FTF FREESCALE TECHNOLOGY FORUM 2015

Increasing Automotive Safety Through Embedded Radar Technologies

FTF-ACC-F1276

Andrew Robertson | Systems and Applications Engineer

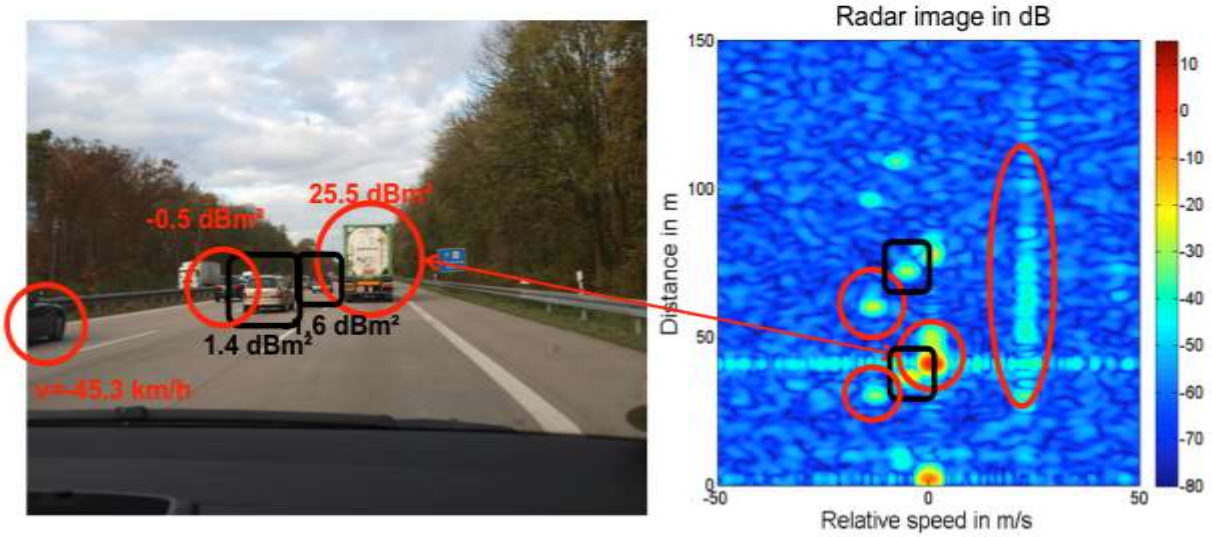
J U N E . 2 0 1 5

freescale™

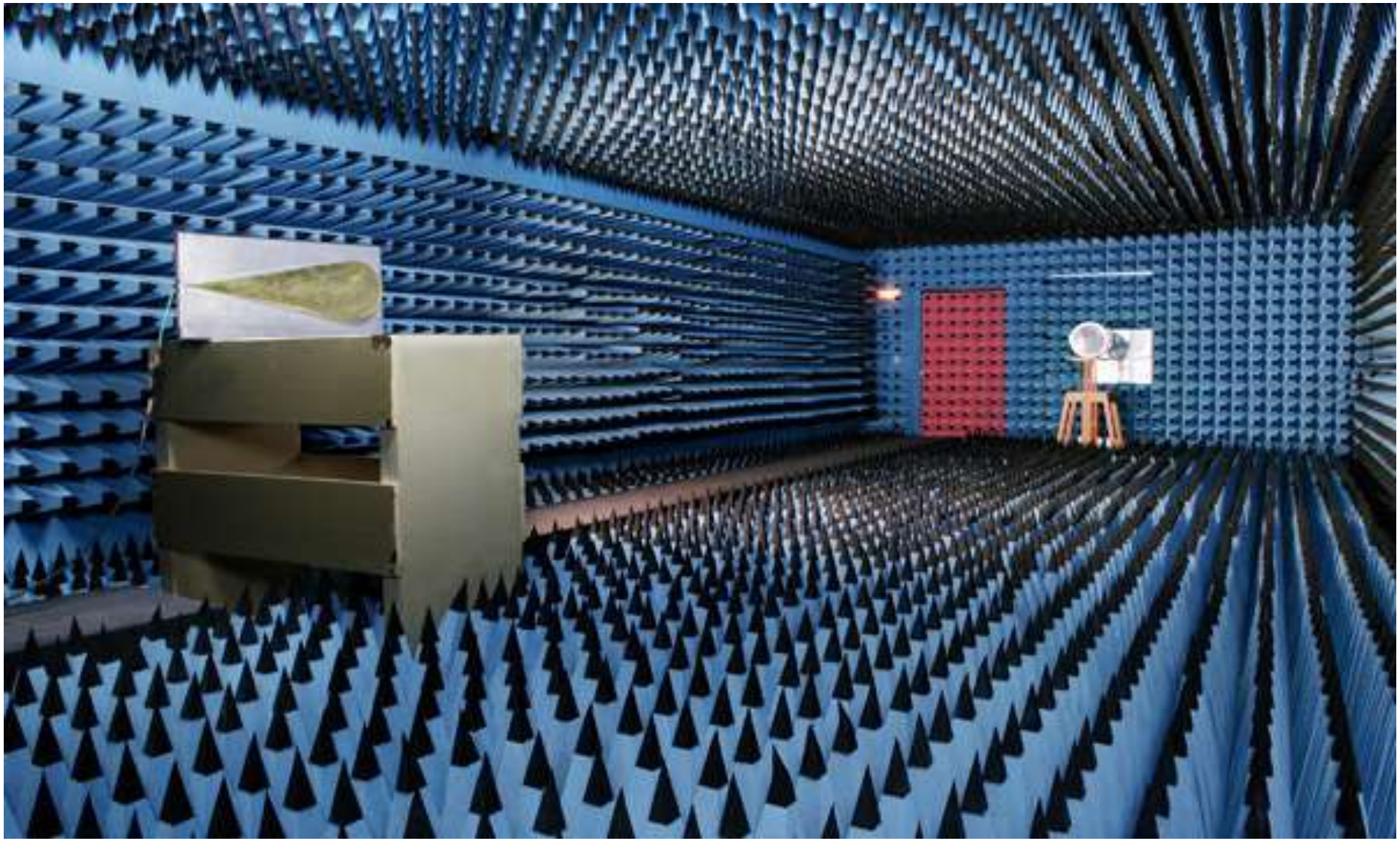
External Use

Agenda

- Radar Fundamentals
- Microcontroller Overview
- Range & Doppler Processing
- Algorithm Mapping
- Summary

