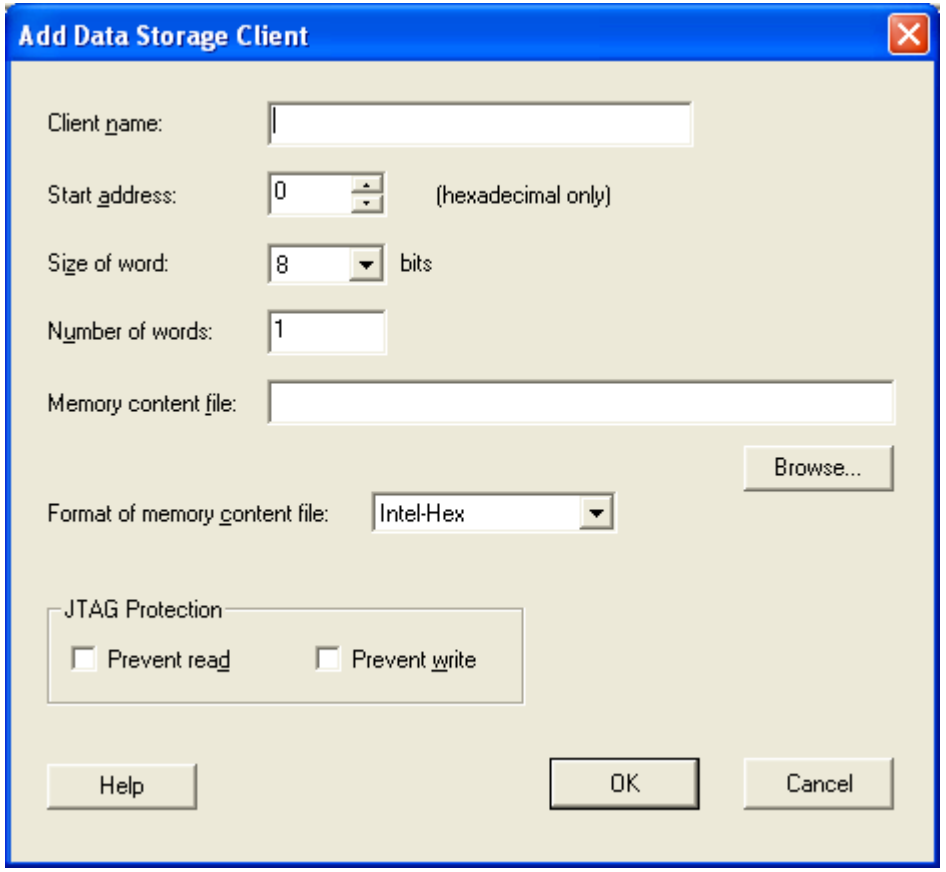
Initialization Client

- Specify Clients to be Initialized at Start-up
- Supported Word Sizes
  - 8 bit & 9 bit
- On-demand Save-back to NVM
- Multiple Memory file Formats Supported



Data Storage Client

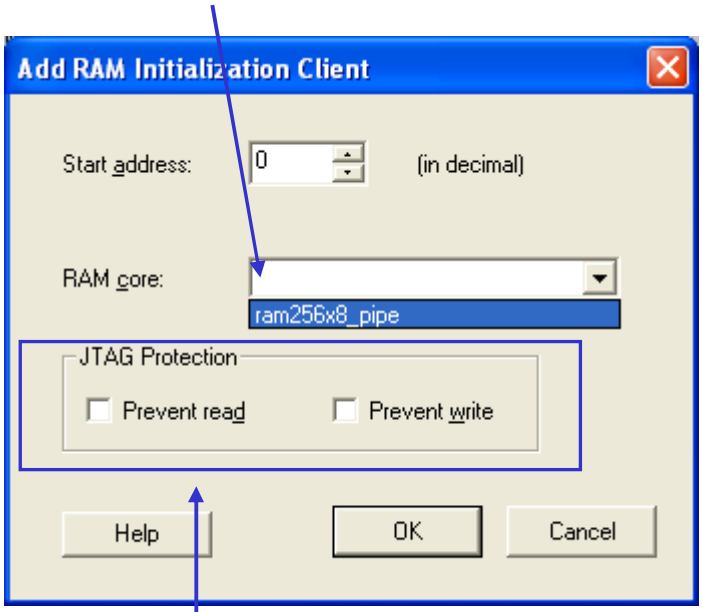
- Create a Partition in the Flash Memory System and Specify the Memory Content for That Partition
- Supported Word Sizes
  - 1 byte, 2 byte, 4 byte
- Start Address
- Ability to Lock Start Address (System Builder Level)



RAM Initialization Client

- The RAM Initialization Client is a Special Type of Initialization Client That Allows RAM to be Initialized at Power-up
  - Memory Editor in SmartGen used to specify RAM content
- Difference from Initialization Client:
  - Cascading Of Multiple RAM Blocks Is Handled Automatically
- Initialization on a 9-bit Data Bus for Optimized Write Cycles
- Logic Automatically Created for Multiplexing Initialization and run time Interfaces

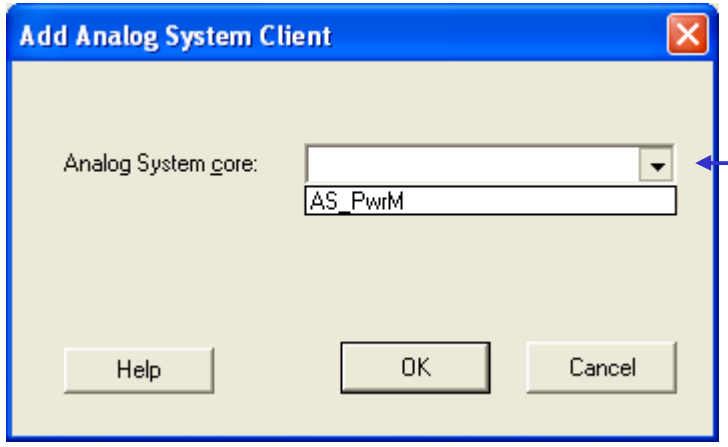
Select Client Name from pull-down menu



Set JTAG Protection

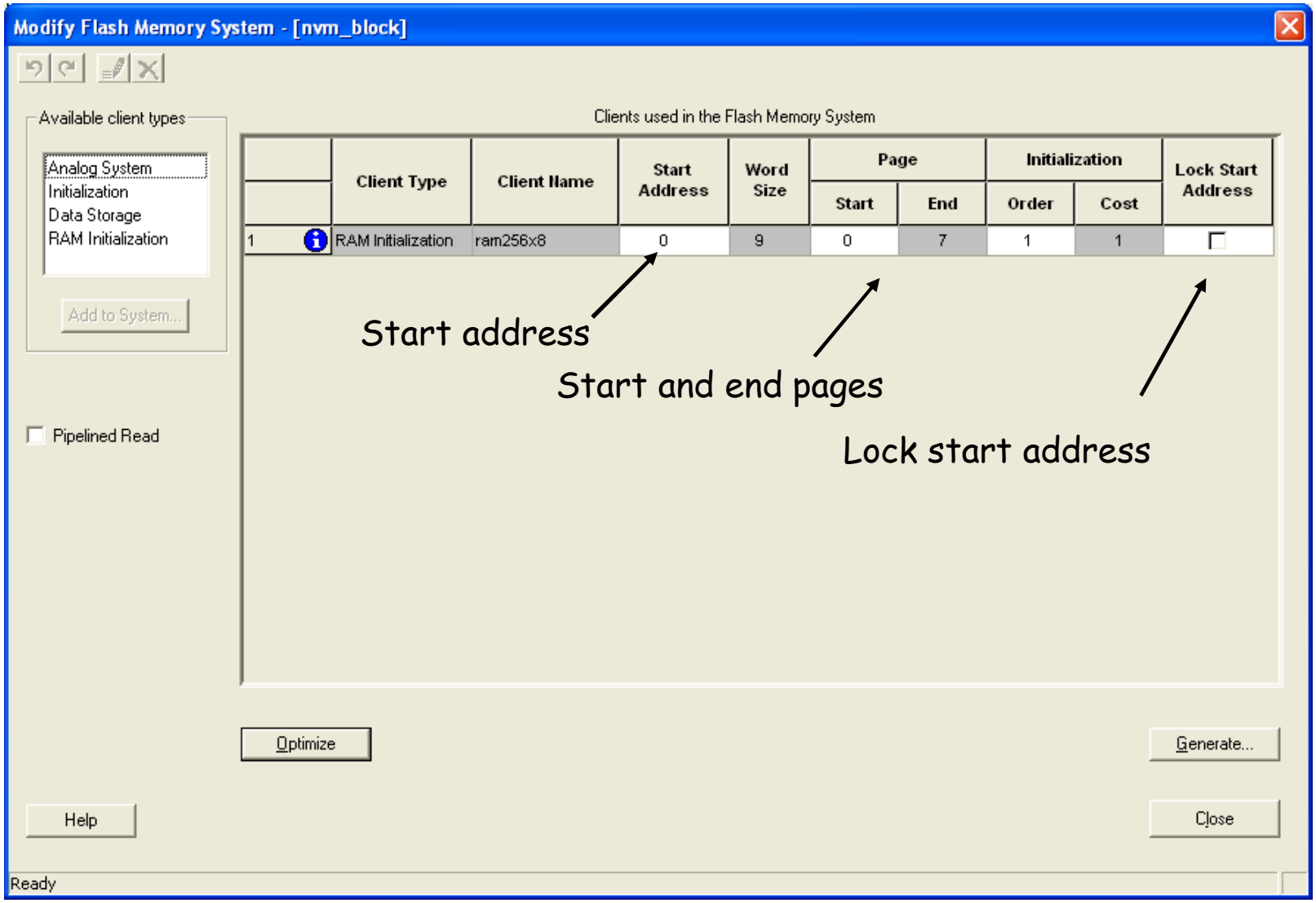
Analog System Client

- Load the Configuration File Generated by the Analog System Builder Into the Flash Memory System Builder
- The Analog System Components can be Initialized by the Flash Memory System at Start Up



Select Client Name from pull-down menu

Flash Memory System Builder with RAM Initialization Client



Flash Memory System
Restrictions

- Minimum Usage per Client: 1 Page
- Maximum Data Storage Clients: 64
- Maximum Initialization Clients: 64
- Analog System takes 6 Initialization Clients
  - 4 Initialization Clients if Calibration IP is not used
- Each RAM takes as many Initialization Clients as Number of RAM Blocks in the RAM

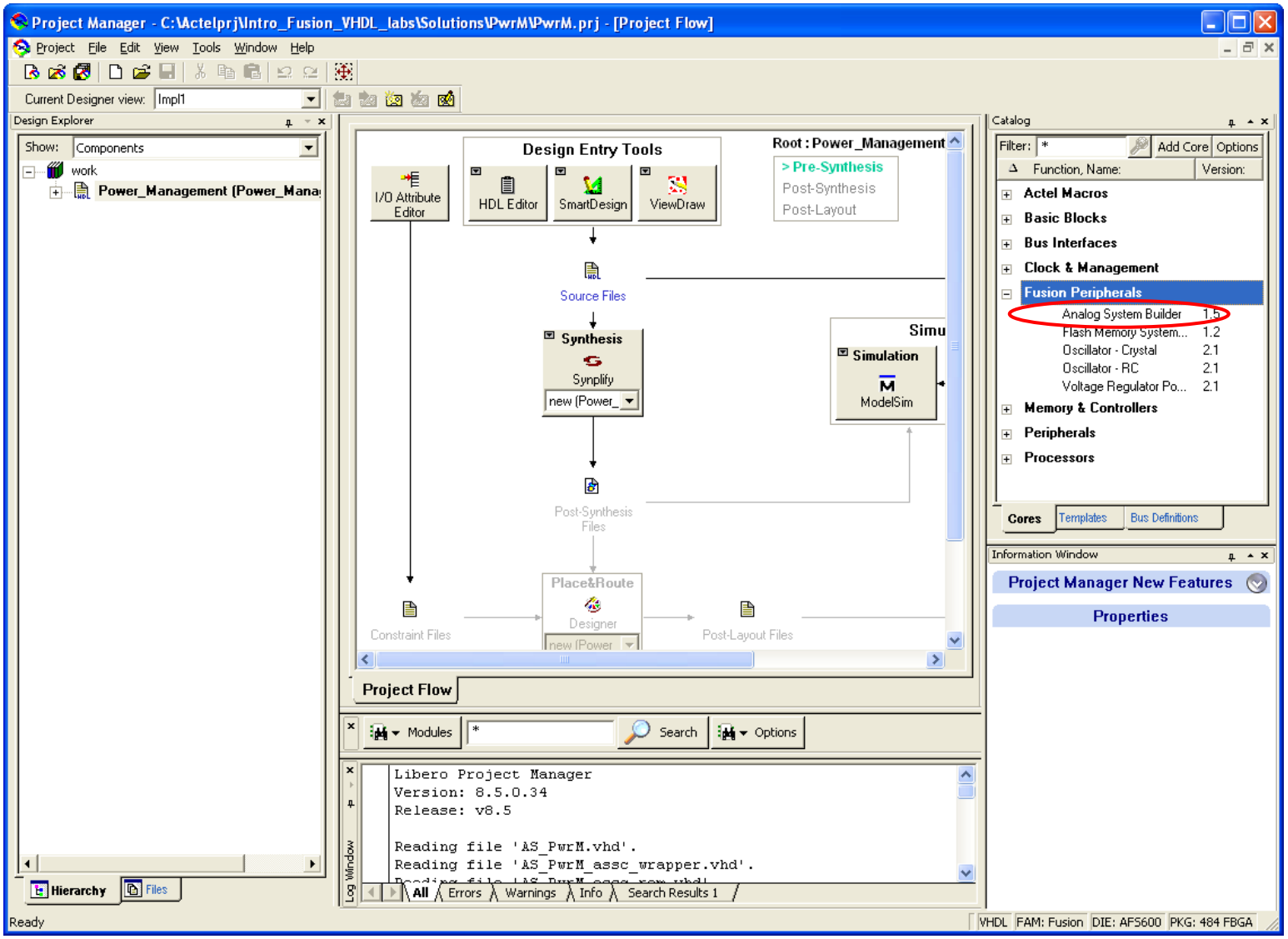
Fusion Design Flow

- Fusion Smart Backbone
- Libero Flow
- SmartGen
  - Flash Memory System Builder
  - Analog System Builder
  - Other SmartGen Cores
- Simulation
- Programming and Security

Analog System Builder

- Creates Complete Analog System Including
  - AB Hard-Macro
  - Analog System Soft-IPs (RTL)
  - Analog System data storage RAMs
  - Memory files for simulation
  - Configuration file for import into NVM System

