Digital Control Applications in Power Electronics

Simone Buso

University of Padova - Italy
Department of Electronics and Informatics
Power Electronics Laboratory

e-mail: simone.buso@dei.unipd.it

Lesson 1

DSP applications in power electronics: ADMC401 and TMS320F240

Introduction

Digital Control Advantages:

- Flexibility
- Ease of upgrade
- Man to Machine Interface (MMI)
- Sophisticated control techniques
- Reduced number of components
- Unsensitivity to components' ageing …

Digital Control Disadvantages:

- Design complexity
- Cost
- Dynamic performance (sampling frequency, quantization...)

Introduction

Available tools for digital control implementation

- Microcontrollers (μC)
  - CISC (Complex Instruction Set Computer) machines (micro-coded instructions)
  - RISC (Reduced Instruction Set Computer) machines (with/without hardware multiplier)
  - 4 to 32 bit CPU and bus
- Digital Signal Processors (DSP)
  - Fixed-point arithmetic
  - Floating-point arithmetic

Introduction

General μC features:

- specifically designed for control tasks
- A/D converters almost always included on chip
- capture and compare input pins available
- several programmable I/O pins
- timers and counters available (PWM)
- different kinds of external memory available (ROM, EEPROM, FLASH)
- different computational powers available (from 32 bit RISC to simple 8 bit CPU's and even less)
- lots of third party development tools (EV-Boards, in-circuit emulators (ICE's))

Introduction

General DSP features:

- not specifically designed for control tasks
- A/D converters not always included on chip
- useful peripheral units (CapCom, timers, PWM) normally not present on chip
- very high computational power available (32 bit RISC CPU's both fixed and floating point)
- relatively few third party development tools (EV-Boards)
- DSP solutions for power electronic applications are now available (TMS320F240, ADMC401)

Digital Signal Processors [1]

First development in the late 70's (TMS320C10 - 1979). Typically designed for open loop digital signal processing e.g.:

- real time FFT calculation;
- digital filtering of sampled signals.

The market for this kind of applications is enormous and steadily growing, including telecom and consumer electronics.

Applications in industrial electronics (control tasks) are rather insignificant in volume. The manifacturers offer very few control oriented solutions.

Digital Signal Processors

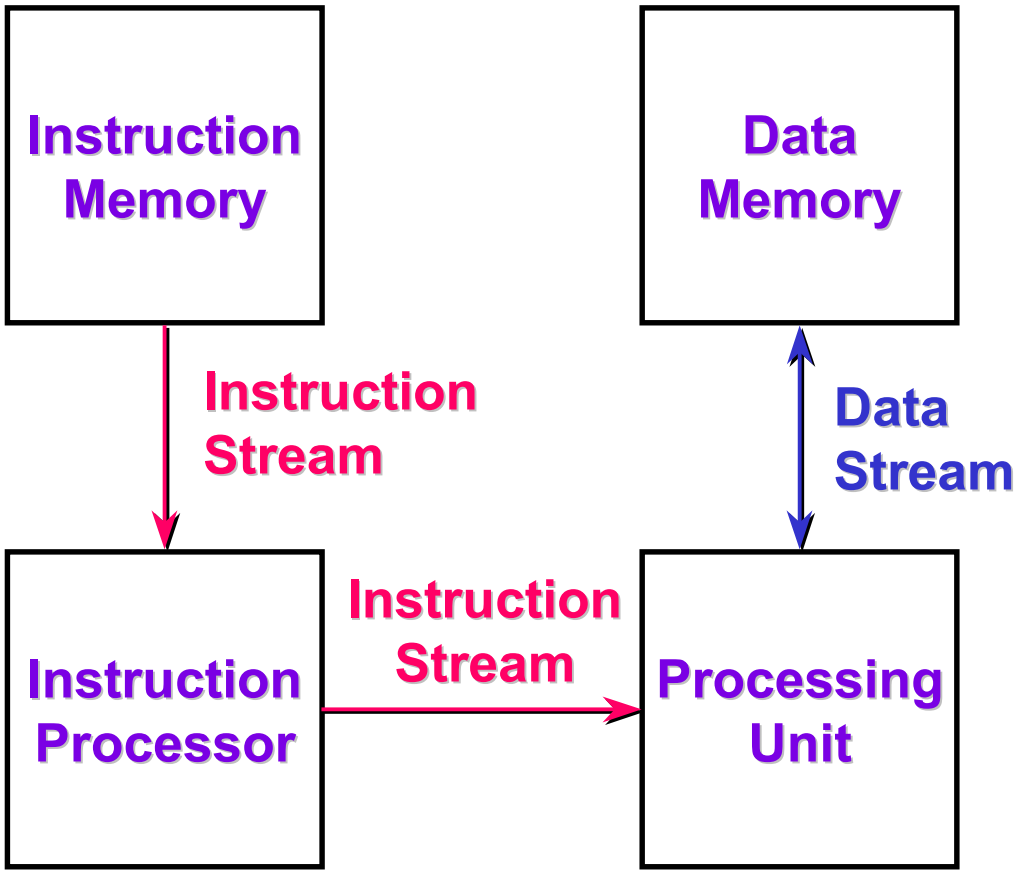
Two types of DSP's can be found:

- fixed point DSP (16, 24 bit);
- floating point (typically 32 bit or more).

For control applications, fixed point DSP's are very close to modern top-level μC's (RISC machines with Harvard architecture and hardware multiplier) in terms of performance. They are normally much less effective in terms of peripherals. Globally, they appear more expensive.

Floating point DSP's are very advantageous, but only for those applications where high cost for the control system can be afforded. The lack of peripheral units is almost total ($\Rightarrow$ you must add them).

Digital Signal Processors



Schematic diagram of a DSP with Harvard architecture.

Parallel processing of instructions and data is allowed by multiple bus architecture.

The instruction processor takes care of address computations.

Digital Signal Processors

Modern DSP's normally adopt modified Harvard architectures featuring:

- improved instruction processors with more than a single address generator;
- improved processing units with multiple independent logic units (ALU, MAC, SHIFT);
- application specific hardware modules (e.g. registers, timers etc.) in the CPU to speed-up typical calculations (e.g. DFT);
- different internal memory structures with multiple data memories and buses or mixed program and data memories (useful for filter coefficients).

New DSP solutions

Recently, new DSP solutions have been proposed by some manufacturers which merge the DSP computational capabilities with some typical μC peripheral units.

The idea is to provide the designer with a control oriented DSP, allowing the direct implementation of closed loop digital control of power converters with minimum additional interface circuitry.

The main advantage with respect to any μC is in the very high computational capability of the DSP.

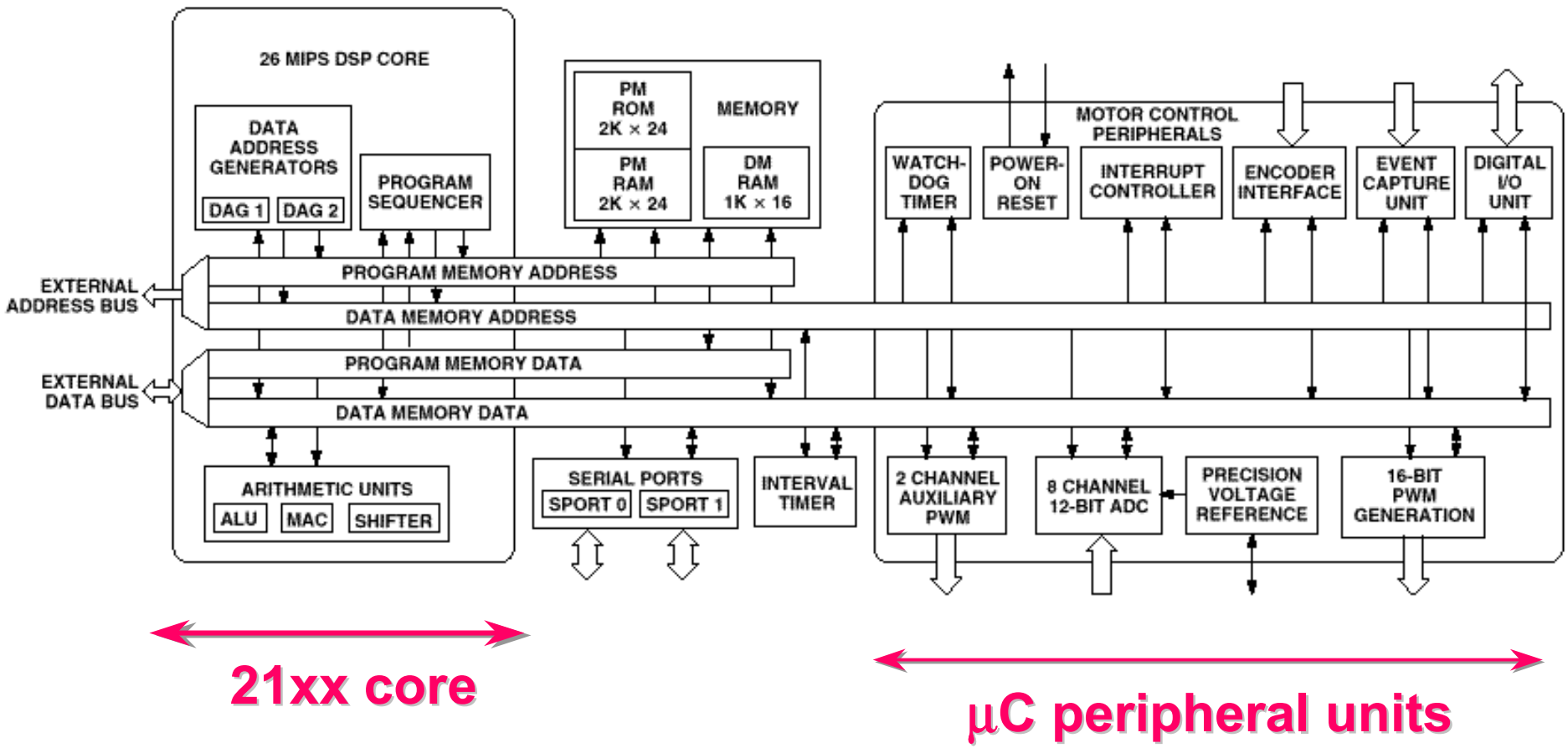
Since these devices are not very popular, the cost factor may be their weak point.

