Embedded System Course 2018-19

DSP

Università Degli Studi dell'Aquila

Lecturer: Dott. Paolo Giammatteo
--
paolo.giammatteo@univaq.it

General Purpose Microprocessor

- A general purpose microprocessor is a processor that is not tied to or integrated with a particular language or piece of software
- They are often used for computation, text editing, multimedia display, and communication over a network
- These provide digital control over practically any technology, such as appliances, automobiles, cell phones, industrial process control, etc…

DSP Microprocessor

- The DSP processor, on the other hand, is a particular type of microprocessor

- A DSP processor is a specialized microprocessor that has an architecture optimized for the operational needs of digital signal processing

- It is characterized by the representation of discrete units, such as discrete time, discrete frequency, or discrete domain signals

DSP vs GPP

- The main difference between a DSP and a microprocessor is that a DSP processor has features designed to support high-performance, repetitive, numerically intensive tasks

- DSP processors are designed specifically to perform large numbers of complex arithmetic calculations and as quickly as possible

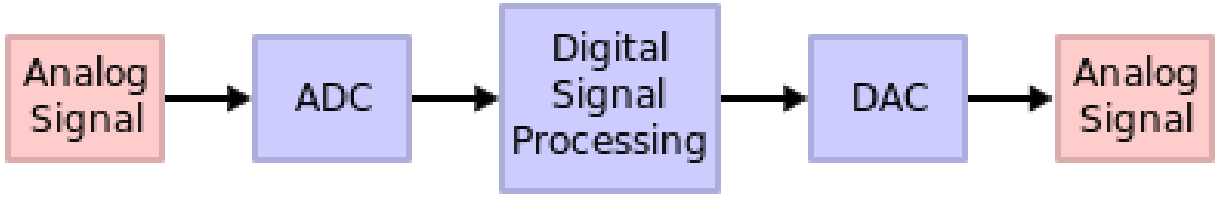
- As compared to general microprocessors, DSP processors are more efficient at performing basic arithmetic operations, especially multiplication

DSP vs GPP

- Most general-purpose microprocessors and operating systems can execute DSP algorithms successfully
- However, they GPP are not suitable for use in portable devices such as mobile phones
- Digital Signal Processors have approximately the same level of integration and the same clock frequencies as general purpose microprocessors, but they tend to have better performance

Common DSP applications

- Communications
- Audio, Video processing
- Graphics, 3-D rendering
- Navigation, radars, GPS
- Controls – Robotics, guidance, Machine Vision
- Filtering
- Frequency-Time transformations



Common DSP tasks

- Modulation-Demodulation, Error correction
- Noise reduction, equalization, echo cancellation
- Audio compression
- Vector and Matrix calculations
- Control algorithms

DSPs Need to do

- Efficient repetitive numerical calculations
- Maintain numeric fidelity
- Provide high memory bandwidth
  - Streaming data
- Real Time processing

DSPs Need to minimize

- Real Time execution unpredictability
- Memory use
- Power consumption
- Cost
- Development time

What do DSP have?

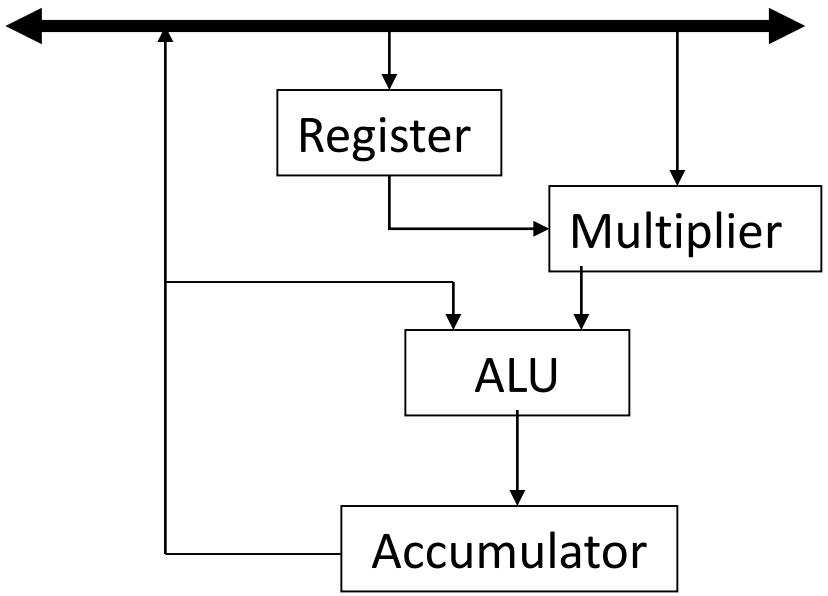
- Specialized memory architecture (Harvard)
- Specialized parallel execution units
- Specialized addressing modes
- Specialized instruction sets for parallel execution
- Specialized peripherals

DSPs Characteristics

- Designed for real-time processing
- Optimized for data streaming operations
- Separation between program memory and data memory (Harvard architecture)
- Special instructions dedicated to SIMD operations (Single Instruction, Multiple Date)
- No hardware support for multitasking operation
- Ability to perform direct access (DMA) to the memory of a host
- Processing of digital signals obtained by converting analog signals, using an analog-to-digital converter (ADC)

Multiply-Accumulate (MAC)

In digital signal processing, the multiply–accumulate operation is a common step that computes the product of two numbers and adds that product to an accumulator.



$$z_{t+1} = z_t + x * y$$

Fixed-Point Arithmetic

The DSP must be able to process data in the necessary format and with precision with which you want to implement the application. The choice of format has a profound influence on arithmetic operations as addition and multiplication, which are the heart of the DSP algorithms.

- Most DSPs use fixed point arithmetic than floating point
- Faster
- Cheaper
- Hardware support for saturation arithmetic, rounding and shifting

Fixed-Point Arithmetic

Observations:

The floating point arithmetic format is easier to program, since it is not necessary to reformulate the application to adapt it to the internal numerical representation of the processor. Typically expensive.

With the fixed point arithmetic format you have opposite characteristics compared to the case previous, complicated to program but low cost of realization.

DSP Architecture

All digital processors contain three fundamental modules within them make up the architecture:

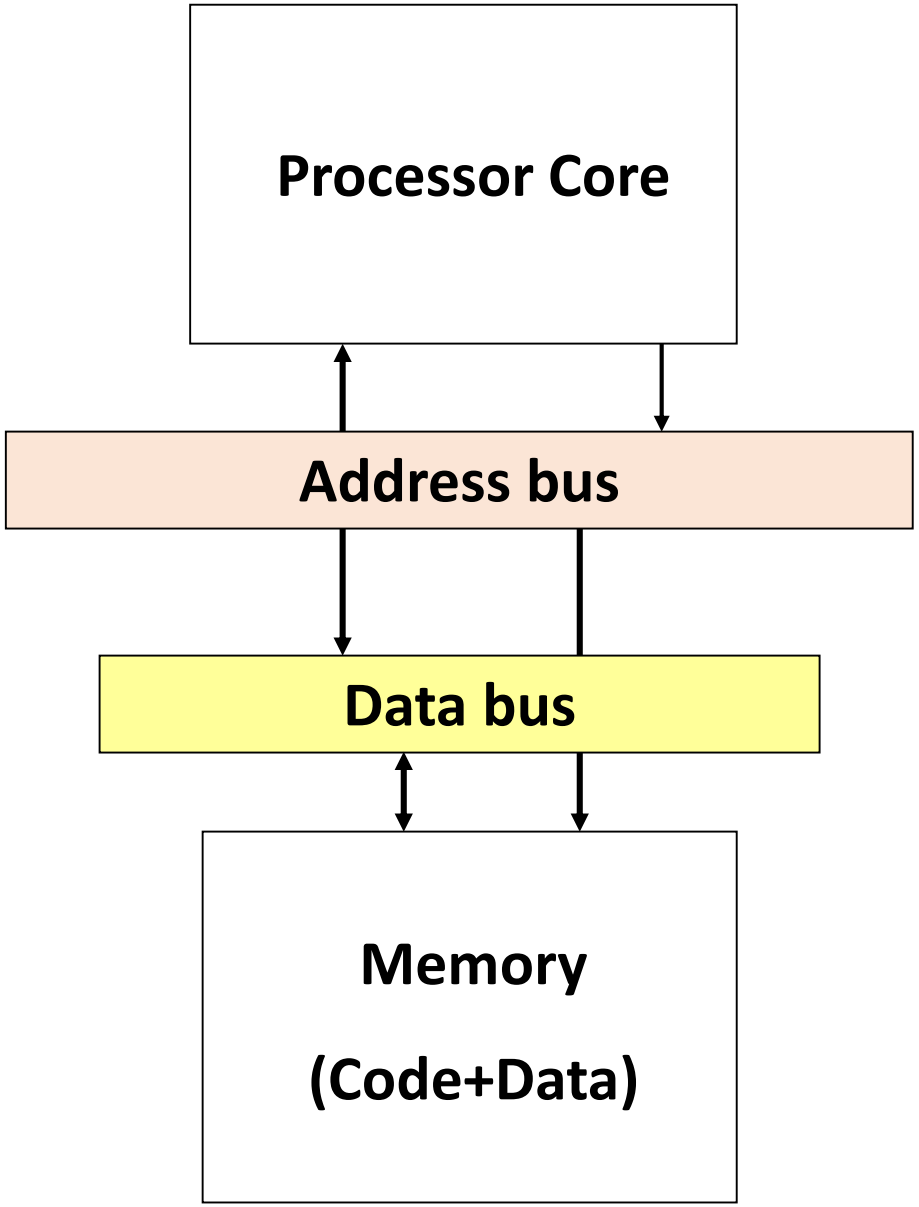
- Processing unit for logical and arithmetic operations (DATAPATH)
- Memories to contain data and program instructions
- Bus to efficiently transfer data and instructions from one memory to one ALU and vice versa

DSP Architecture

Typical operations performed cyclically by a processor are of this type:

1. reading the instruction to be executed (fetch)
2. decoding of the instruction
3. collection of data on which to operate from the memory
4. execution of the requested operations

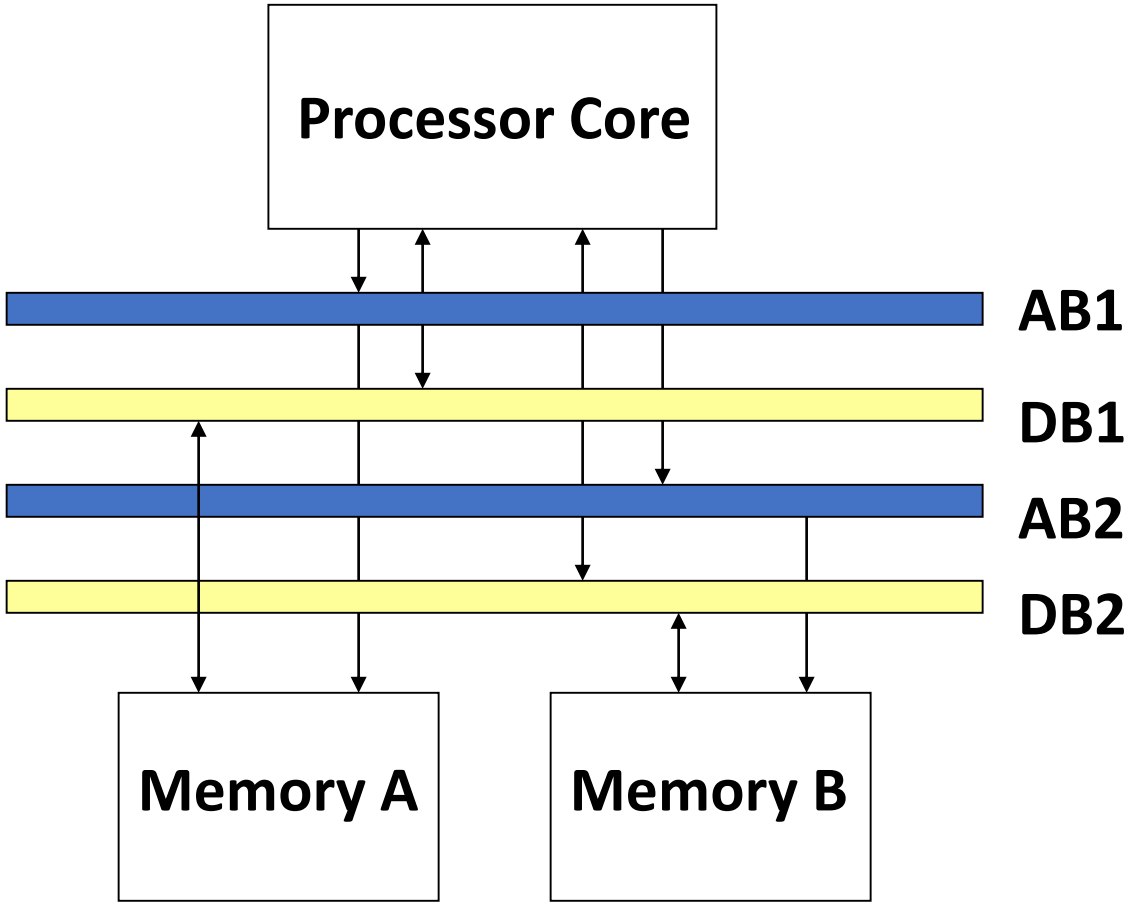
Von Neumann Architecture



1. Fetch MAC instruction
2. Read value of ‘x’
3. Read value of ‘h’
4. Multiply x, h and accumulate
5. Write result to memory

- 4 memory access operations
- One multiplication

Harvard Architecture



1. Data and Code in separate memory segments
2. Multiple address and data buses
3. Double memory bandwidth
4. Simultaneous code and data fetch

DSP Architecture

- The DSPs base their functioning on the Harvard architecture, while the General Purpose on that of like Von Newman
- From a memory point of view, a Von Newman processor performs only one access to each cycle, using other solutions to reduce execution times, for example with the use of so-called cache
- A DSP performs 2 (or more) memory accesses per instruction cycle. So at a high level, the distinction in the architecture of the memory consists in the double access way of the processor to the data memory and to that program

Additional architecture feature

- VLIW: Very Long Instruction Word:

  Systems with this type of architecture have, as main characteristic, to take so many instructions from the program memory at the same time and consider them as a single one. The name VLIW refers to the fact of considering blocks from 4 or 8 instructions at a time as a single instruction, this is made possible by a certain arrangement of the same in memory