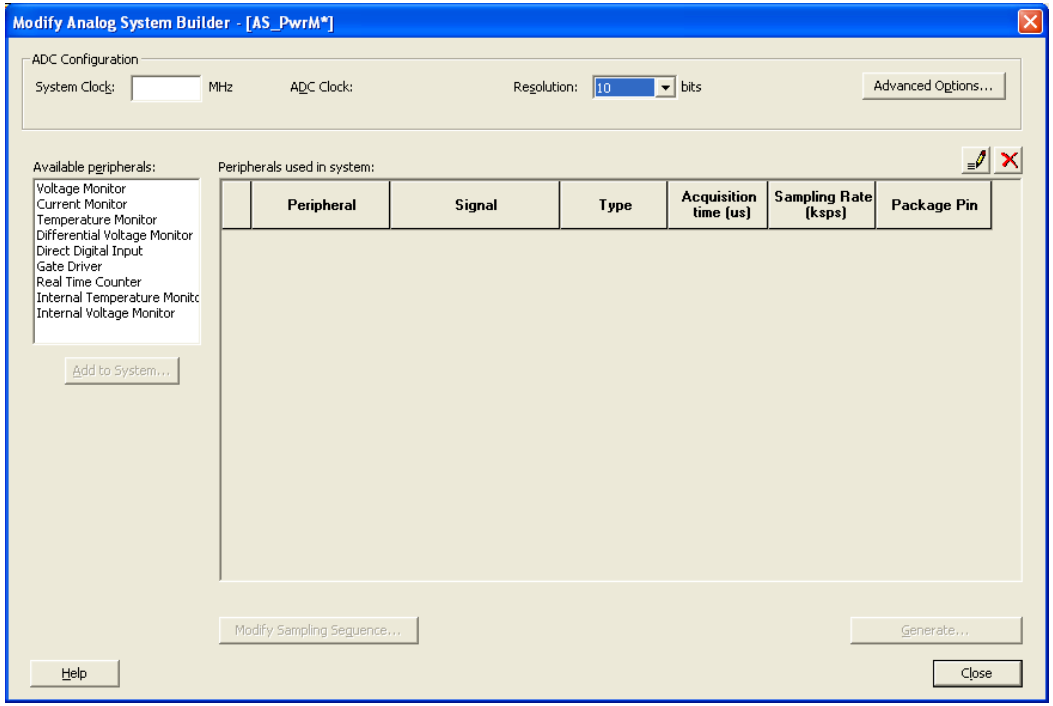
Analog System Builder in Libero



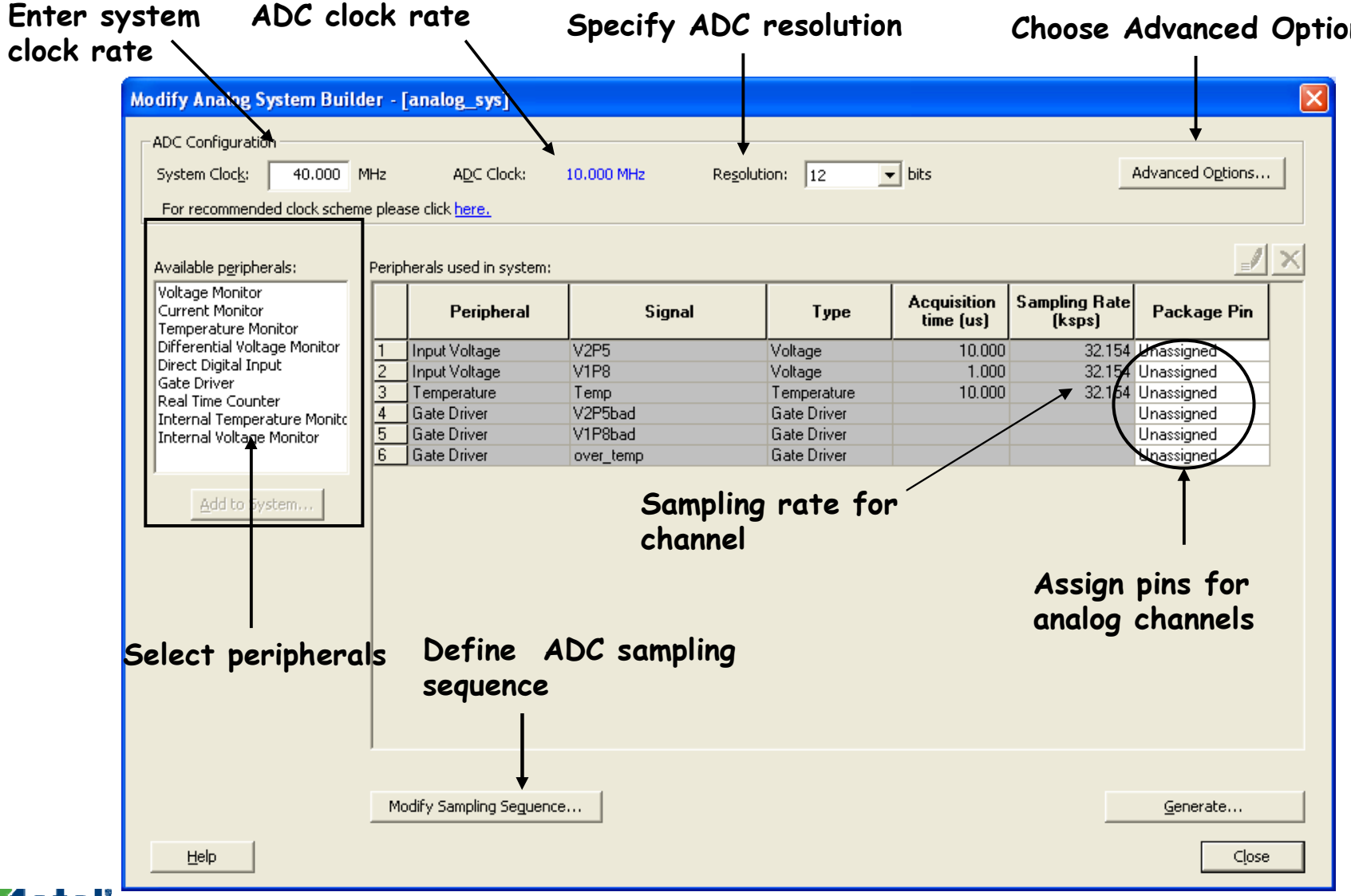
Analog System Builder

Supported Peripherals

- Voltage Monitor
- Current Monitor
- Differential Voltage Monitor
- Temperature Monitor
- Direct Digital Input
- Output Gate Driver
- Internal Temperature Monitor
- Internal Voltage Monitor
- RTC (Real Time Counter)

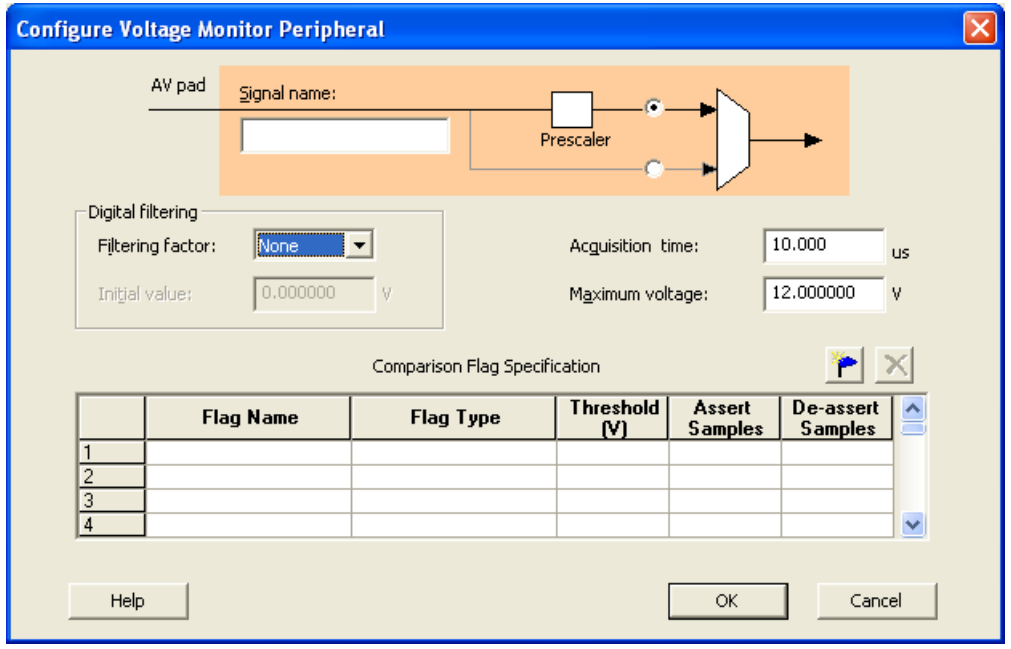


Analog System Builder GUI



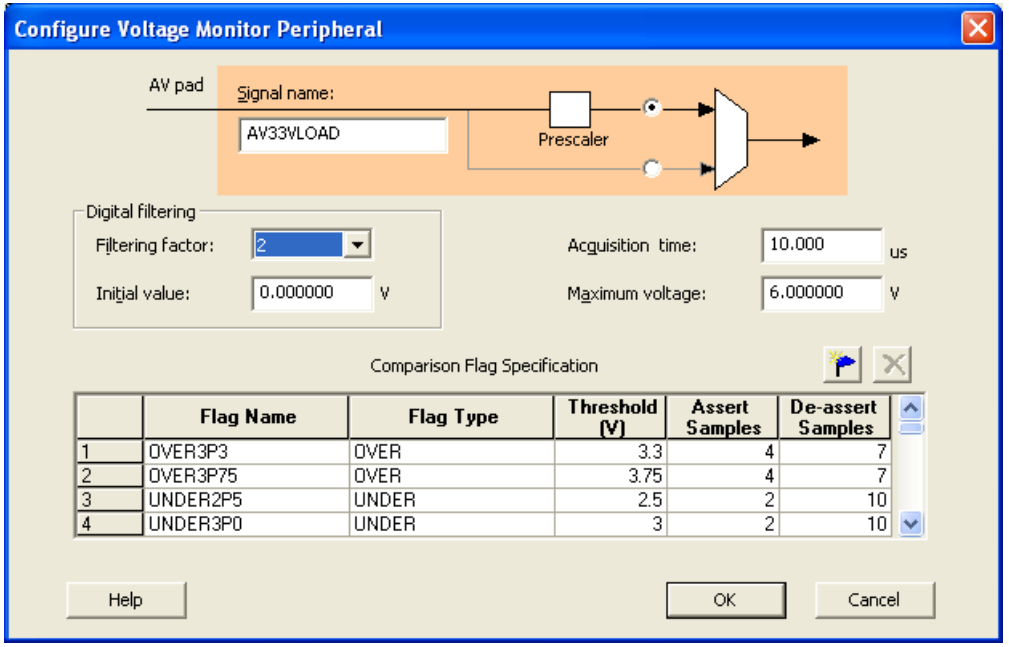
Voltage Monitor Settings

- Signal Name
  - Name of analog pad in top-level design
- Acquisition Time
  - Sample & Hold Duration
- Digital Filtering
  - Digital Averaging Factor
  - Initial Value
- Maximum Voltage
  - Used to set the pre-scaling factor
    - User range 0V to +12 or -12V to 0V
    - ADC range -3V to + 3V



Voltage Monitor Flag Settings

- Voltage Comparison Flag Specification
  - Flag Name
  - Specify Flag Assertion when Under or Over Threshold
  - Threshold level (V)
  - Number of consecutive times the signal is above or below threshold for flag to assert and de-assert
    - Eliminate cases where a single glitch causes the flag to assert
    - Make sure the signal is really above or below range for flag assert
    - Make sure the condition that caused the flag assertion is really gone before de-asserting the flag

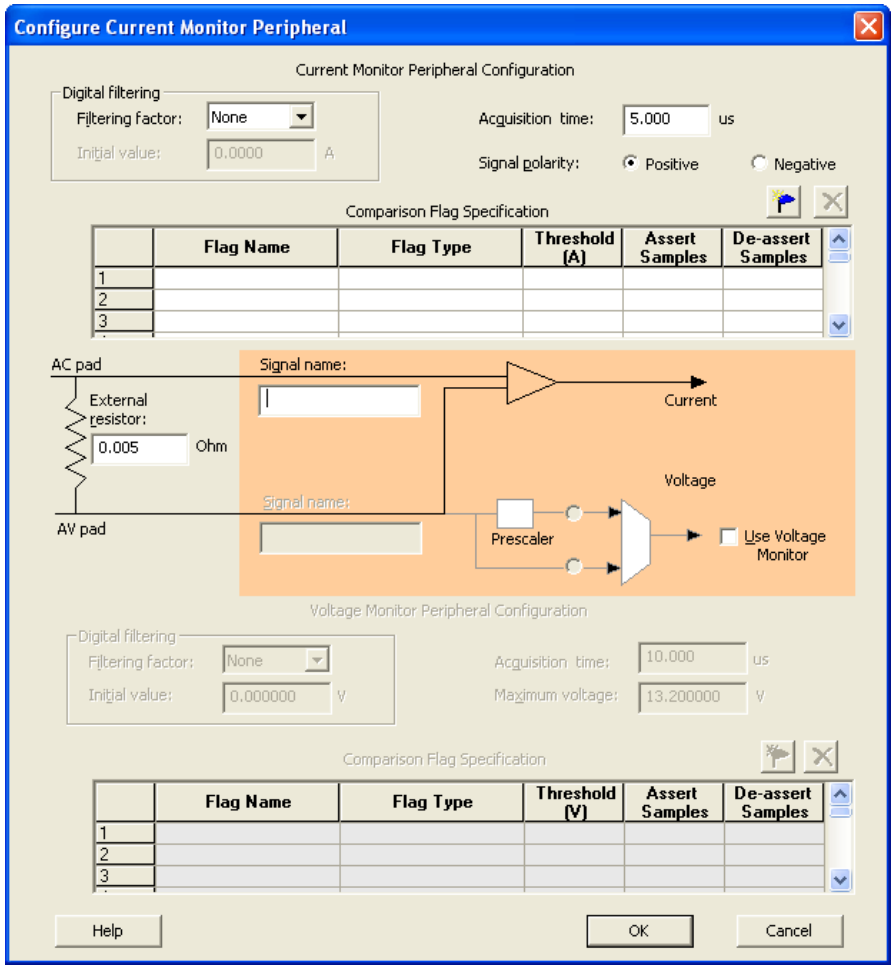


Current Monitor

- Requires a Voltage-current Channel Pair Placed On Adjacent Package Pins
- Measures Differential Voltage Across External Resistor
  - The differential voltage is multiplied by 10x before it is applied to the ADC
  - Choose an external resistor that ensures that the difference in voltages is less than the value of Vref
- Optionally Monitors Voltage on the Voltage Channel in the Pair
  - Options identical to voltage monitoring service
- Voltage Differential Must be Less Than or Equal to VREF

Current Monitor Settings

- Digital Filtering
  - Digital Averaging Factor
  - Initial Value
- Acquisition Time
  - Sample & Hold duration
- Signal Name
  - Name of analog pad in top-level design
- External Resistor
  - Value of the resistor connected across the Current-Voltage pair
- Current Comparison Flag Specification
  - Flag Name
  - Specify Flag Assertion when Under or Over Threshold
  - Threshold level (A)
  - Number of consecutive times the signal is above or below threshold for flag to assert and de-assert
- Voltage Monitor Settings Same as Voltage Monitor

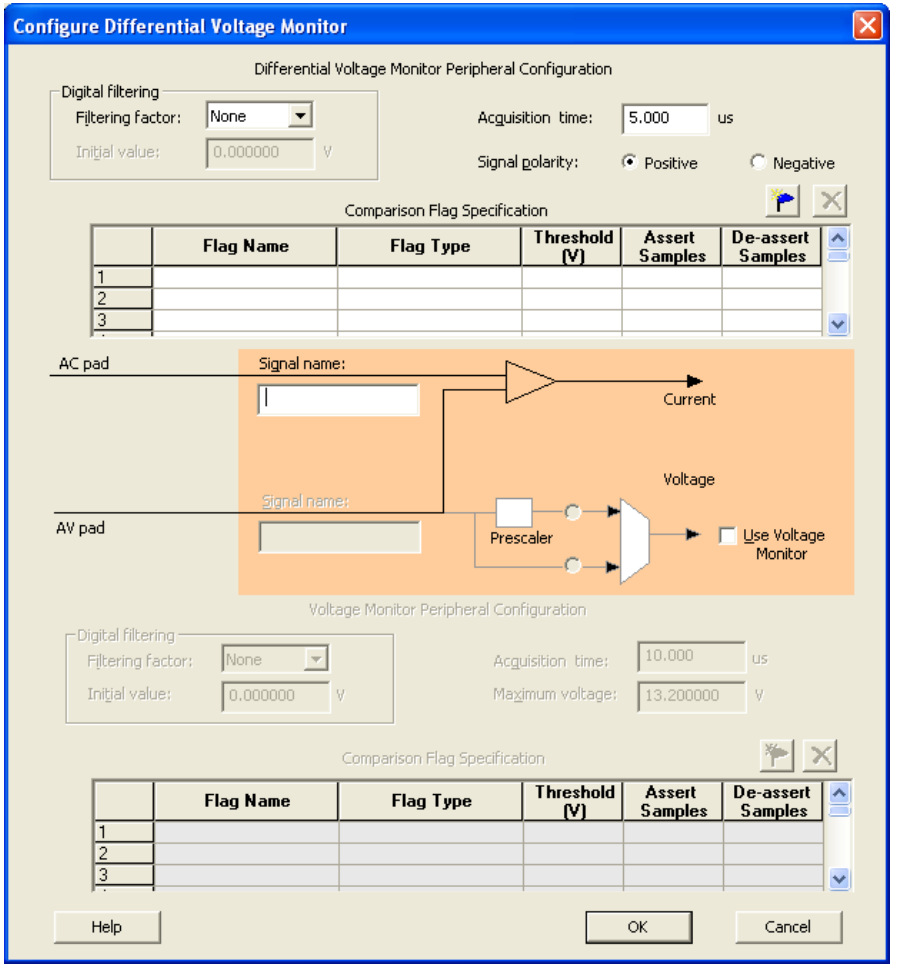


Differential Voltage Monitor

- Measures The Differential Voltage Between A Pair Of Voltage And Current Input Channels
  - Uses same components as Current Monitor
  - Requires two channels (AV and AC)
  - they must be on adjacent package pins
- Optionally Monitors Voltage on the Voltage Channel in the Pair
- Voltage Differential Must be Less Than or Equal to V_{REF}

