Sreenivasa Institute of Technology and Management Studies [SITAMS]
[Autonomous]
Chittoor, Andhra Pradesh

Department of Electronics and Communication Engineering

23ECE361T – Digital Signal Processing

UNIT V – ARCHITECTURES FOR PROGRAMMABLE DSP DEVICES

Architecture of TMS320C5X: Introduction, Bus Structure, Central Arithmetic Logic Unit, Auxiliary Register ALU, Index Register, Block Move Address Register, Parallel Logic Unit, Memory mapped registers, program controller, some flags in the status registers, On- chip memory, On-chip peripherals.

1.TMS320C5X Architecture Details: Categories of Texas Instruments DSP's

What is Word?

A word is basically a fixed-sized group of bits that are handled as a unit by the instruction set and/or hardware of the processor.

• The majority of the registers in a processor are usually word-sized, and the largest piece of data that can be transferred to and from the working memory in a single operation is a word in many (but not all) architectures.

TMS320 Family OverviewThe TMS320 family consists of two types of

single-chip DSPs: 16-bit fixed-point and floating-point processors.

DSP's Categories	Characteristic	Applications
TMS320 C1s, C2x, C24x	Low Cost, Fixed-Point, 16-Bit Word Length	Motor control, disk head positioning, control
TMS320 C5x, C54x, C55x	Power Efficient, Fixed-Point, 16 Bit Words	Wireless phones, modems
TMS320 C62x (16-bit fixed-point) C3x, C4x, C64x, C67x (32-bit)	High Performance DSP's (32-bit floating-point and Fixed-Point)	Communications infrastructure, xDSL, imaging, sound, video

Table 1. Characteristics of the 'C5x Processors

TMS320 DEVICES	ON-CHIP MEMORY (16-BIT WORDS)				I/O PORTS		POWER SUPPLY (V)	CYCLE TIME (ns)	PACKAGE TYPE QFP‡
	DARAM		SARAM	ROM					
	DATA	DATA + PROG	DATA + PROG	PROG	SERIAL	PARALLEL†			
TMS320C50	544	512	9K	2K§	2	64K	5	50/35/25	132 pin
TMS320LC50	544	512	9K	2K§	2	64K	3.3	50/40/25	132 pin
TMS320C51	544	512	1K	8K§	2	64K	5	50/35/25/20	100/132 pin
TMS320LC51	544	512	1K	8K§	2	64K	3.3	50/40/25	100/132 pin
TMS320C52	544	512	–	4K§	1¶	64K	5	50/35/25/20	100 pin
TMS320LC52	544	512	–	4K§	1¶	64K	3.3	50/40/25	100 pin
TMS320C53	544	512	3K	16K§	2	64K	5	50/35/25	132 pin
TMS320LC53	544	512	3K	16K§	2	64K	3.3	50/40/25	132 pin
TMS320C53S	544	512	3K	16K§	2¶	64K	5	50/35/25	100 pin
TMS320LC53S	544	512	3K	16K§	2¶	64K	3.3	50/40/25	100 pin
TMS320LC56	544	512	6K	32K	2#	64K	3.3	35/25	100 pin
TMS320LC57	544	512	6K	32K	2#	64K + HPI‖	3.3	35/25	128 pin
TMS320C57S	544	512	6K	2K§	2#	64K + HPI‖	5	50/35/25	144 pin
TMS320LC57S	544	512	6K	2K§	2#	64K + HPI‖	3.3	50/35	144 pin

† Sixteen of the 64K parallel I/O ports are memory mapped.
‡ QFP = Quad flatpack
§ ROM boot loader available
¶ TDM serial port not available
# Includes auto-buffered serial port (BSP) but TDM serial port not available
‖ HPI = Host port interface

These DSPs possess two unique qualities: the operational flexibility of high-speed controllers and the numerical capability of array processors (vector processors).

Combining these two qualities, the TMS320 processors are inexpensive alternatives to custom-fabricated VLSI and multichip bit-slice processors.

The following characteristics make this family the ideal choice for a wide range of processing applications:

1. Very flexible instruction set.
2. Inherent operational flexibility.
3. High-speed performance.
4. Innovative, parallel architectural design.
5. Cost-effectiveness.

TMS320C5x Overview

- The ’C5x generation consists of the ’C50, ’C51, ’C52, ’C53, ’C53S, ’C56, ’C57, and ’C57S DSPs, which are fabricated using CMOS integrated-circuit technology.

- The operational flexibility and speed of the ’C5x result from combining an advanced Harvard architecture (which has separate buses for program memory and data memory), a CPU with application-specific hardware logic, on-chip peripherals, on-chip memory, and a highly specialized instruction set.

- The ’C5x is designed to execute up to 50 million instructions per second (MIPS).

