

11 Introduction to DSP Processors

11.1 INTRODUCTION TO PROGRAMMABLE DSPs

A digital signal processor (DSP) is a specialized microprocessor designed specifically for digital signal processing, generally in real time computing. They contain special architecture and instruction set so as to execute computation-intensive DSP algorithms more efficiently. Some advanced microprocessors may have performances close to that of P-DSPs. However, in terms of low power requirements, cost, real time I/O compatibility and availability of high-speed on-chip memories, the P-DSPs have an advantage over the advanced microprocessors or RISC processors. The programmable DSPs (P-DSPs) can be divided into two broad categories. They are (i) General purpose DSPs and (ii) Special purpose DSPs.

1. **General purpose DSPs:** These are basically high speed microprocessors with architecture and instruction sets optimized for DSP operations. They include fixed point processors such as Texas instruments TMS320C5X, TMS320C54X and Motorala DSP563X and floating point processors such as Texas instruments TMS320C4X, TMS320C67XX, and analog devices ADSP21XXX.
2. **Special purpose DSPs:** This type of processors consist of hardware (i) designed for specific DSP algorithms such as FET, (ii) designed for specific applications such as PCM and filtering. Examples for special purpose DSPs are MT93001, PDSP16515A and UPDSP16256.

The factors that influence the selection of DSP for a given application are architectural features, execution speed, and type of arithmetic and word length. Some of the areas of the applications of P-DSPs are: communication systems, audio signal processing, control and data acquisition, biometric information processing, image/video processing, etc.

www.Jntufastupdates.com

11.2 ADVANTAGES OF DSP PROCESSORS OVER CONVENTIONAL MICROPROCESSORS

1. The architecture of DSP processors supports fast processing of arrays.
2. The DSP processors require single clock cycle to execute instructions.
3. The DSP processors support parallel execution of instructions.
4. The DSP processors have separate data and program memories.
5. The DSP processors support simultaneous fetching of multiple operands.
6. The DSP processors have three separate computational units. Arithmetic logic unit, multiplier and accumulator, and shifter.
7. The DSP processors consist of powerful interrupt structure and timers.
8. The DSP processors have multiprocessing ability.
9. The DSP processors have on chip program memory and data memory.

11.3 MULTIPLIER AND MULTIPLIER ACCUMULATOR (MAC)

Array multiplication is one of the most common operations required in digital signal processing applications. An example is: convolution and correlation which require array multiplication operation. Implementation of the convolver with single multiplier/adder is shown in Figure 11.1. One of the important requirements of these array multipliers is that they have to process the signals in real time. The array multiplication should be completed before the next sample of the input signal arrives at the input to the array. This requires the multiplication as well as accumulation to be carried out using hardware elements. There are two approaches to solve this problem.

1. A dedicated MAC unit may be implemented in hardware, which integrates multiplier and accumulator in a single hardware unit. Example for this approach is Motorola DSP5600X.
2. Have separate multiplier and accumulator. Example for this approach, is TIDSP320C5X. In this approach, the output of the multiplier is stored into the product register and the content of the product register is added to accumulator register in the central ALU.

In both the approaches, the MAC operation can be completed in one clock cycle. The presence of H/W multipliers and or multiplier accumulator is one of the mandatory requirements of a P-DSP.

In Figure 11.1, the array corresponding to the present and the past $M - 1$ samples of the input is given by

$$x_n = \{x_n x_{n-1} x_{n-2} \ \ldots \ x_{n-M+3} x_{n-M+2} x_{n-M+1}\}$$

and the array corresponding to the impulse response of the sequence is given by

$$h = \{h_0 h_1 h_2 \ \ldots \ h_{M-3} h_{M-2} h_{M-1}\}$$

The output at the nth sampling instant, y_n is obtained by multiplying the array x_n with the array h. To obtain y_{n+1}, the input signal array x_{n+1} is multiplied with the array h. The vector

www.Jntufastupdates.com

x_{n+1} is obtained by shifting the array x_n towards right so that the $(n + 1)$th sample of the input data x_{n+1} becomes the first element and all the elements of x_n are shifted right by 1 position so that the ith element of x_n becomes $(i + 1)$th element of x_{n+i}. The content of the product register is added to the accumulator before the new product is stored. Further, the content of 'dma' is copied to the next location whose address is 'dma + 1'.

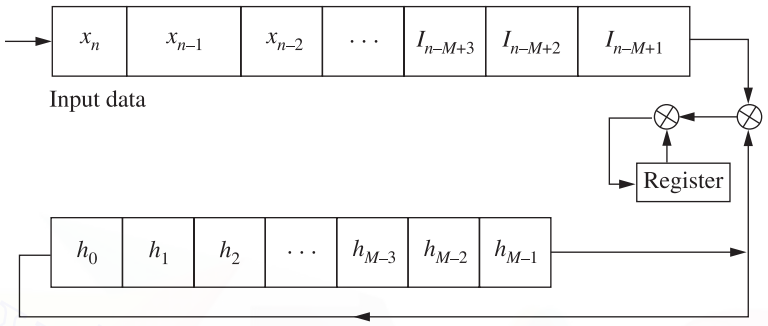


Figure 11.1 Implementation of convolver with single multiplier/adder.

11.4 MODIFIED BUS STRUCTURES AND MEMORY ACCESS SCHEMES IN P-DSPs

The MACD instruction, that is, the MAC operation with data move requires four memory accesses per instruction cycle. The four memory accesses/clock period required for the MACD instructions are as follows:

1. Fetch the MACD instruction from the program memory.
2. Fetch one of the operands from the program memory.
3. Fetch the second operand from the data memory.
4. Write the content of the data memory with address 'dma' into the location with the address 'dma + 1'.

The relatively static impulse response coefficients are stored in the program memory and the samples of the input data are stored in the data memory.

Figure 11.2 shows the Von Neumann architecture. The Von Neumann architecture is most widely used in majority of microprocessors. This architecture consists of 3 buses: The data bus, the address bus and the control bus. In this architecture, the CPU can be either reading an instruction or reading/writing data from/to memory. Both cannot occur at the same time since the instruction and data use the same signal path ways and memory. The execution of each instruction requires four clock cycles because there is a single address bus and there is a single data bus for accessing the program as well as data memory area.

The number of clock cycles required for the memory access can be reduced by using more than one bus for both address and data. Figure 11.3 shows the Harvard architecture. In this, there are two separate buses for the program and data memory. Hence the content of program memory and data memory can be accessed in parallel. The instruction code is fed from the program memory to the control unit and the operand is fed to the processing unit

www.Jntufastupdates.com

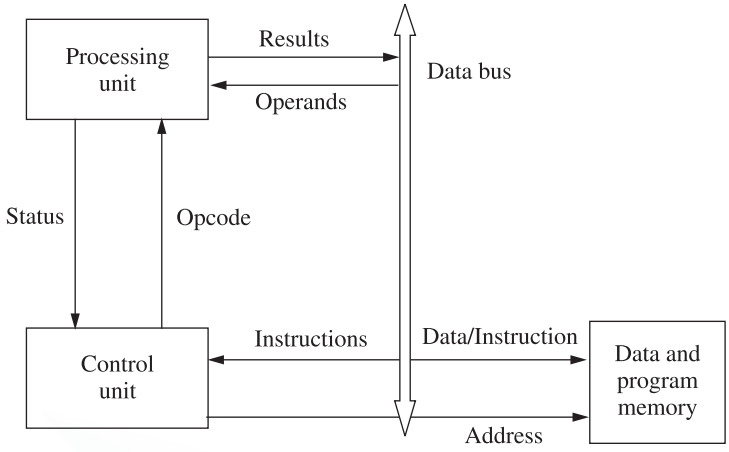


Figure 11.2 Von Neumann architecture.