Differential Voltage Monitor Settings

- Digital Filtering
  - Digital Averaging Factor
  - Initial Value
- Acquisition Time
  - Sample & Hold duration
- Signal Name
  - Name of analog pad in top-level design
- Differential Voltage Comparison Flag Specification
  - Flag Name
  - Specify Flag Assertion when Under or Over Threshold
  - Threshold level (V)
  - Number of consecutive times the signal is above or below threshold for flag to assert and de-assert
- Voltage Monitor Flags same as Voltage Monitor

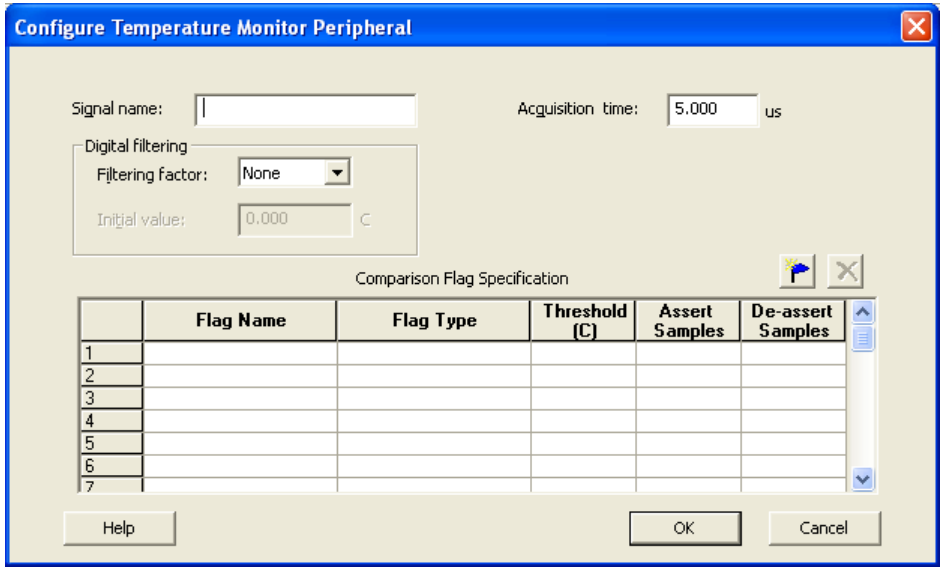


Temperature Monitor

- Measures Differential Voltage Across External Diode
- Configuration Settings
  - Identical to the voltage monitoring service, except maximum supply voltage.
- Voltage Differential Must be Less Than or Equal to V_{REF}

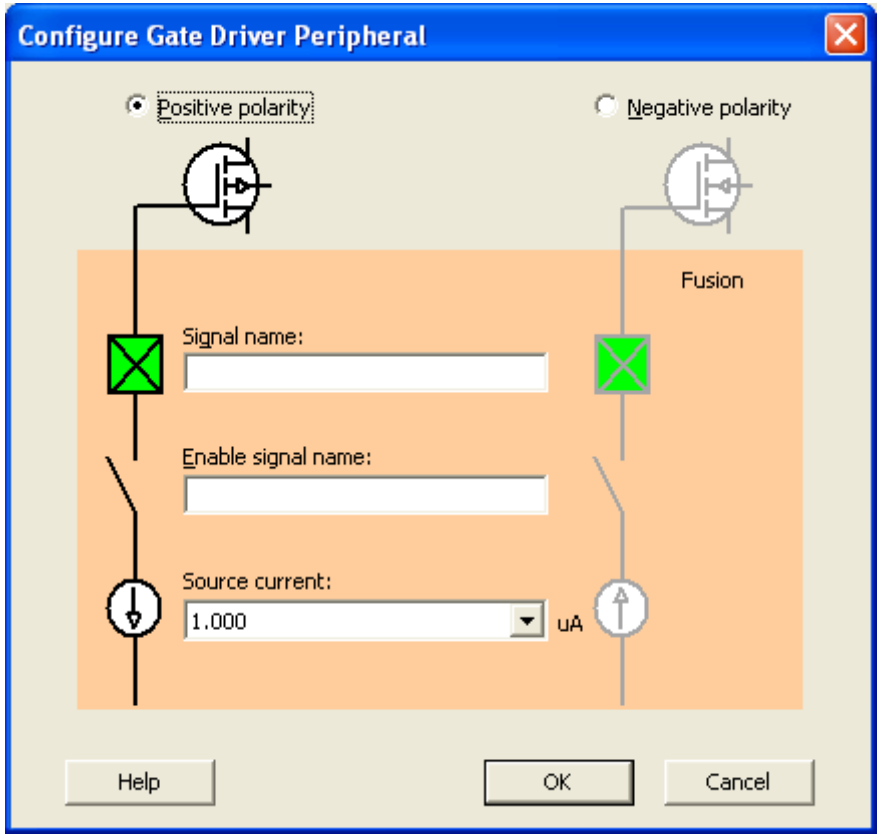
Temperature Monitor Settings

- Digital Filtering Factor
  - Digital Averaging Factor
- Acquisition Time
  - Sample & Hold duration
- Signal Name
  - Name of analog pad in top-level design
- Temperature Comparison Flag Specification
  - Flag Name
  - Specify Flag Assertion when Under or Over Threshold
  - Threshold level (^{o}C)
  - Number of consecutive times the signal is above or below threshold for flag to assert and de-assert



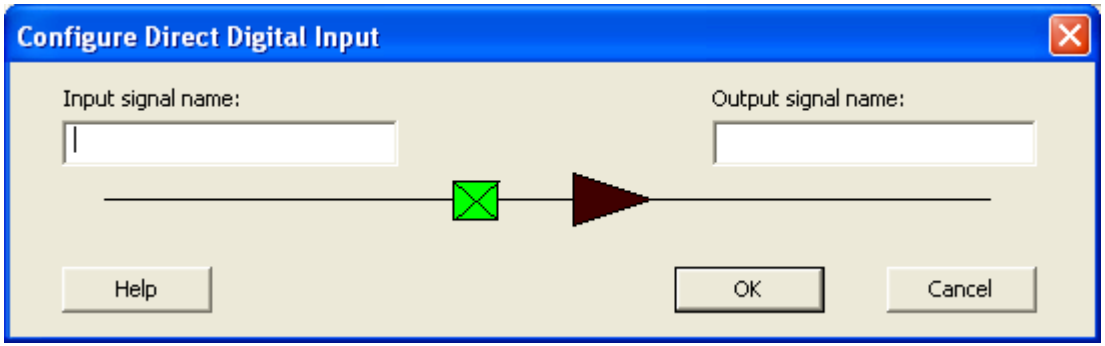
Output Gate Driver

- Analog Output from the FPGA
- Controlled via the Gate Driver Enable
  - Active High
  - Generated with FPGA gates
- Turn External PMOS or NMOS Transistor ON or OFF (Gate Driver Polarity)
- Can be Driven by Flag Logic from Analog System to Build a Self-controlled Design



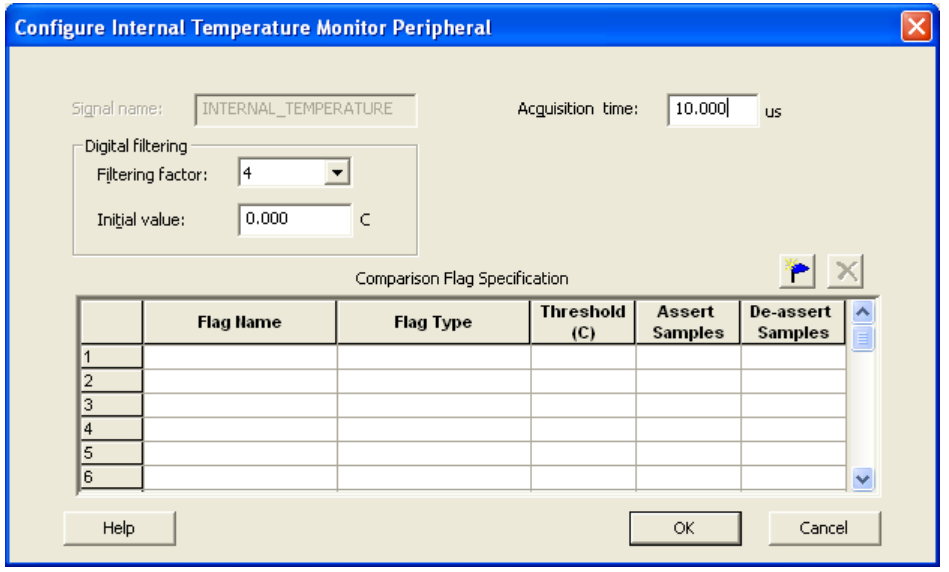
Direct Digital Input

- Enables Use of Unused Analog Inputs as Slow Digital Inputs
- Useful if Running Out of I/Os in a Design
- Does Not Need to be Sequenced Via Sample Sequencer
- Does Not Impact Throughput of the Analog System



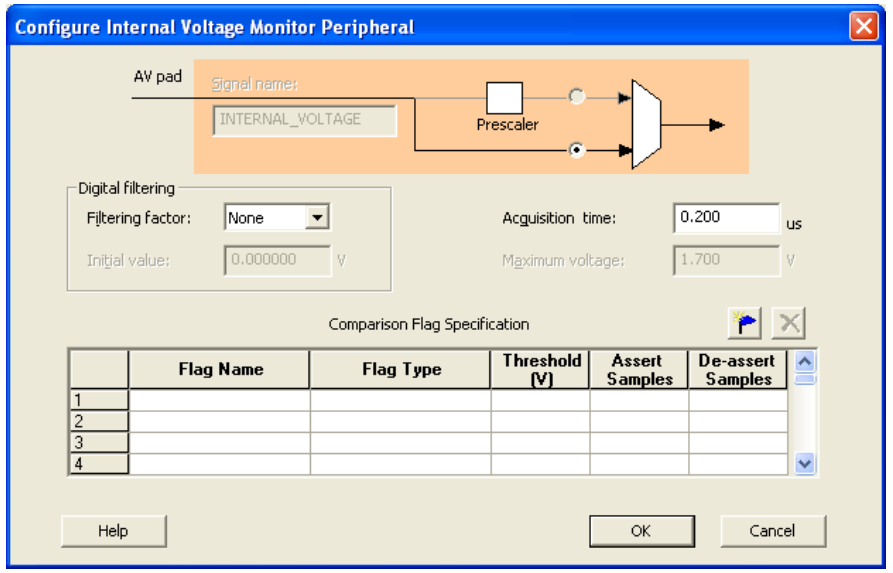
Internal Temperature Monitor

- Used to Monitor the Chip Temperature
- 32nd Channel in the Analog Mux
- Similar to External Temperature Monitor
- Flags Can be Used to Detect Overheating of the Device
- One Per Device



Internal Voltage Monitor

- Used to Monitor 1.5V Supply to FPGA Fabric and NVM
- Channel 0 on the Analog Mux
- Similar to Voltage Monitoring
- Flags Can be Used to Detect and Handle Brown-out Conditions
- One Per Device

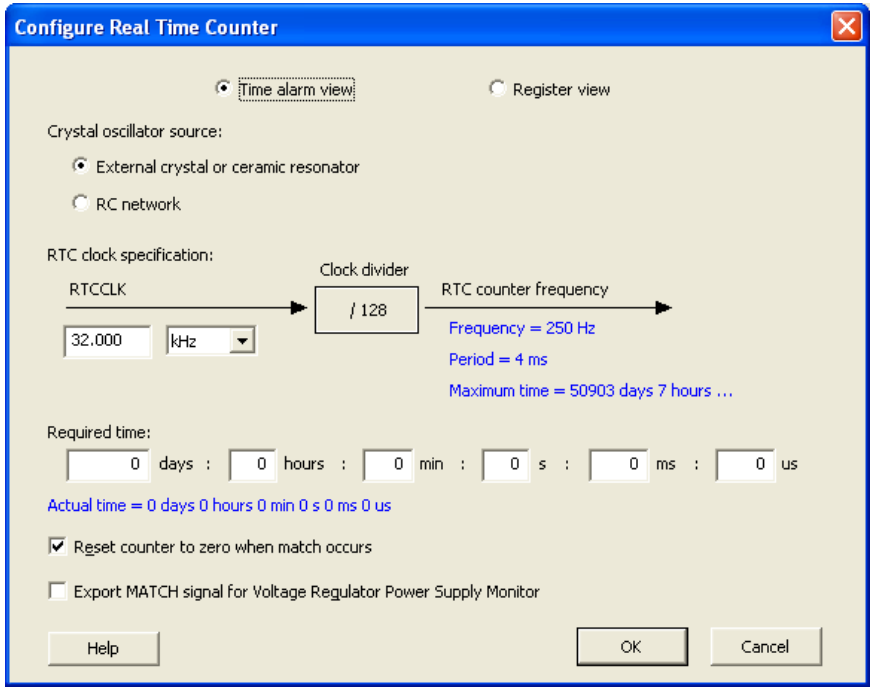


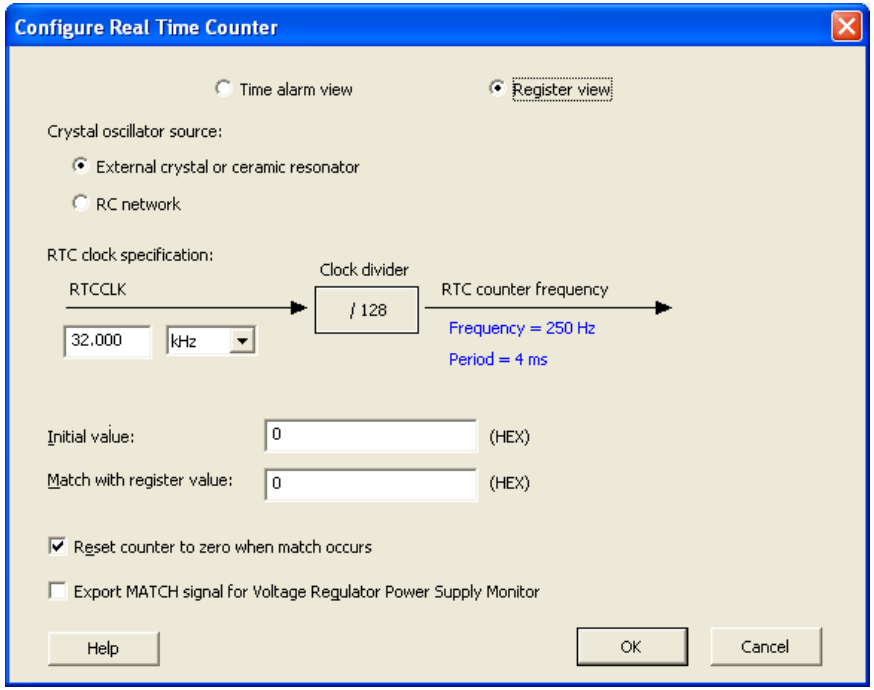
RTC Configuration in SmartGen

- Crystal Oscillator Mode Control
  - RTC CLK must be driven by XTLOSC
  - RTC controls XTLOSC Mode
    - Low Gain (32 kHz to 200 kHz)
    - Medium Gain (200 kHz to 2 MHz)
    - High Gain (2 MHz to 20 MHz)
- Match with Register Value
  - Triggers MATCH when counter equals 40-bit match value
- Initial value
  - Can specify non-zero value
- Reset Counter to Zero
  - Works as a timer application when chosen
  - Works as an elapsed time record if not chosen
- Export MATCH Signal
  - Asserts RTCPSMMATCH to activate Voltage Regulator Power Supply Monitor (VRPSM)