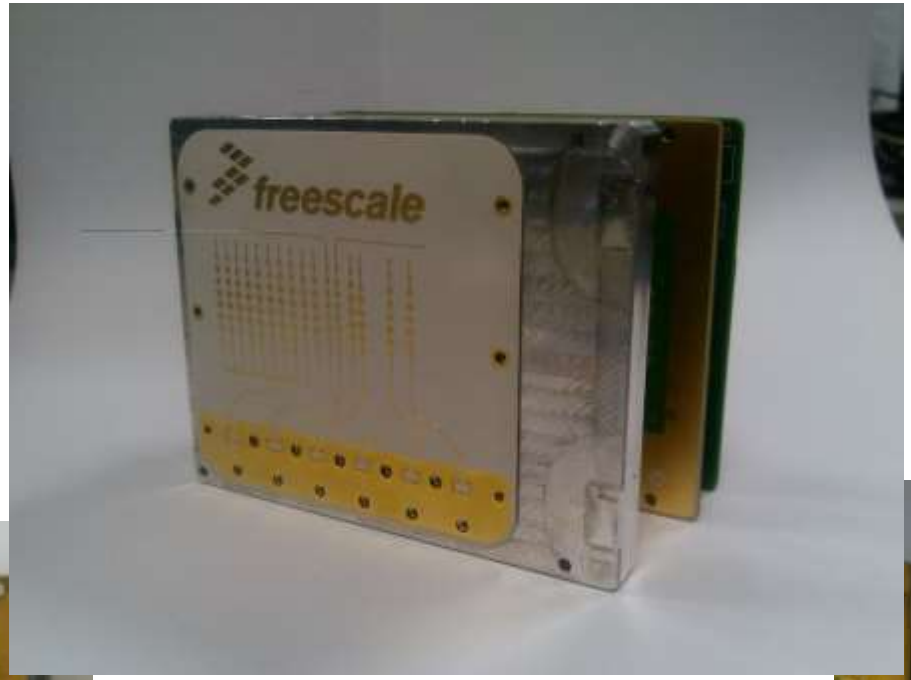


Anechoic Chamber



Freescale Radar Sensor

Wave Guide and Antenna



Analogue Front End
MR2001

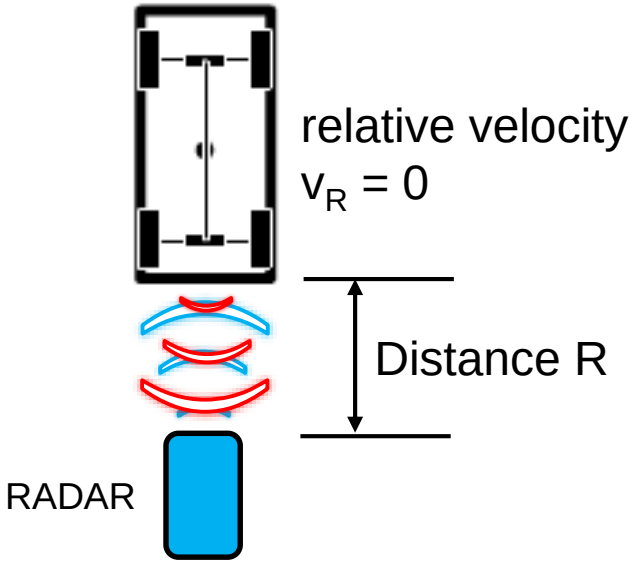
Microcontroller
MPC5775K

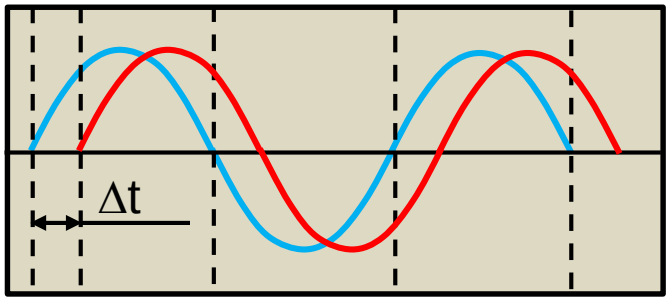
Radar - Fundamentals

- **Radar Overview**

RaDAR (Radio Detection And Ranging)

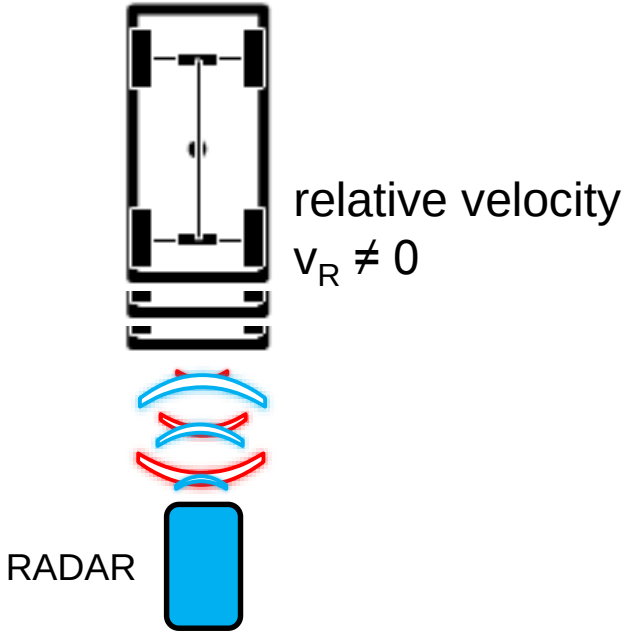
Distance

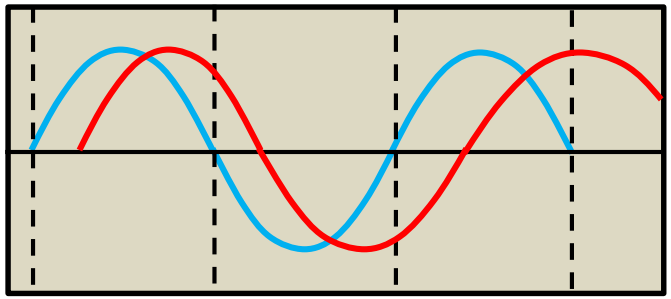




$\Delta t = 2\ R / c_0$

Speed





Doppler Shift

RADAR Processing

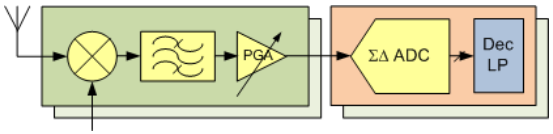
MMIC
⇒ MR3003

ADC/DSP
⇒ MPC577xK

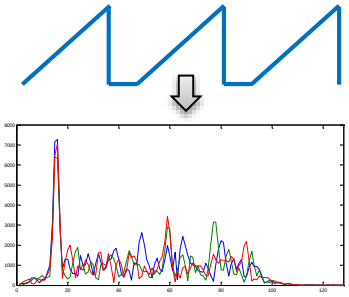
Re-ceiver | Signal condi-tioning | Signal analysis | Detec-tion | Tracking