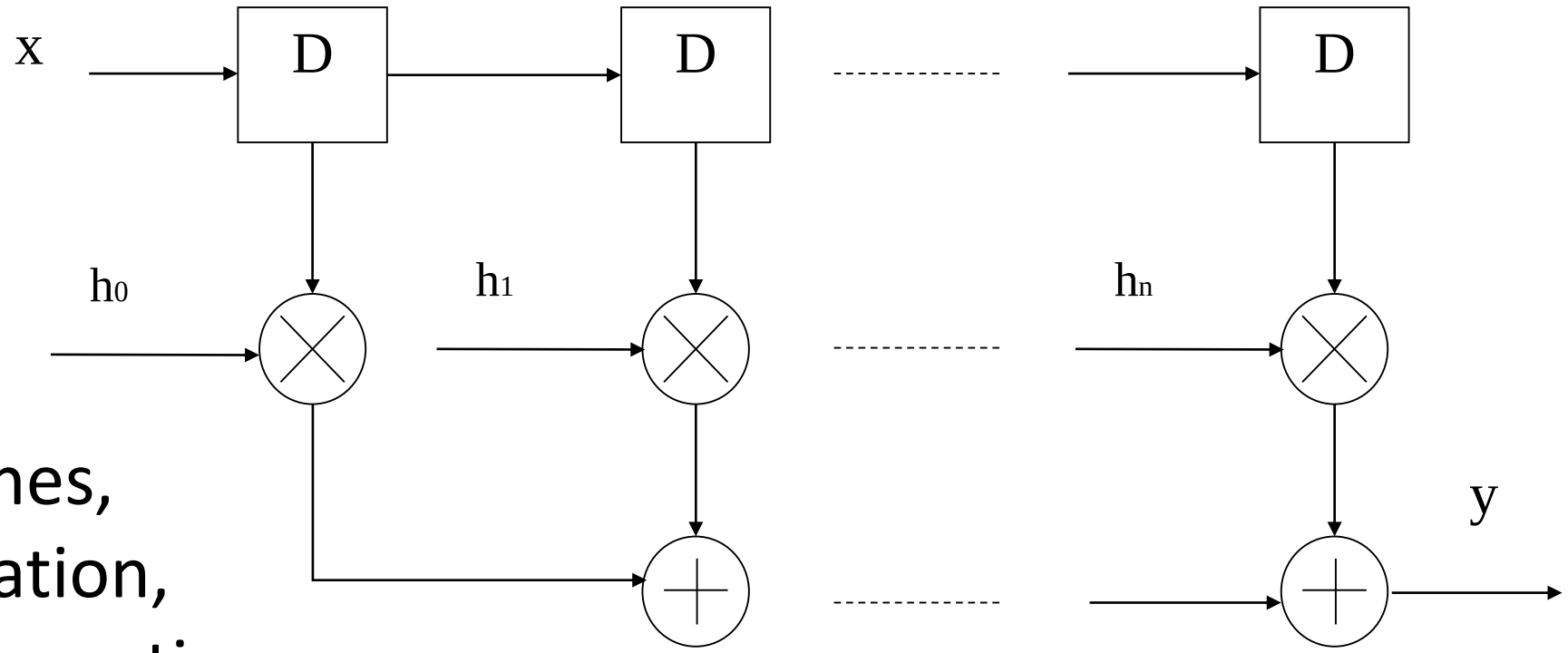
- More MACs, ALUs and other execution units

Cache in DSP

- GPPs normally contain two on-chip caches – one for data and the other for instructions
- Allows full speed retrieval of instructions and data without accessing slower off-chip memory
- DSPs contain a very small instruction cache and no data cache
- GPPs use control logic to determine what code and data goes into cache, while in DSPs it is programmer's job to make a decision

Example 1: FIR Filtering

$$y[n] = h_0 x[n] + h_1 x[n-1] + \cdots + h_N x[n-N] = \sum_{i=0}^{N} h_i x[n-i]$$

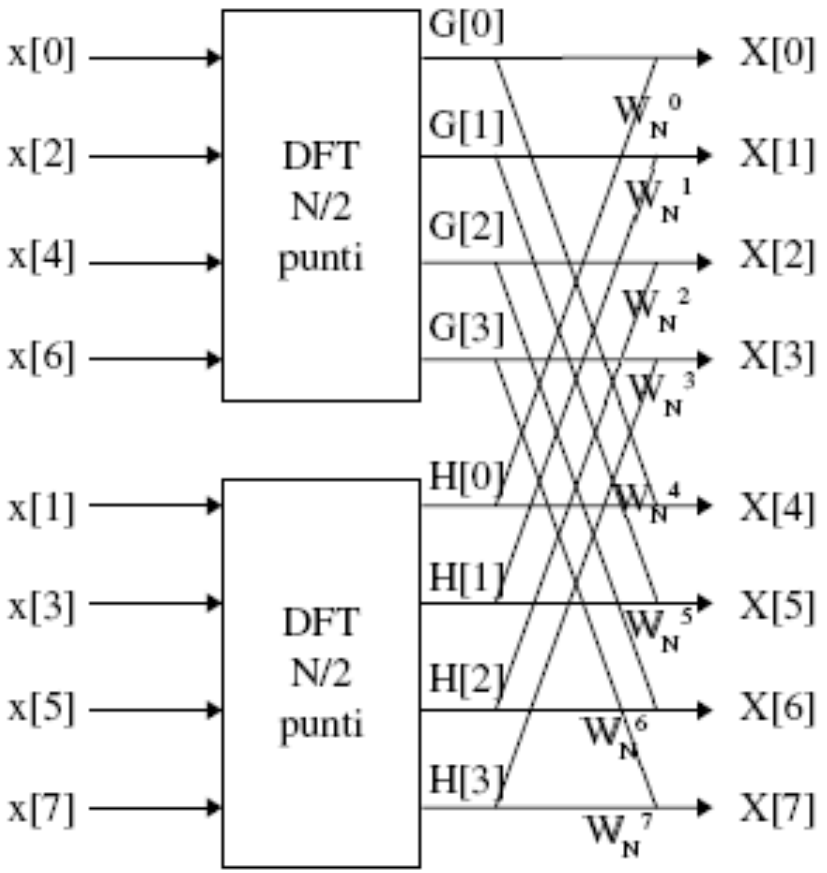


1. Two data fetches,
2. Multiply operation,
3. Accumulate Operation,
4. Input vector shifting

Example 2: FFT Transform

$$X_q = \mathcal{F}_d(x_n) = \sum_{k=0}^{N-1} x_k e^{-i\frac{2\pi}{N}kq} \qquad q = 0, 1, \ldots, N-1$$

$$W_N^{kq} = e^{-i\frac{2\pi}{N}kq}$$



Butterfly scheme

Texas Instruments TMS320

Texas Instruments TMS320 is a blanket name for a series of digital signal processors (DSPs) from Texas Instruments

The processor is available in many different variants, some with fixed-point arithmetic and some with floating point arithmetic

Three DSP families:

- C2000 (C28x + ARM Cortex M3 series, C28x Delfino floating-point series, C28x Piccolo series, C28x fixed-point series, and C240x)
- C5000 (C54x and C55x)
- C6000 (C62x, C64x, C66x and C67x)