RTC Configuration in SmartGen

- Two Views in SmartGen
  - Enter desired time or match value



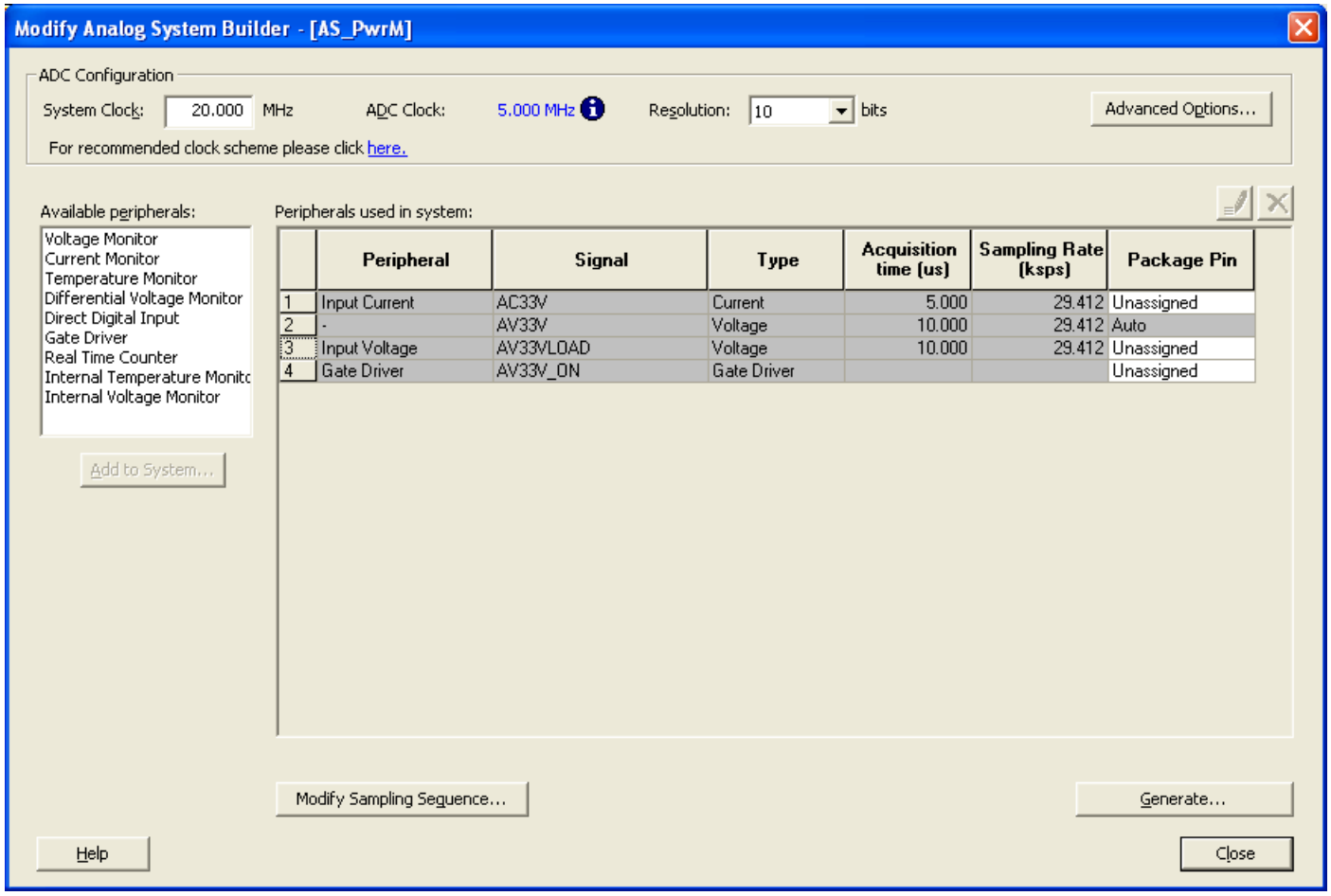


RTC Design Considerations

- Uses Analog System Configuration Bus for Setting the Various Values
- Can be Used with VRPSM & XTLOSC to Create a Self Wake-up Application
- Analog Block System Clock F_{MAX} = 100 MHz *but*
  - RTC Initialization clock F_{MAX} = 10 MHz

Analog System Builder with Peripherals

- Analog System Builder After Adding Peripherals



Sample Sequencer

- ADC has 32 Input Channels
  - 30 external inputs + Internal Temperature + Internal Voltage
  - Each channel must be sampled individually
  - TDM Sequencing Used (64 Time Slots)
- SmartGen Sample Sequence “Procedures”
  - “Procedure” is a named group of samplings
    - (Power Up; Calibrate; Run; Standby; Power Down, etc.)
  - “Procedure” can be as small as one and as many as 64 samplings
  - Support for Multiple Independent “Procedures” and Repeat “Procedures”
  - Jump to New Procedure Caused by Terminating Operation in a Sequence or an External Trigger
  - Supports use models where different channels are sampled at different times during system operation
    - Example: power-up sequence and normal operation sequence

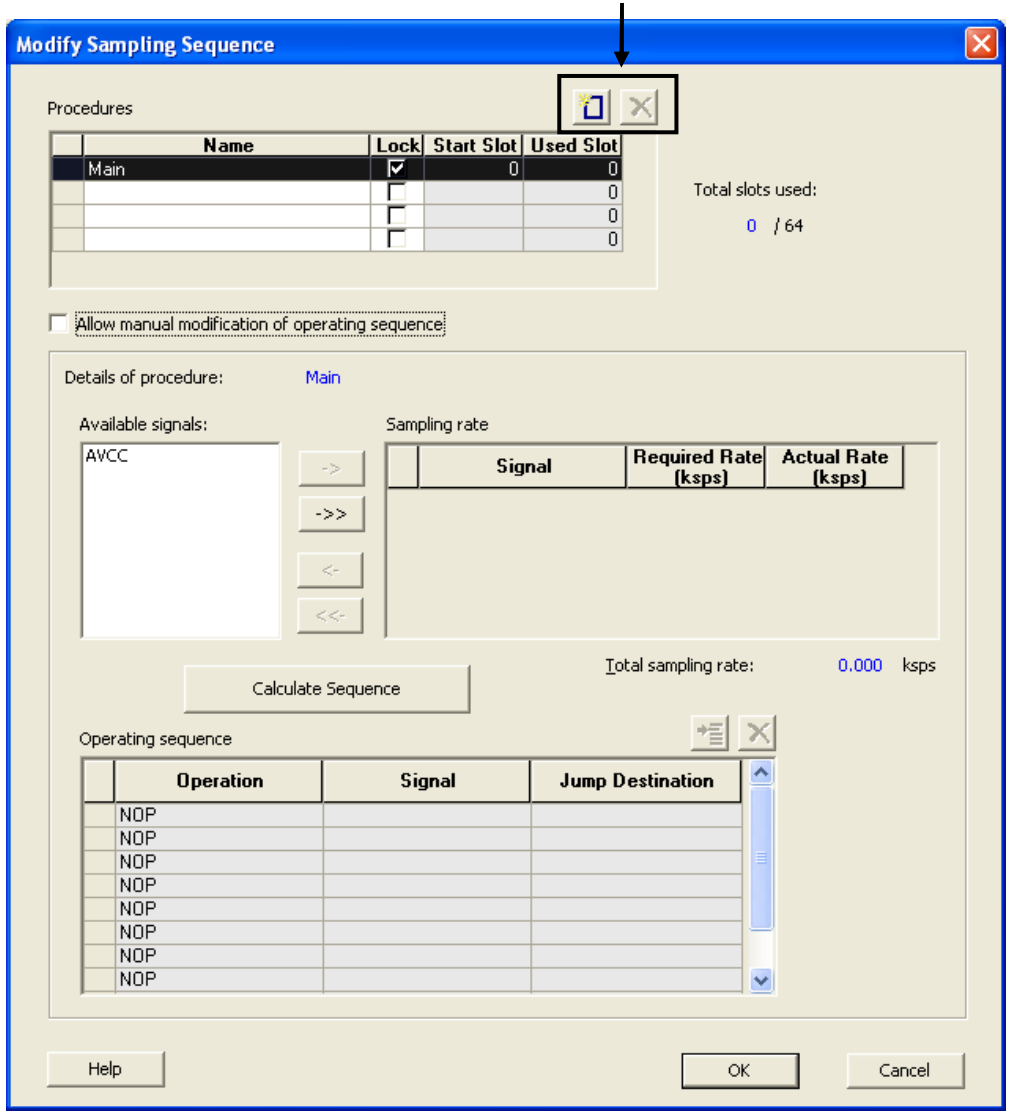
Sample Sequence Specification

- Main Procedure – System Default
  - Always starts at slot 0 and always executed upon reset
  - Cannot be deleted or unlocked
- Additional Procedures
  - Define additional procedures based on system requirements
  - Procedures Must be Triggered by External User Logic Through the External ASSC Control ports
  - Two modes:
    - Jump and continue
      - Assert ASSC_SEQJUMP for 1 clock to execute slot at ASSC_SEQIN
    - Single Step
      - Set ASSC_XMODE = 1, to enable single step through sequencer
      - Assert ASSC_XTRIG for 1 clock to execute slot at ASSC_SEQIN

SmartGen

Sample Sequencer GUI

Add or Delete a Procedure



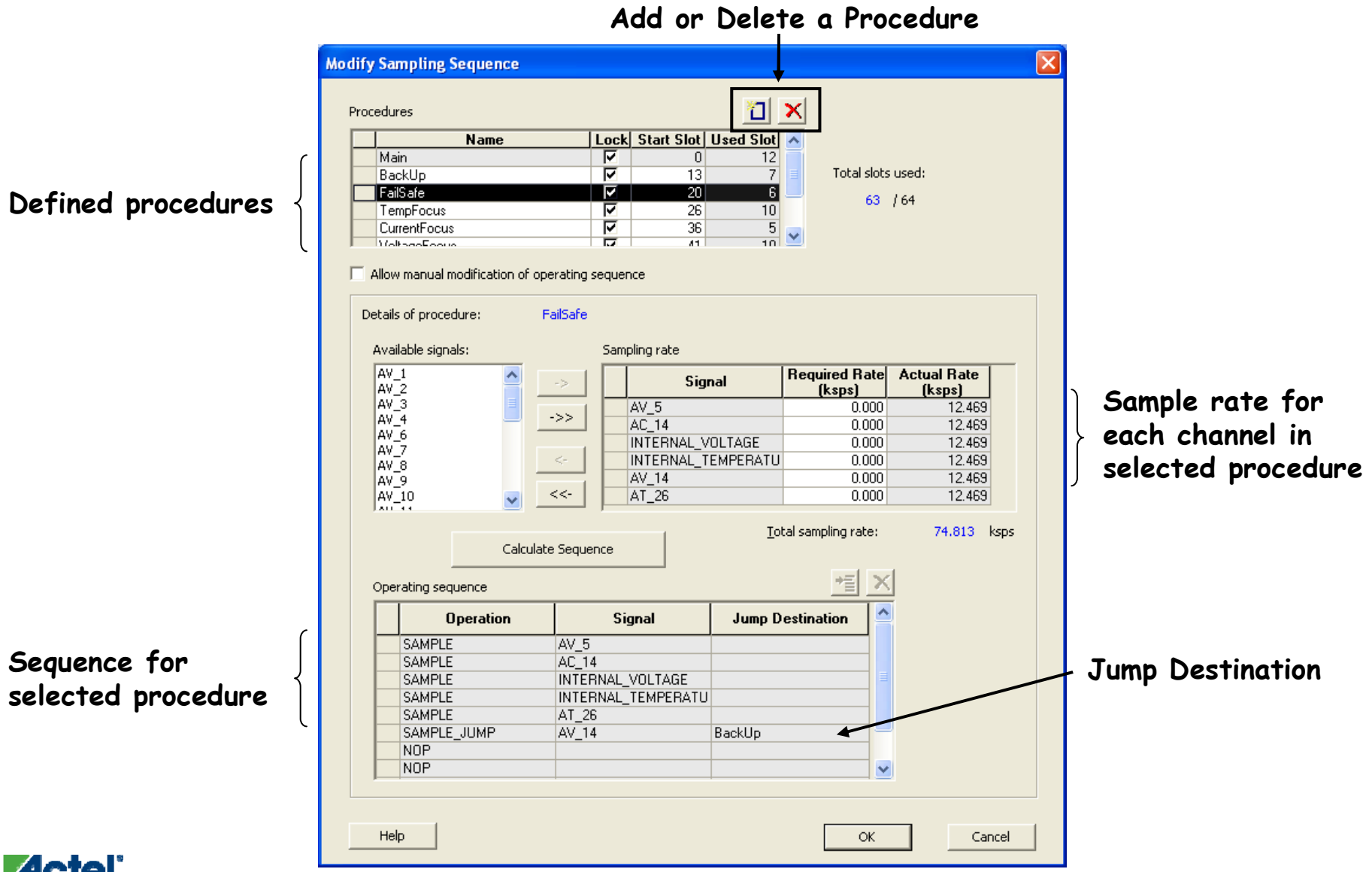
Sequence for selected procedure

Sampling Sequence Operations

- Available Operations:
  - SAMPLE - Sample a channel that is configured in the system and proceed to the next slot
  - SAMPLE_JUMP – Sample a channel that is configured in the system and jump to the start of the specified procedure
  - CALIBRATE – Perform a full calibration of the Fusion ADC and proceed to the next slot
    - Takes 3840 ADC Clock cycles
  - CALIBRATE_JUMP – Perform a full calibration of the Fusion ADC and jump to the start of the specified procedure
  - JUMP – Jump to the start of the specified procedure
  - POWERDOWN – Perform a powerdown operation on the ADC;
    - After a powerdown is initiated, a calibration operation is required to resume sampling
  - STOP – Stop the sequencer
    - An external trigger is required to re-start the sequencer
  - NOP – No operation is performed and proceed to the next slot
    - NOP's in the middle of a sequence use up a time slot; NOP's after the end of the last functional slot do not use up a time slot