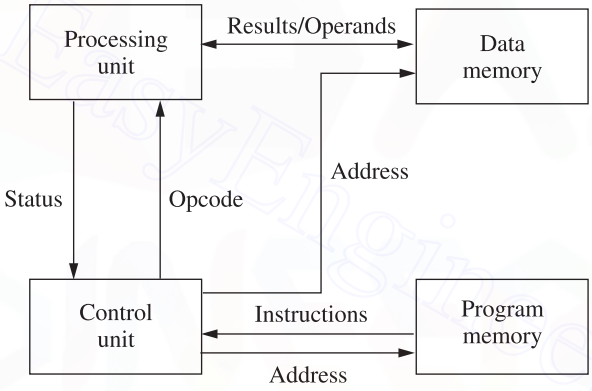


Figure 11.3 Harvard architecture.

from the data memory. Since it has two memories, it is not possible for the CPU to mistakenly write codes into the program memory and, therefore, compute the code while it is executing. However, it is less flexible. It needs two independent memory banks. These two resources are not interchangeable. The processing unit consisting of the registers and processing elements such as MAC units, multiplier, ALU, shifters, etc., are also referred to as data path.

Figure 11.4 shows the modified Harvard architecture. The P-DSPs normally follow this. One set of bus is used to access a memory that has both program and data and another set of bus is used to access data alone. In modified Harvard architecture, data can also be transferred from one memory to another memory. This architecture is used in several P-DSPs.

It may also have multiple bus system for program memory alone or for data memory alone. These multiple bus system increases complexity of CPU, but allow it to access several memory locations simultaneously, thereby increasing the data throughput between memory and CPU.

www.Jntufastupdates.com

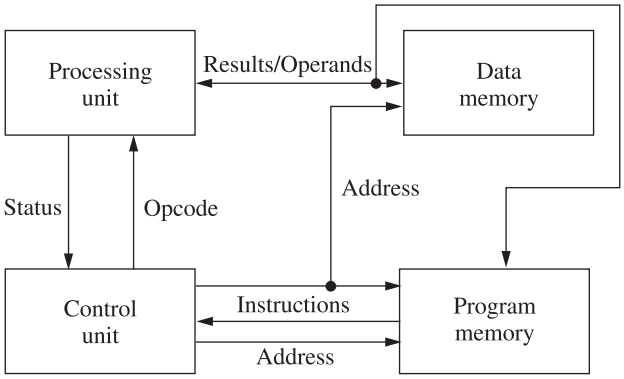


Figure 11.4 Modified Harvard architecture.

By using more number of buses, the number of memory accesses/clock cycle can be increased. Motorola DSP5600X, DSP96002, etc. have three separate buses and so have three memory accesses/clock cycle. TMS320C54X has four address buses and so has four memory accesses per clock cycle.

11.5 MULTIPLE ACCESS MEMORY AND MULTIPORTED MEMORY

The different techniques adopted for increasing the number of memory accesses/instruction cycle are: (i) Multiple access memory and (ii) Multiported memory.

Multiple access memory

The number of memory accesses/clock period can be increased by using a high speed memory that permits more than one memory access/clock period. The memory that permits more than one access/clock period is called multiple access memory. Two memory accesses per clock period can be achieved using the DRAM, the dual access RAM. By using the Harvard architecture, multiple access RAM may be connected to the processing unit of the P-DSP. Four memory accesses/clock period can also be achieved with dual access RAM, when it is connected to a programmable DSP with two independent address and data buses.

Multiported memory

The number of accesses/clock period can also be increased by using multiported memory. The dual port memory shown in Figure 11.5 has two independent data and address buses and hence two memory accesses can be achieved in one clock period. Multiported memories dispense with the need for storing the program and data in two different memory chips in order to permit simultaneous access to both program and data memory.

One of the major limitations of the dual ported memory is the increase in the cost compared to two single port memories of the same total capacity. This is due to the increased number of pins and larger chip area required for the dual port memory. Larger and more expensive package and a larger die size is required for larger number of I/O pins.

www.Jntufastupdates.com

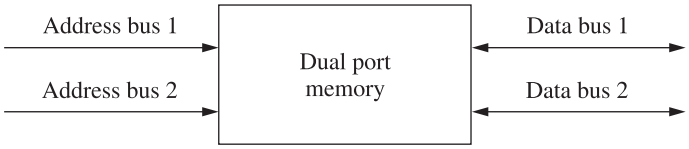


Figure 11.5 Block diagram of a dual ported memory.

Some P-DSPs combine the modified Harvard architecture with the dual ported memories. For example, the Motorola DSP56IXX processors have a single ported program memory and a dual ported data memory. Hence one program memory access and two data memory access can be achieved per clock period.

11.6 VLIW ARCHITECTURE

Very long instruction word (VLIW) architecture is another architecture used for P-DSPs (Example: in TMS320C6X). The VLIW processor consists of architecture that reads a relatively large group of instructions and executes them at the same time. These P-DSPs have a number of processing units (data paths). In other words, they have a number of ALUs, MAC units, shifters etc. The VLIW is accessed from memory and is used to specify the operands and operations to be performed by each of the data paths. The VLIW processing increases the number of instructions that are processed per cycle. Figure 11.6 shows the block diagram of the VLIW architecture. The multiple functional units share a common multiported register file for fetching the operands and storing the results.

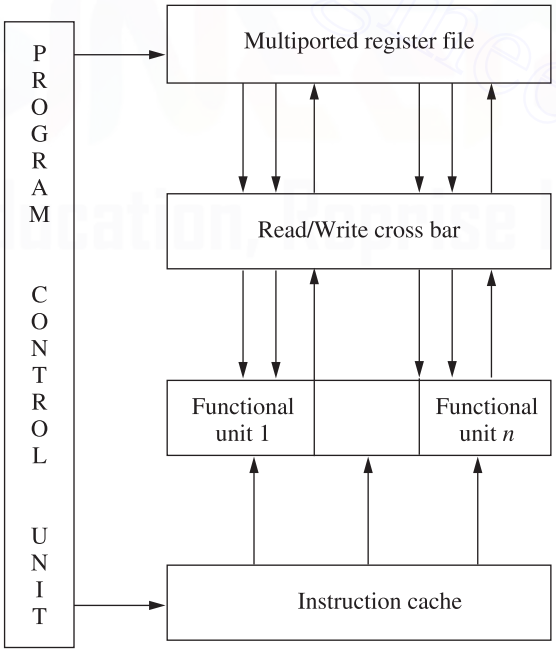


Figure 11.6 VLIW architecture.

www.Jntufastupdates.com

The read/write cross bar provides parallel random access by multiple functional units to the multiported register file. Execution of the operations in the functional units is carried out concurrently with the load/store operation of data between a RAM and the register file.