- Antenna
- Mixer

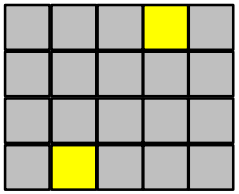
- LNA
- HP/LPF
- AAF
- ADC



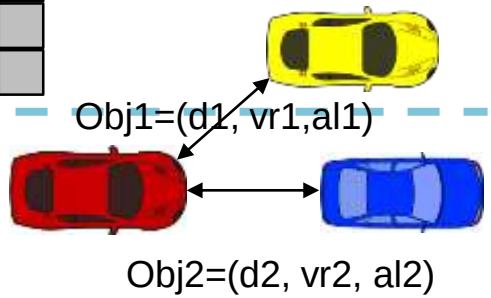
- Gain
- Window
- FFT
- Filter



- Power
- CFAR



- Clustering
- Kalman Filter



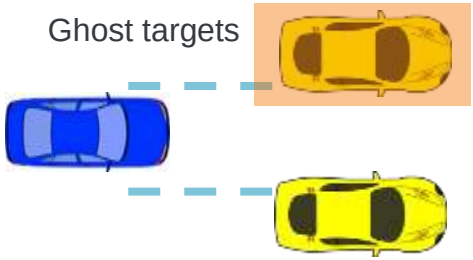
Road Scenarios

AEBS City

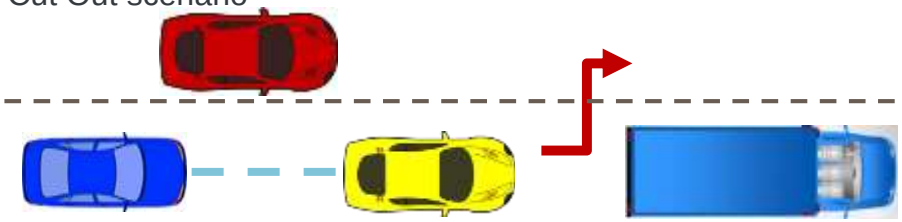
AEBS Urban decelerating

ACC up to 200m

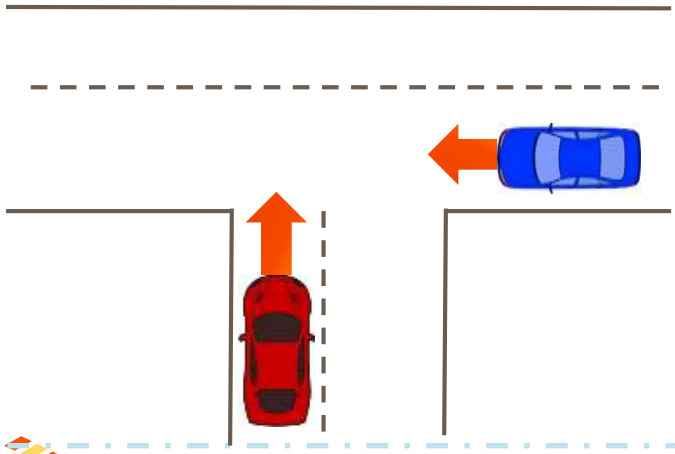
Ghost targets



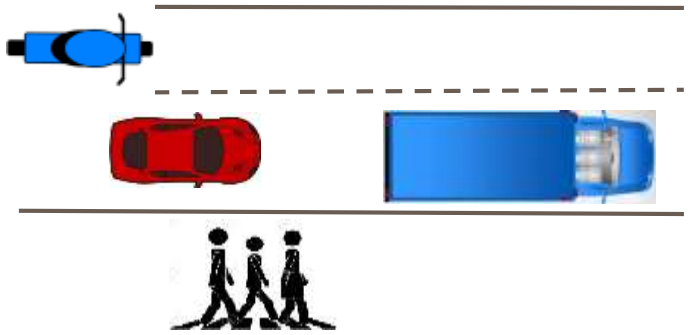
Cut Out scenario



T- Intersection



Bikes and pedestrian

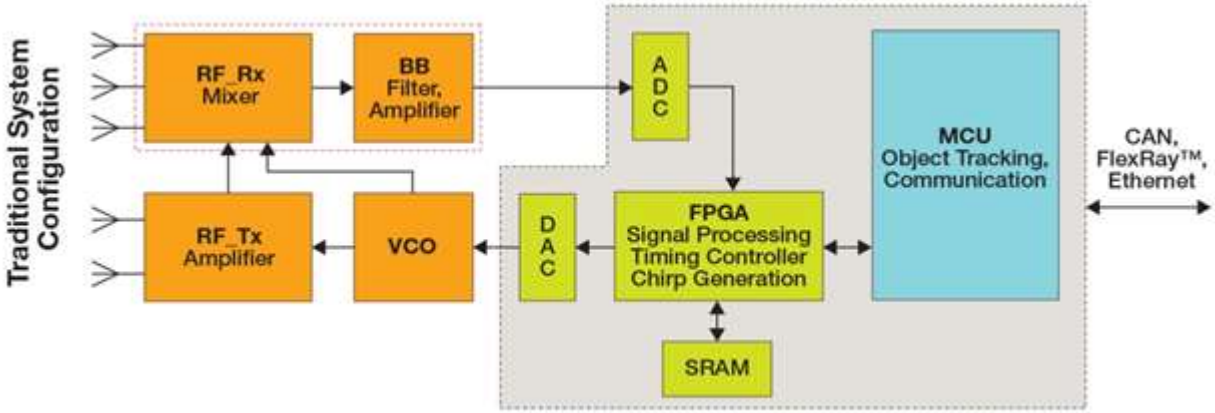


MPC577xK – Microcontroller Overview

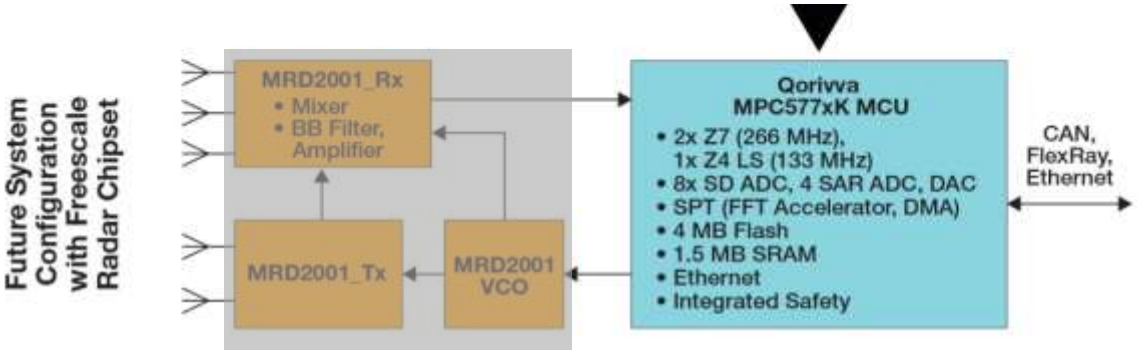
- Microcontroller Overview
- Analog Front End (Sigma Delta ADC)
- Chirp Generation (DAC)
- Signal Processing Tool Box (SPT)

Benefits of Integration

MPC5675K System



MPC577xK System



Performance– MPC5775K offers top-performance for intense computational tasks with key integrated digital accelerators

Safe –Built on proven safe technology it delivers a scalable, well documented, process compliant safe architecture and safe Software

Integration & Cost – Right balance of memory, large number of Analogue IP designed for Radar, FFT accelerator. Drive Miniaturization and BOM saving

Flexible – can be used in all applications and with all Front End Radar sensor technology and types.

Qorivva MPC5775K MCU Overview

CPU Platform

- 266 MHz Power ISA Dual Issue core multi core system
  - Two z4 Cores in permanent delayed Lockstep for high safety integrity level
  - Two z7 cores for application execution
  - I-cache – 16 KB (2 ways) / D-Cache 16 KB (2 ways)
  - Core Local D-memory (64kB at each core) with local MPU
- Vector Floating Point Unit & SIMD (z7)
- 64 bit BIU with E2E ECC

Radar Processing Platform

- Signal Processing Toolbox (SPT) FFT accelerator, SDMA, PDMA
- 8x Integrated ΣΔ-ADC with 5 MHz bandwidth and internal sampling clock of 320 MHz.
- 12-bit resolution DAC with maximum of 2Msps
- Low jitter 320Mhz PLL for RADAR

Memory

- Up to 4 MBytes byte Flash with EE Emulation and ECC
- Up to 1.5 MBytes SRAM with ECC
- Safe Crossbar (E2E ECC) with system MPU

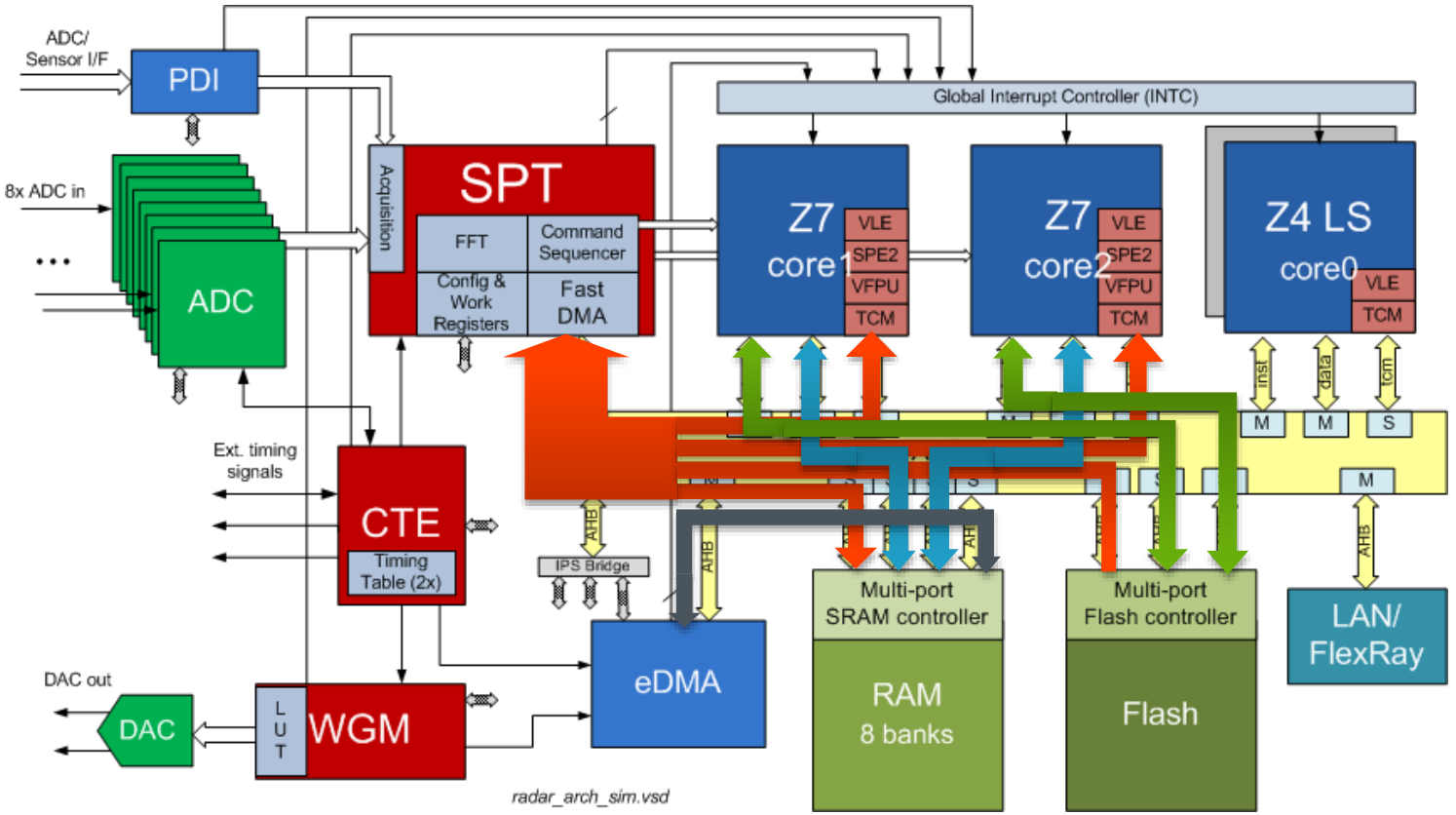
Vehicle & ECU communication

- 4 x FlexCAN (64 message buffers)
- 1 x FlexRay (Dual Channel 128 msg. buffers)
- 1 x Ethernet Controller (ENET)
- 4 x LINFlex (SCI) & 3x IIC
- 4 x dSPI (4cs std / 8cs in larger v package version only)
- 3 x eTimer
- 2 x FlexPWM (2x 12 channel) & 2x CTU