Analog Devices ADMC401 [2]

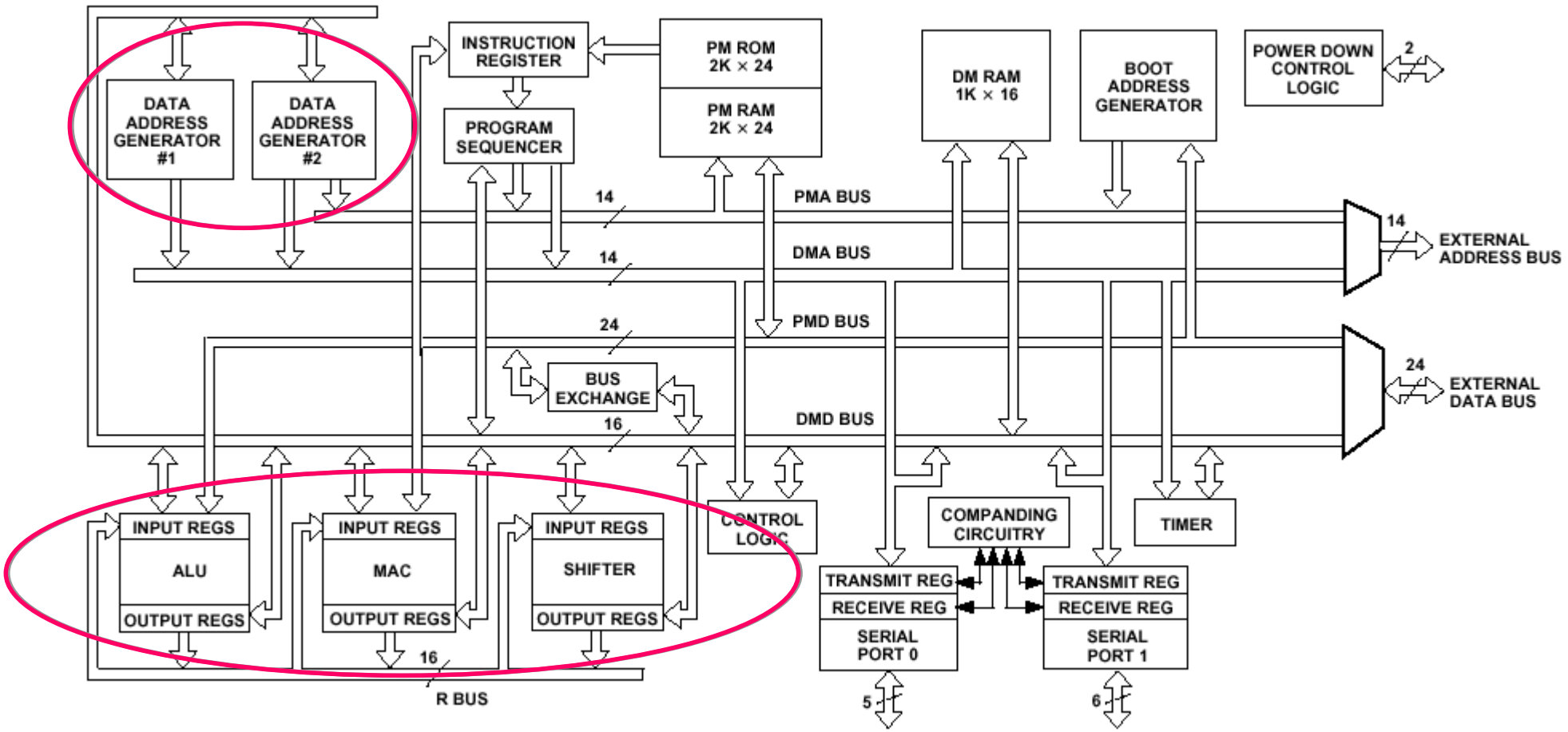
- Fixed-point DSP core features:
  - 26 MIPS performance;
  - ADSP 21xx compatible;
  - Single cycle instruction execution (38.5 ns @ 13 MHz clock frequency);
  - 16 bit arithmetic and logic unit;
  - Single Cycle 16 bit X 16 bit MAC.
- Built-in peripheral units:
  - High resolution multi-channel ADC;
  - Three-phase 16 bit PWM generation unit;
  - dual channel event timer unit (CAPCOM);
  - Incremental encoder interface unit.

Analog Devices ADMC401

Functional Block Diagram



ADMC401: Central Processing Unit



ADMC401: Central Processing Unit

In one processor cycle the DSP core can:

- Generate the next program address.
- Fetch the next instruction.
- Perform one or two data moves.
- Update one or two data address pointers.
- Perform a computational operation.

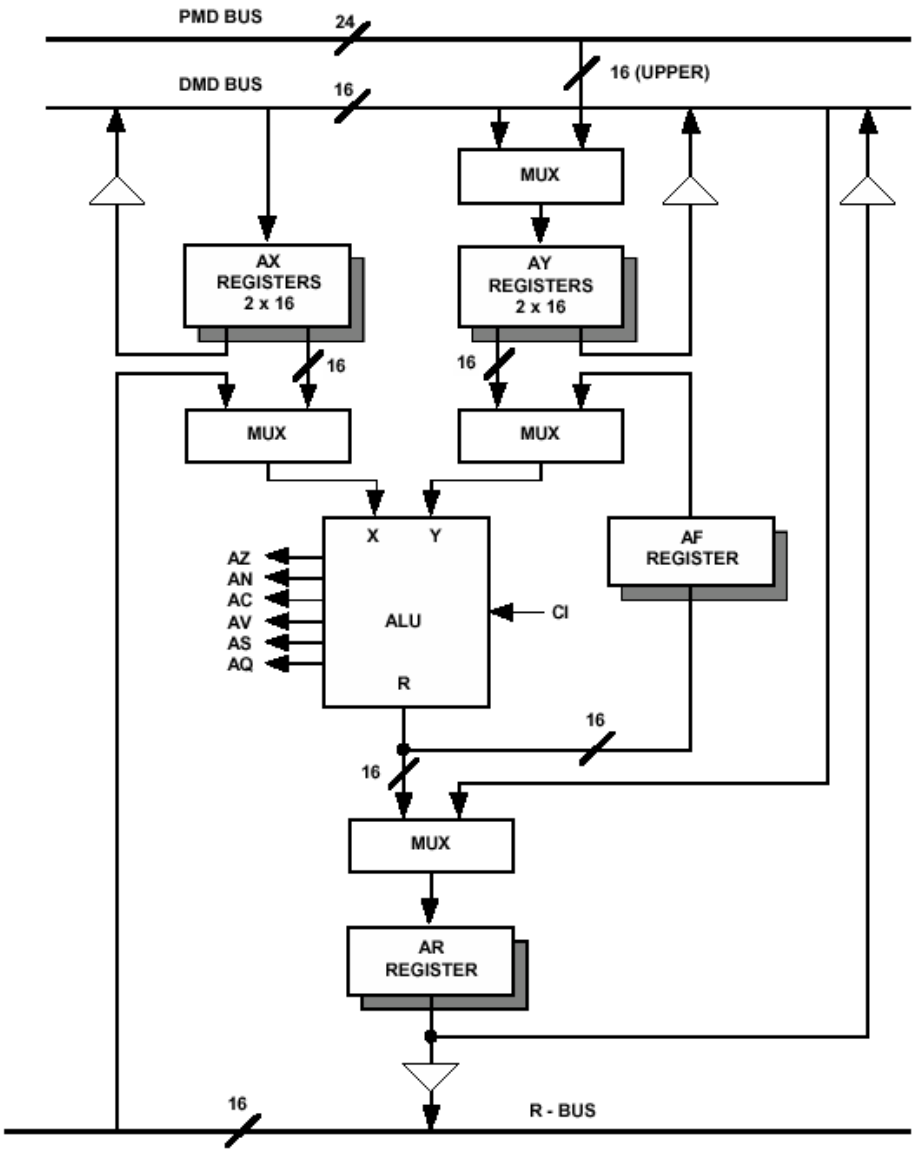
At the same time, the ADMC401 continues to:

- Receive and transmit through the serial ports.
- Decrement the interval timers.
- Generate PWM signals.
- Convert the ADC input signals.
- Operate the encoder interface unit.
- Operate all other peripherals.

