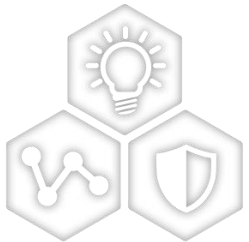


World's Smallest FPGA SoC



A Leading Provider of Smart, Connected and Secure Embedded Control Solutions

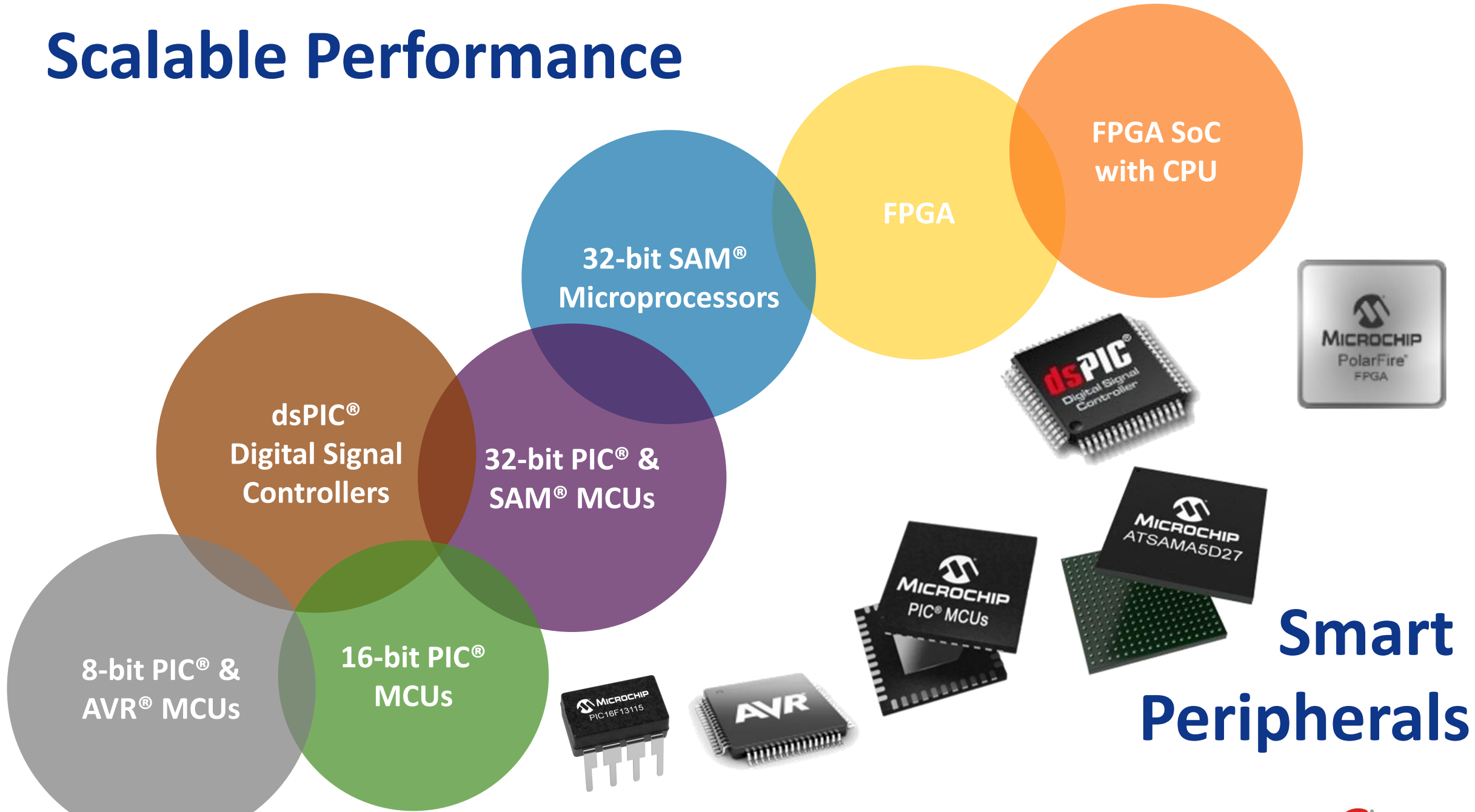


SMART | CONNECTED | SECURE