System

- Highly stable Oscillator for Radar ASIC to A/D synchronization
- SIPI (~300MBaud) for interprocessor or mc to ASIC communication
- Safe DMA Engines
- Autonomous Fault Collection and Control Unit
- CRC computing unit
- Junction temperature sensor
- Nexus Class 3+ debug interface (Aurora extension)

RADAR System Integration

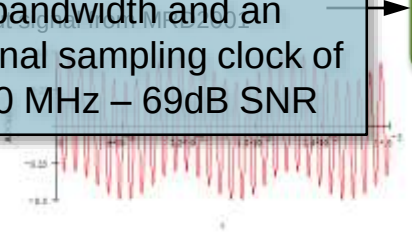


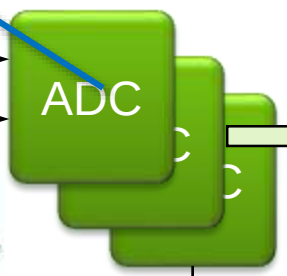
- High-speed multi-master bus connects modules: 64-bit @ 133MHz
- Multi-ported SRAM and Flash support concurrent transfers

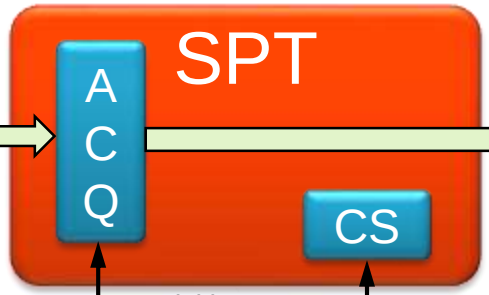
RADAR Timing Generation

Sample received RADAR echoes
10MSps/12-bit

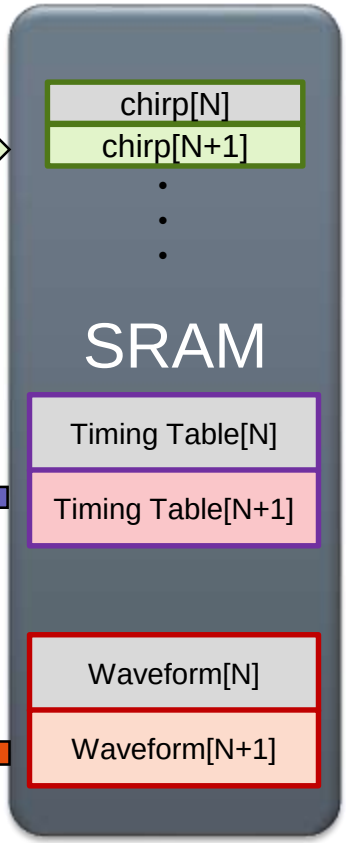
8x ΣΔ-ADC with 5 MHz bandwidth and an internal sampling clock of 320 MHz – 69dB SNR







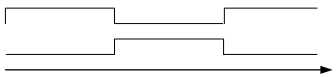
fast DMA



acquisition window

events

Control signal to MRD2001

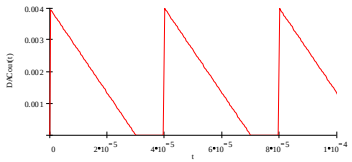


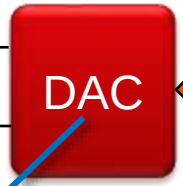
ctep

eDMA

run, hold, reset

Output signal to MRD2001

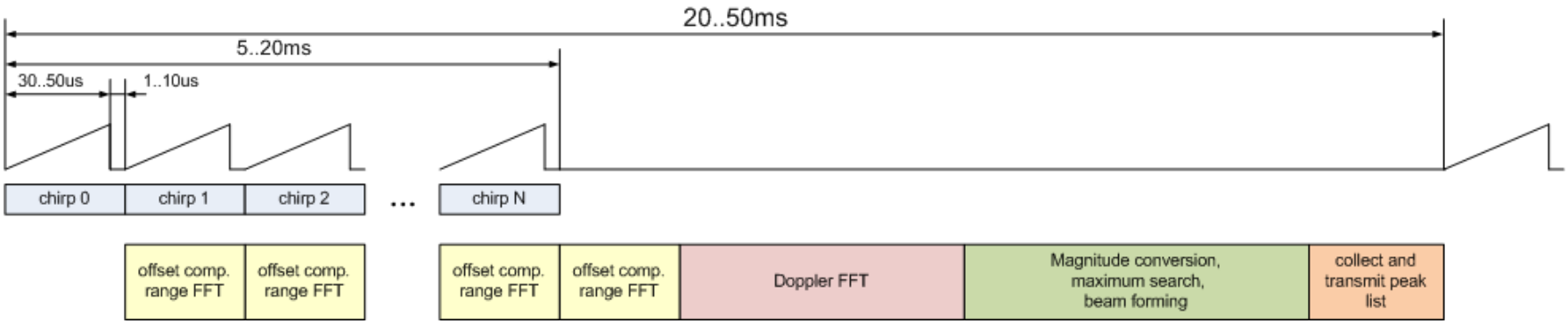




eDMA

A new best-in-class 12-bit resolution DAC which has maximum of 2Msps

Chirp Sequence RADAR Timing

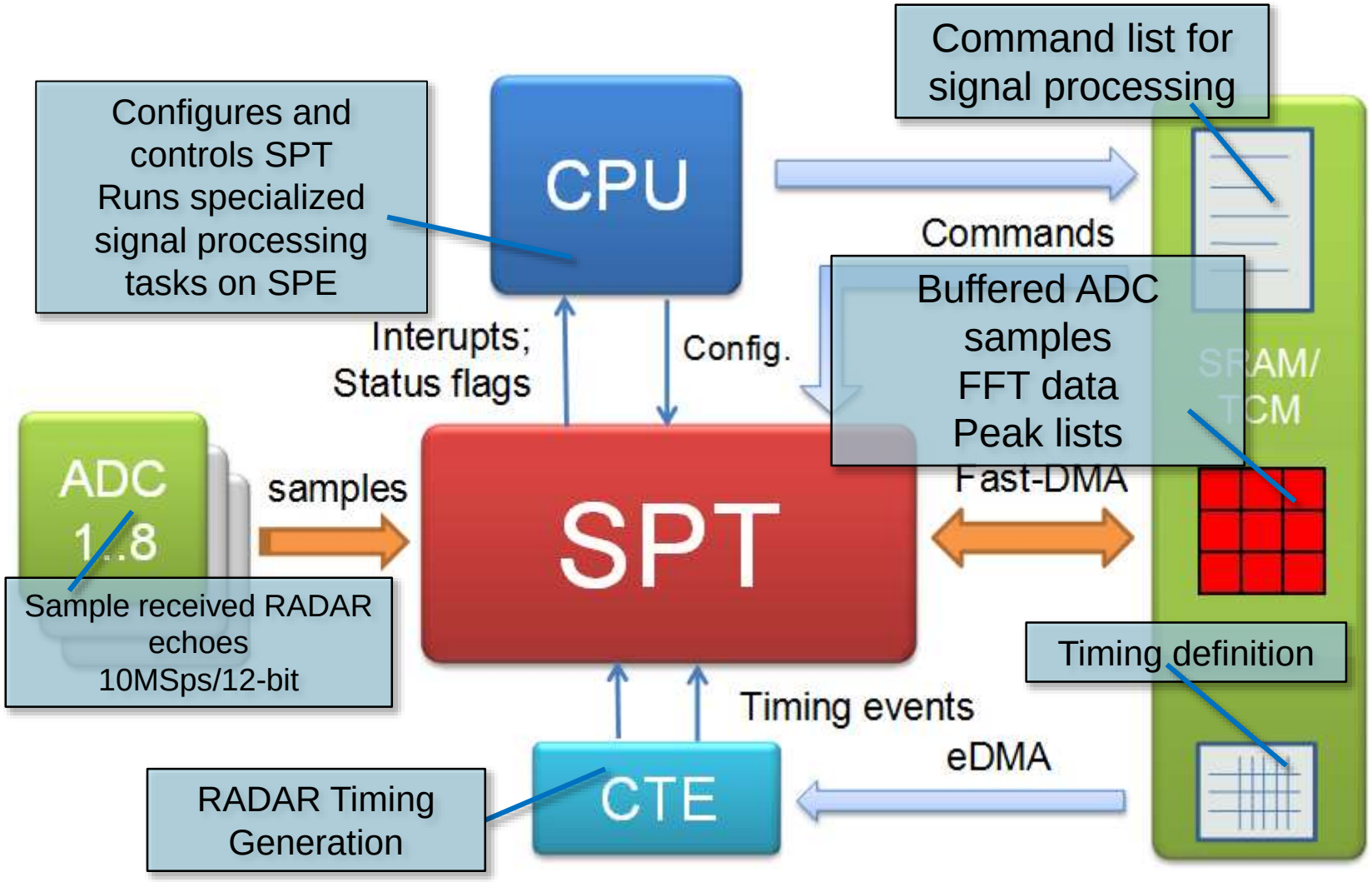


- 1 frame comprises up to 512 chirps and the processing algorithm
  - ~20 frames per second possible dep. on complexity of processing
- Range FFT performed on-the fly
- Doppler FFT and post-processing run in gaps (transmit idle) between chirps