ADMC401: Central Processing Unit

- The ALU performs a standard set of arithmetic and logic operations; division primitives are also supported.
- The MAC performs single-cycle multiply, multiply/add, multiply/subtract operations with 40 bits of accumulation.
- The shifter performs logical and arithmetic shifts, normalization, denormalization and derive exponent operations. The shifter can be used to implement numeric format control efficiently.
- The internal result (R) bus directly connects the computational units so that the output of any unit may be the input of any unit on the next cycle.

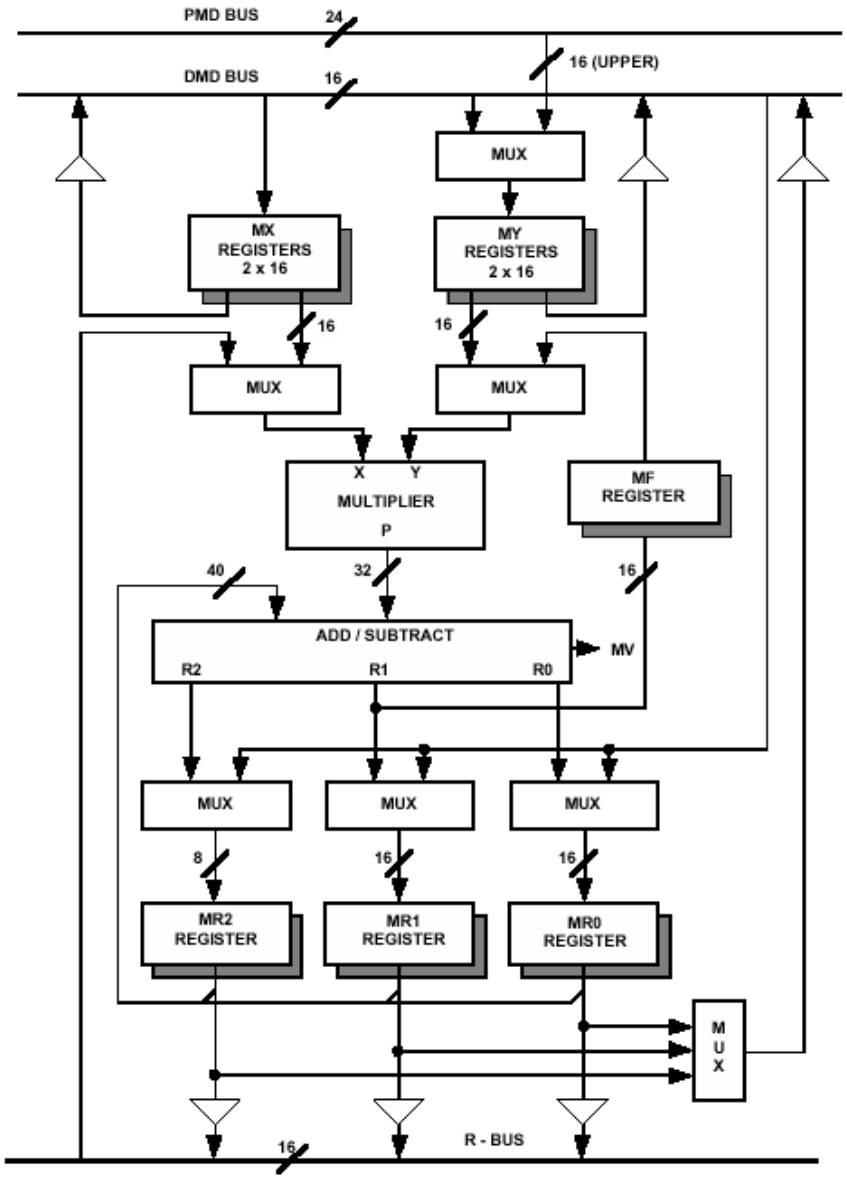
ADMC401: Central Processing Unit



The ALU block diagram allows to visualize the function of each ALU register.

- AX1 AX2: X operand;
- AY1 AY2: Y operand;
- AR: result (used also as X operand);
- AF: copy of AR (used also as Y operand);

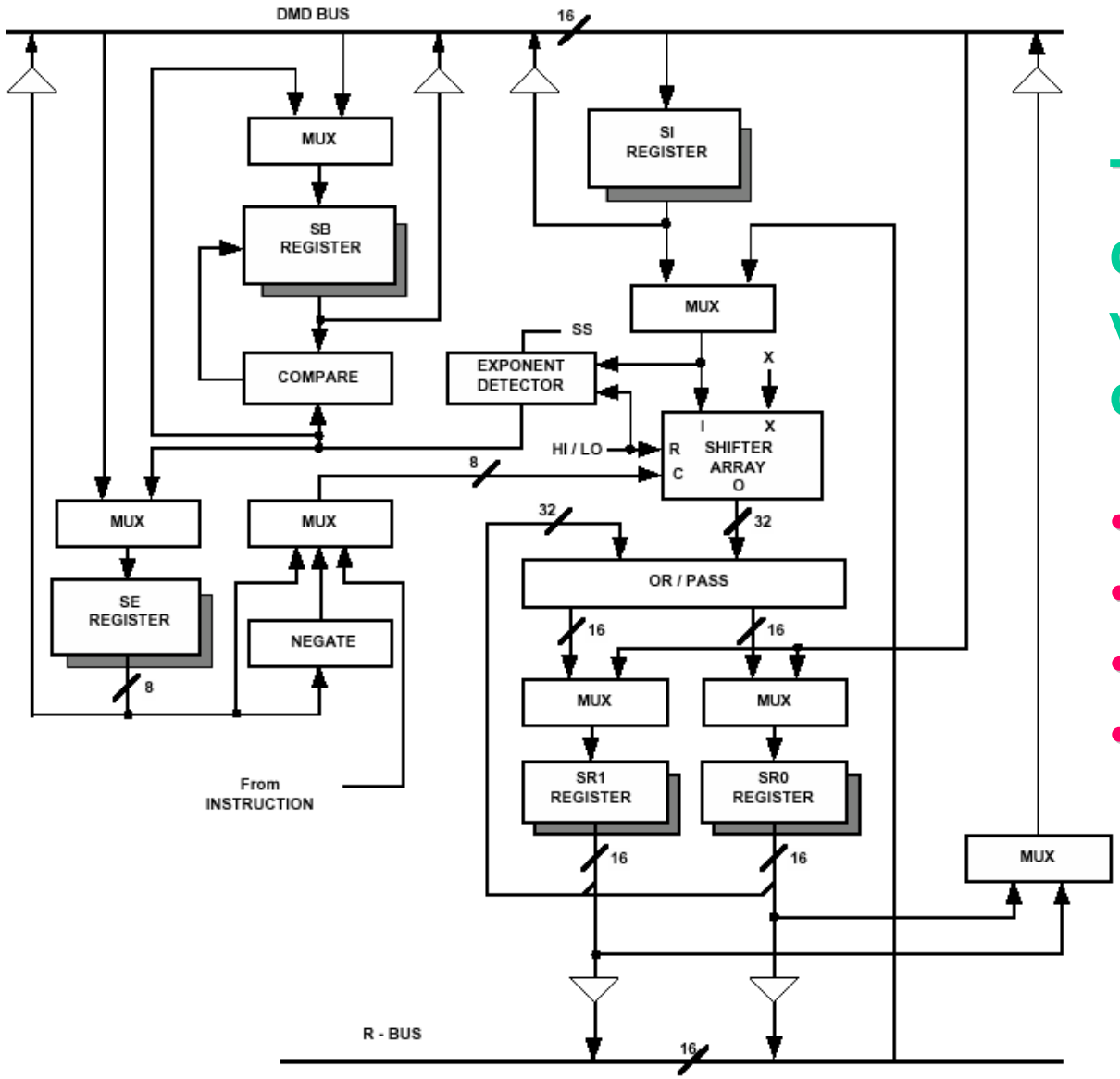
ADMC401: Central Processing Unit



The MAC block diagram allows to visualize the function of each MAC register. The structure replicates that of the ALU.