The performance gains that can be achieved with VLIW architecture depends on the degree of parallelism in the algorithm selected for a DSP application and the number of functional units. The throughput will be higher only if the algorithm involves execution of independent operations.

11.7 PIPELINING

Instruction pipelining is a mechanism used for increasing the efficiency of the advanced microprocessors as well as P-DSPs. Pipelining a processor means breaking down its instruction into a series of discrete pipeline stages which can be completed in sequence by specialized hardware.

An instruction cycle starting with the fetching of an instruction and ending with the execution of instruction including the time for storage of the results can be split into a number of microinstructions. Execution of each of the microinstructions is also referred to as one phase of an instruction. For example, an instruction cycle requiring four micro-instructions can be said to be in four phases as follows.

1. *Fetch phase:* In this phase, the instruction is fetched from program memory.
2. *Decode phase:* In this phase, the instruction is decoded.
3. *Memory read phase:* In this phase, the operand required for the execution of the instruction may be read from the data memory.
4. *Execution phase:* In this phase, the execution as well as storage of the results in either one of the register or memory is carried out.

In a modern processor, the above four steps get repeated over and over again till the program is finished executing. Each one of the above microinstructions may be carried out separately by four functional units. If we assume that each of the above phases take equal time for completion, then if there is no pipelining, each of the functional units is busy only for 25% of the time. The functional units can be kept busy almost all the time by using pipelining and processing a number of instructions simultaneously in the CPU. If one clock cycle of the processor corresponds to *T*, in a period of 12T, only three instructions can be executed in a machine without pipelining, whereas in the same period, nine instructions can be carried out with pipelining. Hence the throughput is increased by a factor of 3 in this case.

The performance and programming simplification is achieved by pipeline operation with the elimination of pipeline interlocks; the control of pipeline is simplified. The bottlenecks in the program fetch, multiply operations and data access are eliminated by increased pipelining.

The number of instructions that are processed simultaneously in the CPU, also referred to as depth of the instruction pipeline, differs in different families of DSPs.

www.Jntufastupdates.com

11.8 SPECIAL ADDRESSING MODES IN P-DSPs

In addition to the addressing modes such as direct, indirect and immediate supported by the conventional microprocessors, P-DSPs have special addressing modes that permit single word/instruction format and thereby speed up the execution by making effective use of the instruction pipelining. Further there are also special addressing modes such as cyclic addressing and bit reversed addressing that are specifically tailored for DSP applications. The special addressing modes in P-DSPs are as follows:

1. Short immediate addressing
2. Short direct addressing
3. Memory mapped addressing
4. Indirect addressing
5. Bit reversed addressing
6. Circular addressing

Short immediate addressing

In short immediate addressing mode, the operand is specified as a short constant that forms part of a single word instruction. The length of the short constant depends on the programmable DSP and the instruction type. In TMS320C5XDSPs, an 8 bit constant can be specified as one of the operands in the single word instruction like AND, OR, addition, subtraction, etc.

Short direct addressing

In short direct addressing mode, the lower order address of the operand is specified in the single word instruction. In TITMS320DSPs, the higher order 9 bits of the memory are stored in the data page pointer and only the lower 7 bits are specified as part of the instruction. Using short direct addressing in the Motorola DSP5600X processor, an instruction is specified with a 6 bit address.

Memory-mapped addressing

In this addressing mode, the CPU registers and the I/O registers are accessed as memory locations by storing them in either the starting page or the final page of the memory space. For example, in TMS320C5X, page 0 corresponds to the CPU registers and I/O registers.

In the case of Motorola DSP5600X, the last page of the memory space containing 64 locations is used as the memory map for the CPU and I/O registers.

Indirect addressing

This addressing mode has a number of options in P-DSPs. This mode permits an array of data to be efficiently processed, fetched and stored. The address of the operands can be stored in one of the registers called indirect address registers. In the case of TI processors, the indirect address registers are called auxiliary registers ARs. When the operands fetched by these registers are being executed, these registers can be updated. This is made possible

www.Jntufastupdates.com

by having an additional ALU in the CPU core specifically for the indirect address registers of ARs. The increment or decrement of ARs can be performed either in steps of 1 or in steps specified by the content of an offset register. In P-DSPs from Texas instruments, the offset register is known as INDEX register and in the case of analog devices the offset register is known as modifier register. The content of the indirect address registers may also be updated by a constant using bit reversed addressing mode.

Bit reversed addressing

The binary pattern corresponding to a particular decimal number is obtained by writing the natural binary equivalent of the number in the reverse order. Therefore, the least significant bit of the bit reversed number becomes the most significant bit of the natural binary number and vice versa. In this addressing mode, the address is incremented or decremented by the number represented in the bit reversed form.

Circular addressing mode

In real time processing of signals, the input signal is continuously stored in memory. The processed data is stored in another memory space continuously and may be written on to the output device. In this case, the input as well as output program will be simple. However, since the input as well as the output memory space is finite it would be exhausted after processing the input signal for some time, if the data is written into the memory by using linear addressing mode. This problem may be overcome by checking continuously whether the range of either the input or the output memory space is exceeded. In that case, the new data is to be stored starting from the beginning of the particular memory space. Checking this condition is an overhead that can be overcome using the circular addressing mode. In this mode, the memory can be organized as a circular buffer with the beginning memory address and the ending memory address corresponding to this buffer designed by the programmer. In the circular addressing mode, when the address pointer is incremented, the address will be checked with the ending memory address of the circular buffer. If it exceeds that, the address will be made equal to the beginning address of the circular buffer.

11.9 ON-CHIP PERIPHERALS

The P-DSPs have a number of on-chip peripherals that relieve the CPU from routine functions. Further they also help to reduce the chip count on the DSP system based around P-DSP. Some of the on-chip peripherals in P-DSPs and their functions are as follows.

11.9.1 On-chip Timer

Two of the common applications of timers are generation of periodic interrupts to the DSPs and generation of the sampling clocks for A/D converters. The timer can be programmed by the P-DSPs.

www.Jntufastupdates.com

www.EasyEngineering.net

11.9.2 Serial Port

The serial port enables the data communication between the P-DSP and an external peripheral such as A/D converter, D/A converter or an RS232C device. These ports normally have input and output buffers so that the P-DSP writes or reads from the serial port in parallel form and the serial port sends and receives data to the peripherals in serial form.

11.9.3 TDM Serial Port

The TDM serial port is a special serial port the P-DSPs have. This port permits a P-DSP to communicate with other devices or P-DSPs by using time division multiplexing (TDM).

11.9.4 Parallel Ports

Parallel ports enable communication between the P-DSP and other devices to be faster compared to the serial communication by using a number of lines in parallel. In addition they have additional lines, which are for strobing or for hand shaking purpose.

11.9.5 Bit I/O Ports

These are additional I/O ports the P-DSPs have that are single bit wide. These port bits may be individually set, reset or read. These bits are normally used for control purposes, but they can also be used for data transfer.

11.9.6 Host Ports

Host port is a special parallel port the P-DSPs have. This enables the P-DSPs to communicate with a microprocessor or a PC, which is called a host. In addition to data communication, the host can generate interrupts and also cause the P-DSP to load a program from ROM to the RAM on reset.