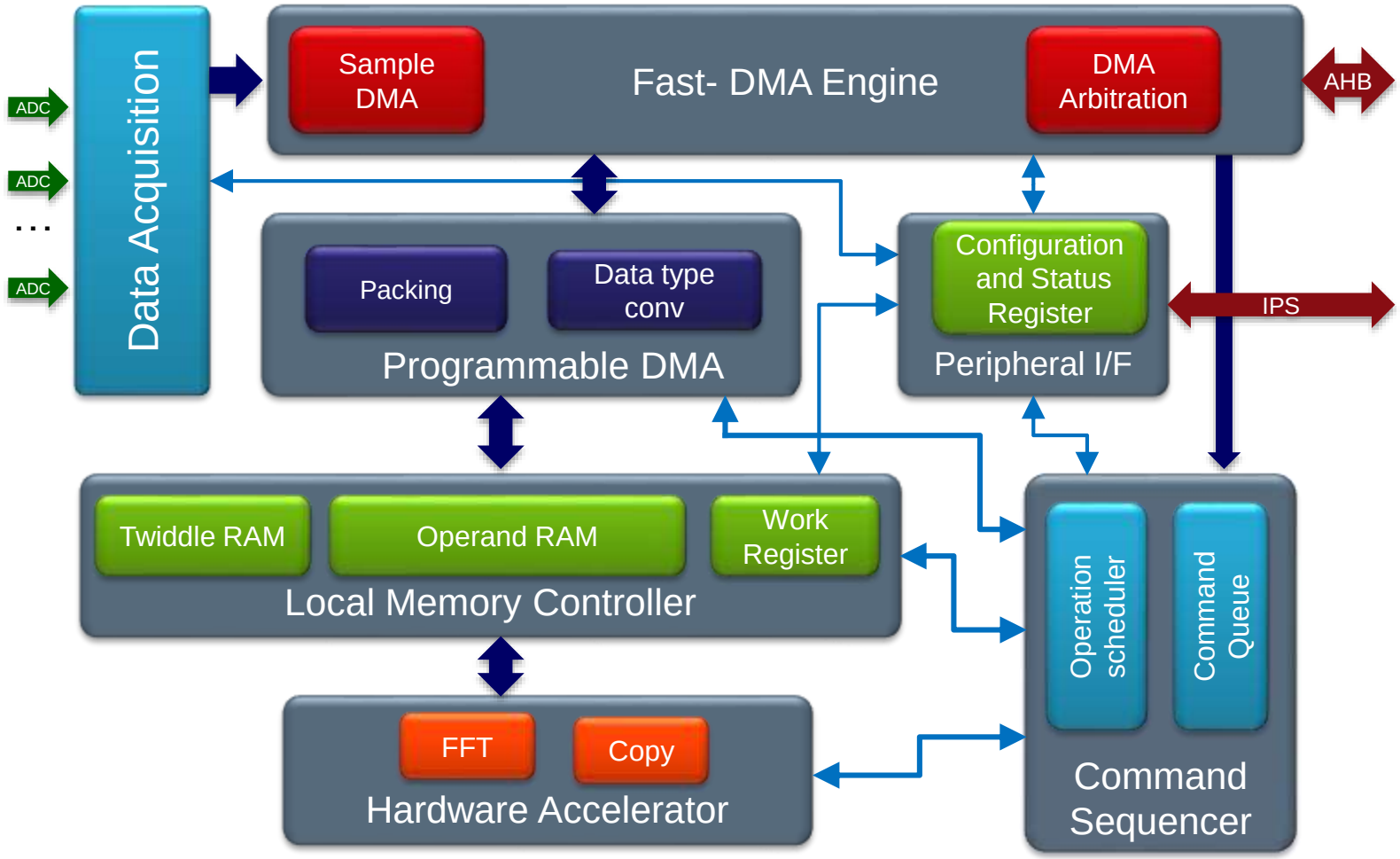
SPT - Range and Doppler Processing

- **Range FFT (Distance)**
- **Doppler FFT (Speed)**

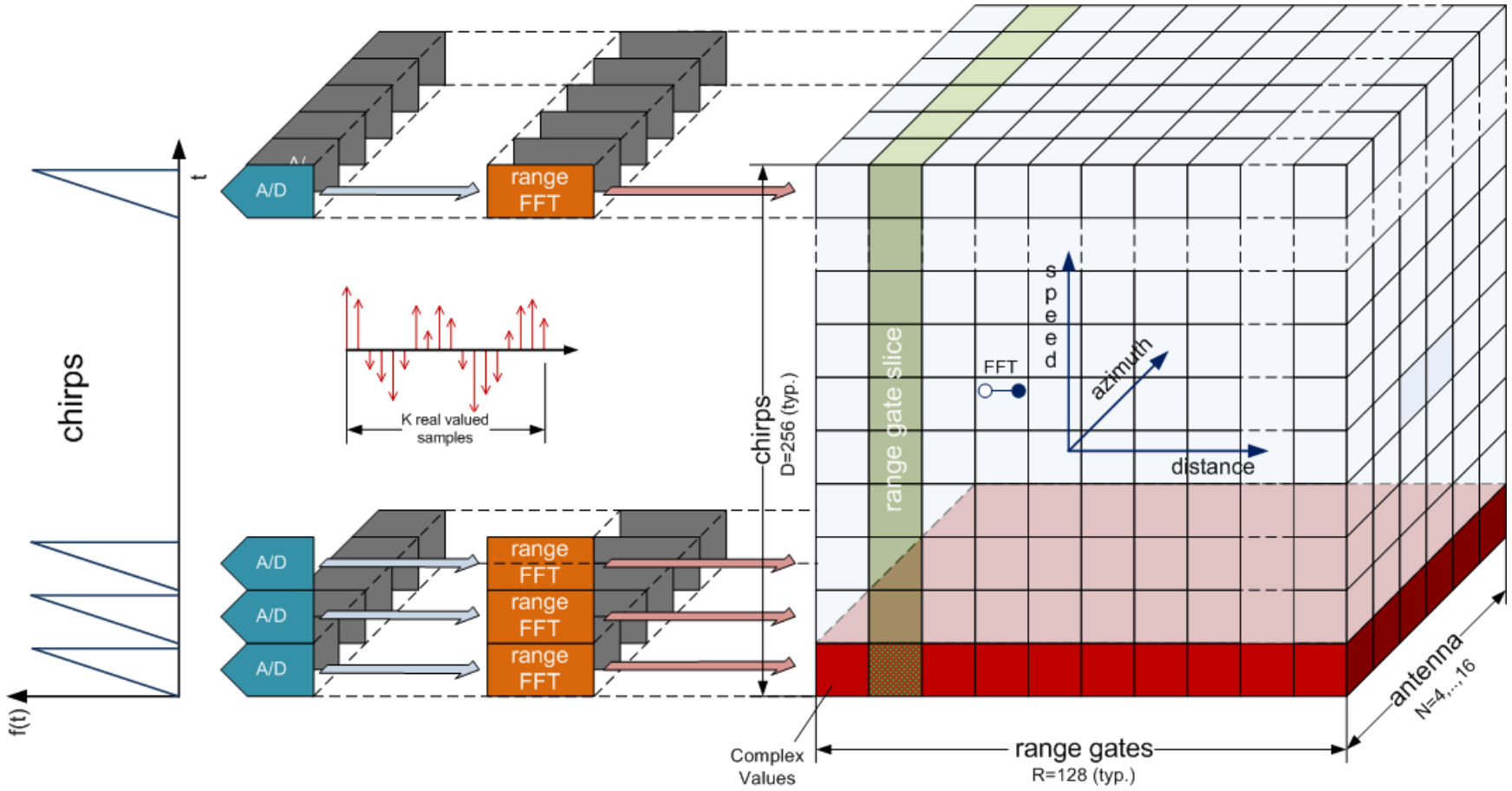
SPT Operation Principle



FFT – Block Diagram



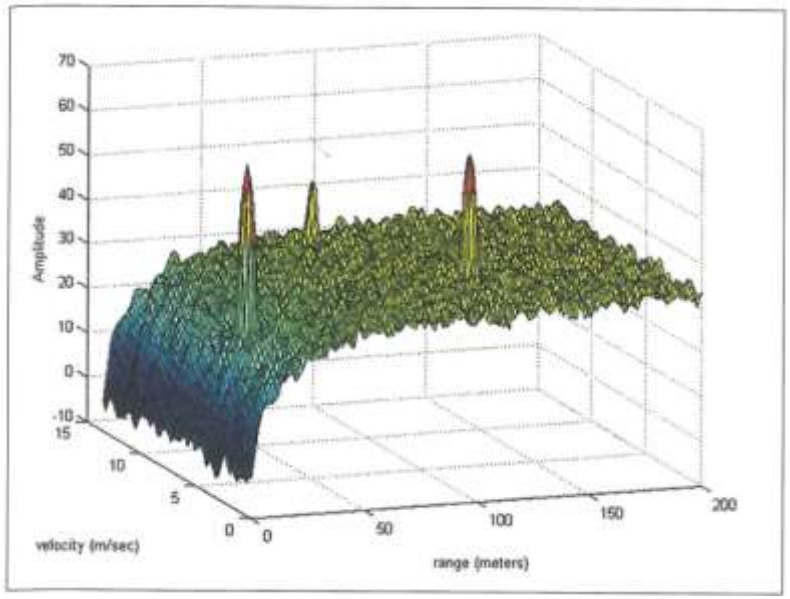
Signal Analysis – Range FFT

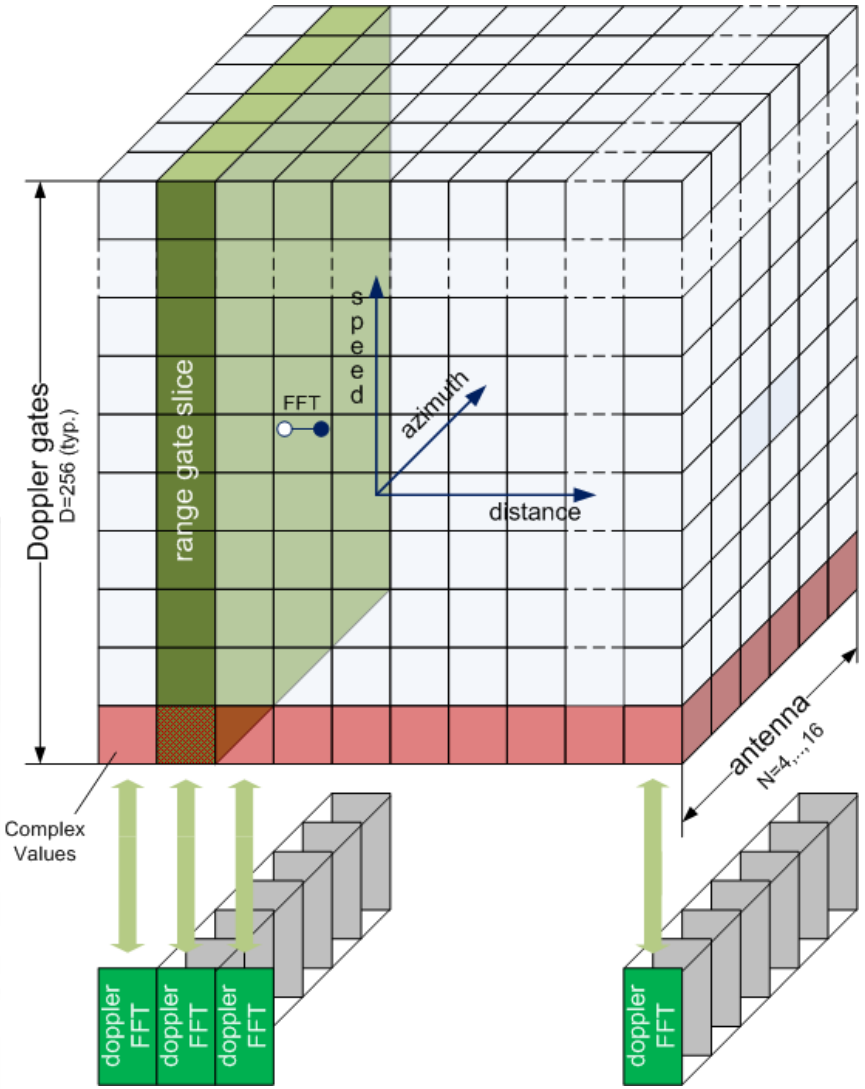


- Range FFTs
  - real to complex transform, provide SNR gain

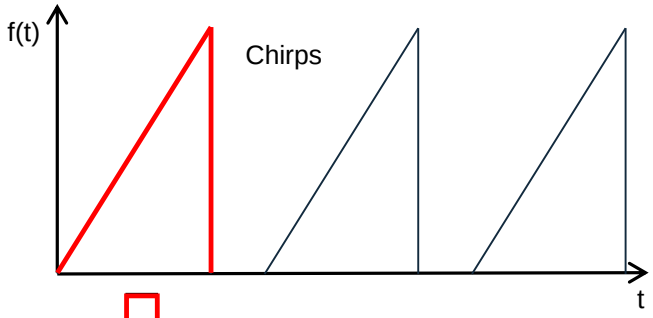
Signal Analysis – Doppler FFT

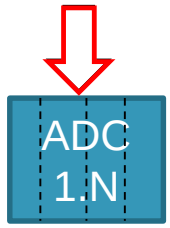
- Doppler FFTs
  - Complex to complex
  - Provide SNR gain
  - Determine the relative speed (Doppler gates)