- **MX1 MX2:** X operand;
- **MY1 MY2:** Y operand;
- **MR0 MR1 MR2:** result (used also as X operand);
- **MF:** copy of MR (used also as Y operand);

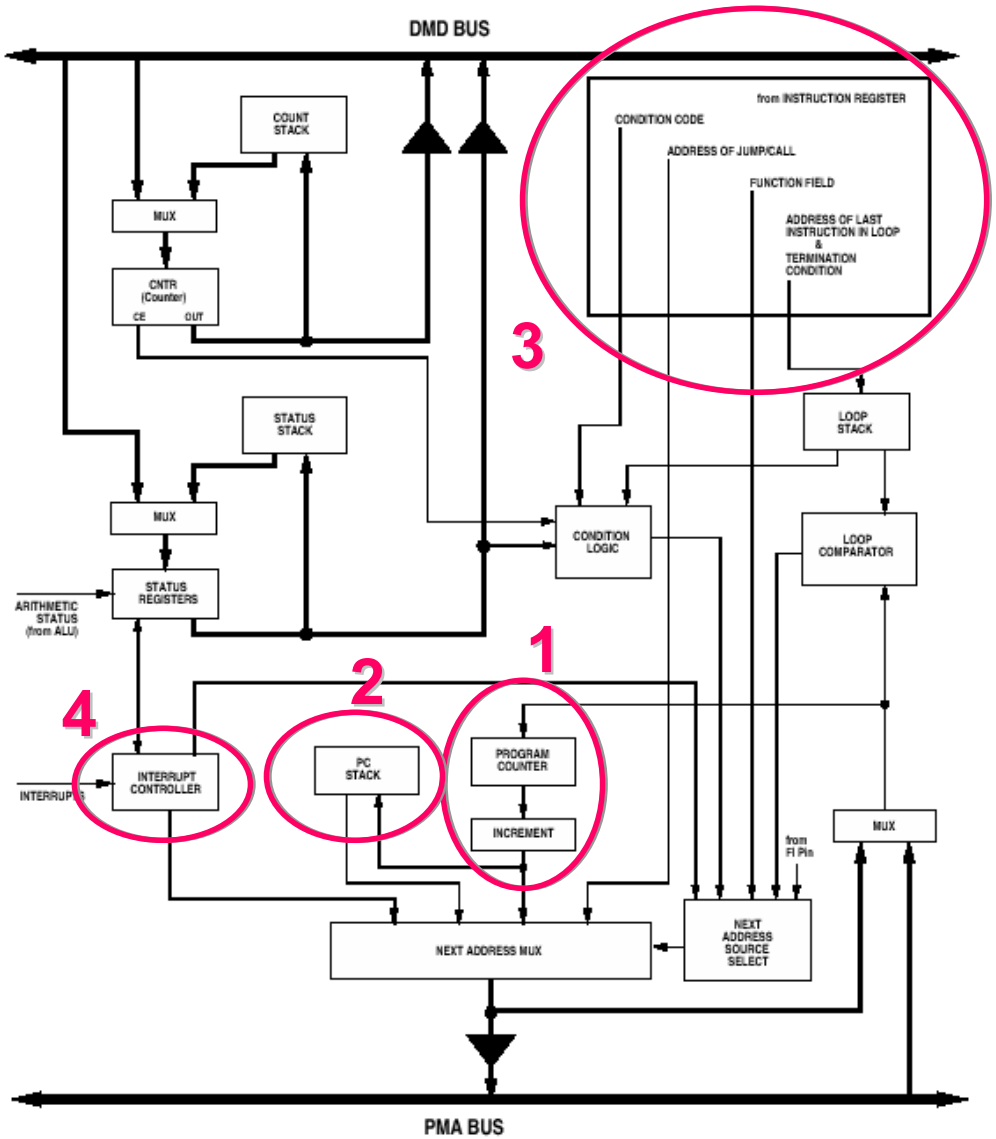
ADMC401: Central Processing Unit



The Shifter block diagram allows to visualize the function of each register.

- SI: input;
- SB: block exponent;
- SR0 SR1: result;
- SE: exponent register.

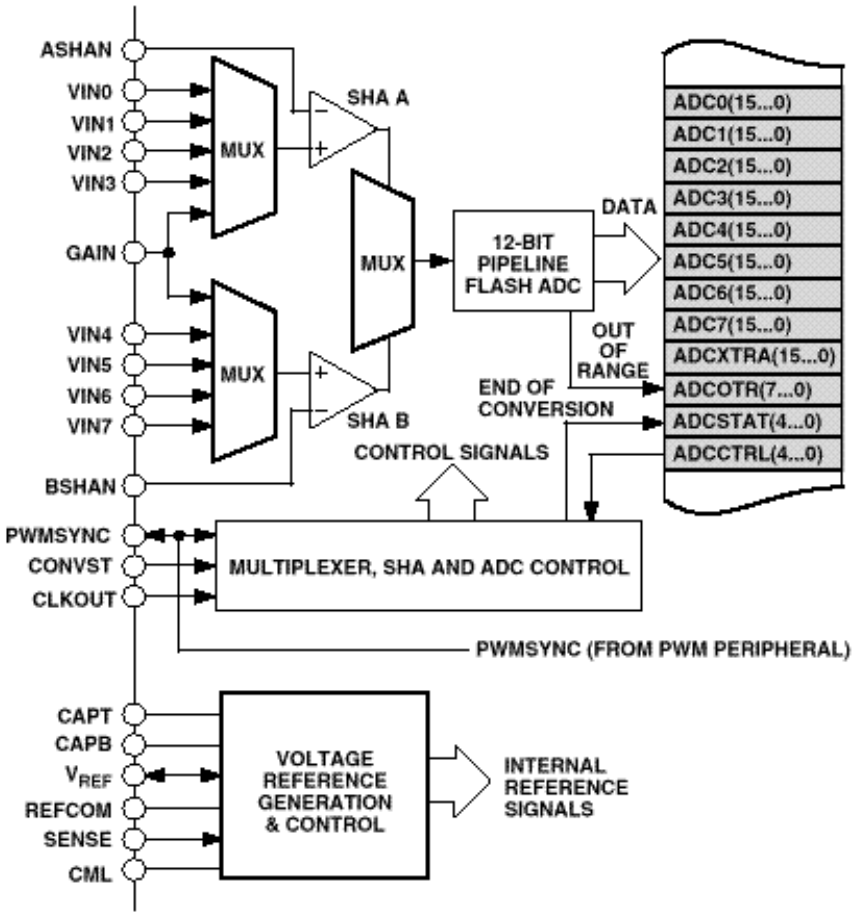
ADMC401: Central Processing Unit



The program sequencer pre-fetches the next instruction, generating the address from one of four sources:

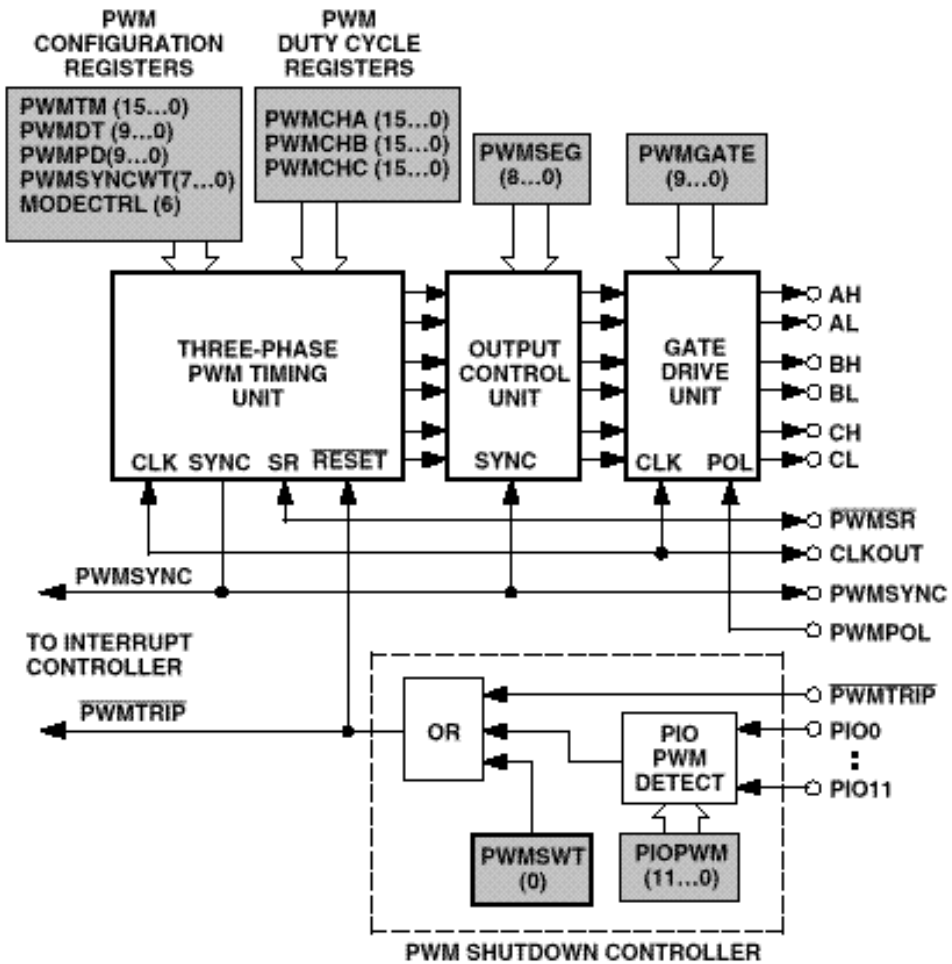
- PC incrementer
- PC stack
- instruction register
- interrupt controller