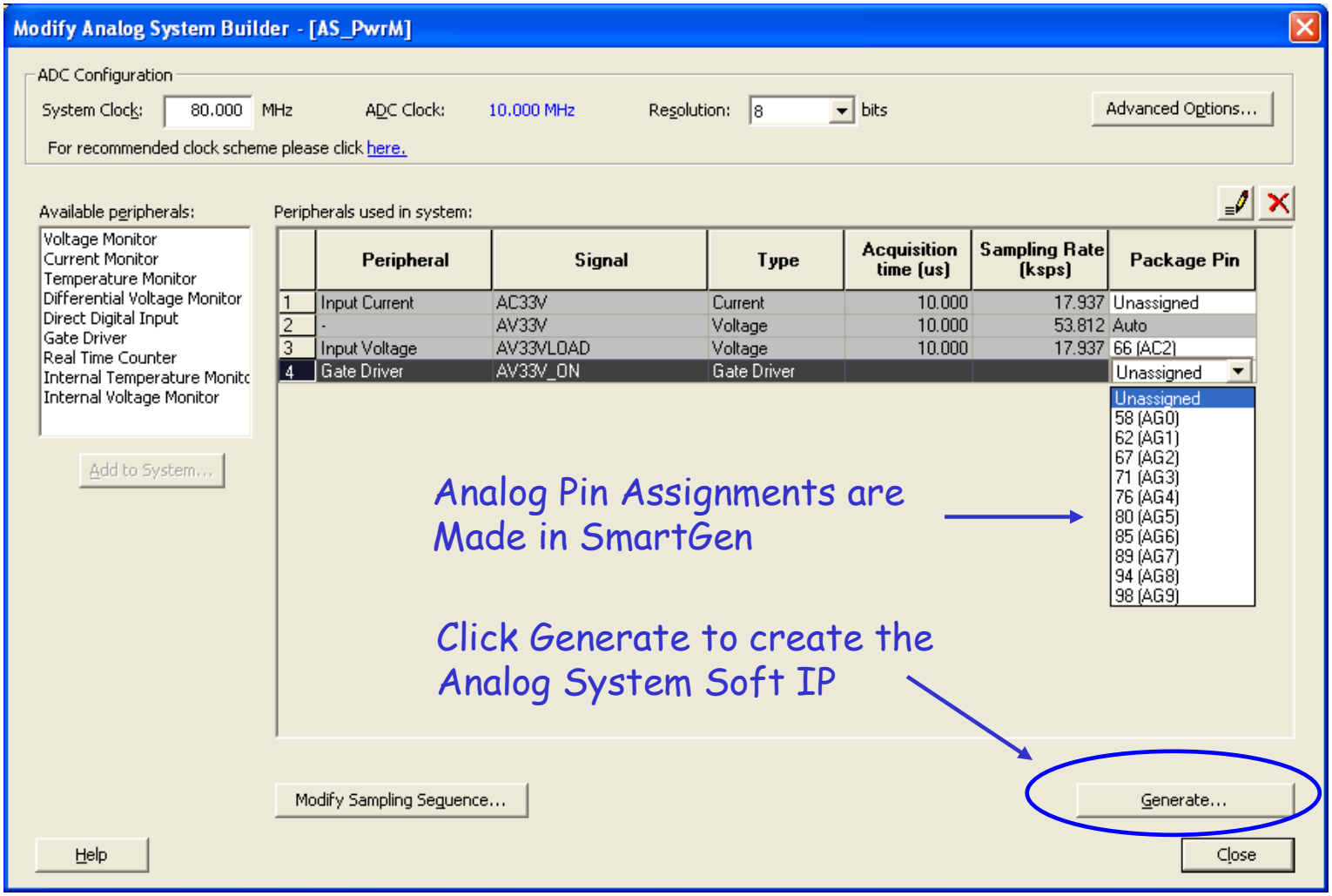
SmartGen

Sample Sequencer with Procedures



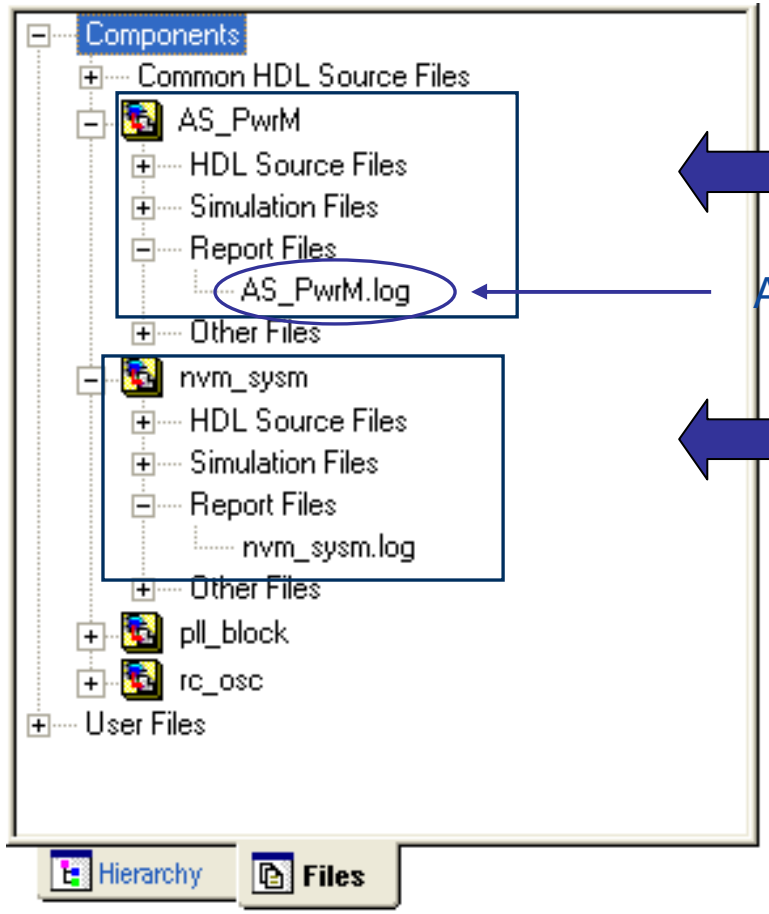
Analog System Builder

Pin Assignments and Generate



Fusion Backbone

SmartGen Output Files



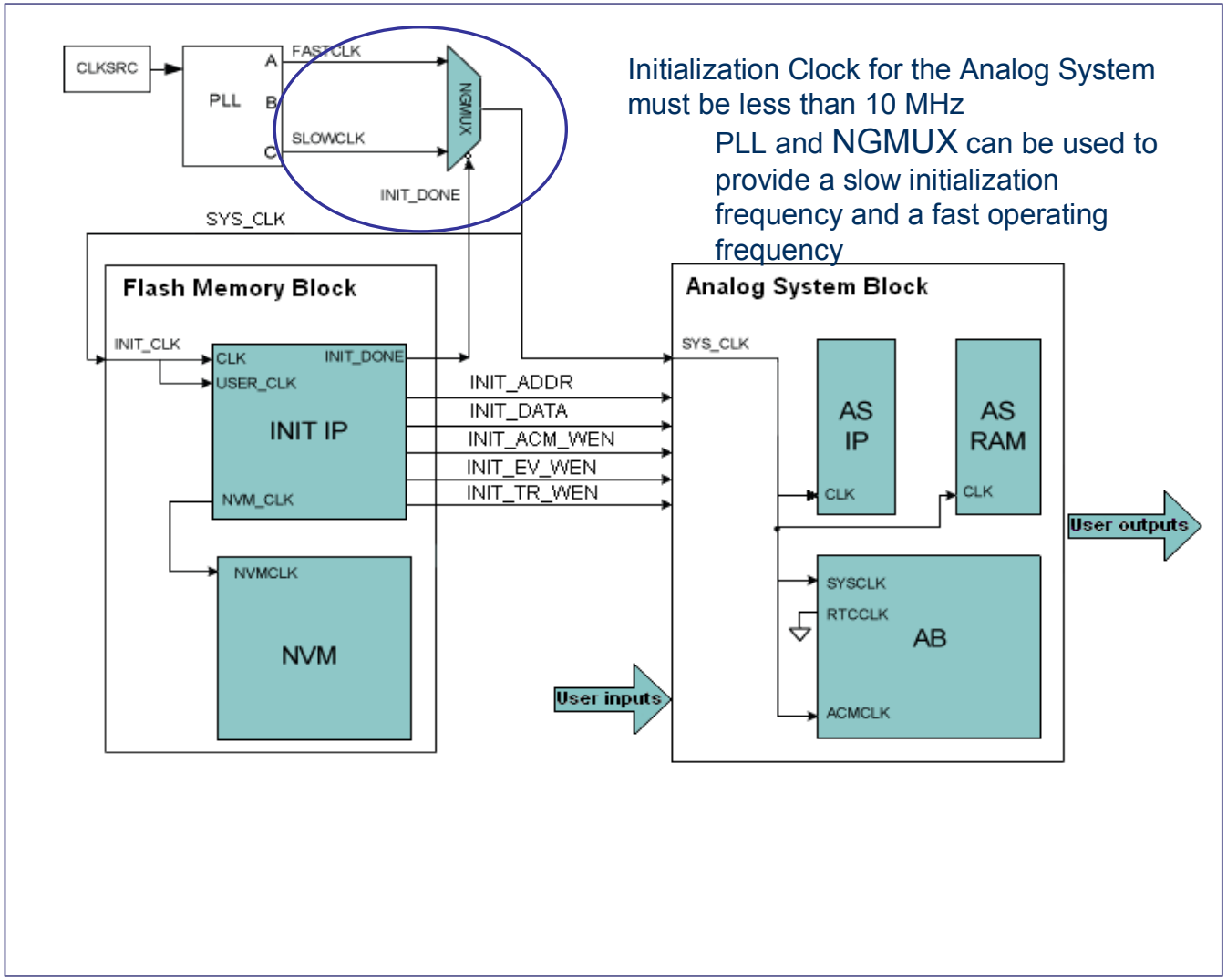
Analog System Builder Output

Analog System Log File

Flash Memory System Builder Output

Fusion Analog Block IP

Basic Configuration



Initialization Clock for the Analog System must be less than 10 MHz

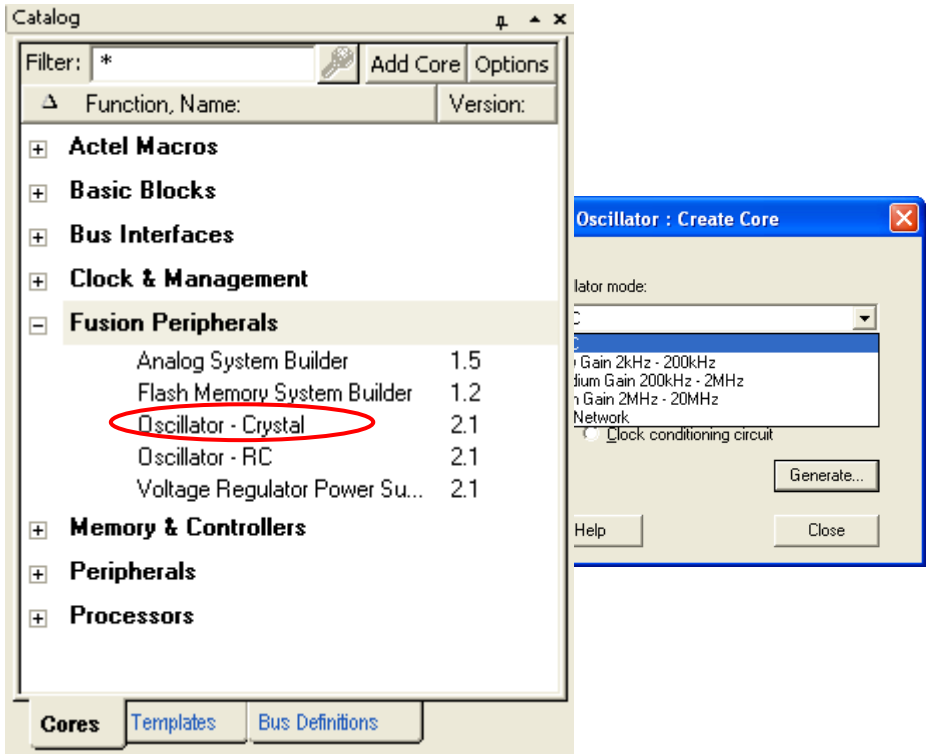
- PLL and NGMUX can be used to provide a slow initialization frequency and a fast operating frequency

Fusion Design Flow

- Fusion Smart Backbone
- Libero Flow
- SmartGen
  - Analog System Builder
  - Flash Memory System Builder
  - Other SmartGen Cores
- Simulation
- Programming and Security

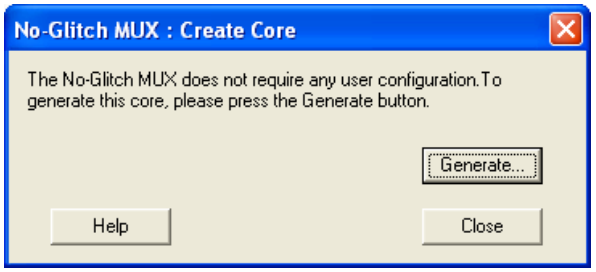
Crystal Oscillator

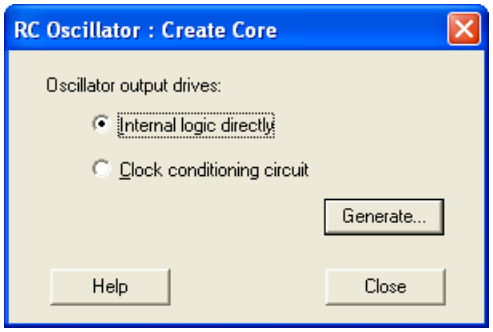
- New for Fusion
- Mode control from FPGA or RTC
- Mode selection for Gain
  - Low Gain (32 – 200KHz)
  - Medium Gain (200KHz - 2MHz)
  - High Gain (2MHz - 20MHz)

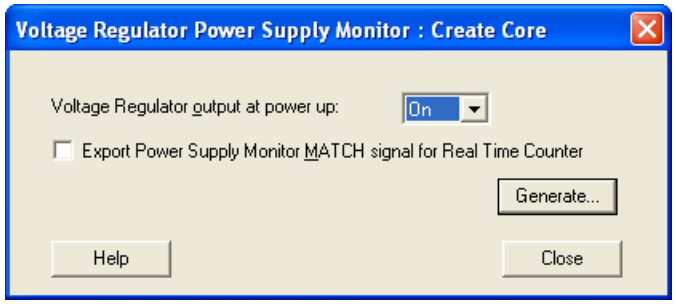


VRPSM, NGMUX and RCOSC

- Very few configuration options
- NGMUX and RCOSC available under Clock Conditioning Cores
- VRPSM in Voltage Regulator Category





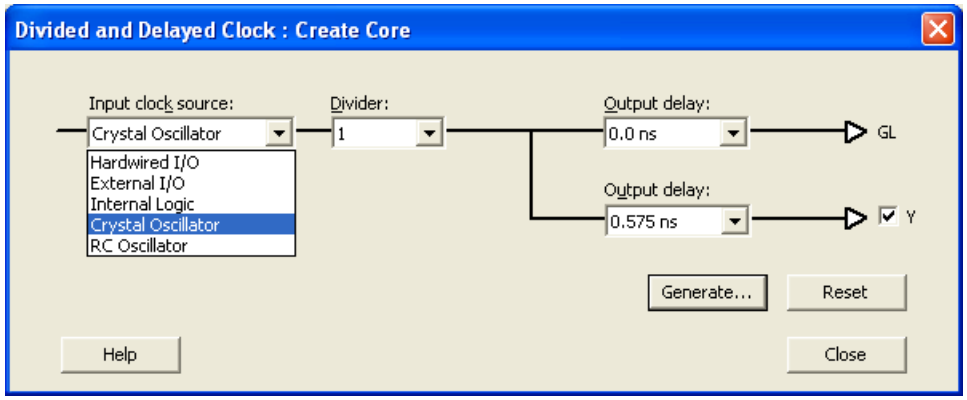


Static PLL

- Similar to ProASIC3 PLL
  - Dividers, Delays, Phase-shift & MUX selections
- New Input Clock Sources
  - Crystal Oscillator
  - RC Oscillator
- GLA Output Divider Available in PLL Bypass Mode
  - Divider values 1 - 32
- RC Oscillator Clock Source Considerations
  - Input Clock frequency is 100MHz
  - Extra Divide-by-half feature available
    - Dividers with values 0.5, 1.5, 2.5 …. 31.5
- SmartGen does not Include the XTL/RCOSC Library Macro as part of PLL Generation
  - User must manually connect XTLOSC or RCOSC to CLKA

Divided and Delayed Clock

- New for Fusion
- Divide a Clock and Optionally Delay by a Given Amount
- Clock Source Options Same as Static PLL
- Same divider Options as static PLL Including the RC Oscillator Divide-by-half
- Divider RESET for Predictable Edge Synchronization Between Input and Output Clock



Fusion Design Flow

- Fusion Smart Backbone
- Libero Flow
- SmartGen
- Simulation
  - Analog Block Simulation
  - Flash Memory Block Simulation
- Programming and Security

Supported Simulators

- The Analog Block Simulation Flow Works With All Digital Simulators That Actel Supports
  - ModelSim VHDL
  - Cadence NC-VHDL
  - ModelSim Verilog
  - Cadence NC-Sim
  - Cadence NC-Verilog
- The Digital Simulation Flow Correctly Handles All 37 Analog Inputs
  - VAREF, GNDREF, AV0-AV9, AC0-AC9, AT0-AT9, ATRETURN01, ATRETURN23, ATRETURN45, ATRETURN67, ATRETURN89
- Analog Block Simulation Requires Using Actel-supplied VHDL Procedures or Verilog Modules

Input Stimulus for Analog System

- Create Analog System Netlist Using SmartGen
  - Each analog pin will be configured as either an analog or a digital input
- Use analog_io Package Contained in

```
<drive>:\Actel\Libero_v8.5\Designer\lib\vtl\95\fusion.vhd
library fusion;
use fusion.analog_io.all;
```

- For Input AV0 Configured as an Analog Input

```
drive_analog_input( 0.3, AV0 );
```

  - *This converts the real number 0.3 into a serial-bit-stream which drives AV0*
- For Input AV1 Configured as a Digital Input

```
AV1 <= '1';
```

VHDL Simulation

Analog IO Procedures

- Use the Procedures Contained in Package **analog_io**

```
procedure drive_analog_input(analog_val       : in  real;
                             signal serial    : out std_logic);
   procedure drive_current_inputs(volt_real   : in  real;
                                  resistor    : in  real;
                                  current     : in  real;
                                  signal av   : out std_logic;
                                  signal ac   : out std_logic );

    procedure drive_differential_inputs(volt_real : in  real;
                                  delta       : in  real;
                                  signal av   : out std_logic;
                                  signal ac   : out std_logic );

    procedure drive_temperature_quad(temp_celsius : in  real;
                                     signal serial: out std_logic );
```

Driving Analog Input
VHDL Simulation

- Pass a Constant into Procedure
  - drive_analog_input(0.3, AV0);
  - This converts the real number 0.3 into a serial-bit-stream which drives AV0
- Pass a Real Variable into Procedure

```
variable  AV0real  :  real;
AV0real  :=  0.3;
drive_analog_input( AV0real,  AV0 );
```

VHDL Simulation

Driving Analog Input Using Signal (Not Recommended)

- VHDL Signal Assignments do not Occur Instantly
  - Example of poorly written code with unexpected behavior

```
signal  AV0real  :  real;
AV0real  <=  0.4;
wait on AV0real;
AV0real  <=  0.3;
drive_analog_input( AV0real,AV0 );   -- AV0 will be assigned 0.4 not 0.3
```

- Make Sure that the Signal Value has been Updated when Passing a Signal into drive_analog_input
  - A wait statement can be used for this

```
signal AV0real:  real;
serialize_AV0:process
begin
 wait on AV0real;
 drive_analog_input (  AV0real,  AV0  );
end process serialize_AV0
```

VHDL Simulation

Restrictions

- You Cannot Call **`drive_analog_input`** From a Process With a Sensitivity List
  - `drive_analog_input` contains wait statements
  - NC-VHDL does not allow a subprogram with wait statements to be called from a process with a sensitivity list

Input Stimulus for Analog System

Verilog Simulation

- Create Analog System Netlist Using SmartGen
  - Each analog pin will be configured as either an analog or a digital input
- Use the Verilog Modules drive_analog_io and read_analog_io contained in:

```
<drive>:\Actel\Libero_v8.5\Designer\lib\vlog\fusion.v
```