11.9.7 Common Ports

These are parallel ports that are used for interprocess communication between a number of identical P-DSPs in a multiprocessor system.

11.9.8 On-chip A/D and D/A Converters

Some of the P-DSPs targeted towards voice applications such as cellular telephones and tapeless answering machines have A/D and D/A converters inside the P-DSPs.

www.Jntufastupdates.com

www.EasyEngineering.net

11.10 P-DSPs WITH RISC AND CISC

P-DSPs may be implemented using either the RISC processor or the CISC processors. The relative advantages of each of these processors is given below.

11.10.1 Advantages of Restricted Instruction Set Computer (RISC) Processors

1. In RISC processor, the control unit uses only around 20% of the chip area because of the reduced number of instructions. Hence the remaining area can be used for incorporating other features.
2. The delayed branch and call instructions are used to improve the speed of the RISC processors.
3. The execution time required for all the instructions of RISC processor is same because all the instructions are of uniform length.
4. The RISC processors have smaller and simpler control units, which have fewer gates.
5. The speed of the RISC processor is high because of smaller control unit and smaller propagation delays.
6. Since a simplified instruction set allows for a pipelined superscalar design, RISC processors often achieve two to four times the performance of CISC processor using comparable semiconductor technology and the same clock rates.
7. Because the instruction set of a RISC processor is simpler, it uses much less chip space than a CISC processor. Extra functional units such as memory management units or floating point arithmetic units can be placed on the same chip.
8. The throughput of the processor can be increased by applying pipelining and parallel processing.
9. Since RISC processors can be designed more quickly, they can take advantage of new technological developments sooner than corresponding CISC design.
10. High level language (HLL) support; The programs can be written in C and C^{++}. It relieves the programmer from learning the instruction set of a P-DSP which in turn increases the throughput of the programmer.

11.10.2 Advantages of Complex Instruction Set Computer (CISC) Processors

1. The CISC processors have a very rich instruction set that even support high level language constructs similar to "if condition true then do", "for" and "while".
2. The CISC processors have instructions specifically required for DSP applications such as MACD, FIRS, etc.
3. The assembly language program of a CISC processor is very short and easy to follow.
4. For RISC architecture compliers are essential. So this becomes costly. For CISC compilers are not required. Hence they are of low cost.
5. New CISC processors are designed to be upward compatible with the older processors. This makes the learning curve steeper.
6. Microprogramming is easier to implement and much less expensive than hardwiring a control unit.

www.Jntufastupdates.com

7. Since microprogram instruction sets can be written to match the constructs of high-level languages, the complier need not be very complicated.
8. As each instruction is more capable, fewer instructions could be used to implement a given task. This made more efficient use of the relatively slow main memory.

11.11 ARCHITECTURE OF TMS320C50

The TMS320C5X generation of the Texas instruments TMS320C50 digital signal processor is fabricated with CMOS IC technology. It is a fixed point, 16 bit processor running at 40 MHz. The single instruction execution time is 50 nsec. Its architectural design is based on the combination of advanced Harvard architecture, on-chip peripherals and on-chip memory.

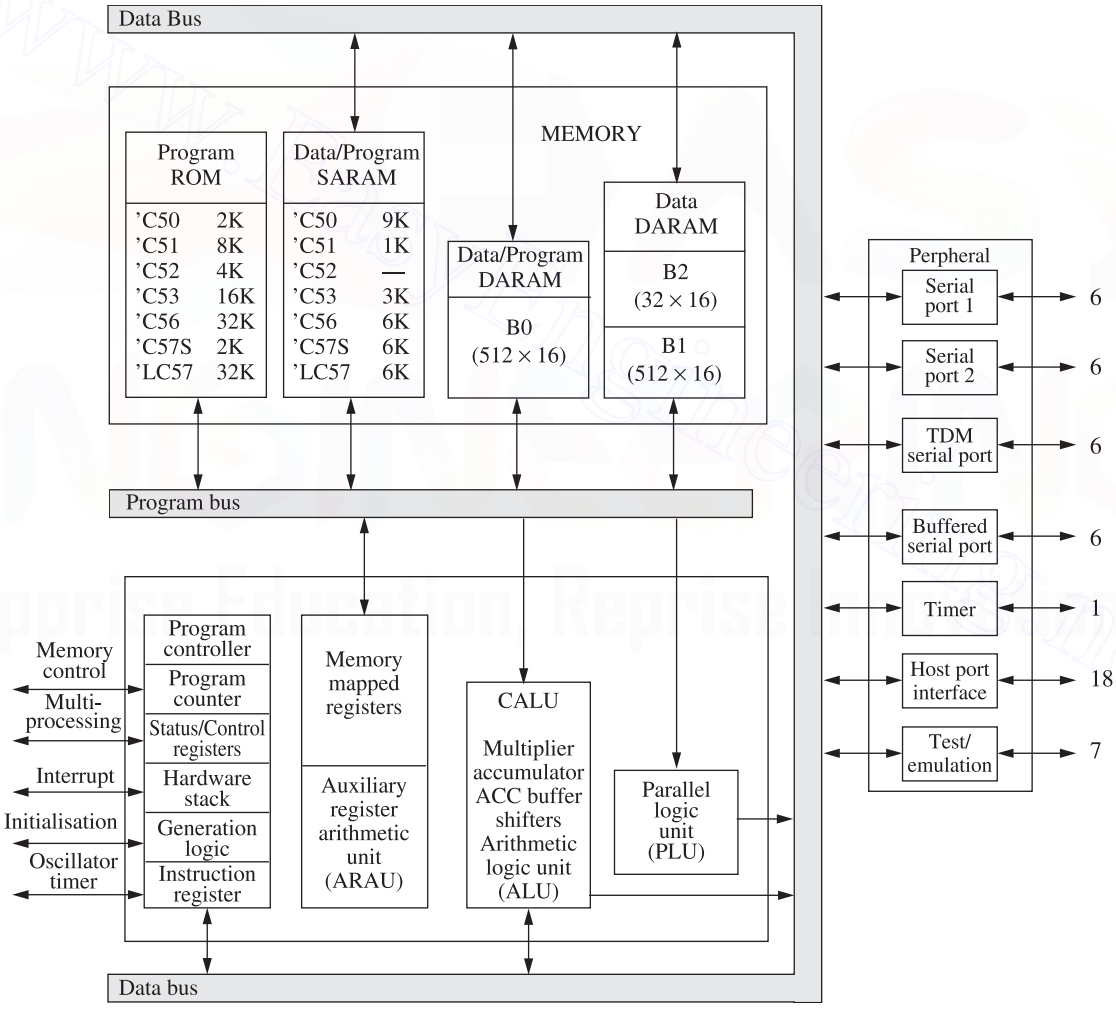


Figure 11.7 Architecture of TMS320C50.

Moreover, the TMS320C50 has a highly specialized instruction set. These features enable the operational flexibility and the device speed, which together with the cost effectiveness make the signal processor as the suitable device for a wide range of applications.

The TMS320C50 has a programmable memory map which can vary for each application. On-chip memory includes 10K words of the RAM and 2K words of the ROM. All C5X DSPs have the same CPU structure. However, they have different on-chip memory configuration and on-chip peripherals.