ADMC401: Analog to Digital Converter



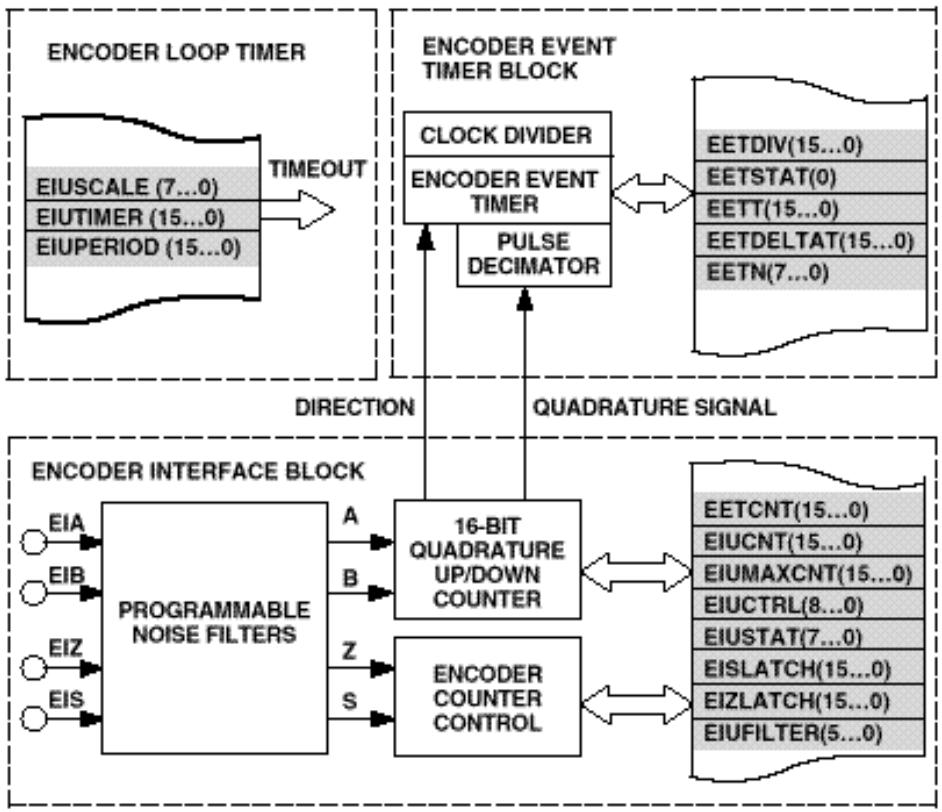
- 8 analog inputs.
- 12 bit resolution.
- conversion time: 2 μs (all channels thanks to four stage pipeline architecture).
- 4 V p-p input voltage range
- 2 channels can be simultaneously sampled.
- conversion can be synchronized to PWM or externally triggered.

ADMC401: PWM Generation Unit



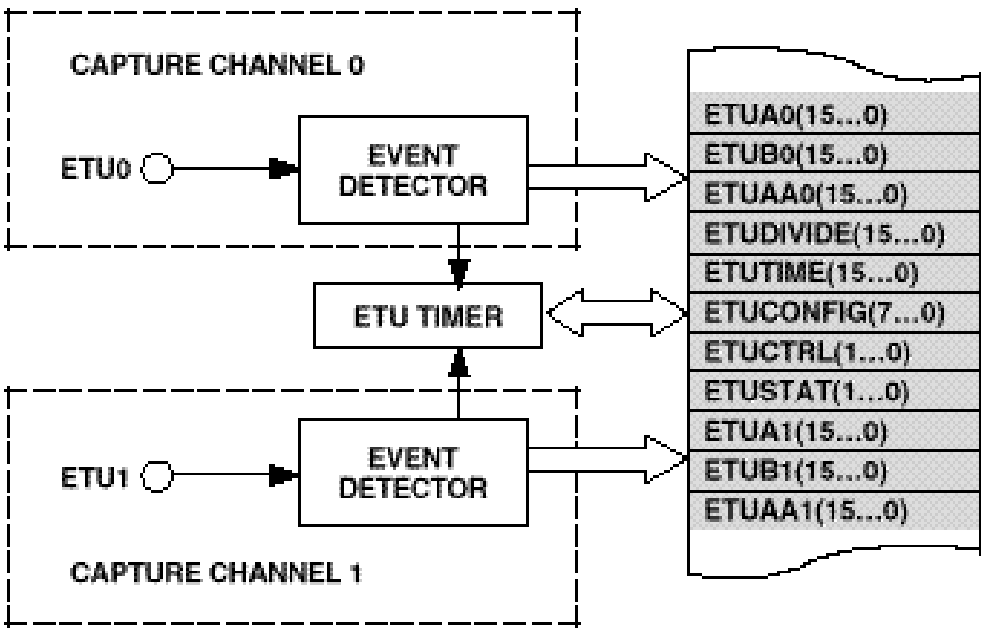
- 6 PWM outputs.
- 16 bit counter resolution.
- programmable output polarity.
- static or chopped output signals.
- programmable dead-time and minimum pulse width.
- single update and double update mode (for asymmetrical PWM patterns).
- switching frequency from 198 Hz to 102 kHz.

ADMC401: Incremental Encoder Interface



- **16 bit** up-down counter (frequency and direction of rotation detection).
- programmable **input noise filter** (to avoid spurious triggering).
- **two** additional **strobe inputs** (to latch the counter contents into registers).
- **16 bit loop timer** (position and speed loops set point generation).

ADMC401: Event Timer Unit



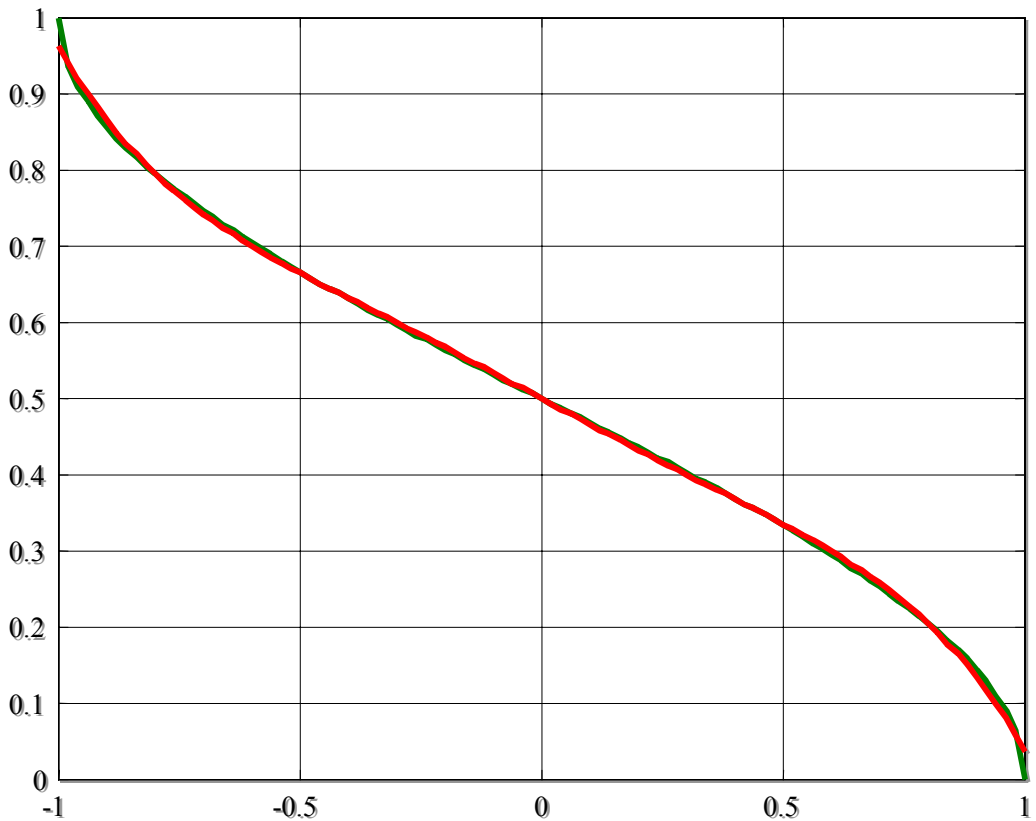
- 16 bit dedicated timer with programmable frequency.
- 2 independent channels.
- 2 programmable (rising edge or falling edge) events for each channel.
- single shot and free-running modes of operation.
- only capture function, no possibility of compare mode.