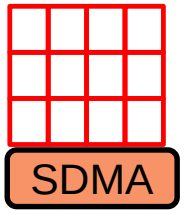


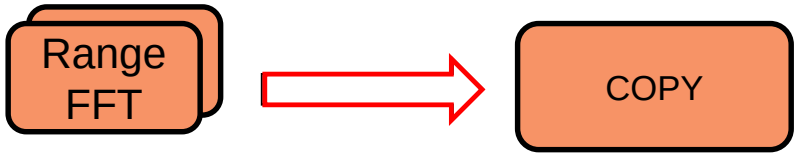


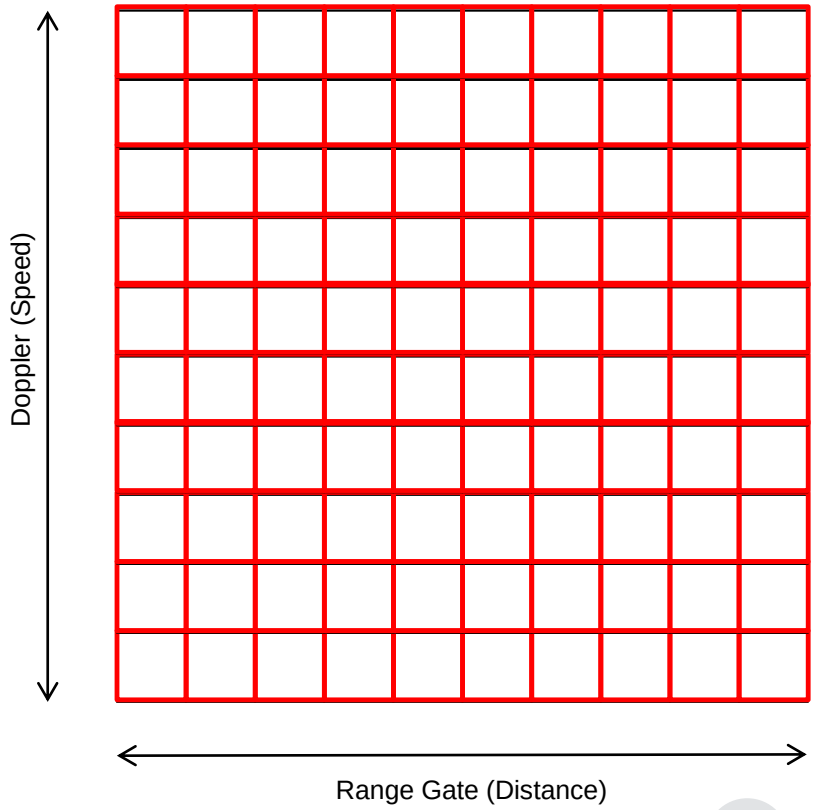
Fast Chirp Sequence Range FFT



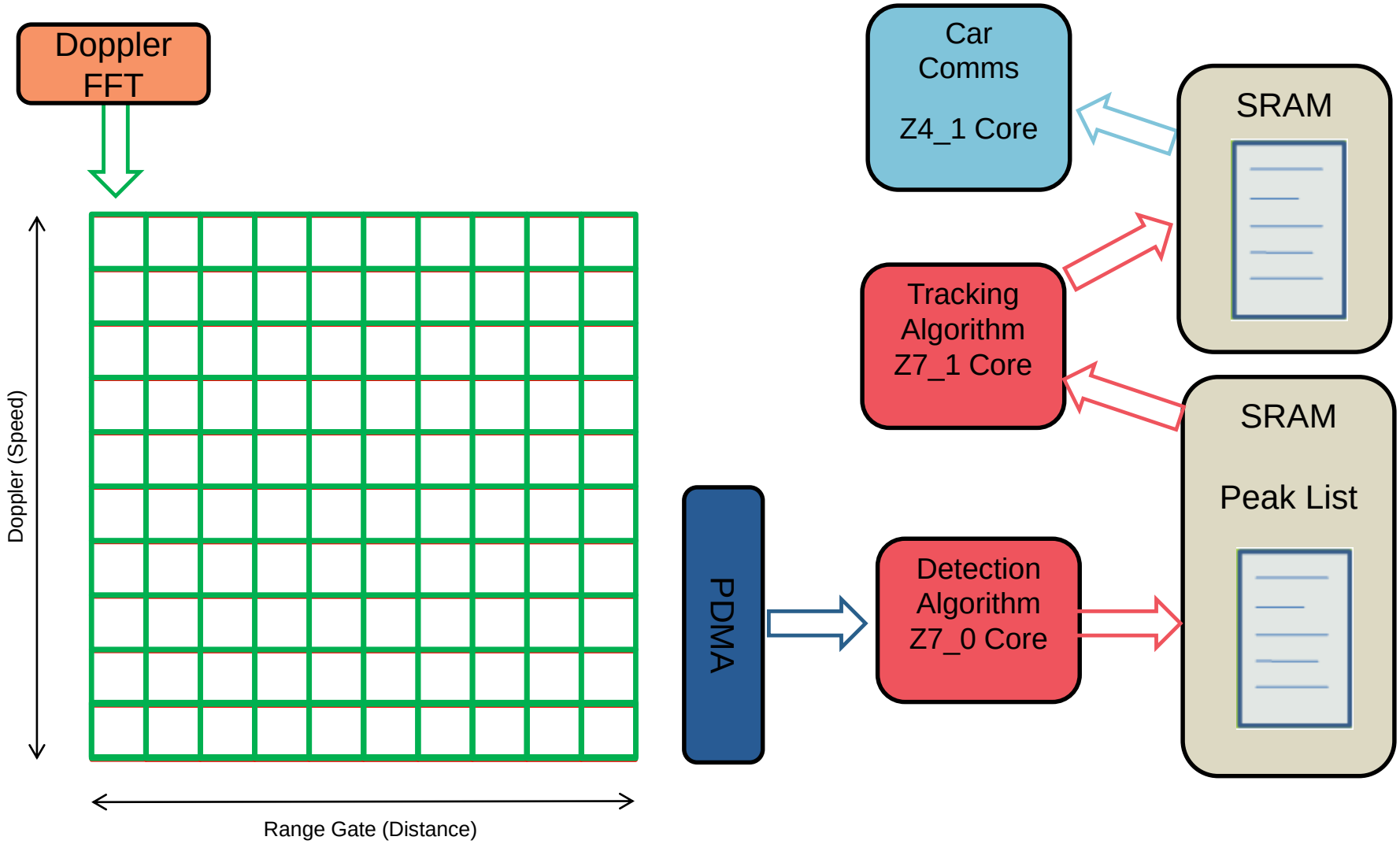








Fast Chirp Sequence Doppler FFT



Algorithm Mapping

- CFAR and Beamforming Algorithms
- Kalman Filters
- Signal Processing Extension (SPE2)

CFAR – Constant False Alarm Rate

Constant False Alarm Rate (CFAR) algorithms are used in digital signal processing applications to extract targets from background in noisy environments.

- Cell-Averaging (CA-CFAR): Detection Threshold

$$Tx \sum_{n=1}^{2N} X_n$$

- Smaller-Of (SO-CFAR)

$$T\ x\ min\left[\sum_{n=1}^{N} Xn\,,\ \sum_{n=N+1}^{2N} Xn\right]$$

- Greater-Of (GO-CFAR)

$$T\ x\ max\left[\sum_{n=1}^{N} Xn\,,\ \sum_{n=N+1}^{2N} Xn\right]$$

- Ordered-Statistic (OS-CFAR)

$$Tx\ X(k)$$

Multiple Signal Classification (MUSIC) Beamforming

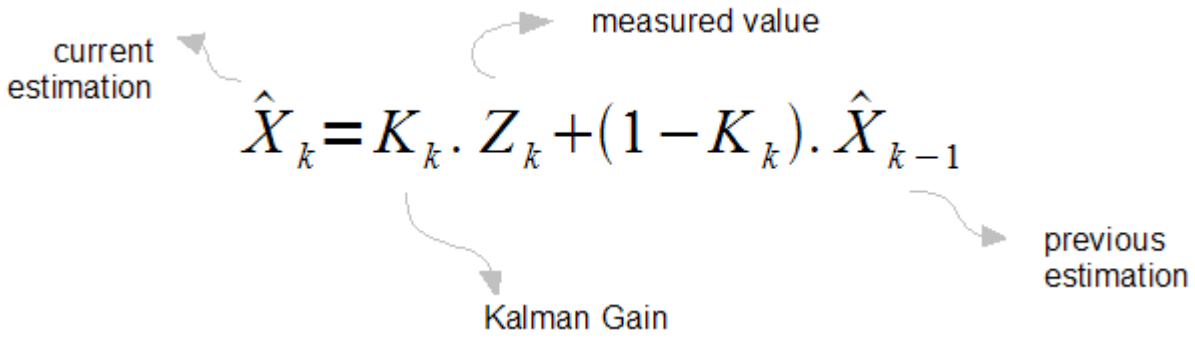