The functional block diagram of TMS320CX is shown in Figure 11.7. It can be divided into four sub blocks. They are: (1) Bus structure, (2) Central processing unit, (3) On-chip memory and (4) On-chip peripherals.

11.12 BUS STRUCTURE

Separate program and data buses in the advance Harvard architecture of C5X maximize the processing power and provide a high degree of parallelism. Many DSP applications are accomplished using single cycle multiply/accumulate instruction with a data move option. The C5X included the control mechanism to manage interrupts, repeated operations and function calling. The 'C5X' architecture has four buses:

1. Program bus (PB)
2. Program read bus (PRB)
3. Data read bus (DB)
4. Data read address bus (DRB)

The program bus carries the instruction code and immediate operands from program memory to the CPU. The program address bus provides address to program memory space for both read and write.

The data read bus interconnects various elements of the CPU to data memory space. The data read address bus provides the address to access the data memory space.

11.13 CENTRAL PROCESSING UNIT

The CPU consists of the following elements:

1. Central arithmetic logic unit (CALU)
2. Parallel logic unit (PLU)
3. Auxiliary register arithmetic unit (ARAU)
4. Memory mapped registers
5. Program controller

11.13.1 Central Arithmetic Logic Unit (CALU)

The CPU uses the CALU to perform 2's complement arithmetic. It consists of the following:

1. Parallel multiplier (16 × 16 bit)
2. Accumulator (32 bit)

www.Jntufastupdates.com

3. Accumulator buffer (ACCB) (32 bit)
4. Product register (PREG)
5. Shifters
6. Arithmetic logic unit (ALU)

All 32 bit signed/unsigned multiplication operations can be performed in parallel multiplier within one machine cycle. All multiply instructions except the MPYU (multiply unsigned) instruction perform a signed multiply operation in the multiplier. One of the operands to the multiplier is from the 16 bit temporary register O (TREGO) and the second input is from the program bus or data bus. The product register (PREG) holds the product.

The 32 bit ALU along with 16 bit accumulator carries out arithmetic and logic operations executing most of them in one machine cycle. Here the accumulator provides one of the inputs to the ALU, whereas the product register, accumulation buffer, or scaling shifter output provides the second input. The results of operations performed in ALU are stored in accumulator.

The scaling shifter has a 16 bit input connected to the data bus and a 32 bit output connected to the ALU. The scaling shifters produce a left shift of 0 to 16 bits on the input data. A 5 bit register TREGI specifies the number of bits by which the scaling shifter should shift or the shift count is specified by a constant embedded in the instruction word.

11.13.2 Parallel Logic Unit (PLU)

The parallel logic unit (PLU) is another logic unit that executes logic operations on data without affecting the contents of the accumulator. The multiplier bit in a status/control register or any memory location can be directly set, clear, test or toggled by the PLU. After executing the logical operation the PLU writes the result of the operation to the same memory location from which the first operand was fetched.

11.13.3 Auxiliary Register Arithmetic Unit (ARAU)

The C5X consists of a register file containing eight auxiliary registers (ARO-AR7) each of 16 bit length, a 3 bit auxiliary register pointer (ARP) and an unsigned 16 bit ALU. The auxiliary register file is connected to the auxiliary register arithmetic unit.

Auxiliary registers

The eight 16 bit auxiliary registers (ARO-AR7) can be accessed by the CALU and modified by the ARAU or the PLU. The primary function of ARs is to provide 16 bit address for indirect addressing to data space or for temporary data storage. The ARs can also be used as general purpose registers or counters. The contents of the ARs can be stored in the data memory or used as inputs to the CALU.

Index registers (INDX)

The index register (INDX) is a 16 bit register used by the ARAU as a step value to modify the address in the ARs during indirect addressing. The INDX can be added to or subtracted

www.Jntufastupdates.com

from the current AR or any AR update cycle. The INDX can be used to increment or decrement the address in steps larger than 1.

Auxiliary register compare register (ARCR)

The ARCR is a 16 bit register used for address bound comparison. It limits blocks of data and supports logical comparisons between the current AR and ARCR in conjunction with the CMPR instruction. The result of this comparison is placed in the TC bit of STI.

Block move address register (BMAR)

The BMAR is a 16 bit register that holds the address value of a source destination space of the block move. It can also hold the address value of an operand in program memory for a multiply accumulate operation.

11.13.4 Memory Mapped Registers

The C5X has 96 registers mapped into page 0 of the data memory space. This memory mapped register space contains various controls and status registers including those for CPU, serial port, timer and software wait generators. Additionally the first sixteen I/O port locations are mapped into this data memory space, allowing them to be accessed either as data memory using single word instruction or as I/O locations with two word instruction.

Instruction registers (IREG)

The 16 bit instruction register (IREG) holds the op code of the instruction being executed.

Interrupt register (IMR, IFR)

The 16 bit interrupt mask register (IMR) individually masks specific interrupts at required time. The 16 bit interrupt flag register indicates the current status of the interrupts.

Status registers

The two 16 bit status registers contain status and control bits for the CPU.

11.13.5 Program Controller

The program controller contains logic circuitry that decodes the operational instructions, manages the CPU pipeline, stores the status of CPU operation and decodes the conditional operations. It consists of the following elements:

(i) Program counter (PC)
(ii) Status and control registers
(iii) Hardware stack

www.Jntufastupdates.com

(iv) Program memory addresses generation
(v) Instruction registers

(i) *Program counter (PC)*

It is a 16 bit counter which contains the address of internal or external program memory used to fetch instructions. The PC addresses program memory either on chip or off chip, via the program address bus. Through the PAB an instruction is loaded into the instruction register (IREG). Then the PC is ready to start the next instruction fetch cycle.

(ii) *Status and control registers*

The C5X has four status and control registers. They are circular buffer control register, process mode status register, status registers ST0 and ST1.

(iii) *Hardware stack*

The stack is 16 bit wide and 8 levels deep and is accessible via the push and pop instructions. The stack is used during interrupts and subroutine to save and restore the PC contents.

(iv) *Program memory addresses generation*

It contains the code for application and holds table information and immediate operands. The program memory is accessed only by the program address bus.

(v) *Instruction registers*

The 16 bit instruction registers (IREG) hold the op code of the instruction being executed.

11.14 SOME FLAGS IN THE STATUS REGISTERS

The bit assignment details for flags in status registers ST0 and ST1 are shown in Figure 11.8.

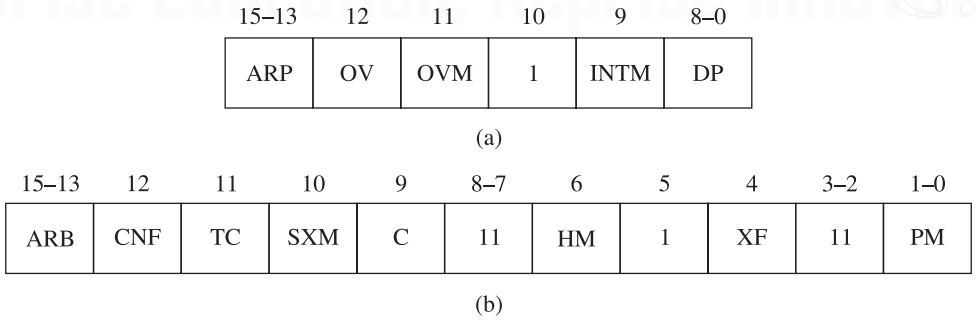


Figure 11.8 (a) Status Register 0 (ST0) bit assignment, (b) Status Register 1 (ST1) bit assignment.