- For Input AV0 Configured as an Analog Input:

```
drive_analog_io drive_AV0( 0.3, AV0 );
```

  - This converts the real number 0.3 into a serial-bit-stream which drives AV0
- For Input AV1 Configured as a Digital Input:

```
assign AV1 = 1'b1;
```

Verilog Simulation

Analog IO Modules

- **Use the Modules Provided by Actel**

```
module drive_analog_input (parallel_in, serial_out);

module drive_current_inputs (volt_vect, resistor_vect, current_vect, av,
                             ac);

module drive_differential_inputs (volt_vect, delta_vect, av, ac);

module drive_temperature_quad (temp_celsius, serial_out);
```

Verilog Simulation
Restrictions

- You Cannot Instantiate **`drive_analog_io`** in a Procedural Block
  - Must instantiate macro outside the procedural block

WaveFormer Lite

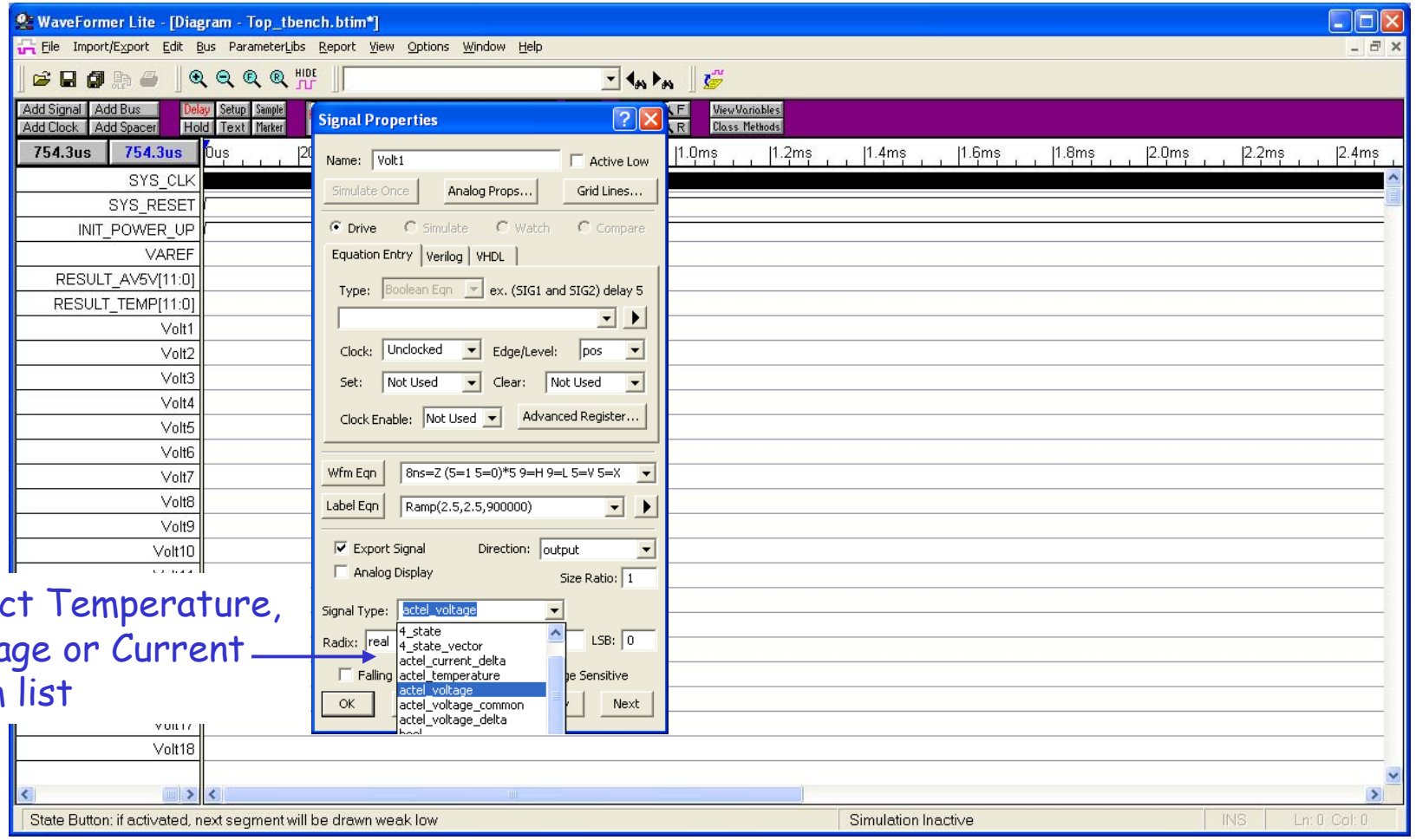
- Supports Creation of Analog Stimulus for Fusion
  - Pre-defined analog waveforms are available such as Sinusoidal, Step, Increment, Random, and RC discharge
  - VHDL Testbench includes VHDL procedures in analog_io package

    drive_current_monitor(volt_real, resistor, current, signal_serial);
    drive_current_inputs(volt_real, resistor, current, signal_av, signal_ac);
    drive_differential_inputs(volt_real, delta, signal_av, signal_ac);
    drive_temperature_quad(temp_celsius, signal_serial);
  - Verilog Testbench includes modules for analog inputs provided by Actel

    module drive_analog_input (parallel_in, serial_out);
    module drive_current_inputs (volt_vect, resistor_vect, current_vect, av, ac);
    module drive_differential_inputs (volt_vect, delta_vect, av, ac);
    module drive_varef_out (parallel_in, en_out, serial_out);
    module drive_current_monitor (volt_vect, resistor_vect, current_vect, serial_out);
    module drive_temperature_quad (temp_celsius, serial_out);

Fusion Analog Signals

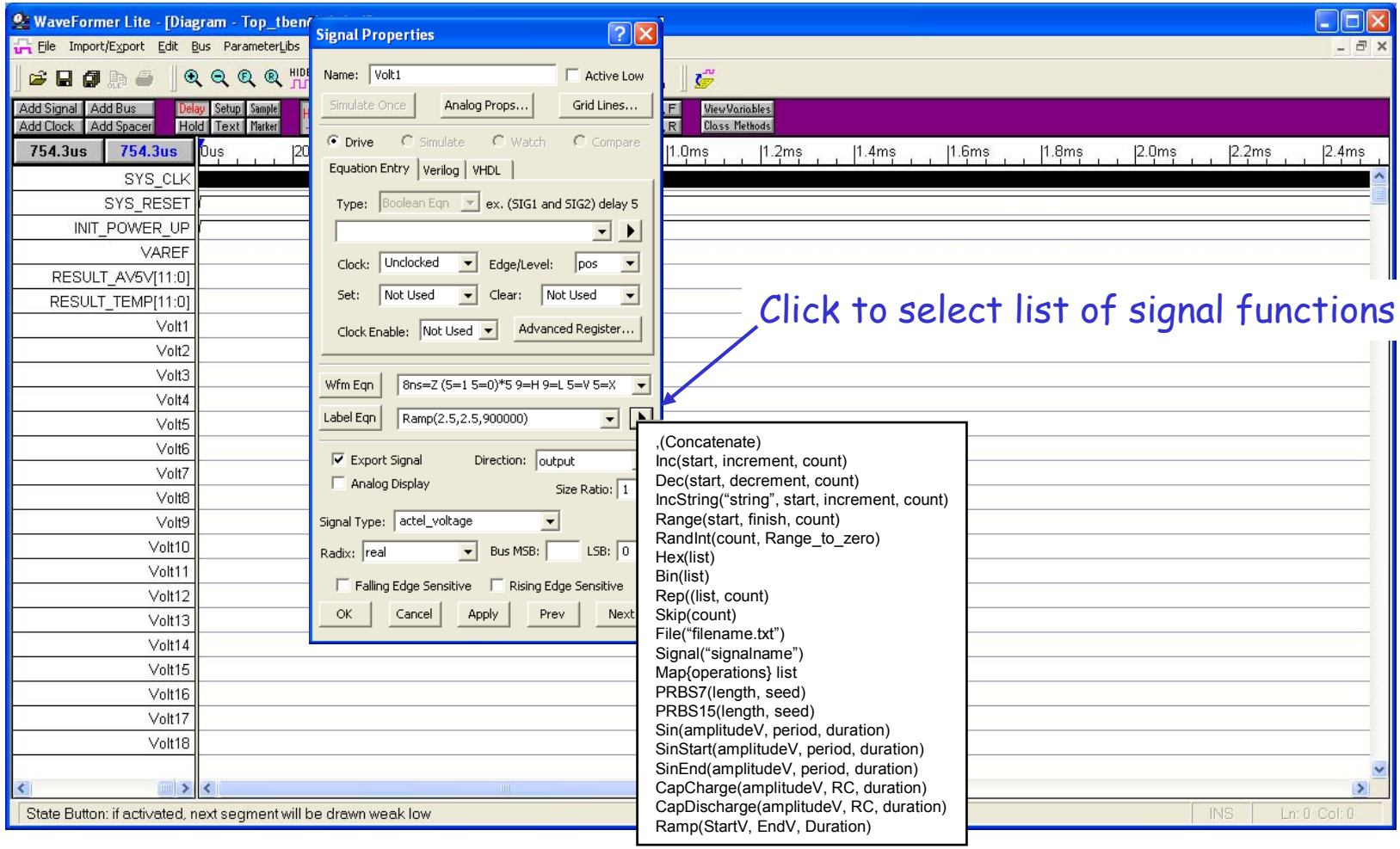
Specify Signal Type



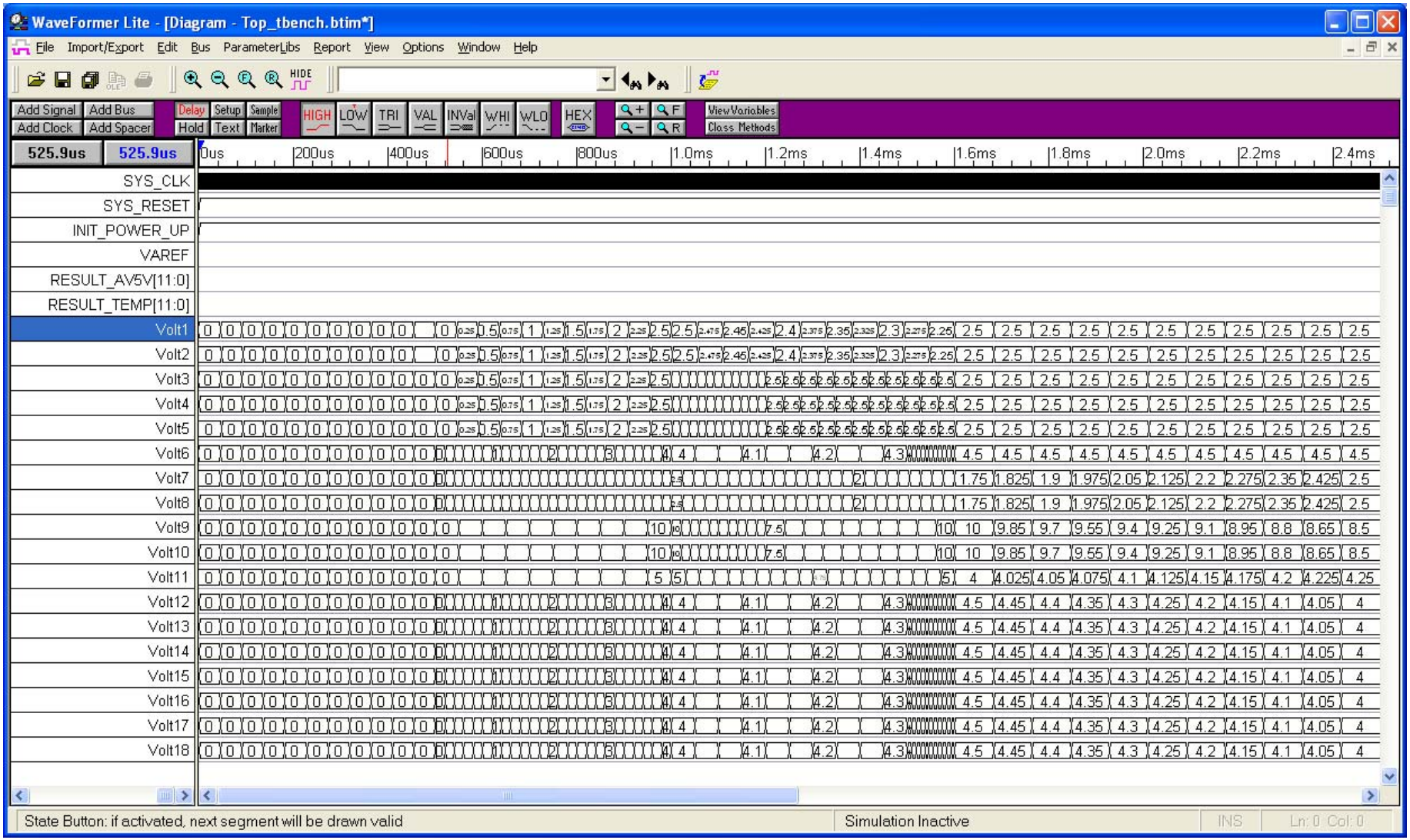
Select Temperature, Voltage or Current from list

Fusion Analog Signals

Using Built-In Waveforms



Fusion Analog Signals



WaveFormer Lite

VHDL Testbench

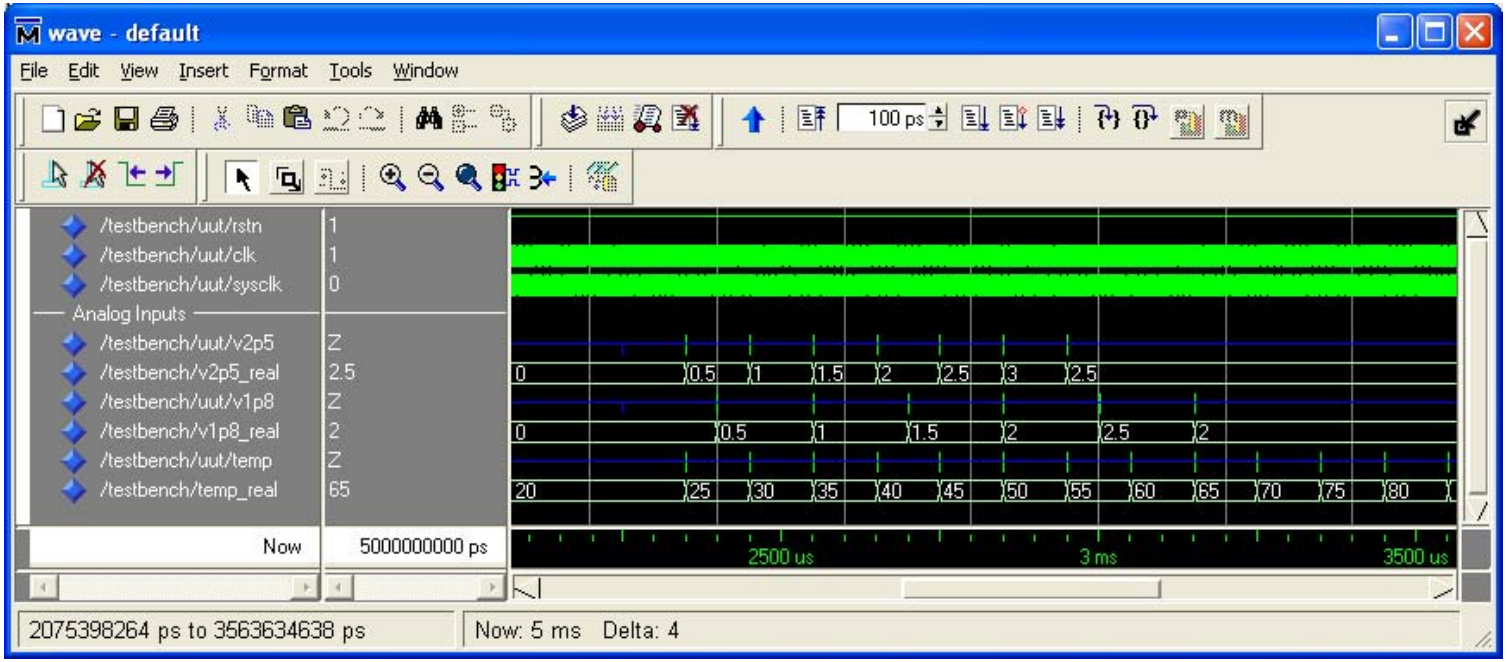
```
architecture STIMULATOR of stimulus is
  -- Control Signal Declarations
  signal AV5V_driver : real;
  signal Temp_driver : real;

 . . .

begin

 . . .

  -- Actel Analog Drivers Block
  drive_analog_input(AV5V_driver, AV5V);
  drive_temperature_quad(Temp_driver, Temp);

 . . .

 -- Sequence: Unclocked
  Unclocked : process
  begin
    AV5V_driver <= 0.0;
    Temp_driver <= 5.0;
    wait for 12 ns;
    INIT_POWER_UP <= '1';
    wait for 28 ns;
    SYS_RESET <= '1';
    wait for 379907 ns;
    AV5V_driver <= 0.1;
    wait for 20132 ns;
. . .
```