Attila Kolinger

Apr 2024

Scalable Performance



Smart Peripherals

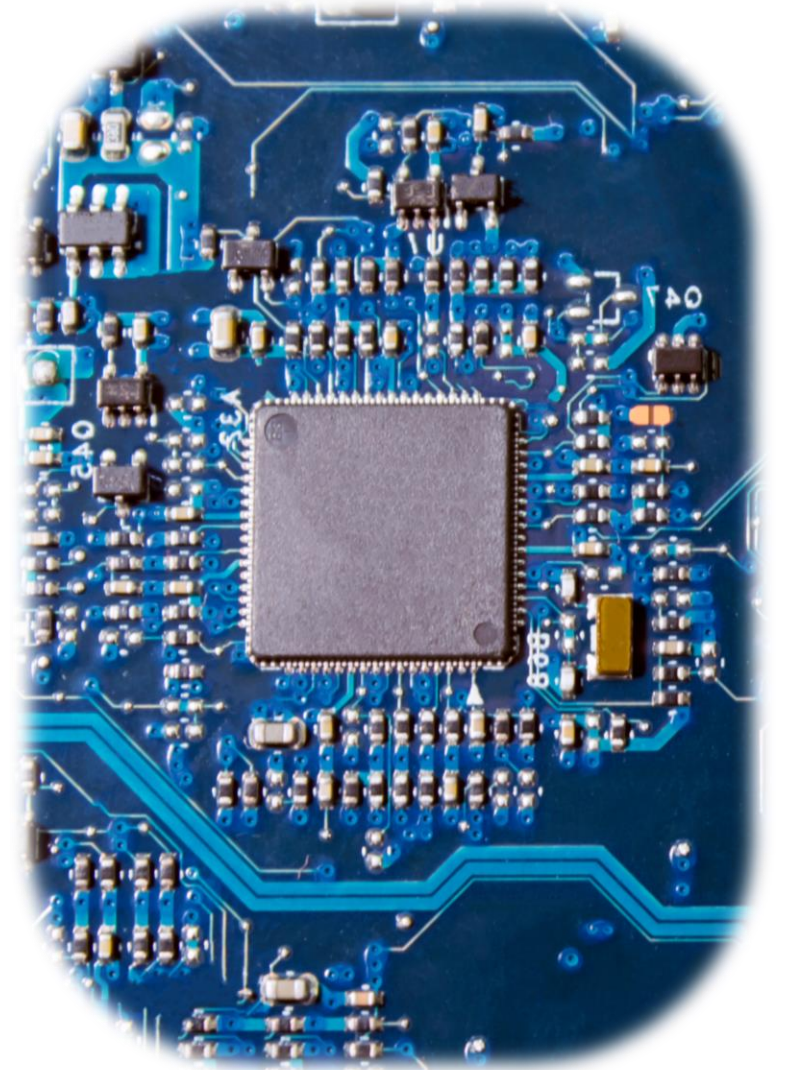
Why Use FPGA SoCs?

Safer CI/CD

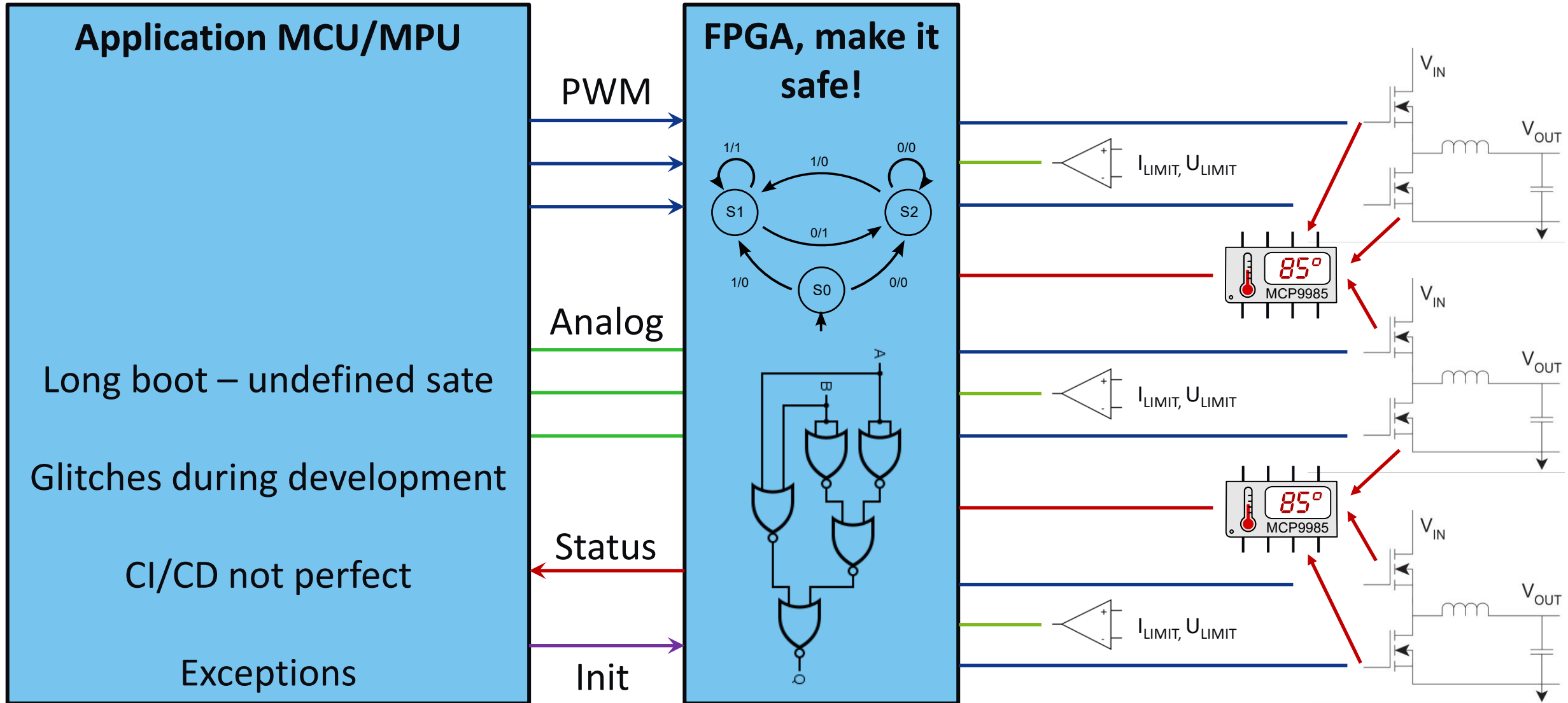
Should You Use an FPGA SoC?