ADMC401: Other Ancillary Fuctions

- 2 auxiliary 8 bit PWM timers
- 16 bit watchdog timer
- Programmable digital I/O port (12 pin)
- 2 synchronous serial ports

ADMC401: Software examples (1)

Approximation of function y = arcos(x)



—— $y = \arccos(x)$

—— $y = \sum_i c_i x^i$

c_0	5.0004e-001	4000
c_1	-3.3887e-001	D49F
c_2	-1.3448e-005	0000
c_3	8.8821e-002	0B5E
c_4	-3.5333e-005	0000
c_5	-2.1278e-001	E4C3

ADMC401: Software examples (1)

```
.VAR/DM/RAM/SEG=USER_DM3    ACOS_COEFF[5];
.INIT   ACOS_COEFF: 0xD49F, 0x0000, 0x0B5E, 0x0000, 0xE4C3;

Arcos:
        AR=-AR;                     { invert sign x=-x }
;
        M0=1; L0=0;
        I0=^ACOS_COEFF;     { pointer to coefficient vector }
;
        MY1=AR;                     { writes AR to MY1. Now MY1 = x}
;
        MF=AR*MY1 (SS),     MX1=dm(I0,M0);    {MF = x^2}
        MR=MX1*MY1 (RND),   MX1=dm(I0,M0);    {MR = c1x}
```

parallel instructions

ADMC401: Software examples (1)

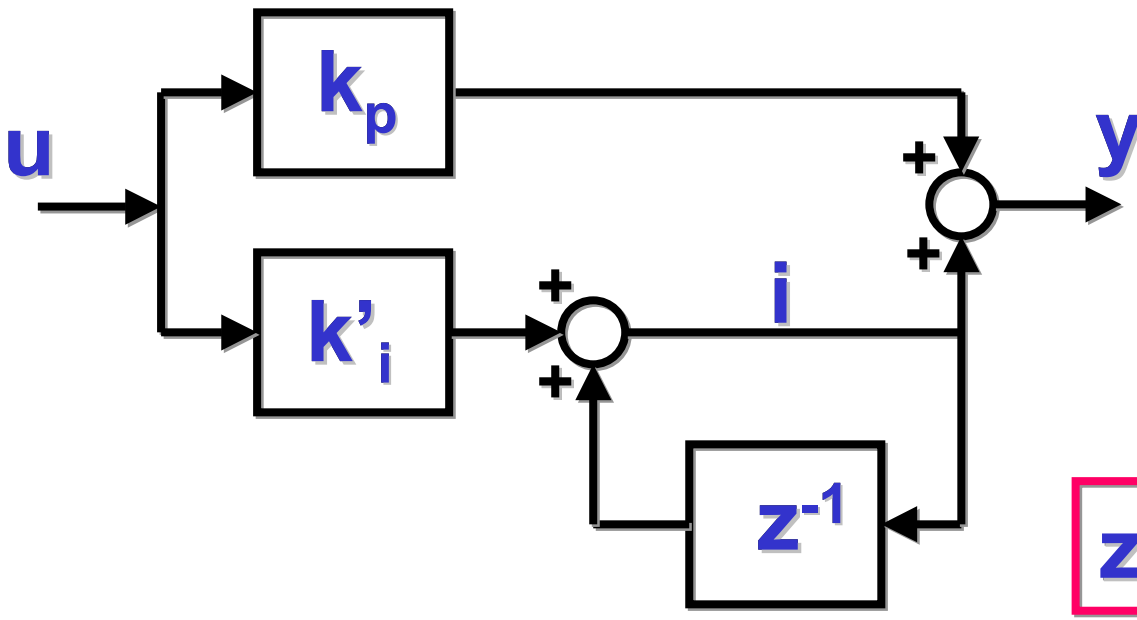

```
        CNTR=0x0003;                          { order -2 }
        DO appr UNTIL CE;                     { executes Σ }
        MR=MR+MX1*MF (RND);                   { MR=c1x+c2x2 }
appr:   MF=AR*MF (SS), MX1=dm(I0,M0);         { MF=MF*x }
        MR=MR+MX1*MF (RND);
        AY0=0x4000;                           { add c0 }
        AR=MR1+AY0;                           { AR = y }
```

Complete execution requires less than 1 μs (≈800 ns)

ADMC401: Software examples (2)

PI controller

Equivalent block diagram:



z^{-1} = unity delay

where $k'_I = k_i \cdot T$ (sampling period).

ADMC401: Software examples (2)

PI controller implementation:

$k_p = 16 \cdot 6000h$ $\quad$ $k'_i = (1/8) \cdot 200h$

```
AX0 = dm(Vout);
AY0 = dm(Vref);
AR = AX0 - AY0;
dm(u) = AR;
MX0 = AR;
MY0 = kp;
MR  = MX0 * MY0 (SS);
IF MV SAT MR;
SR = ASHIFT MR1 BY 4 (LO);
{
{
```

```
{ u(k) = Vout - Vref }
{ compute error u(k) }
{ store u(k) in memory }
{ MX0 = u(k) }
{ load kp gain }
{ MR = kp*u(k) }
{ saturate if necessary }
{ multiply proportional }
   part p by 16 to scale  }
   kp gain as required   }
```

ADMC401: Software examples (2)

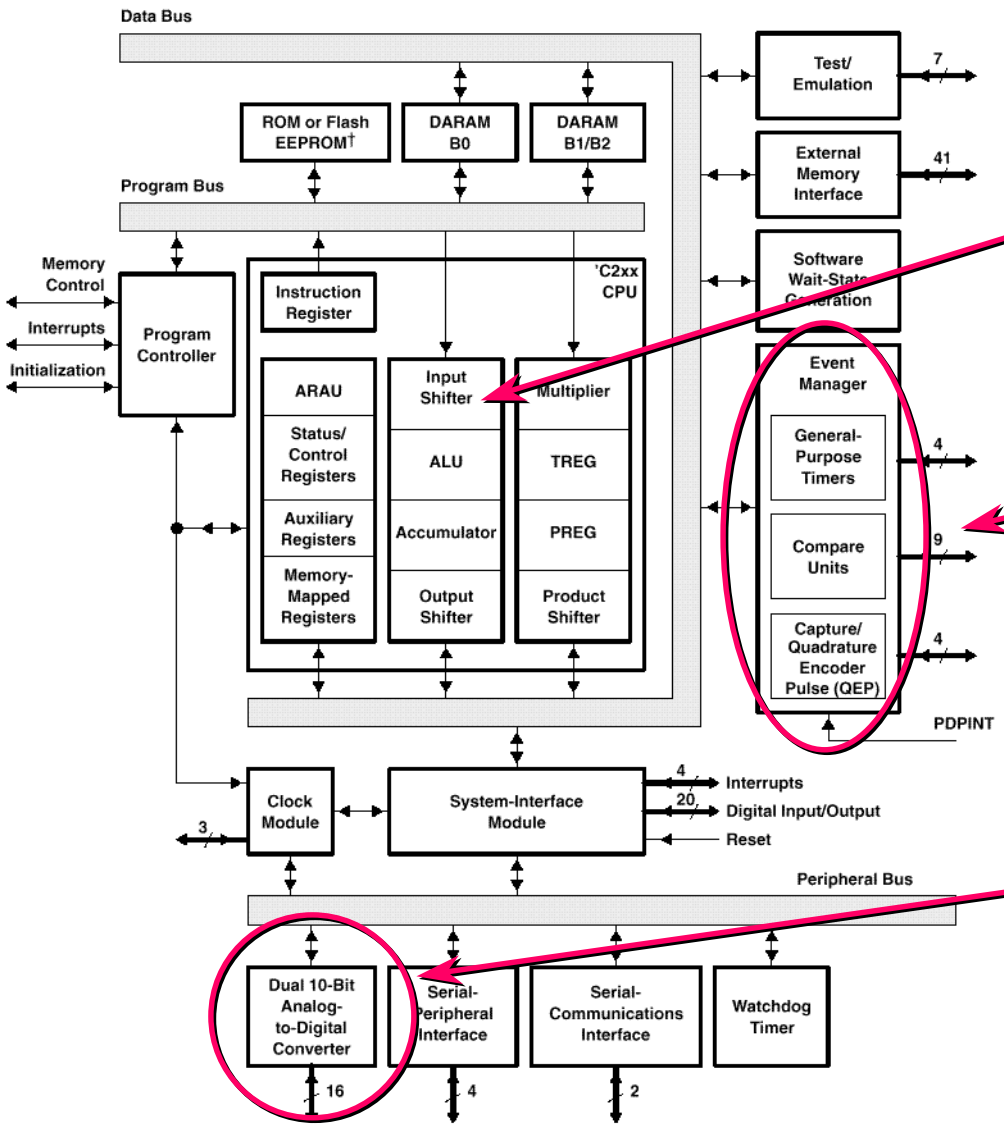
```
dm(p) = SR0;                        { store result in memory}
AF  = PASS SR0;                     { AF = p(k) }
AR = dm(u);                         { AR = u(k) }
SR = ASHIFT AR BY -2 (LO);          { divide input  variable u }
{                                     by 4 to scale ki gain   }
MY0 = ki;                           { load ki gain }
MR1 = dm(i);                        { load i(k-1) }
MR  = MR + SR0 * MY0 (SS);          { i(k) = i(k-1) + ki/4·u(k) }
IF MV SAT MR;                       { saturate if necessary  }
AR = MR1;
dm(i) = AR;                         { store integral part }
AR = AR + AF;                       { AR = p(k) + i(k)}
```