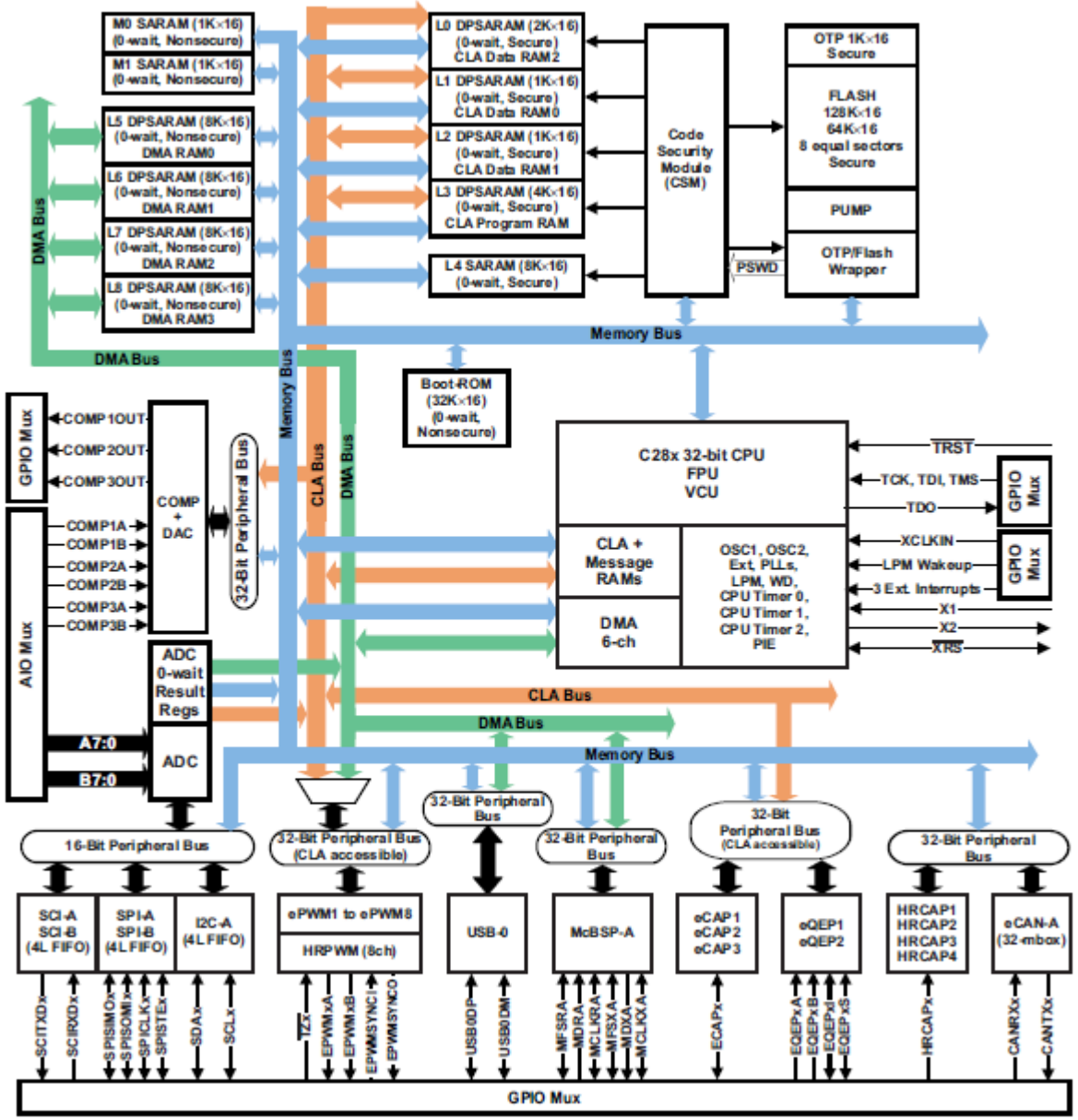
Core Benchmarks

A widespread way to evaluate performance of processors for generic use is through benchmarking, or the use of programs sample. This method has the advantage of globally evaluate the performance of a processor, including management of memory.

Function	C66x (n cycles)	C28x (n cycles)	ARM CORTEX (n cycles)
Complex FFT (256 pts)	1782	2073	8644
Real block FIR - fixed point 128 samples, 16 coeff	1345	-	6971
Log10	14	29	56
Square Root	6	16	5
ArcTan2	24	41	49