YES! If you ...

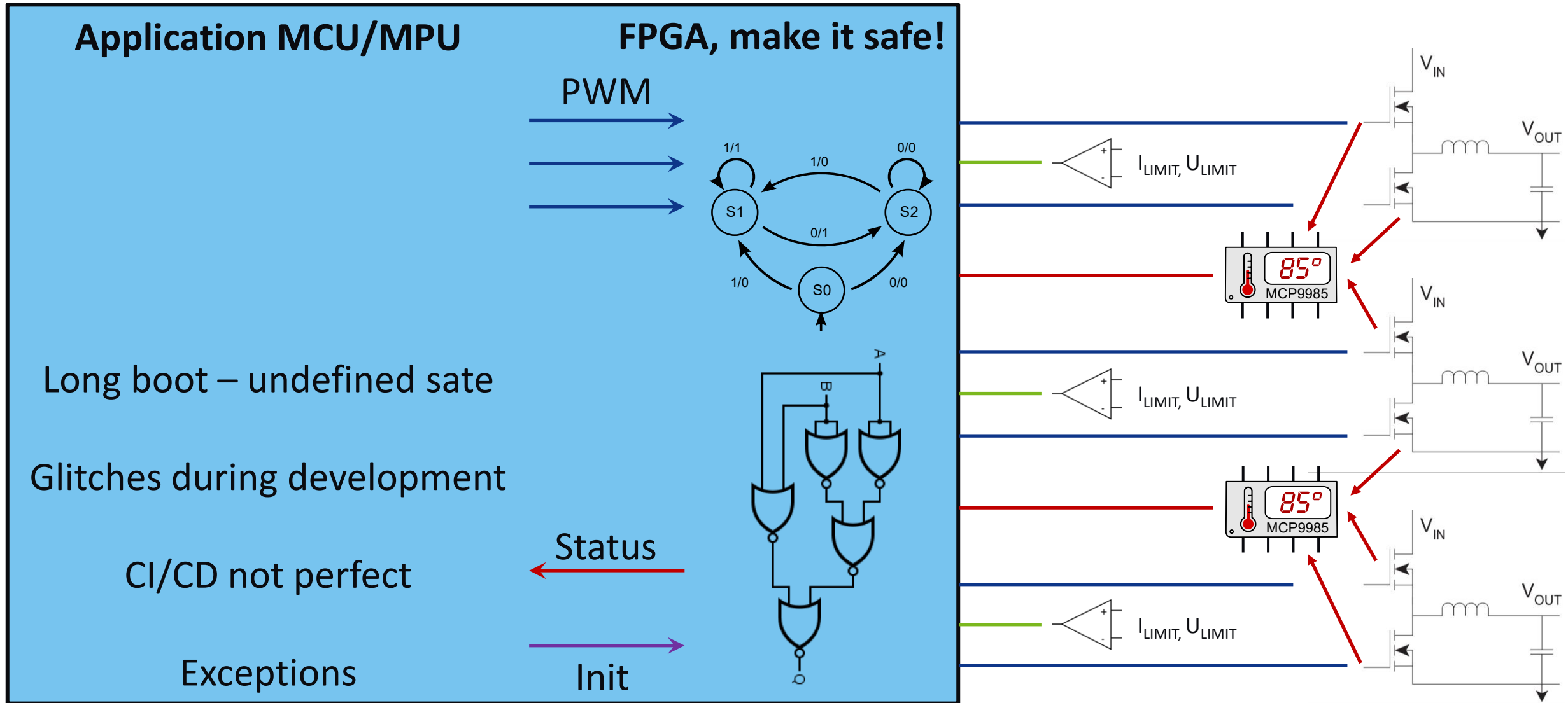
- need parallel processing
- might need PCB rewiring
- need a lot of glue logic
- have hardware that is safety critical
- demand software independent HW
- are doing pre-development
- create new MCU peripherals
- must change existing peripheral behaviour
- run “C” code in an FPGA