Complete execution requires ≈ 700 ns

Texas Instruments TMS320F240 [3]

- Fixed-point DSP core features:
  - ♦ 20 MIPS performance;
  - ♦ TMS320C25 source code compatible;
  - ♦ Single cycle instruction execution (50 ns @ 20 MHz CPU clock frequency);
  - ♦ 16 bit arithmetic and logic unit;
  - ♦ Single Cycle 16 bit X 16 bit signed product.
- Main built-in peripheral units:
  - ♦ 10 bit resolution, 16 channel ADC;
  - ♦ 12 channel 16 bit PWM generation unit;
  - ♦ 3 16 bit general purpose timers;
  - ♦ 4 independent capture circuits.

TMS320F240 Functional Block Diagram

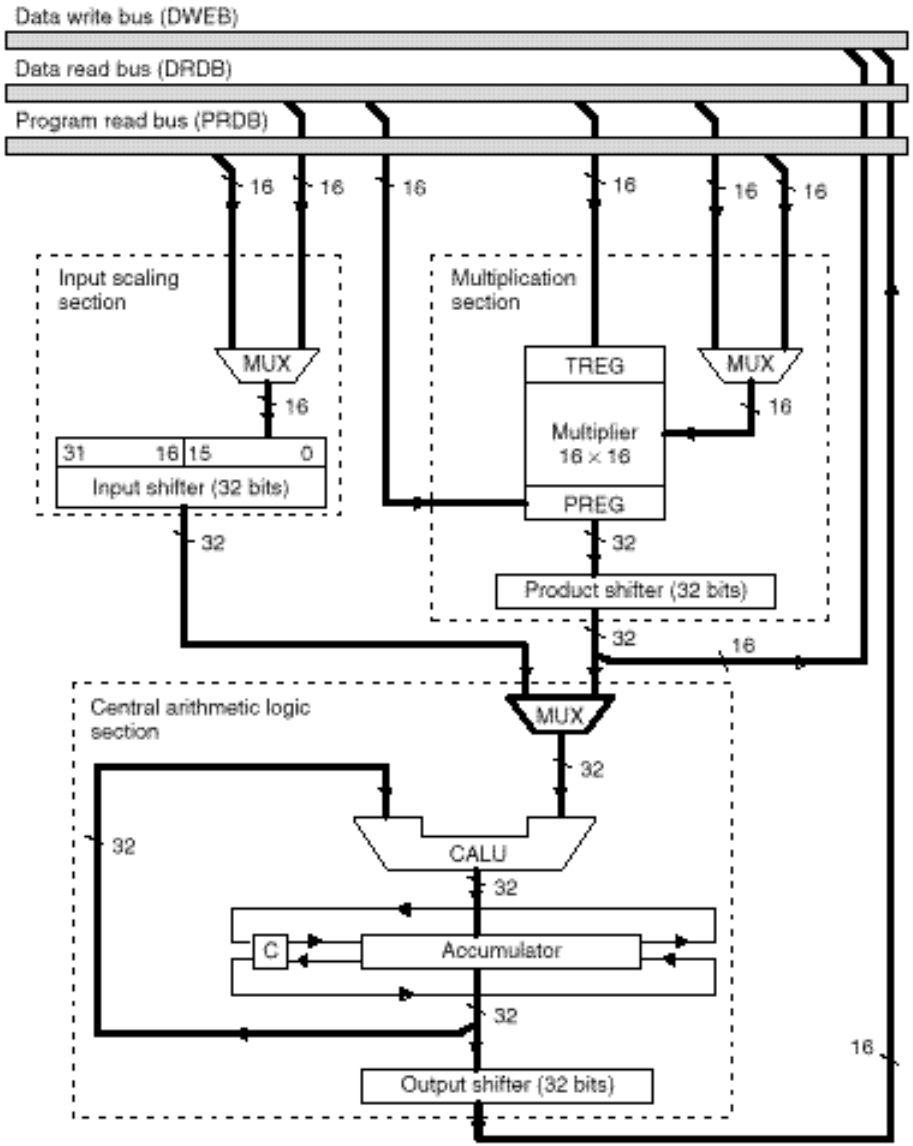


Central Arithmetic Logic Unit (CALU)

Event Manager: Timers Compare Units ...