- Beamforming computes the desired angle of incidence of the source signal on the antenna arrays so that the processed beam can be directed to the same source at the computed angle of arrival.

$$\begin{pmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_{n-1}(t) \\ x_n(t) \end{pmatrix} = \begin{pmatrix} & & & \\ a(\theta_1) & a(\theta_2) & \cdots & a(\theta_k) \\ & & & \end{pmatrix} \begin{pmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_k(t) \end{pmatrix} + \begin{pmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_{n-1}(t) \\ n_n(t) \end{pmatrix}$$

$$a(\theta_k) = \begin{pmatrix} 1 \\ e^{j2\pi(d/\lambda)\sin\theta_k} \\ \vdots \\ e^{j2\pi(d/\lambda)(n-2)\sin\theta_k} \\ e^{j2\pi(d/\lambda)(n-1)\sin\theta_k} \end{pmatrix}$$

Kalman Filters

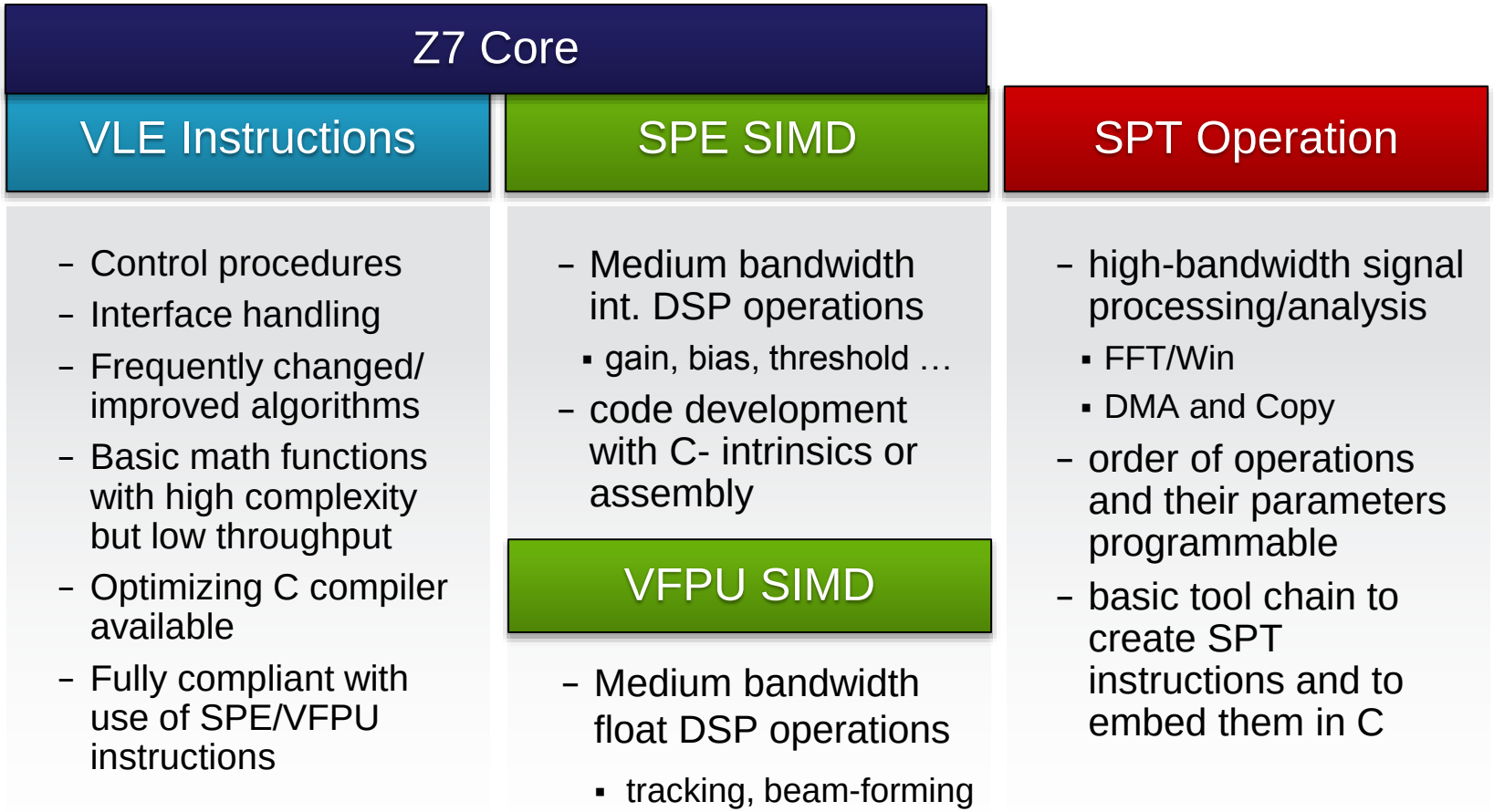
- Kalman Filters uses sequential matrix operations to predict the position and velocity of a Vehicle whose motion is linear but has unknown position and velocity.