WaveFormer Lite
Verilog Testbench

```
module stimulus(SYS_CLK, SYS_RESET, INIT_POWER_UP, VAREF, Supply_good,
                Over_temp, ATRETURN01, AV5V, Temp);
  output SYS_CLK;
  output SYS_RESET;
  . . .
  reg SYS_RESET_driver;
  reg INIT_POWER_UP_driver;
  reg VAREF_driver;
  reg ATRETURN01_driver;
  real AV5V_driver;
  real Temp_driver;
  . . .
  drive_analog_input analog_AV5V_driver($realtobits(AV5V_driver), AV5V);
  drive_temperature_quad temperature_Temp_driver($realtobits(Temp_driver), Temp);
  . . .
task Unclocked;
    begin
    #12;
    INIT_POWER_UP_driver <= 1'b1;
    #26;
    SYS_RESET_driver <= 1'b1;
    #380193;
    AV5V_driver <= 0.1;
    #19;
    Temp_driver <= 5.75;
    #19725;
    Temp_driver <= 15.0;
    #25;
```

Viewing Values of Serialized Signals in Simulator

- Waveform Viewer Shows the Values of the Serialized Signals as ‘Z’
  - The serialization procedure in the simulation models occurs in zero simulation time, using delta delays
  - The remainder of the time the serialized signal is ‘Z’
  - ModelSim displays both the time and the delta during simulation



Restrictions on AV/AC/AT Voltages for Current and Temperature Monitors

- Analog Voltage Reference
  - Selected by VAREFSEL
    - 0 = voltage supplied from internal reference = 2.56V
    - 1 = external voltage applied = VAREF
- Current Monitor
  - Maximum value is voltage reference
  - 10 times the difference in the absolute voltages applied on the AV and AC quads
  - If exceeded then the current monitor output will saturate at the reference voltage.
    - If internal reference used:
      - Saturates at 2.56V when difference between AV and AC is 0.256V or greater
- Temperature Monitor
  - Maximum value is voltage reference
  - 12.5 times the absolute voltage applied on the AT quad
  - If exceeded then the temperature monitor output will saturate at the reference voltage.
    - If internal reference used:
      - Saturates at 2.56V when AT is 0.2048V or greater

ADC
Clock Period and Conversion Time

- ADC_CLK
  - ADC internal clock
  - Minimum period is 100ns (10MHZ)
  - Maximum period is 2,000ns (0.5 MHZ)
  - Generated from user-controlled signals:
    - SYSCLK
    - TVC[7:0] (programming divider)
- Conversion Time
  - Dependent on user design requirements:
    - ADC_CLK period
    - STC[7:0] (sample time control)
    - ADC mode (8, 10, or 12-bit conversion)
  - Minimum conversion time is 1.2μs
    - 10MHZ ADC_CLK, 8-bit conversion, and sampling time of 2 ADC_CLK periods
  - Maximum conversion time is 542μs
    - 0.5MHZ ADC_CLK, 12-bit conversion, and sampling time of 257 ADC_CLK periods
  - RESULT will not change until conversion is finished.
  - When the conversion is finished, DATAVALID is asserted and RESULT is driven

Fusion Design Flow

- Fusion Smart Backbone
- Libero Flow
- SmartGen
- Simulation
  - Analog Block Simulation
  - Flash Memory Block Simulation
- Programming and Security

Addressing

- The NVM Array is Word-Addressable
  - DATAWIDTH signal used to select the size of the word
    - 1, 2, or 4 bytes
- Address bits are MSB Justified
  - Different from RAMs
  - For 1 byte access:
    - ADDR[17:0] are significant
  - For 2 byte access:
    - ADDR[17:1] are significant
    - ADDR[0] is ignored
  - For 4 byte access:
    - ADDR[17:2] are significant
    - ADDR[1:0] are ignored

Data

- Data bits are LSB Justified
  - Same as RAMs
  - For 1-byte access:
    - DO[7:0] are significant
    - DO[31:8] are 0
  - For 2-byte access:
    - DO[15:0] are significant
    - DO[31:16] are 0
  - For 4-byte access:
    - DO[31:0] are significant
- WEN Updates User Data into Page Buffer only
  - To store data into the NVM array, WEN should be followed by a PROGRAM of the same page

Cycle Accuracy

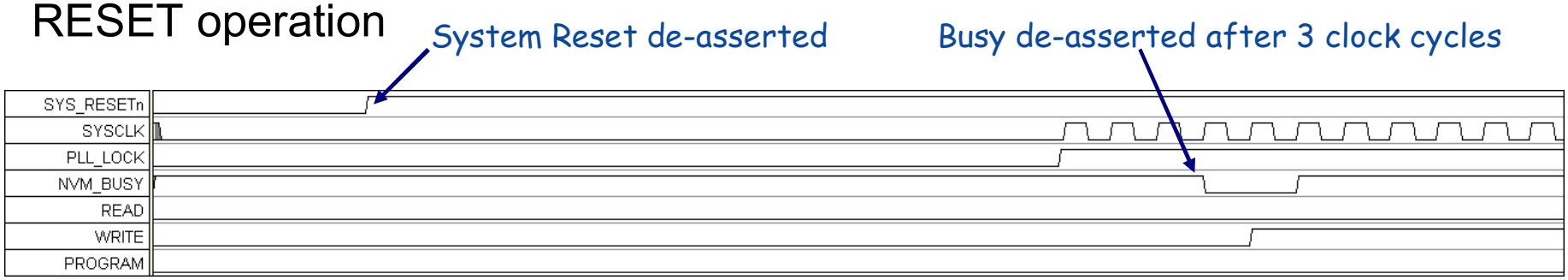
- Copy Page is Cycle-approximate
  - Affects any user operations which involve copy page operations
  - Cycle time is non-deterministic in silicon
    - Takes 63-67 clock cycles
  - Simulation models always execute copy page in 65 clock cycles
- Update Page Cycle Accuracy Controlled by Generic (VHDL) / Parameter (Verilog) FAST_SIM
  - Affects RESET, PROGRAM and ERASE operations

Cycle	Duration of Busy	
	FAST_SIM = 1(default)	FAST_SIM = 0
Reset	3 SYSCLK cycles	25 us
Program	4 us	8.4 ms
Erase		

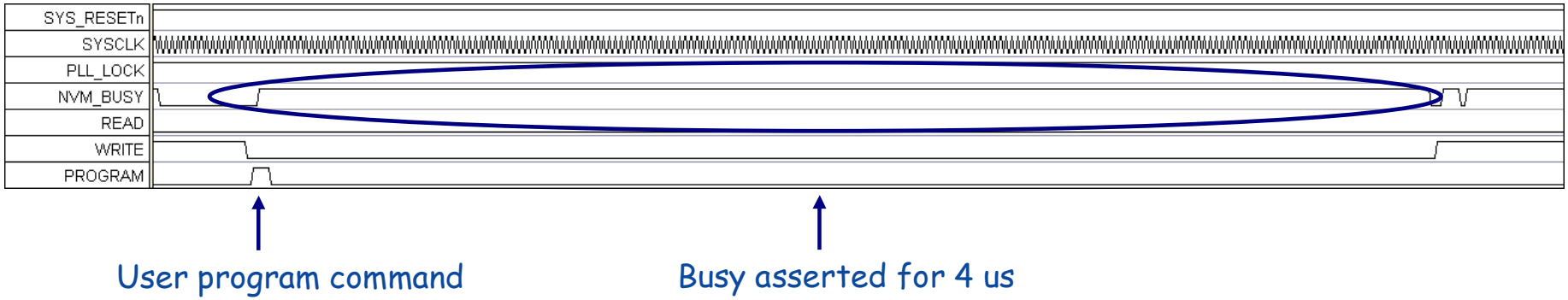
Busy Operation

FAST_SIM = 1 (default)

RESET operation



PROGRAM operation



Using FAST_SIM Generic / Parameter

- Modify NVM file from SmartGen

```
module nvm_block(
        . . .
    );
. . .
    NVM #( .MEMORYFILE("nvm_block.mem"), .FAST_SIM(0) )
    NVM_INST (.RD({ USER_DOUT[31], USER_DOUT[30], USER_DOUT[29], USER_DOUT[28],
        USER_DOUT[27], USER_DOUT[26], USER_DOUT[25], USER_DOUT[24],
        . . .
 USER_DATA[3], USER_DATA[2], USER_DATA[1], USER_DATA[0]}),
        . . .
    VCC VCC_power_inst1 (.Y(VCC_power_net1));
endmodule
```

add

nvm_block.v

```
architecture DEF_ARCH of nvm_block is
component NVM    generic (MEMORYFILE:string := "";
                          FAST_SIM: integer := 0
                 port( RD: out   std_logic_vector(31 downto 0);
                       BUSY: out   std_logic;
                       . . .
                      LOCKREQUEST: in std_logic := 'U');
end component;
. . .
begin
NVM_INST : NVM generic map(MEMORYFILE => "nvm_block.mem“)
               port map(RD(31) => USER_DOUT(31),
              . . ., LOCKREQUEST => USER_LOCK);
. . .
end DEF_ARCH;
```

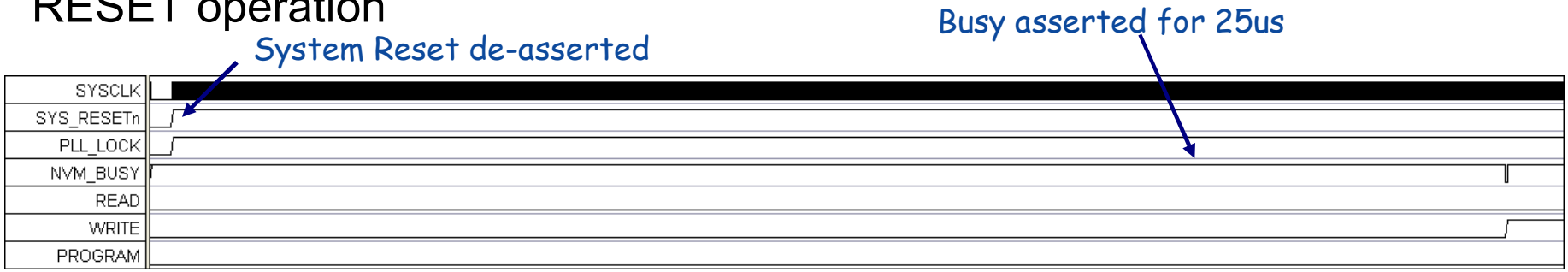
add

nvm_block.vhd

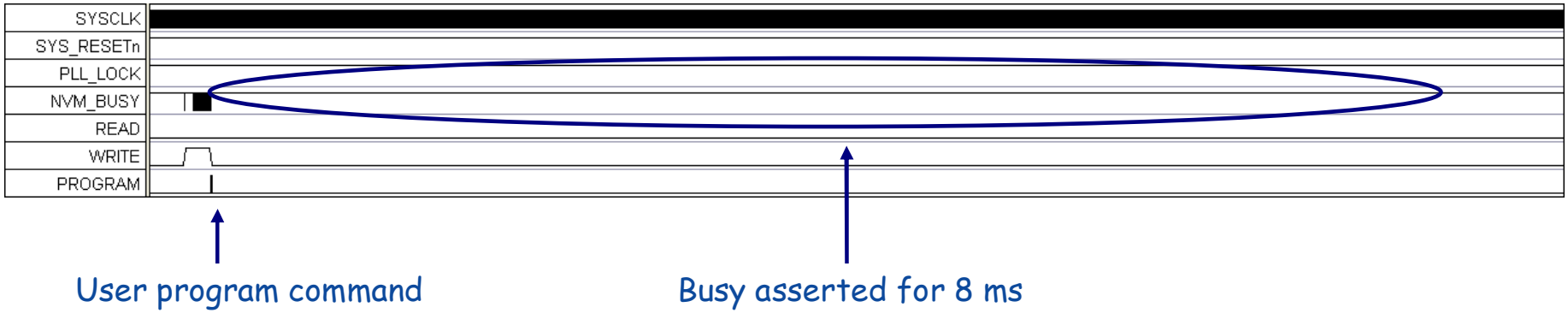
Busy Operation

FAST_SIM = 0

RESET operation



PROGRAM operation



Lab 2

- Complete Lab 2 in the Lab Guide
  - Open a project in Libero
  - Configure a Voltage Monitor, Current Monitor and Gate Driver
  - Simulate the Design

Fusion Design Flow

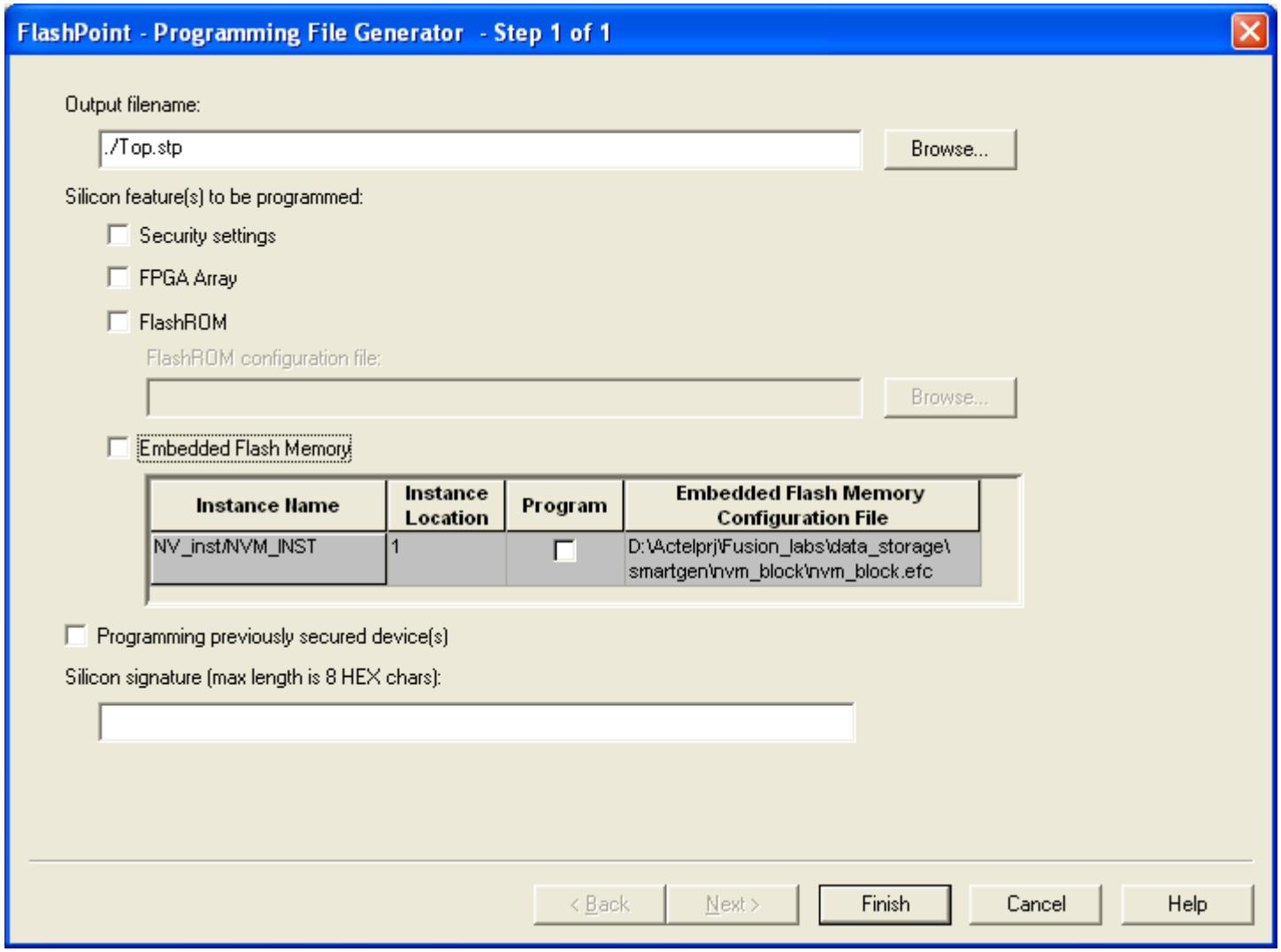
- Fusion Smart Backbone
- Libero Flow
- SmartGen
- Simulation
- Programming and Security