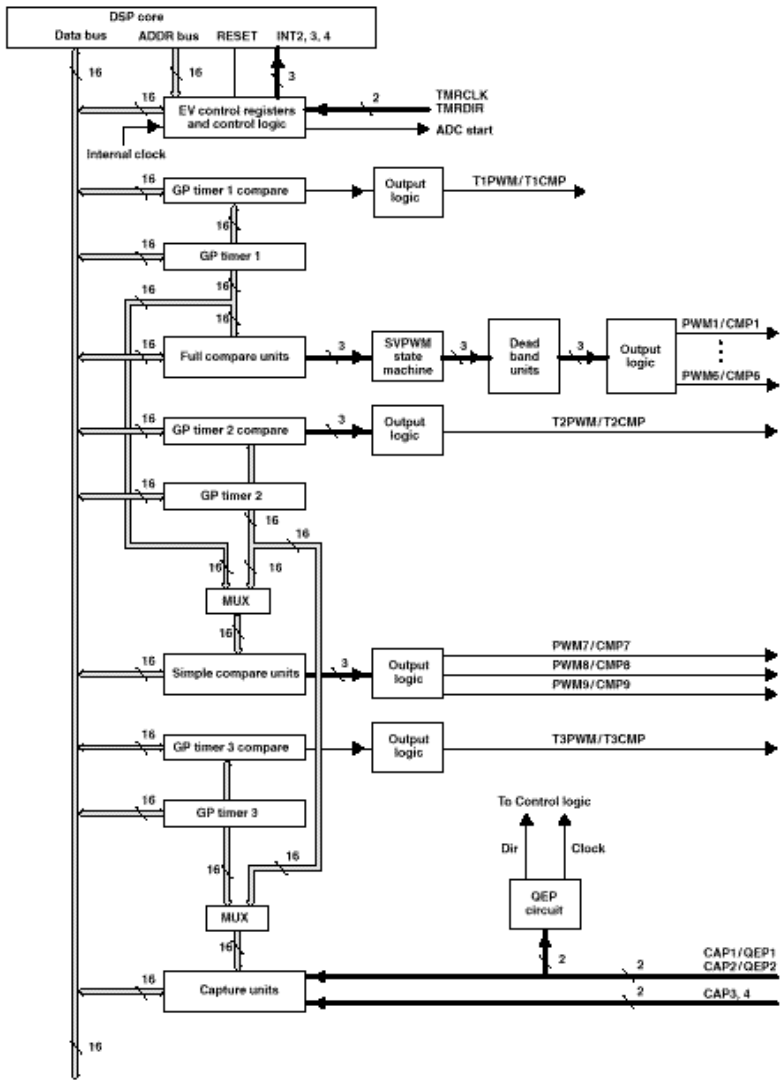
Dual 10 bit ADC

TMS320F240 Central Arithmetic and Logic Unit



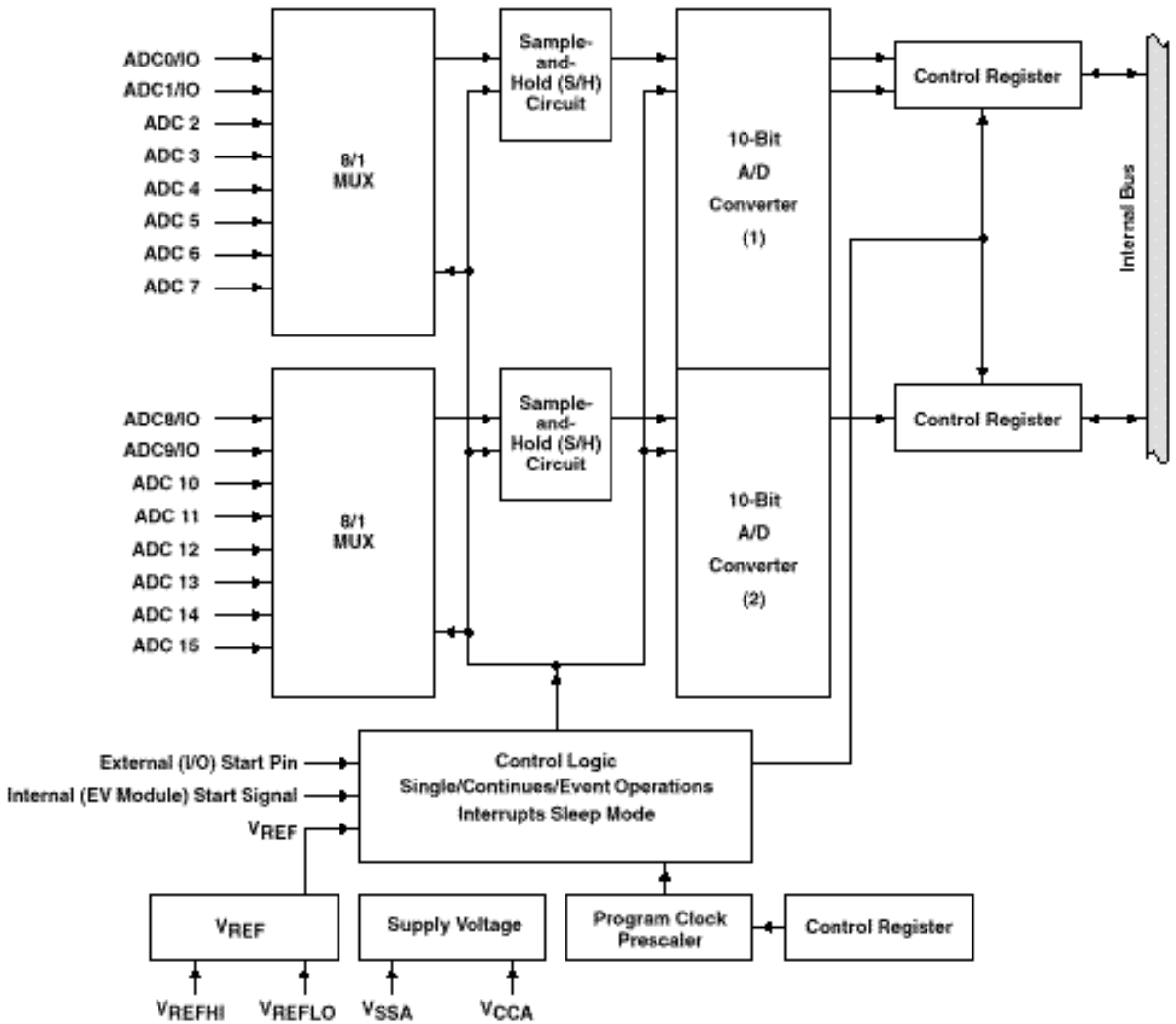
- Advanced Harvard architecture with multiple bus.
- 16 bit X 16 bit hardware multiplier with 32 bit accumulator (1 cycle execution).
- 16 bit input scaling shifter (used for data alignment before logic or arithmetic operations).

TMS320F240 Event Manager



- 3 general purpose timers for 16 bit up/down counts and compare functions (including additional PWM generation).
- 6 full compare ouputs for PWM applications with dead-time generators (0 to 102μs).
- 3 simple compare units.
- The 12 available PWM outputs have all 50 ns resolution and 16 bit dynamic range.
- 4 channel capture unit operating on GPT1 or GPT2, with 2 level FIFO.

TMS320F240 Analog to Digital Converter



- 2 10 bit ADC's with built-in S/H circuits.
- 16 input channels available, but only 2 can be sampled simultaneously.
- Minimum conversion time 6.1 μs.
- Single shot or continuous mode of operation.
- 2 level FIFO for result storing.

TMS320F240 Ancillary Peripheral Functions

- 4 pin serial peripheral interface
- watchdog timer
- Quadrature - encoder pulse circuit
- 28 programmable I/O pins

TMS320F240: Comparison with ADMC401

CPU:

- TMS has a slightly lower throughput (20 MIPS, 50 ns cycle @ 20 MHz vs 26 MIPS, 38.5 ns cycle @ 13 MHz).
- Similar pipelined architecture for single cycle execution of instructions.
- The TMS shifter has lower flexibility (only left shift is programmable).
- Both allow saturated arithmetic mode of operation.

TMS320F240: Comparison with ADMC401

CPU:

- Only two registers are associated to the multiplier in the TMS. The ADMC has four. Multiplication by a 13-bit constant is available as a single instruction in the TMS.
- TMS allows four different automatic shift strategies after multiplication, ADMC only fractional and integer mode.

TMS320F240: Comparison with ADMC401

Timers / Counters:

- Both have three timers. TMS has three GP timers with capture and compare functions. ADMC has also three timers, but two of them are dedicated to specific functions (PWM, Event Timing Unit).
- Both have 16 bit timer resolution and wide frequency programmability.
- The ADMC PWM unit provides double update mode of operation, while TMS doesn't.

TMS320F240: Comparison with ADMC401

Timers / Counters:

- TMS offers an embedded SVM hardware module to automatically implement a flat-top type of PWM modulation. This simplifies the software [4] to some extent (T_1 and T_2 have to be computed).

TMS320F240: Comparison with ADMC401

A/D converter:

- The TMS provides up to 16 analog inputs thanks to two 8 to 1 multiplexers. ADMC only provides 8 inputs.
- Both allow simultaneous sampling of two channels.
- The TMS has a minimum conversion time of 6.2 μs per channel. The ADMC converts all the 8 inputs in 2 μs.

TMS320F240: Comparison with ADMC401

A/D converter:

- The TMS has 10 bit resolution with 5V peak to peak dynamic range, the ADMC has 12 bit resolution with 4V peak to peak dynamic range.

TMS320F240: Software example

PI controller implementation:

voltage reference is #6DD0h

```
LDP #0          ; select memory page
CLRC SXM        ; disable sign extension
LACC Vout       ; load voltage sample
SFR             ; right shift (alignment)
SETC SXM        ; enable sign extension
SUB  #6DD0h     ; calculate error (Vout - Vref)
NEG             ; invert sign ACC = u = Vref - Vout
LDP #6          ; select memory page
SACL u          ; save voltage u(k) error in memory
```

TMS320F240: Software example

```
LT u(k)        ; load TREG with u(k)
MPY ki         ; compute  ki*u(k)
PAC            ; ACC = ki*u(k)
RPT #15        ; repeat following instruction 15 times
SFR            ; discard 16 LSB's
ADD  i         ; ACC = ki*u(k) + i(k-1)
SACL i         ; store lower ACC into variable i(k)
;
MPY kp         ; compute kp*u(k), u(k) is in TREG
PAC            ; ACC = kp*u(k)
SACH p         ; store higher ACC into variable p
```

TMS320F240: Software example

```
LACC i          ; Load ACC with variable i(k)
ADD p           ; ACC = i(k) + p(k)
RPT #4          ; shift to scale the gains
SFR             ; 4 bits right
LDP #0h         ; select memory page
SACL y          ; store higher ACC in y = PI output
```

Complete execution requires ≈ 1.5 μs

References

[1] P. Pirsch, “Architectures for Digital Signal Processing”, 1998, John Wiley and Sons (ISBN 0-471-97145-6).

[2] Analog Devices Winter 1999 Designers’ Reference Manual.

[3] Texas Instruments 1999 TMS320 DSP Solutions CD-ROM.

[4] Texas Instruments, Application Report SPRA524, ”Space-Vector PWM With TMS320C24x/F24x Using Hardware and Software Determined Switching Patterns”.