Unleash FW Developers



FPGA SoC Solution



FPGA SoC with CPU

The Two Extremes

FPGA SoC Redefined

- **PolarFire SoC FPGAs**

- Deterministic, coherent 64-bit multi-core RISC-V CPU
- 25K to 460K LEs in FLASH
- 12.7 Gbps transceivers
- Up to 50% lower power than alternatives
- Integrated DDR3/4, LPDDR3/4 controller and PHY

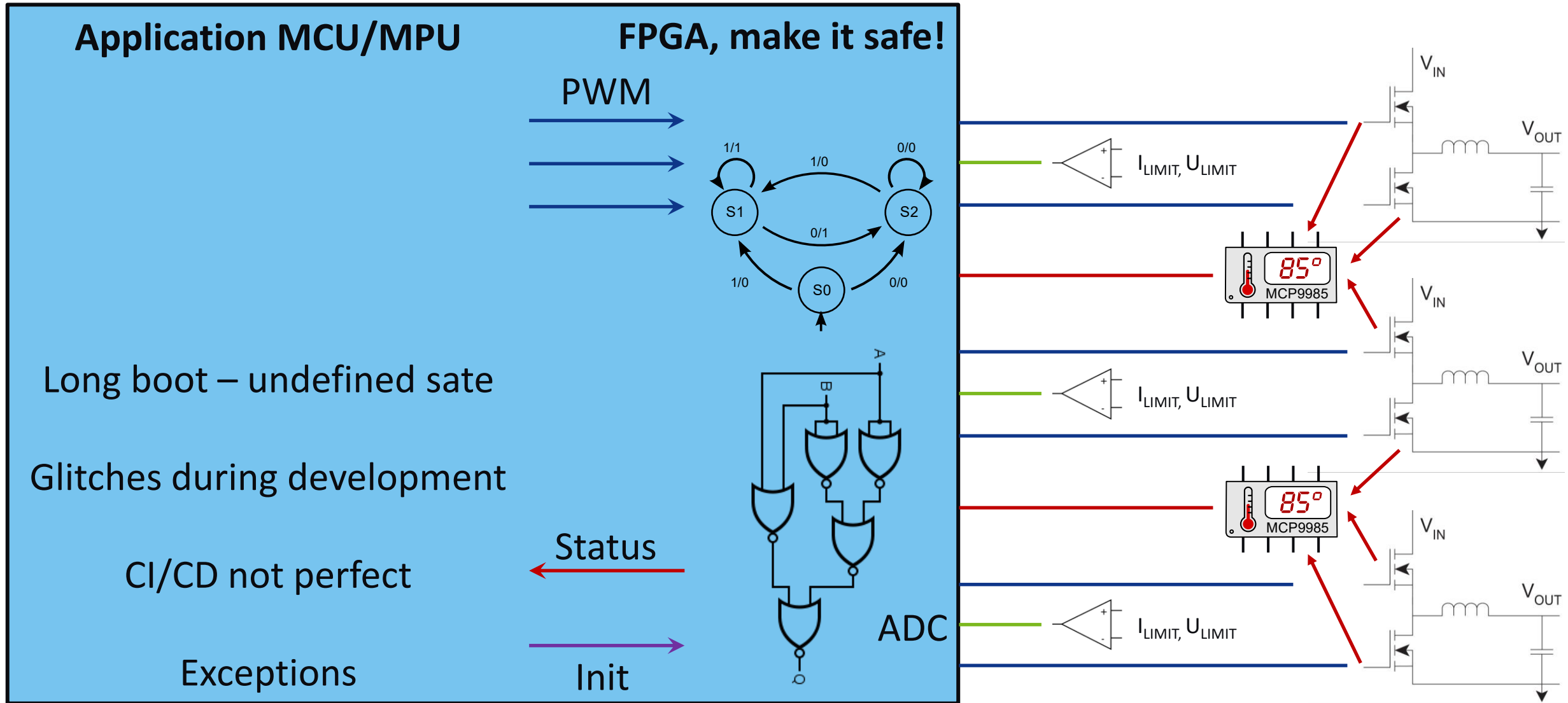


- **PIC16F131xx SoC**

- Deterministic 8-bit CPU