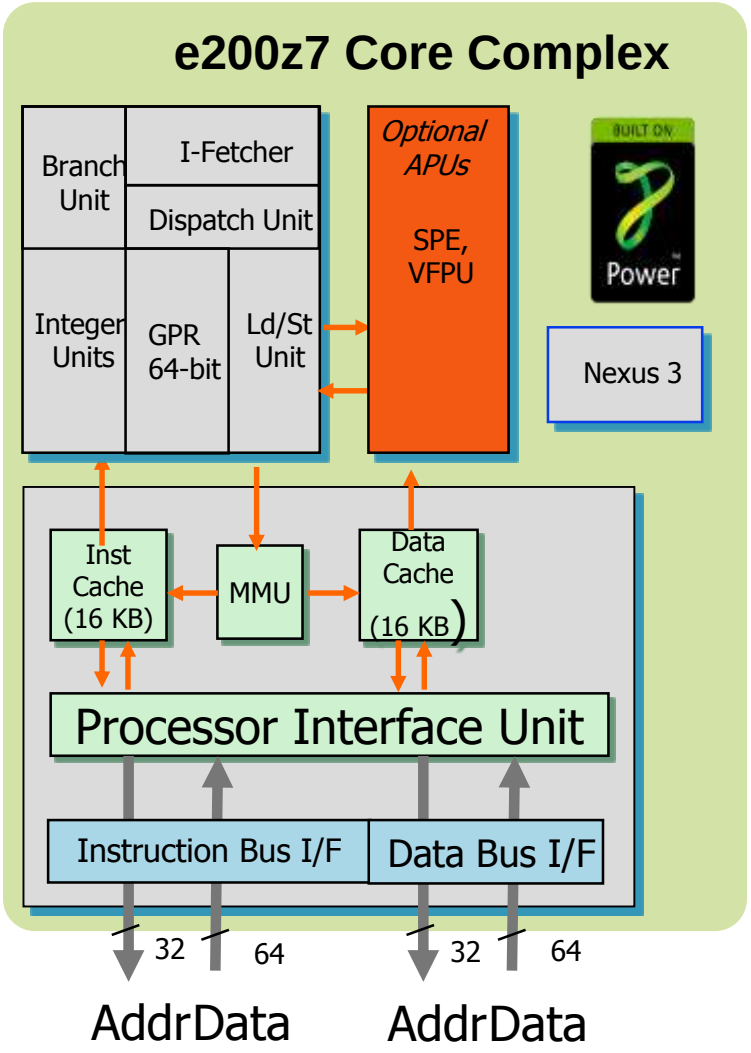
$$x_k = Ax_{k-1} + Bu_k + w_{k-1}$$

$$z_k = Hx_k + v_k$$

Algorithm Partitioning

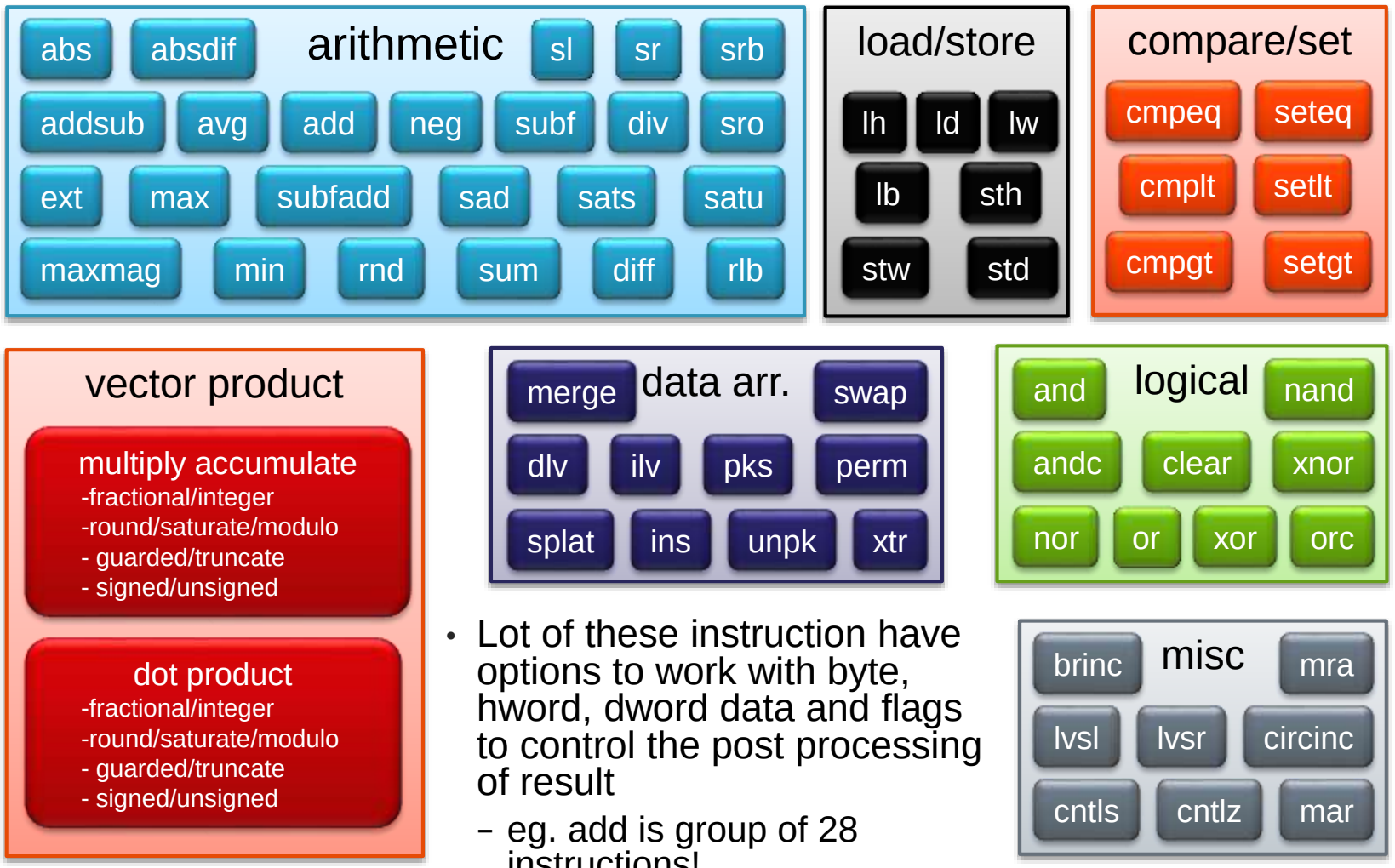


E200z7 Core Overview



- 32-bit Power Architecture processor core
  - Dual-Issue for increased throughput
  - VLE APU for reduced code footprint
  - SPE/VFPU SIMD for DSP support
  - 16KB 4-way instruction and data caches
- Features
  - Dual-issue, 10-stage pipeline
  - Single cycle simple operations
  - Multiply 1 clock throughput
  - Divide 6-16 clocks
  - 1-6-cycle branches
- Implementation
  - Low power design
  - Extensive clock gating
  - Dynamic power management of execution units, caches and MMU

SPE Instruction Set (Integer)



- Lot of these instruction have options to work with byte, hword, dword data and flags to control the post processing of result
  - eg. add is group of 28 instructions!

Summary

- Next Generations
- Q&A

www.Freescale.com