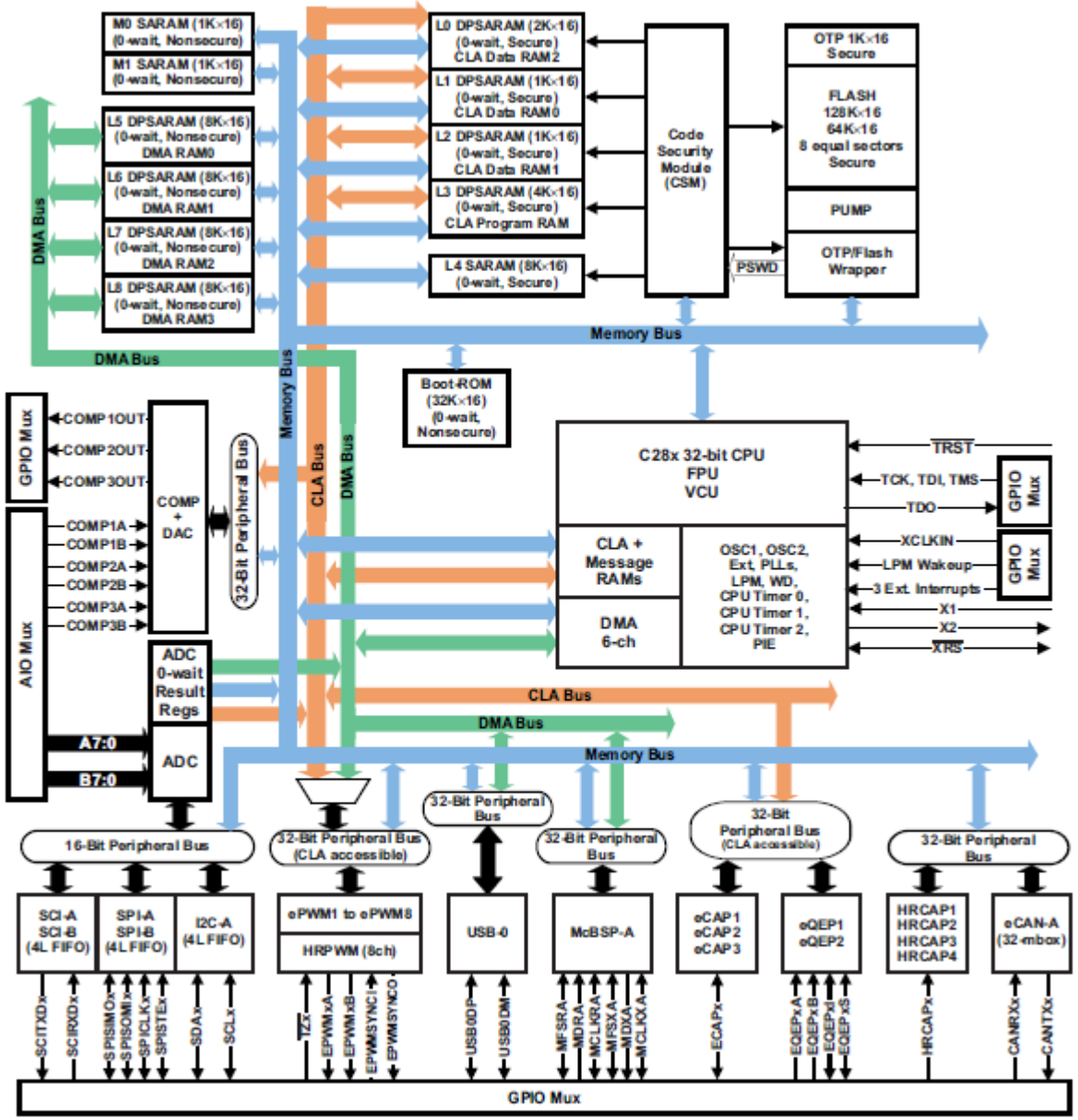
Texas Instruments TMS320C28069

- High-Efficiency 32-Bit CPU (TMS320C28x)
  - 90 MHz (11.11-ns Cycle Time)
  - 16 × 16 and 32 × 32 Multiply and Accumulate (MAC)
  - 16 × 16 Dual MAC
  - Harvard Bus Architecture
  - Code-Efficient (in C/C++ and Assembly)
- Floating-Point Unit (FPU)
  - Native Single-Precision Floating-Point Operations
- Programmable Control Law Accelerator (CLA)
  - 32-Bit Floating-Point Math Accelerator
  - Executes Code Independently of the Main CPU
- Viterbi, Complex Math, CRC Unit (VCU)
- Embedded Memory
  - Up to 256KB of Flash
  - Up to 100KB of RAM
  - 2KB of One-Time Programmable (OTP) ROM
- 6-Channel Direct Memory Access (DMA)

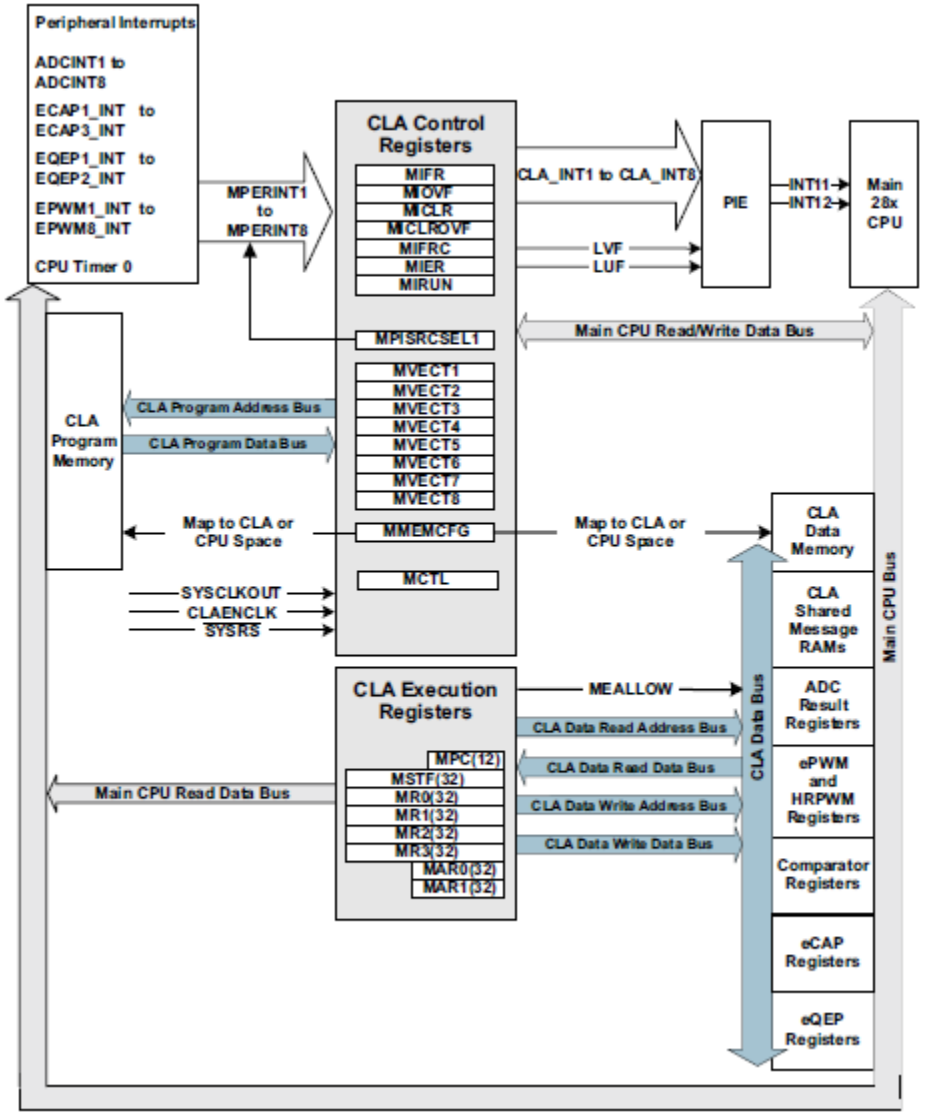


Texas Instruments TMS320C28069

- Three 32-Bit CPU Timers
- Up to 8 Enhanced Pulse-Width Modulator (ePWM) Modules
- Three Input Enhanced Capture (eCAP) Modules
- Up to 4 High-Resolution Capture (HRCAP) Modules
- Up to 2 Enhanced Quadrature Encoder Pulse (eQEP) Modules
- 12-Bit Analog-to-Digital Converter (ADC), Dual Sample-and-Hold (S/H)
- On-Chip Temperature Sensor
- Serial Port Peripherals
  - Two Serial Communications Interface (SCI) [UART] Modules
  - Two Serial Peripheral Interface (SPI) Modules
  - One Inter-Integrated-Circuit (I2C) Bus
  - One Multichannel Buffered Serial Port (McBSP) Bus
  - One Enhanced Controller Area Network (eCAN)
  - Universal Serial Bus (USB) 2.0
  - Full-Speed Device Mode
  - Full-Speed or Low-Speed Host Mode