- 32 logic elements (RAM) Verilog and Schematics
- Frozen FPGA blocks (UART, Timer, I²C, etc.)
- On-board analog peripherals & True 5V (1.8V ... 5.5V)
- From 8pin package (including DIP)



8bit FPGA SoC Solution



Configuration Process - MCC Melody

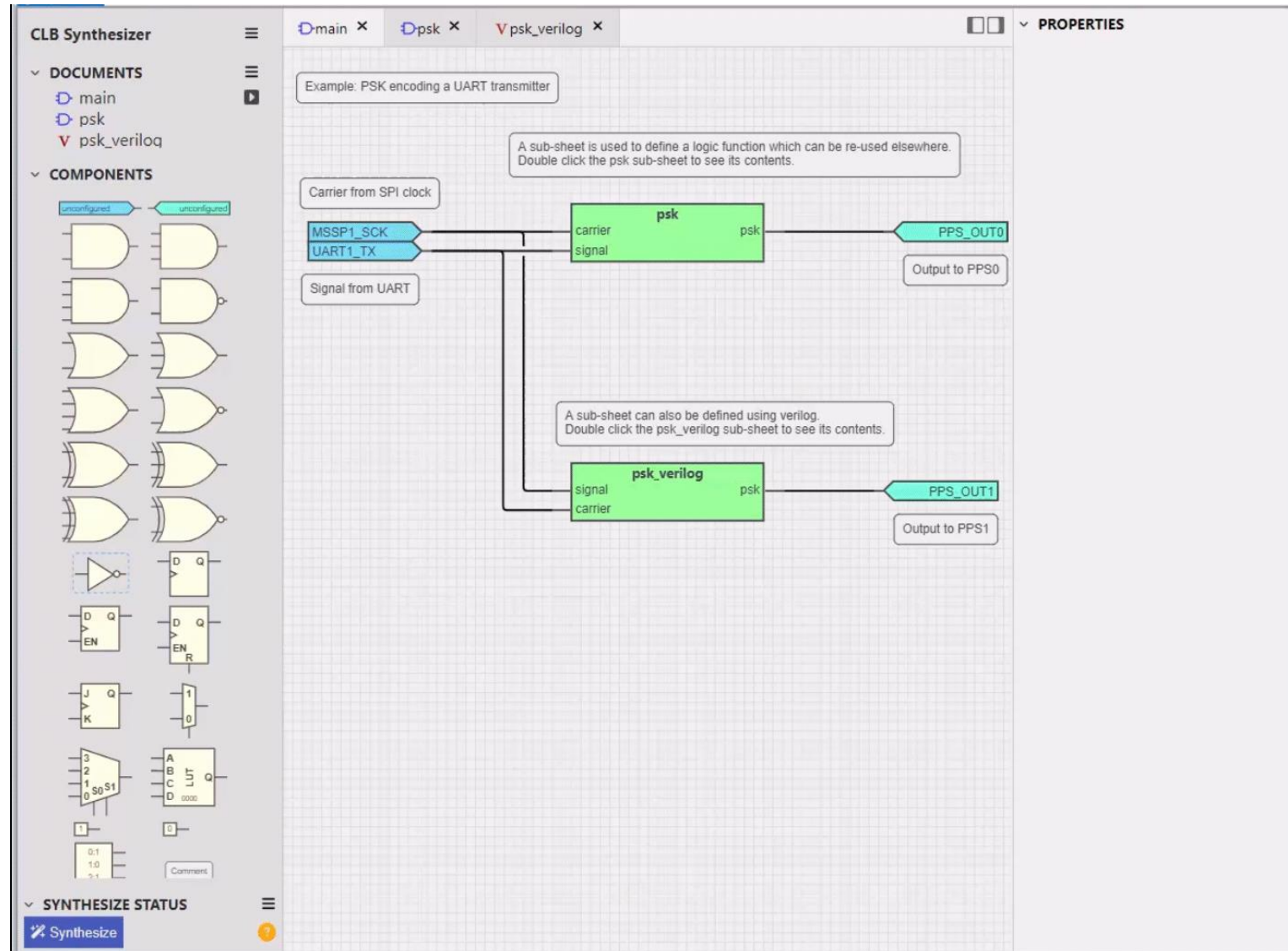
- Clock config
- Add frozen blocks
 - UART, I²C, TMR ...
- Add analog
 - ADCC, DAC, ...
- Add CLB (FPGA)