Significance of the various bits of ST0 and ST1 are as follows:

ARP (Auxiliary register pointer): These bits select the AR to be used in indirect addressing.

OV (Overflow) flag bit: This bit indicates an arithmetic operation overflow in the ALU.

OVM (Overflow mode): This bit enables/disables the accumulator overflow saturation mode in the ALU.

INTM (Interrupt mode): This bit globally masks or enables all interrupts. The INTM bit has no effect on the non-maskable $\overline{\text{RS}}$ and $\overline{\text{NMI}}$ interrupts.

DP (Data memory page register): These bits specify the address of the current data memory page.

ARB (Auxiliary register buffer): This 3 bit field holds the previous value contained in ARP in ST0.

CNF (On-chip RAM configuration control bit): This 1 bit field enables the on-chip dual access RAM block 0 (DARAM B0) to be addressable in data memory space or programmable memory space.

TC (Test/Control flag bit): This 1 bit flag stores the results of the ALU or PLU test bit operations. The status of the TC bit determines if the conditional branch, call and return instructions are to be executed.

SXM (Sign extension mode bit): This 1 bit field enables/disables sign extension of an arithmetic operation.

C (Carry bit): This 1 bit field indicates whether the CPU stops or continues execution when acknowledging an active $\overline{\text{HOLD}}$ signal.

XF (Pin status bit): This 1 bit field determines the level of the external flag (XF) output pin.

PM (Product shift mode bits): This 2 bit field determines the product shifter mode and shift value for PREG output into the ALU.

11.15 ON-CHIP MEMORY

The C5X structure has a total memory address range of 224K words × 16 bits. The memory space is divided into four memory segments.

64K word program memory space: It contains the instruction to be executed.

64K word local data memory space: It stores data used by the instruction.

64K word input/output ports: It interfaces to external memory mapped peripherals.

32K word global data memory space: It can share data with other peripherals within the system.

The large on-chip memory of C5X includes:

1. Program read only memory
2. Data/Program single access RAM (SARAM)
3. Data/Program dual access RAM (DARAM)

www.Jntufastupdates.com

11.16 ON-CHIP PERIPHERALS

All C5X DSPs have the same CPU structure; however they have different on-chip peripherals connected to their CPUs. A TMS320C50 digital signal processor contains the following on-chip peripherals.

1. Clock generator
2. Hardware timer
3. Software programmable wait stage generators
4. General purpose I/O pins
5. Parallel I/O ports
6. Serial port interface
7. Buffered serial port
8. TDM serial port
9. Host port interface
10. User-maskable interrupts

11.16.1 Clock Generator

The clock generator consists of an internal oscillator and a phase locked loop (PLL) circuit. The clock generator can be driven internally by a crystal oscillator or driven externally by a clock source. A clock source with a frequency lower than that of the CPU can be used because the PLL circuit can generate an internal CPU clock by multiplying the clock source by a specific factor.

11.16.2 Hardware Timer

The programmable hardware timer with 4 bit prescaler clocks at a rate that is between 1/2 and 1/32 of the machine cycle rate, depending upon the timer's divide down ratio. It acts as a down-counter producing interrupts to CPU at regular intervals. The timer can be stopped, restarted, reset or disabled by specific status bits. Three registers namely the timer counter register (TIM), the timer period register (PRD) and timer control register (TCR) control and operate the timer. The timer counter register gives the current count of the timer. The timer period register defines the period for the timer. The timer control register controls the operations of the timer.

11.16.3 Software-programmable Wait-state Generators

Software-programmable wait-state generators can be interfaced without any external hardware with slower off chip memory and I/O devices. This feature consists of multiple wait-state generating circuits. Each circuit is user-programmable to operate in different wait states for off chip memory accesses.

www.Jntufastupdates.com

11.16.4 General Purpose I/O Pins

There are two general purpose I/O pins namely branch control input (BIO) pin and external flag output (XF) pin that are controlled by software. The pin BIO keeps track of the status of external devices and executes conditional branches depending on the requirement. The pin XF communicates with external devices through software. The instruction SETCXF sets this pin to high and CLRCXF resets to 0.

11.16.5 Parallel I/O Ports

There is a total of 64K parallel I/O ports of which 16 are memory mapped in data memory space. All the I/O ports can be addressed through the instructions IN, OUT or through data memory read and write instructions. There is a signal IS which differentiates accessing of memory mapped IO space to that of program and data space. Interfacing with external I/O devices can be done through the I/O ports with minimal off-chip address decoding circuits.

11.16.6 Serial Port Interface

Three different kinds of serial ports are available: a general purpose serial port, a time division multiplexed (TDM) serial port and a buffered serial port (BSP). Each C5X contains atleast one general purpose high-speed synchronous, full duplexed serial port interface. The serial port is capable of operating at upto one fourth the machine cycle rate.

The serial port control (SPC) register, the data receive register (DRR), the data transmit receiver (DXR), the data transmit shift register (XSR) and the data receive shift register (RSR) control and operate the serial port interface.

The serial port control register contains the mode control and status bits of the serial port. The data receive register holds the incoming serial data. The data transmit shift register controls the shifting of the data from the data transmit register to the output pin.

The data receive shift register controls the storing of the data from the input pin to the data receive register.

11.16.7 Buffered Serial Port

The buffered serial port (BSP) is available on the C56 and C57 devices. It is a full duplexed, double-buffered serial port and an auto buffering unit. The BSP provides flexibility on the data stream length. The ABU supports high-speed data transfer and supports internal latencies. Five BSP registers control and operate the BSP.

11.16.8 TDM Serial Port

This is the third type of serial port that is utilized by the devices C50, C51, C53. It is a full duplexed serial port that can be configured by software either for synchronous operation or TDM operation. The TDM serial port is commonly used in multiprocessor applications.

www.Jntufastupdates.com

11.16.9 Host Port Interface

The host port is available on the C57S and LC57. It is an 8 bit parallel I/O port that provides an interface to a host processor. Information is exchanged between the DSP and the host processor through on-chip memory that is accessible to both the host processor and the C57.

11.17 USER MASKABLE INTERRUPTS

Four external interrupt lines ($\overline{\text{INT1}}$, $\overline{\text{INT2}}$, $\overline{\text{INT3}}$, $\overline{\text{INT4}}$) and five internal interrupts, a timer interrupt and four serial port interrupts are user maskable.

When an interrupt service routine (ISR) is executed, the contents of the program counter are saved on an 8 level hardware stack and the contents of 11 specific CPU registers are saved in one deep stack. When a return from interrupt instruction is executed, the CPU contents registers are restored.