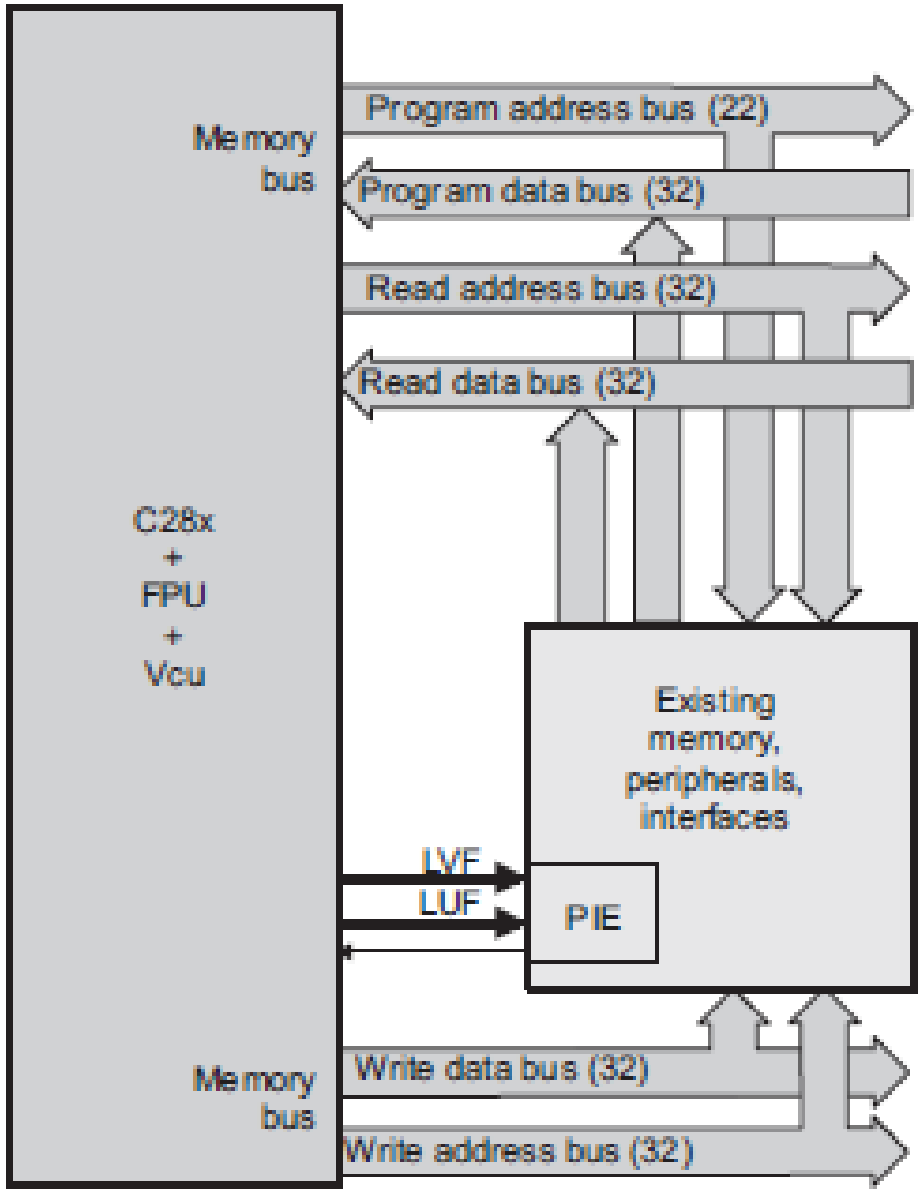


Texas Instruments TMS320C28069



The control law accelerator (CLA) is an independent, fully-programmable, 32-bit floating-point math processor that brings concurrent control-loop execution to the C28x family. The control law accelerator extends the capabilities of the C28x CPU by adding parallel processing. Time critical control loops serviced by the CLA can achieve low ADC sample to output delay. Thus, the CLA enables faster system response and higher frequency control loops.

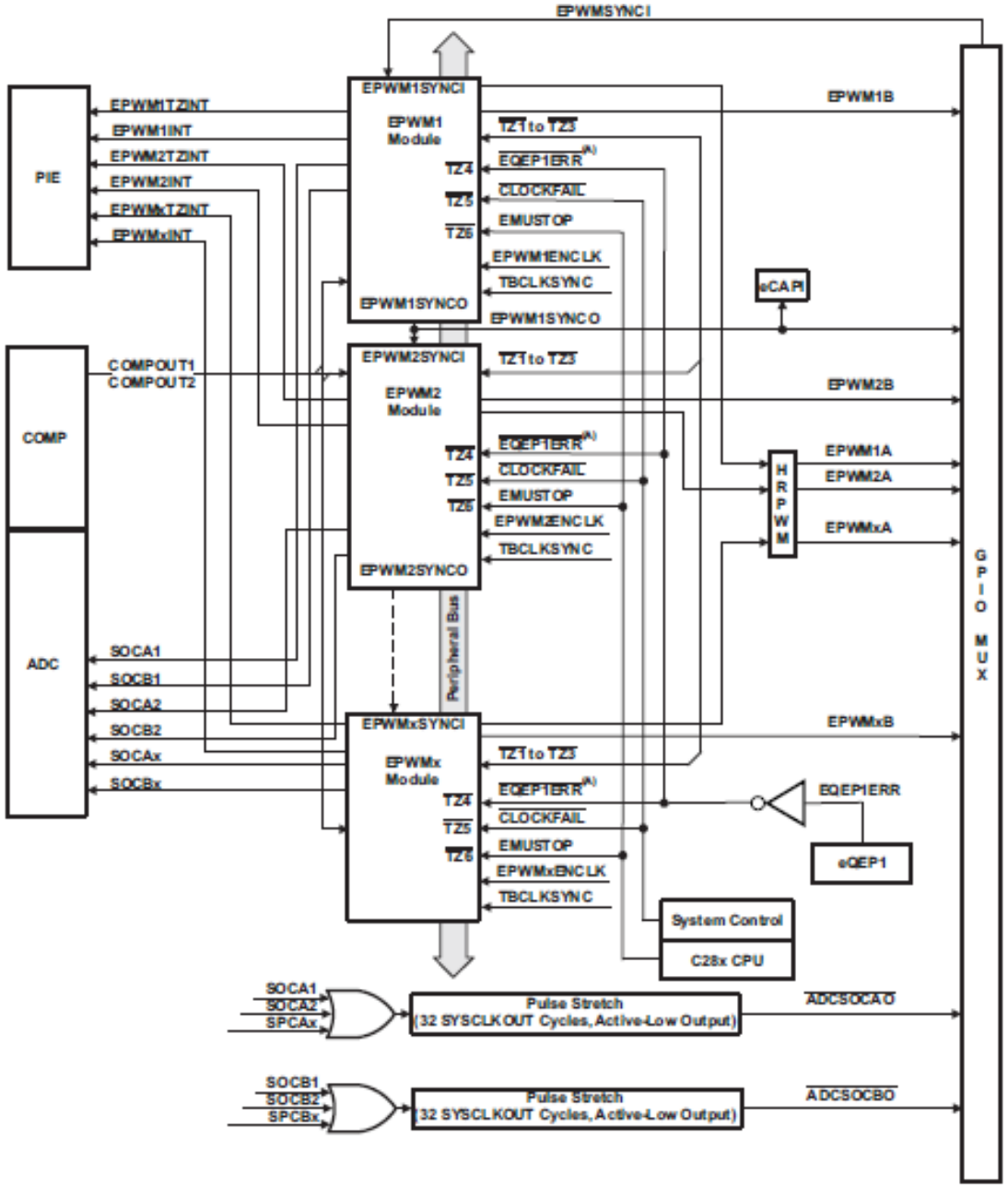
Texas Instruments TMS320C28069



The C28x Viterbi, Complex Math and CRC Unit (VCU) is a fully programmable block which accelerates the performance of communications-based algorithms by up to a factor of 8X over C28x alone.