The screenshot displays the MPLAB X IDE v6.20 interface for configuring a PIC16F13145 device. The main workspace shows the MCC Melody configuration tool with a central Microchip PIC16F13145 component connected to various peripheral blocks: DAC1, CMP1, ADCC, CRC, NVM, and CLC1. The Pin Grid View table below shows the pin configuration for the QFN20 package.

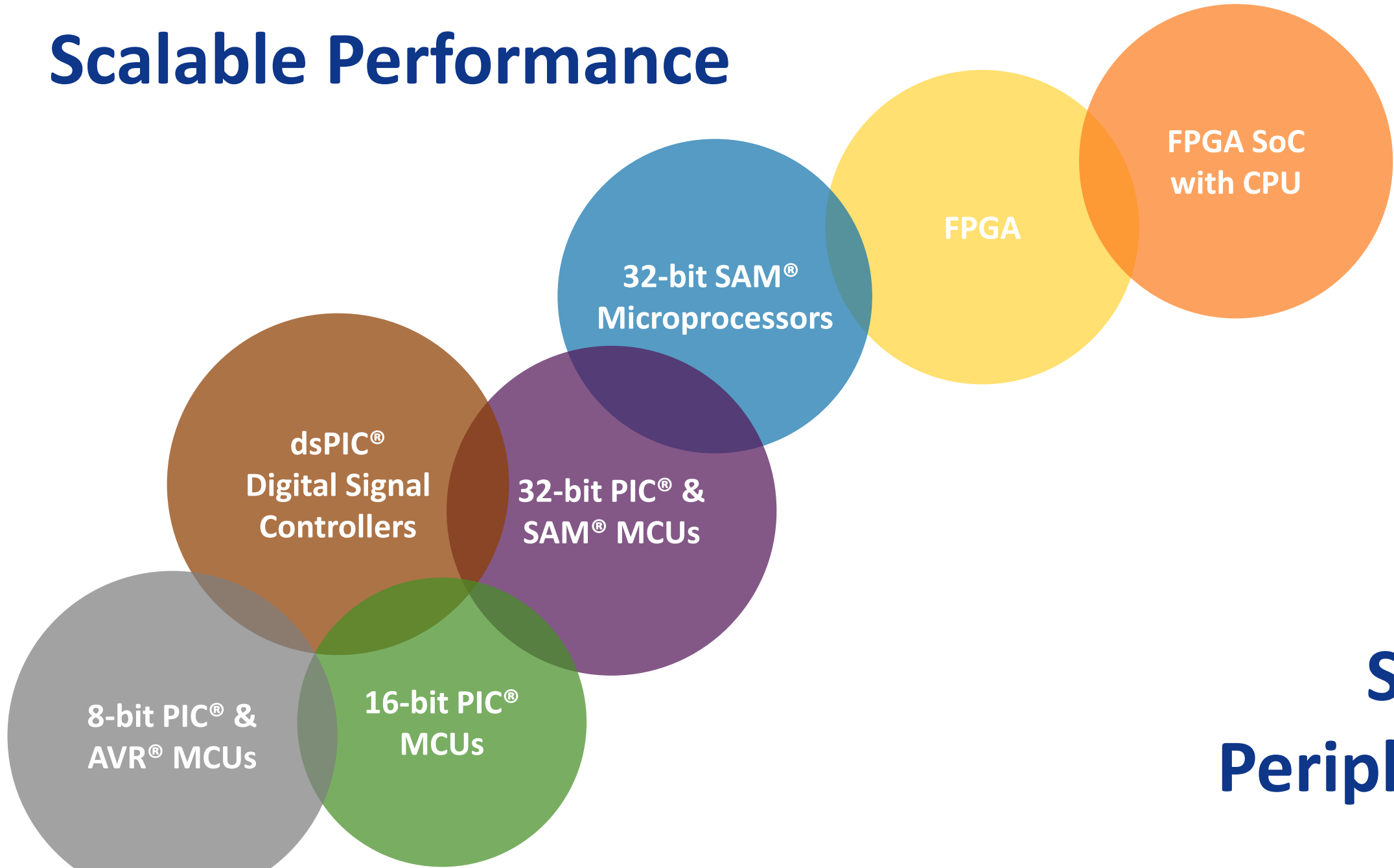
Package:	QFN20	Pin No:	16	15	14	1	20	19	10	9	8	7	13	12	11	4	3	2
			PORTA			PORTB			PORTC									
Module	Function	Direction	0	1	2	3	4	5	4	5	6	7	0	1	2	3	4	5
CMP1	C1OUT	output	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	C1INx+	input	🔒															
	C1INx-	input		🔒									🔒	🔒	🔒			
ADCC	ADGRDA	output	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	ADGRDB	output	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	ANx	input	🔒	🔒			🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒	🔒
	VREF+	input		🔒														
DAC1	DAC1OUTx	output	🔒		🔒													
	VREF+	input		🔒														