11.17.1 Various Interrupt Types Supported by TMS320C5X Processor

The TMS320C5X devices have four external, maskable user interrupts ($\overline{\text{INT4}}$ – $\overline{\text{INT1}}$) that external devices can use to interrupt the processor; there is one external nonmaskable interrupt ($\overline{\text{NMI}}$). The internal interrupts are generated by the serial port (RINT and XINT),

TABLE 11.1 Interrupt locations and priorities

Interrupt type	*Location*		*Priority*	*Function*
	Dec	*Hex*		
$\overline{\text{RS}}$	0	0	1 (Highest)	External reset signal
$\overline{\text{INT1}}$	2	2	3	External interrupt line-1
$\overline{\text{INT2}}$	4	4	4	External interrupt line-2
$\overline{\text{INT3}}$	6	6	5	External interrupt line-3
TINT	8	8	6	Internal timer interrupt
RINT	10	A	7	Serial port receive interrupt
XINT	12	C	8	Serial port transmit interrupt
TRNT	14	E	9	TDM port receive interrupt
TXNT	16	10	10	TDM port transmit interrupt
$\overline{\text{INT4}}$	18	12	11	External user interrupt line-4
—	20–33	14–21	N/A	Reserved
TRAP	34	22	N/A	TRAP instruction vector
$\overline{\text{NMI}}$	36	24	2	Non-maskable interrupt
—	38–39	26–27	N/A	Reserved
—	40–63	28–3F	N/A	Software interrupts

www.Jntufastupdates.com

the timer (TINT), the TDM port (TRNT and TXNT), and the software interrupt instructions (TRAP, NMI, and INTR). Interrupt priorities are set so that reset ($\overline{RS}$) has the highest priority and $\overline{INT4}$ has the lowest priority. The $\overline{NMI}$ has the second highest priority.

Vector-relative locations and priorities for all external and internal interrupts are shown in Table 11.1. No priority is set for the TRAP instruction, but it is included because it has its own vector location. Each interrupt address has been spaced apart by two locations so that branch instructions can be accommodated in those locations.

11.18 ON-CHIP PERIPHERALS AVAILABLE IN TMS320C3X PROCESSOR

The TMS320C3X generation is the first of Texas Instruments 32 bit floating-point DSP. The C3X devices provide an easy-to-use, high performance architecture and can be used in a wide variety of areas including automotive applications, digital audio, industrial automation and control, data communication, and office equipment that include multifunction peripherals, copiers and laser printers.

The C3X processors peripherals include serial ports, timers and on-chip DMA (Direct Memory Access) controllers. The timers and DMA controllers are discussed below.

Timers

The C3X timers are general-purpose 32 bit timers/event counters. Figure 11.9 shows the block diagram of the timer. The timer operates in two signaling modes, internal or external clocking. With internal clock we can use the timer to signal external devices such as A/D converter, D/A converters etc. When external clock is applied to the timer it can count the external events and interrupts the CPU after a specified number of events. Each timer has an I/O pin that can be configured as an input, output or general purpose I/O pin. The three memory mapped registers, global control register, period register and counter register are used by the timer. These registers can be accessed using the memory map address values.

The 32 bit counter present in the timer increments for the rising edge or falling edge of the input clock. The input clock can be half of the internal clock of C3X or it can be external clock signal on TCLKX pin. The counter register holds the value of the counter and its present value is compared with the content of the period register. When the values are equal, the counter is zeroed. This causes the internal interrupt. The pulse generator present generates two types of external clock signals, either pulse or clock.

DMA controller

The DMA controller is an on-chip peripheral present in C3X devices. It can be used to read or write the 32 bit operands in any location in the memory map. The 'C30' and 'C31' devices have only one DMA controller, where as 'C32' has two DMA controllers. The DMA controller consists of address generators, source and destination registers and transfer counter. Figure 11.10 shows the block diagram of DMA controller. The DMA controller has its own address bus and data bus and this minimize the conflict with CPU. The DMA and

www.Jntufastupdates.com

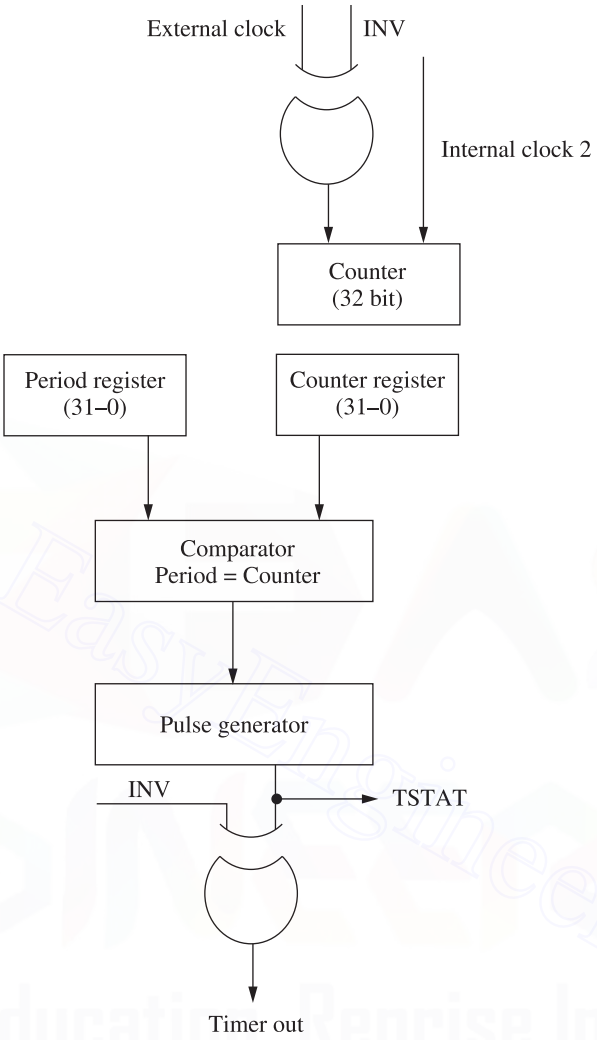


Figure 11.9 Time block diagram.

CPU controller buses can function independently, but when they access the same on chip or the external memory location the priority is provided. As far as C30 and C31 devices are concerned, the highest priority is for the CPU access, but in C32 devices the user can configure the priorities.

www.Jntufastupdates.com

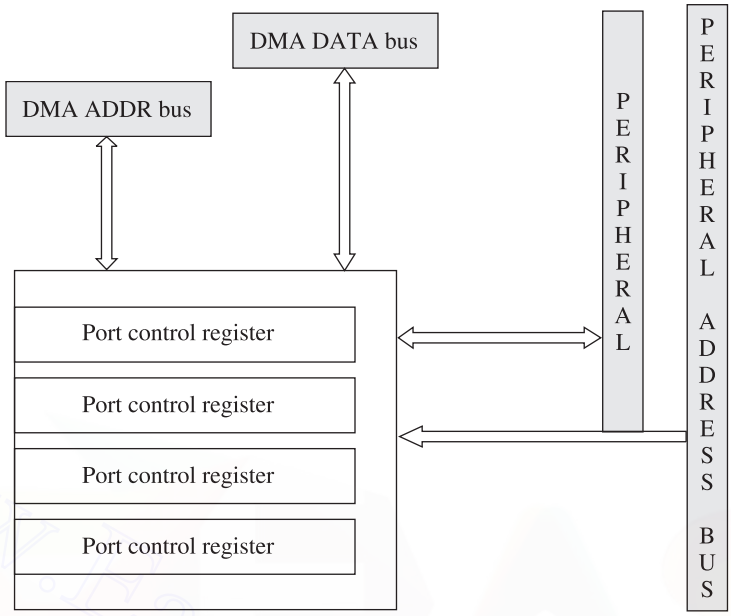


Figure 11.10 Block diagram of DMA controller.