Programming & Security

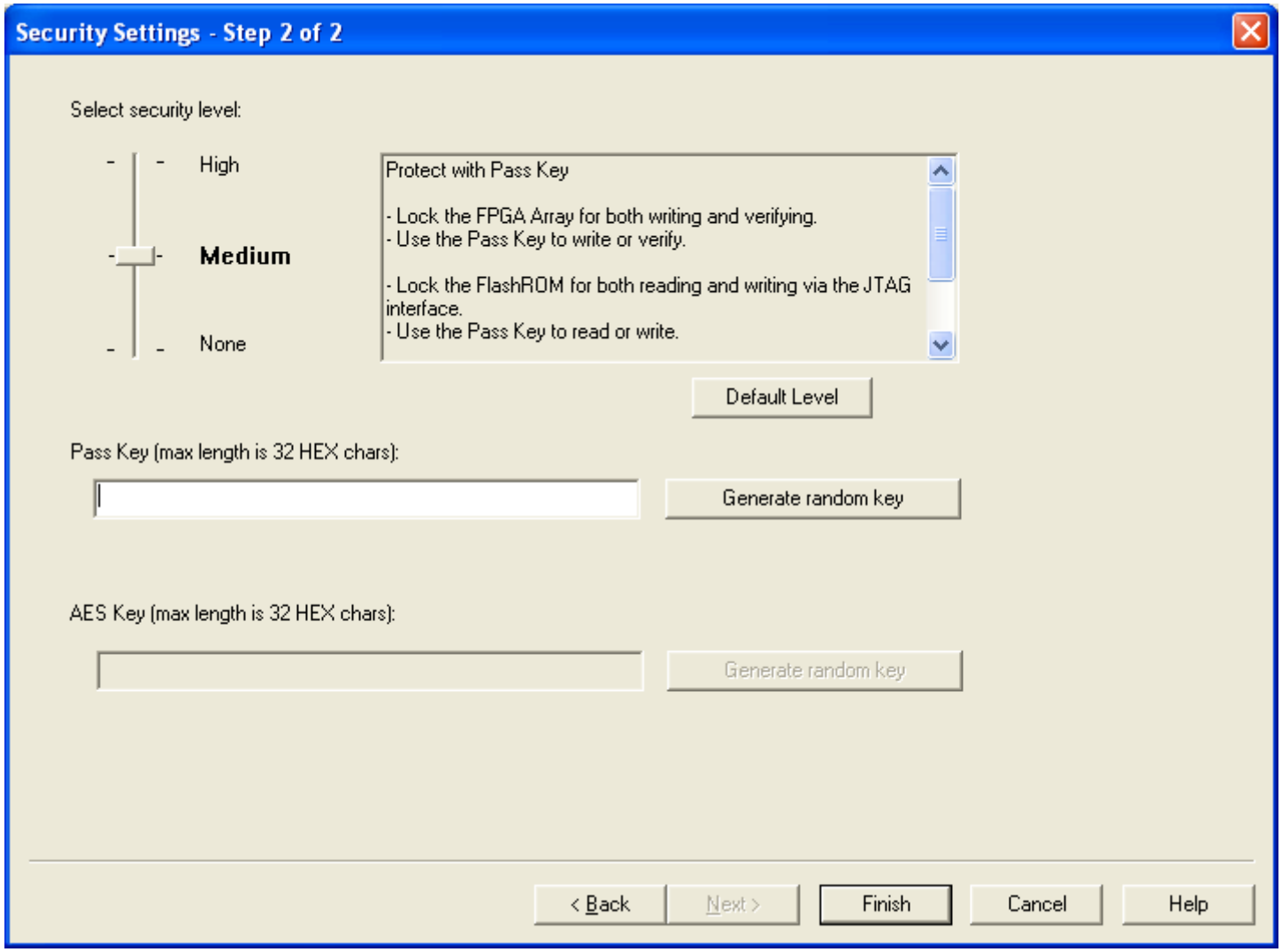
- Device Programming Scenarios
  - First-time Programming
    - Specify Security Information
  - Re-programming
    - Specify Previously-used Security Information
  - Changing Security Settings
- Environments
  - Trusted Programming Environment
    - Users May Have Access to Pass Key and AES Key
  - Un-trusted Programming Environments
    - Never Expose Pass Key or AES Key
    - Never Program Security Settings

FlashPoint

Start-up Screen

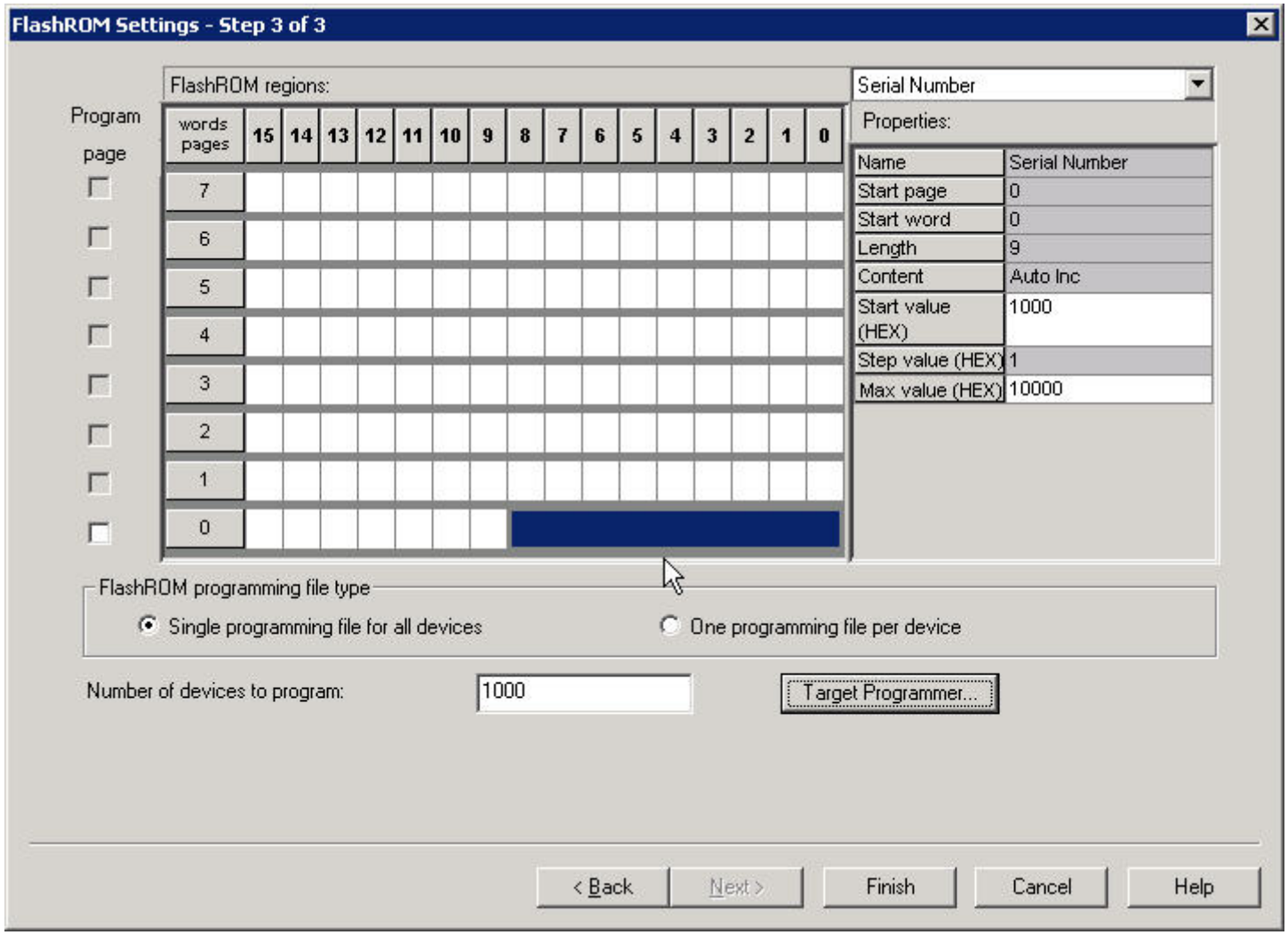


FlashPoint
Security Settings



FlashPoint

FlashROM



Fusion Starter Kit

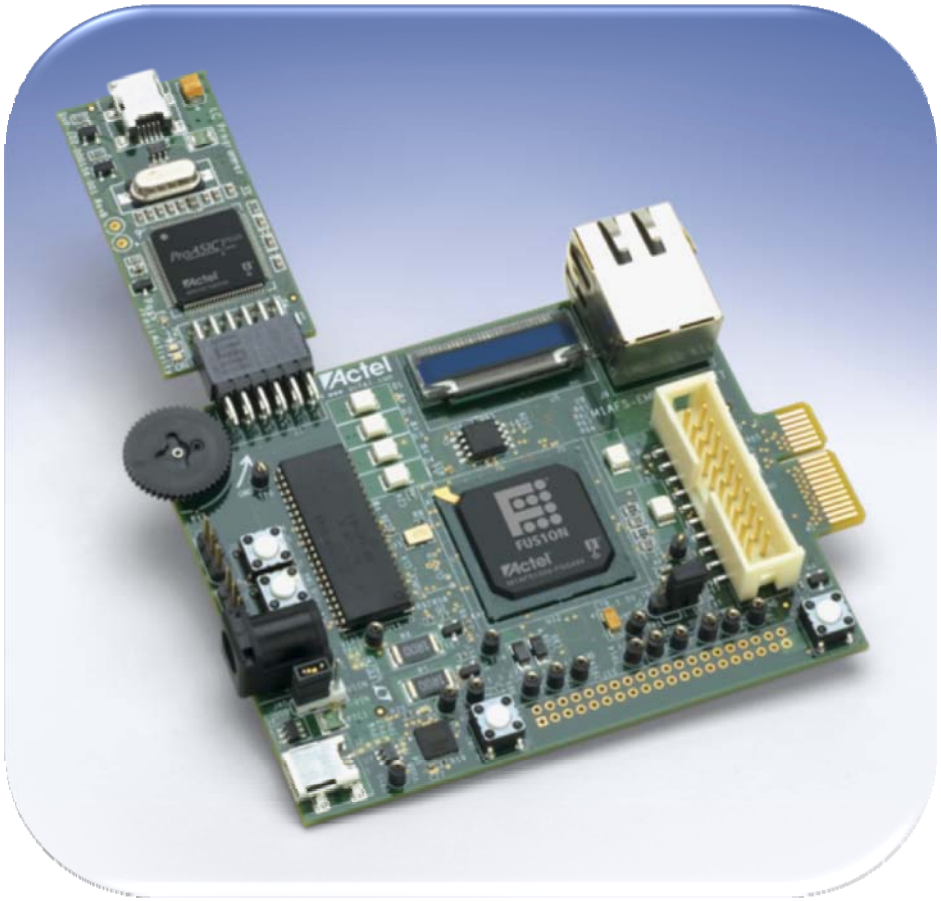
- Designed to Showcase Fusion
  - FPGA – single volt operation
  - Crystal – 32 kHz for RTC
  - Tri-color LED to exhibit PWM
  - MOSFET on board to power control
  - Potentiometer for analog voltages
  - LCD to display values
  - Temperature diode on board
  - Fusion can monitor own current draw!
- Starter Kit Features
  - All I/O brought out to headers
  - JTAG headers for programming & chain
  - Daughter card pins compatible w/PA3
  - Prototyping area
- Contents
  - Libero IDE Gold
  - Fusion evaluation board
  - FlashPro3
  - Tutorial and documentation
  - Logic Navigator debugger
- Available Now

Fusion Embedded Development Kit

- Develop designs using
  - Fusion Mixed-Signal FPGA
    - M1AFS1500-FGG484
  - Cortex-M1 embedded processor
  - 8051s embedded processor
  - Ethernet applications

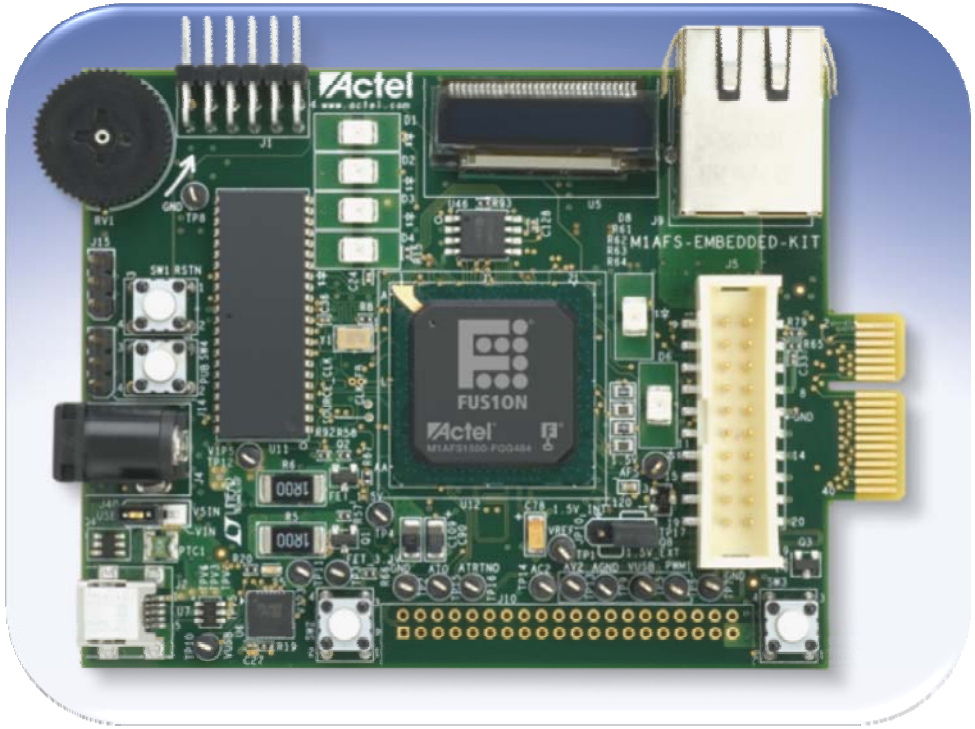
- The development kit includes
  - Low Cost Programming Stick
  - Libero IDE
  - Free Libero IDE Gold license
  - SoftConsole for Compile/Debug
  - On Chip Program and Debug
  - User's guide and tutorial
  - Example designs
  - PCB schematics and layout files

- Ordering code
  - M1AFS-EMBEDDED-KIT @ $199



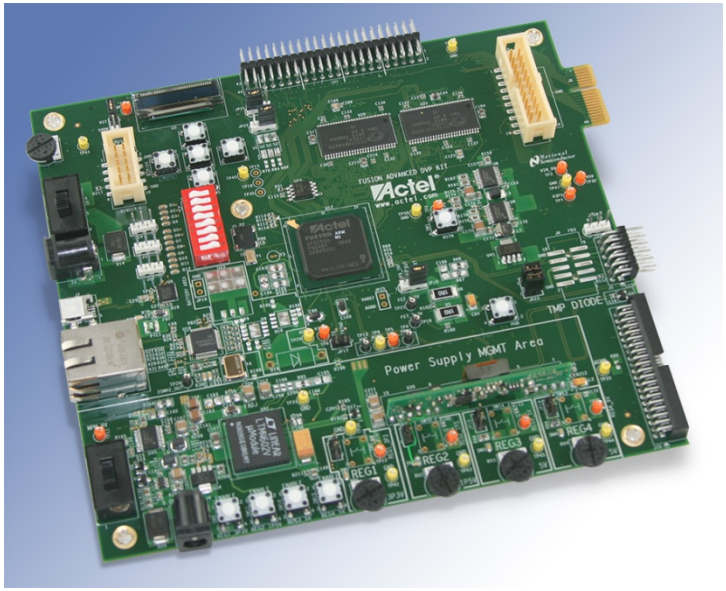
Fusion Embedded Development Board Features

- RoHS compliant
- 10/100 Ethernet interface
- USB-to-UART interface
- I2C interface
- Built-in temperature monitor
- Voltage potentiometer
- RealView debug header
- OLED 96x16 pixel display
- 4,000,000 SRAM



Fusion Advanced Development Kit

- Microprocessor and System Management Applications Development Platform
- Supports the Following Functions:
  - Power-up detection
  - Thermal management
  - Power sequencing
  - Sleep modes
  - System diagnostics
  - Remote communications
  - Clock generation and management



- Kit Contents:
  - FlashPro3programming stick
  - 2 Mini USB cables
  - Fusion Advanced Development Board with ARM Cortex-M1–enabled M1AFS1500-FGG484
  - Libero IDE DVD, including SoftConsole for processor-based designs

Barto's Law

Every circuit is considered

GUILTY

until proven innocent