Configuration Process - MCC Melody

- Clock config
- Add frozen blocks
 - UART, I²C, TMR ...
- Add analog
 - ADCC, DAC, ...
- Add CLB (FPGA)
- **Create FPGA design**
- **Synthesize**
- **Configure GPIO**
- **Generate Melody code**

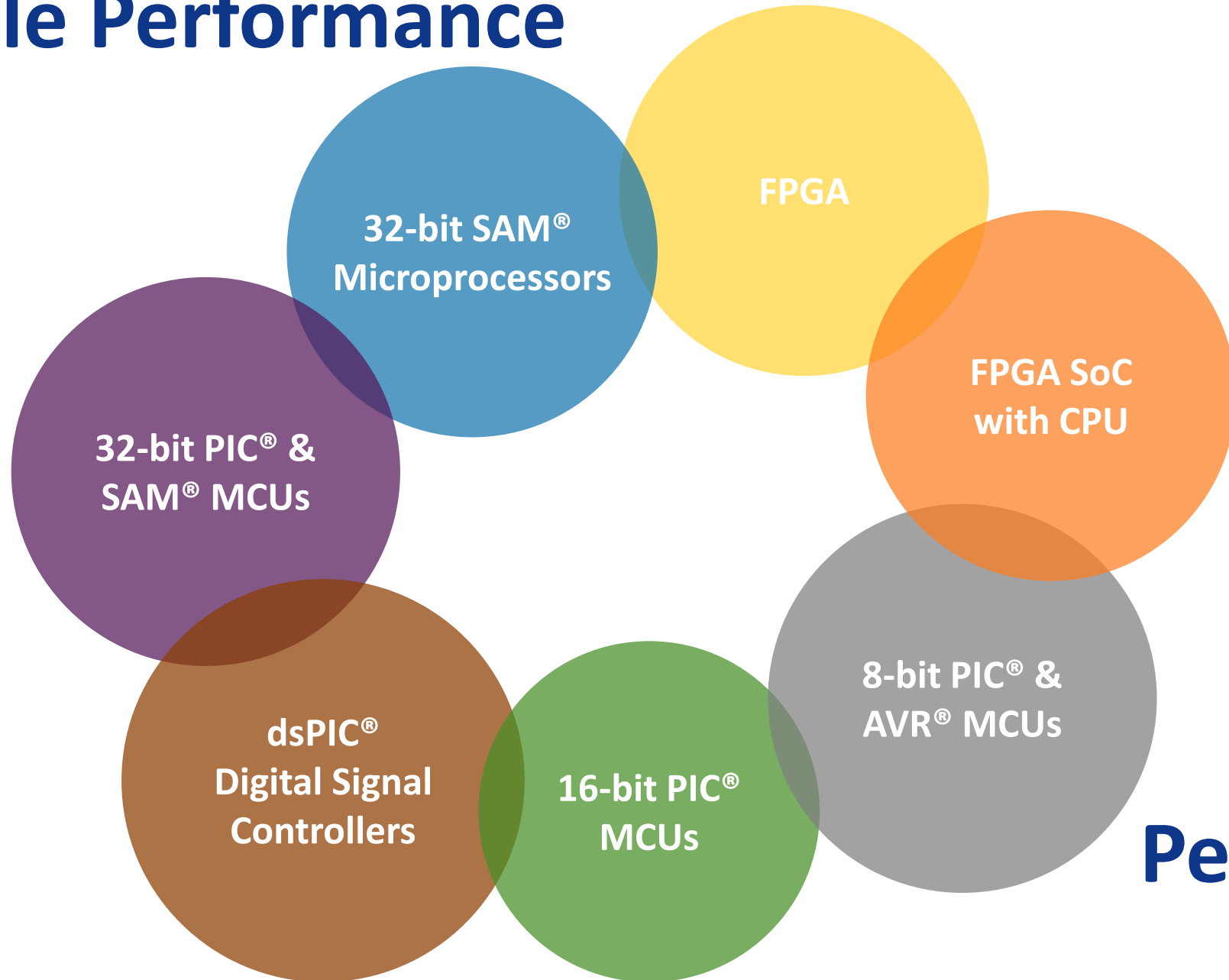


Scalable Performance



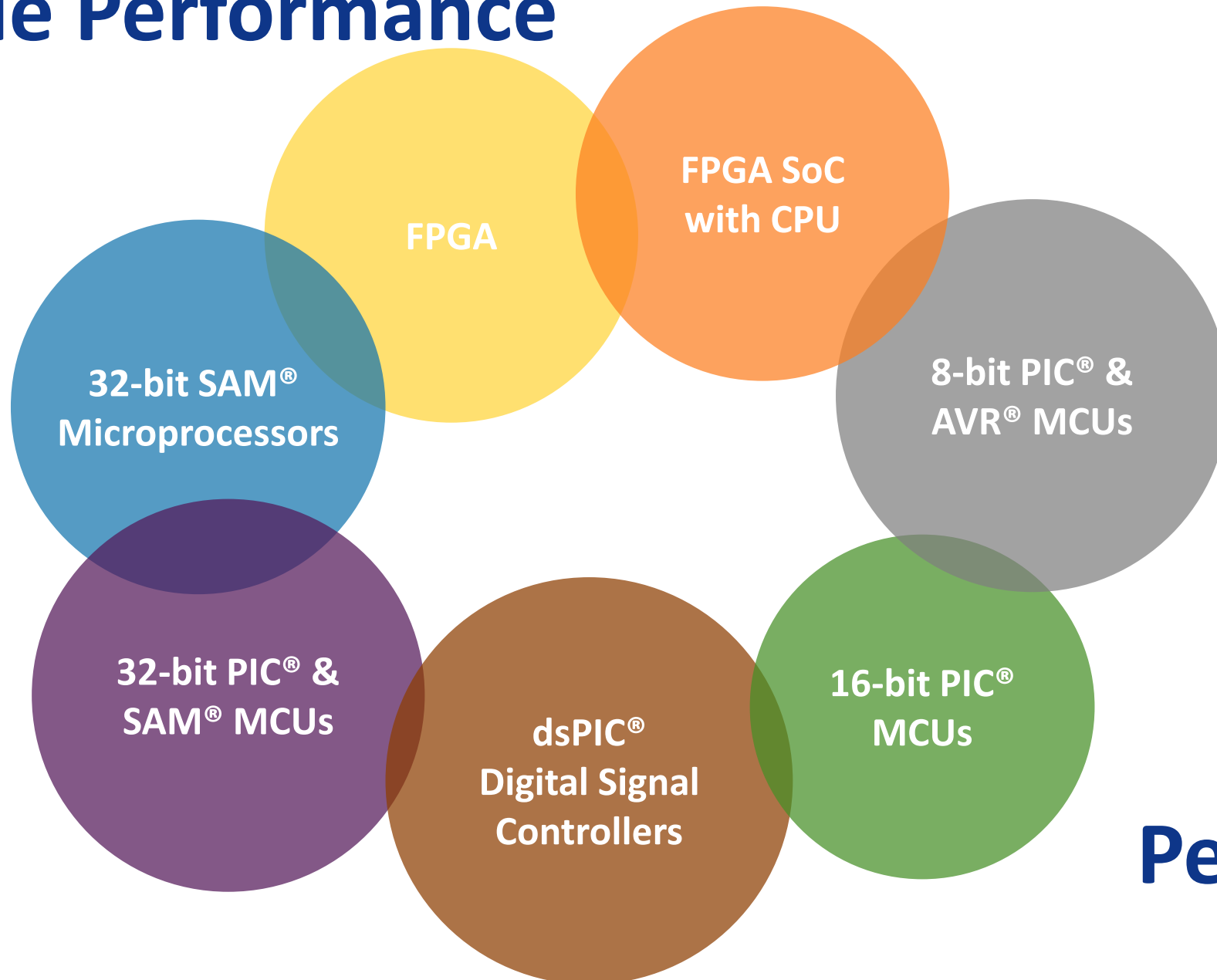
Smart Peripherals

Scalable Performance



Smart Peripherals

Scalable Performance



Smart Peripherals

Thank You!

Any Questions?