SHORT QUESTIONS WITH ANSWERS

1. Name three leading manufacturers of P-DSPs.
Ans. Texas instruments (TI), Analog Devices, and Motorola are the three leading manufacturers of P-DSPs.

2. Why P-DSPs are preferred over advanced microprocessors and the RISC processors?
Ans. Eventhough an advanced microprocessor or RISC processor may use some of the techniques adopted in P-DSPs or may even have instructions that are specifically required for DSP applications and also may have performances close to that of a P-DSP for certain operations, P-DSPs are preferred over advanced microprocessors and the RISC processors because in terms of low power requirement, cost, real-time I/O compatibility and availability of high speed on-chip memories , the P-DSPs have an advantage over them.

3. What are the two categories of DSPs?
Ans. The DSPs are divided into two categories: General purpose DSPs and Special purpose DSPs.

4. What are general purpose DSPs? Name one.
Ans. General purpose DSPs are basically high speed microprocessors with architecture and instruction sets optimized for DSP operations. They include fixed point processors as well as floating point processors. Examples are TMS320C5X, TMS320C4X respectively.

www.Jntufastupdates.com

5. What are the factors that influence the selection of DSP for a given application?
Ans. The factors that influence the selection of DSP for a given application are: architectural features, execution speed and type of arithmetic and word length.

6. Why is any operation that involves an off-chip memory is slow compared to that using the on-chip memory in P-DSPs?
Ans. The P-DSPs use multiple buses only for connecting the on-chip memory to the control unit and data path but for accessing off-chip memory only a single bus is used for accessing both the program memory and the data memory. Hence any operation that involves an off-chip memory is slow compared to that using the on-chip memory in P-DSPs.

7. What is an instruction cycle?
Ans. An instruction cycle is the time that elapses since an instruction is fetched till the particular instruction completes execution including the time taken for writing the result into a register or memory.

8. How can the number of memory access per clock cycle be increased?
Ans. The number of memory access per clock period can be increased by using high speed memory or multiport memory.

9. Name few architectures used for P-DSPs.
Ans. Some architectures used for P-DSPs are: (i) Von Neumann architecture, (ii) Harvard architecture, (iii) Modified Harvard architecture, and (iv) VLIW architecture.

10. Name one P-DSP which uses VLIW architecture.
Ans. One P-DSP which uses VLIW architecture is TMS320C6X.

11. What is data path?
Ans. The processing unit consisting of the registers and processing elements such as MAC units, multiplier, ALU, shifter, etc., is referred to as data path.

12. Which architecture does the P-DSPs follow?
Ans. The P-DSPs follow the modified Harvard architecture.

13. Mention any approach for increasing the efficiency of P-DSPs.
Ans. One approach for increasing the efficiency of P-DSPs is instruction pipelining.

14. What do you mean by depth of the instruction pipeline?
Ans. The number of instructions that are processed simultaneously in the CPU are referred to as depth of the instruction pipeline.

15. Which addressing modes are specifically tailored for DSP applications?
Ans. Cyclic addressing and bit reversed addressing modes are specifically tailored for DSP applications.

16. What are the applications of on-chip timer?
Ans. Two of the common applications of on-chip timer are generation of periodic interrupts to the P-DSPs and generation of the sampling clocks for the A/D converters. The timers can generate a single pulse or a periodic train of pulses. They can also generate a single square wave or a periodic square wave.

www.Jntufastupdates.com

17. What is the use of TDM serial port?
Ans. The TDM serial port permits a P-DSP to communicate with other devices or P-DSPs by using time division multiplexing.

18. What is the use of single bit I/O ports?
Ans. Single bit ports are normally used for control purposes but they can also be used for data transfer. Some of the bits are also used for conditional branching or calls.

19. What do you mean by host port?
Ans. A host port is a special parallel port in P-DSPs that enable them to communicate with a microprocessor or PC.

20. What is the function of the host port?
Ans. In addition to data communication, the host port can generate interrupts and also cause the P-DSP to load a program from ROM to the RAM on reset.

21. What do you mean by comm ports?
Ans. Communication ports are parallel ports that are used for interprocess communication between a number of identical P-DSPs in a multiprocessor system.

22. In RISC processors what percentage of chip area may be used for the control unit and in CISC processors what percentage?
Ans. In RISC processors 20% of the chip area may be used for the control unit and in CISC processors 30 to 40% may be used.

23. How many buses does C5X architecture have? Name them.
Ans. The C5X architecture has four buses and they are: Program bus (PB), Program address bus (PAB), Data read bus (DB), Data read address bus (DAB).

24. What is the function of program bus (PB)?
Ans. The function of program bus is it carries the instruction code and immediate operands from program memory space to the CPU.

25. What is the function of program address bus (PAB)?
Ans. The function of program address bus is it provides addresses to program memory space for both reads and writes.

26. What is the function of data read bus (DB)?
Ans. The function of data read bus is it interconnects various elements of the CPU to data memory space.

27. What is the function of data read address bus?
Ans. The function of data read address bus is it provides the address to access the data memory space.

28. What does CALU consist of?
Ans. The central arithmetic logic unit (CALU) consists of the following elements: Parallel multiplier, arithmetic logic unit (ALU), accumulator (ACC), accumulator buffer (ACCB), product register (PREG), left barrel shifter and right barrel shifter.

www.Jntufastupdates.com

29. What does ARAU consist of?
Ans. The auxiliary register ALU (ARAU) consists of auxiliary register, auxiliary register pointer and an unsigned ALU.

30. What is the use of INDX?
Ans. The index register INDX is used by the ARAU as a step value to modify the address in the ARs during indirect addressing.

31. What is the use of ARCR?
Ans. The auxiliary register compare register ARCR is used for address boundary comparision.

32. What is the use of BMAR?
Ans. The block move address register holds an address value to be used with block moves and multiply/accumulate operations.

33. What is the function of PLU?
Ans. The PLU performs Boolean operations or the bit manipulations required of high-speed controllers. It can set, clear, test or toggle bits in a status register control register, or any data memory location. It allows logic operations to be performed on data memory values directly without affecting the control of the ACC and PREG.

34. What is the use of program controller?
Ans. The program controller contains logic circuitry that decodes the instructions, manages the CPU pipeline, stores the status of CPU operations and decodes the conditional operations.

35. What does program controller consist of?
Ans. The program controller consists of the following elements: program counter (PC), status registers, hardware stack, address generation logic, instruction register and interrupt flag register and interrupt mask register.

REVIEW QUESTIONS

1. What are the advantages of DSP processors over conventional processors?

2. Explain the implementation of convolver with single multiplier/adder.

3. Explain the modified bus structures and memory access schemes in digital signal processors.

4. With the help of a block diagram explain the dual port memory.

5. Explain the different techniques adopted for increasing the number of memory accesses/instruction cycle.

6. Explain the VLIW architecture with its block diagram.

7. Give the advantages and disadvantages of VLIW architecture.

8. Explain how a higher throughput is obtained using VLIW architecture. Give an example of DSP that has VLIW architecture.

www.Jntufastupdates.com