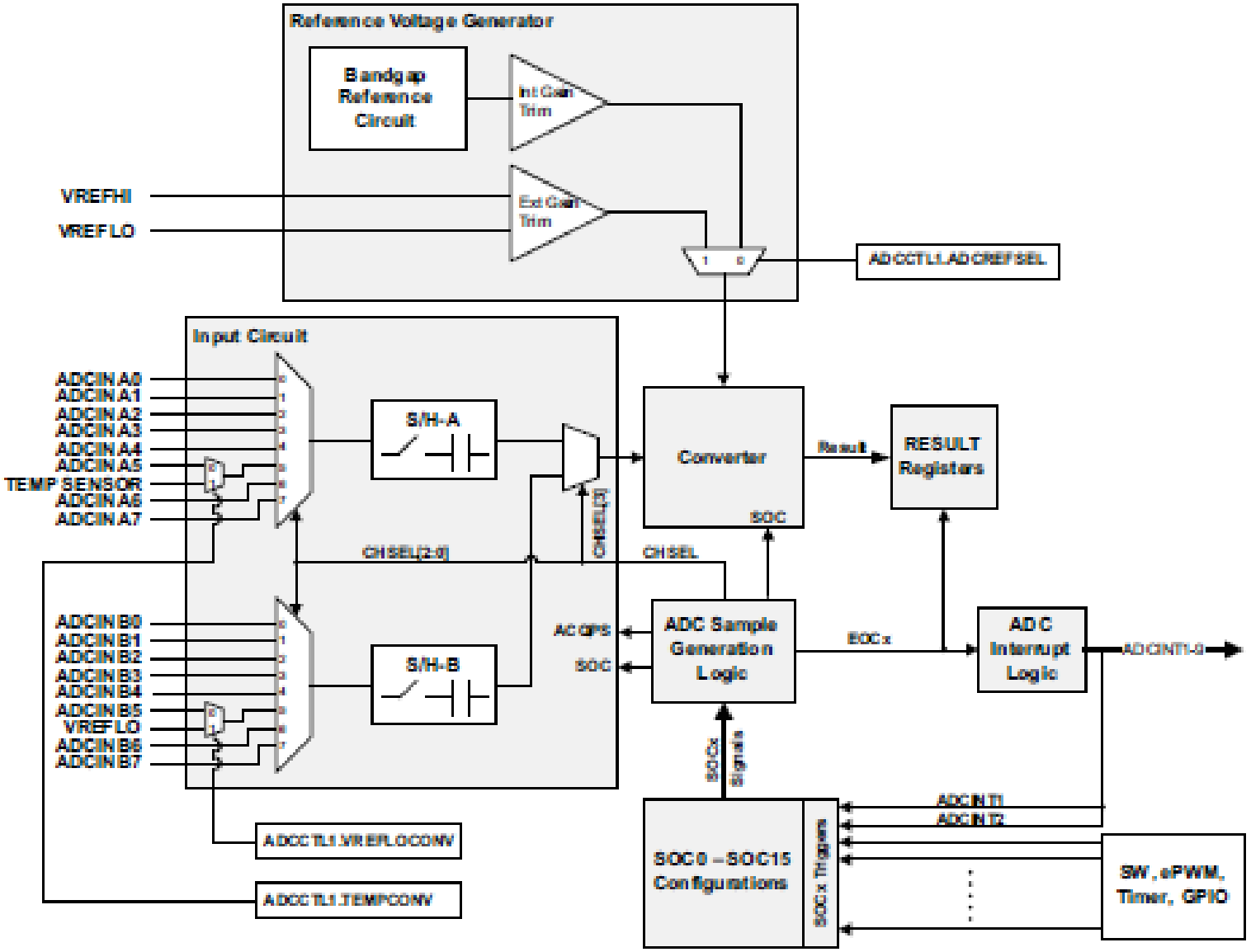
- Viterbi decoding
- Cyclic redundancy check (CRC)
- Complex math

Texas Instruments TMS320C28069



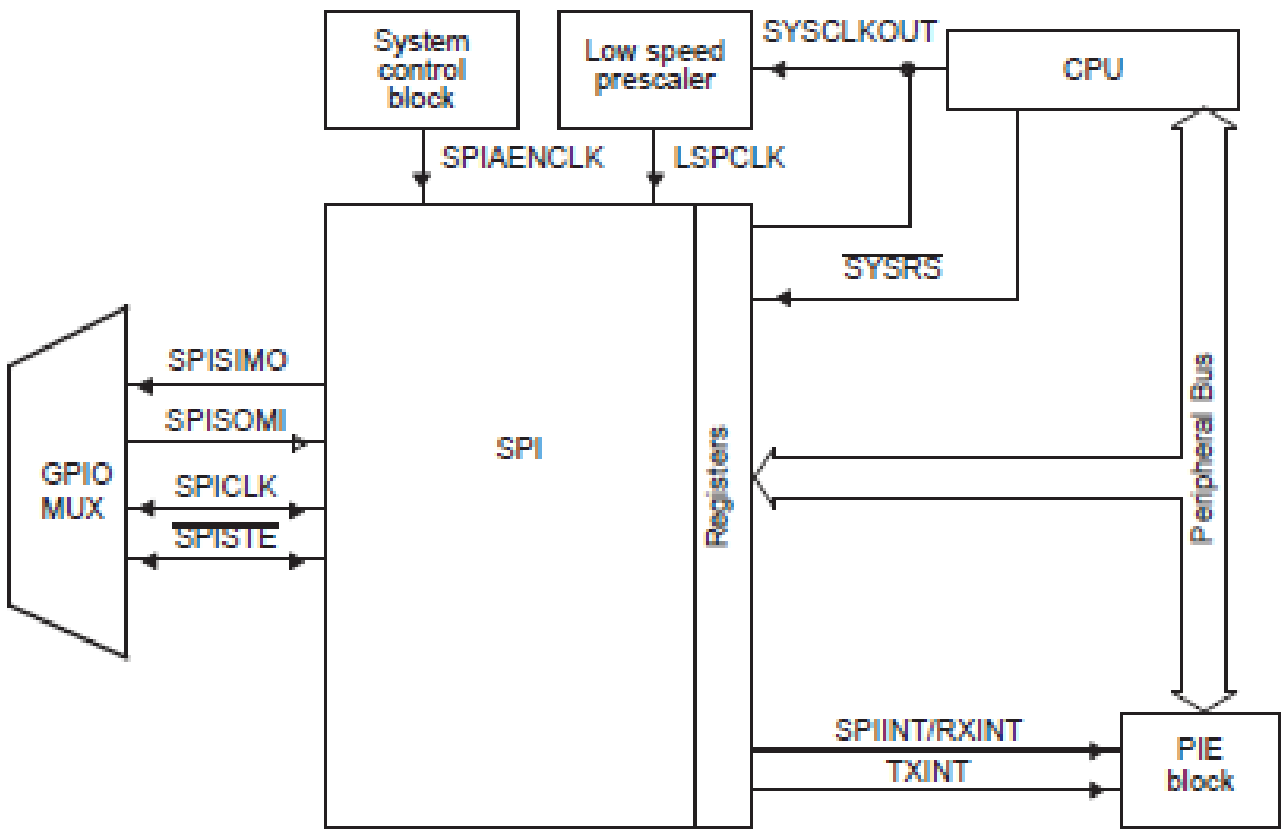
The enhanced pulse width modulator (ePWM) peripheral is a key element in controlling many of the power electronic systems found in both commercial and industrial equipments. These systems include digital motor control, switch mode power supply control, uninterruptible power supplies (UPS), and other forms of power conversion. The ePWM peripheral performs a digital to analog (DAC) function, where the duty cycle is equivalent to a DAC analog value; it is sometimes referred to as a Power DAC.

Texas Instruments TMS320C28069



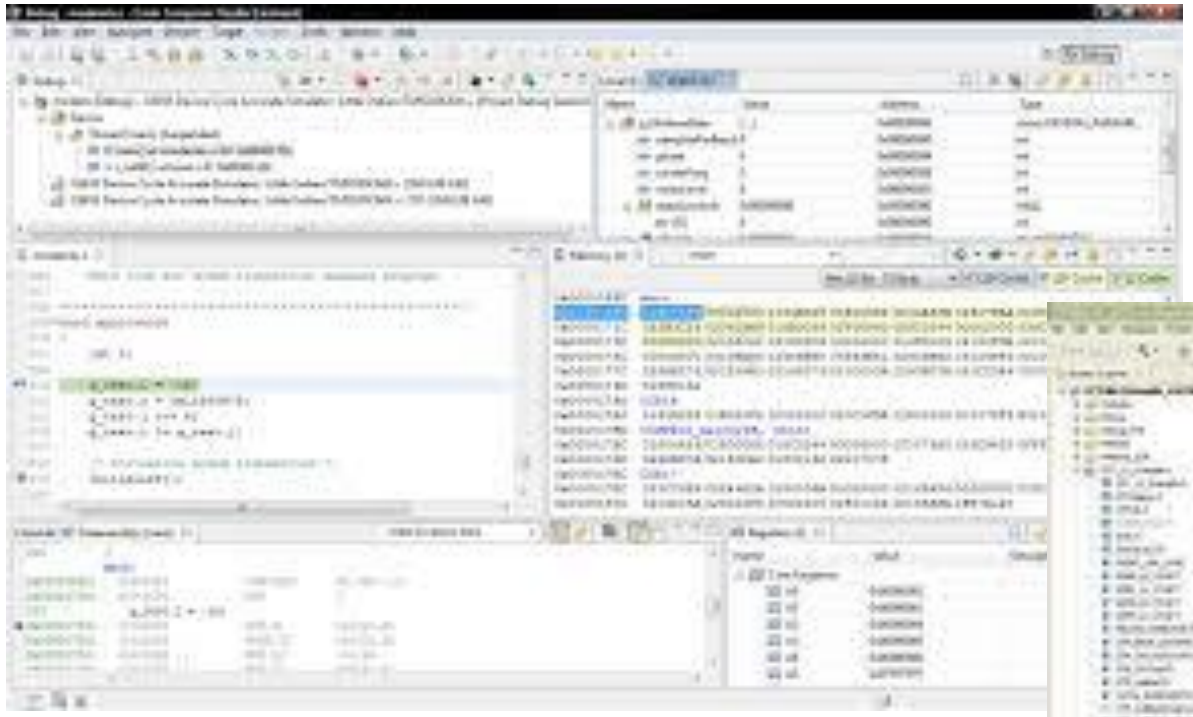
The core of the ADC contains a single 12-bit converter fed by two sample and hold circuits.

Texas Instruments TMS320C28069

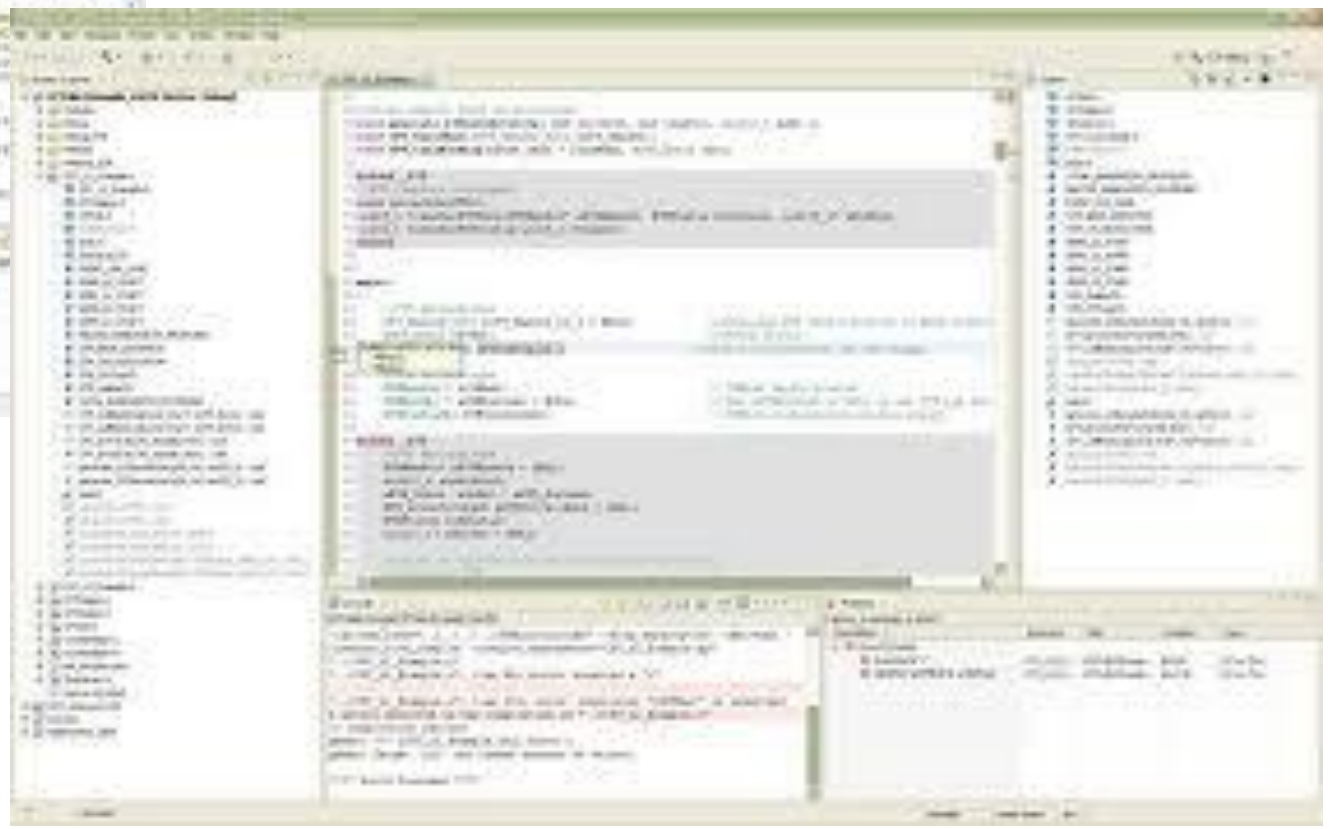


The serial peripheral interface (SPI) is a high-speed synchronous serial input/ output (I/O) port that allows a serial bit stream of programmed length (one to 16 bits) to be shifted into and out of the device at a programmed bit-transfer rate. The SPI is normally used for communications between the DSP controller and external peripherals or another controller.

How it is programmed



Code Composer Studio



http://www.ti.com/tool/CCSTUDIO

Thanks for your attention