Advantages of ’C5x Devices

1. Enhanced TMS320 architectural design for increased performance and versatility.

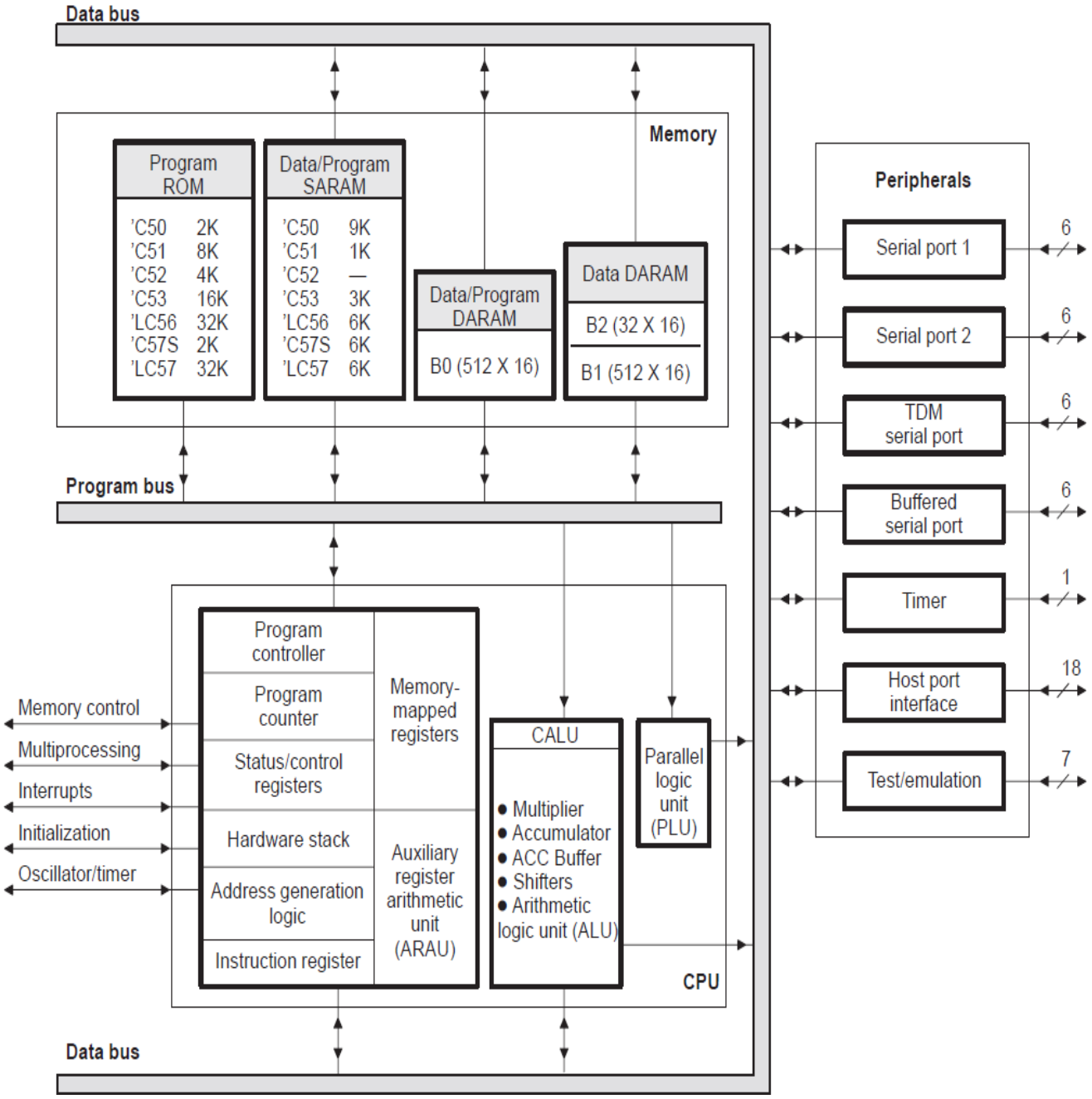
2. Modular architectural design for fast development of spin-off devices. (A spin-off product is a new product that uses the brand name of another product that already has a well-established image.)

3. Advanced integrated-circuit processing technology for increased performance and low power consumption.

4. Source-code compatibility with ’C1x, ’C2x, and ’C2xx DSPs for fast and easy performance upgrades.

5. Enhanced instruction set for faster algorithms and optimized high-level language operation.

Figure 2–1. 'C5x Functional Block Diagram



2.Bus Structure

- Separate program and data buses allow simultaneous access to program instructions and data, providing a high degree of parallelism.

- For example, while data is being multiplied by the multiplier, a previous product can be loaded into, added to, or subtracted from the accumulator. At the same time, a new address can be generated through the program bus.

- Such parallelism supports a powerful set of arithmetic, logic, and bit-manipulation operations that can be performed in a single machine cycle.

- In addition, the ’C5x includes control mechanisms to manage interrupts, repeated operations, and function calls.

Four Major Buses of the ’C5x Architecture

The ’C5x architecture is built around four major buses:

1. Program Bus (PB)
2. Program Address Bus (PAB)
3. Data Read Bus (DB)
4. Data Read Address Bus (DAB)

- PAB: Provides addresses for program memory space for both read and write operations.
- PB: Carries instruction codes and immediate operands from program memory to the CPU.
- DB: Interconnects various CPU elements with data memory. It carries data from data memory to the Central Arithmetic Logic Unit (CALU) and Auxiliary Register Arithmetic Unit (ARAU).
- DAB: Provides the data address used by the CPU.

In Simple Words

- PAB — Provides program memory addresses to the CPU.
- PB — Transfers instructions and operands from program memory to the CPU.

- DAB — Provides data memory addresses to the CPU.
- DB — Transfers data between data memory and the CPU.

The program and data buses can work together to transfer data from on-chip data memory and internal or external program memory to the multiplier for single-cycle multiply-accumulate operations.

Central Processing Unit (CPU)

The ’C5x 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 (MMR)
5. Program Controller

- The ’C5x CPU maintains source-code compatibility with the ’C1x and ’C2x generations while achieving high performance and greater versatility.
- Improvements include a 32-bit accumulator buffer (ACCB), which temporarily stores the contents of the accumulator (ACC), additional scaling capabilities, and new instructions.
- Data management is improved through block-move instructions, which move blocks of data, and memory-mapped-register instructions.

3.Central Arithmetic Logic Unit (CALU)

The CPU uses the CALU to perform two’s-complement arithmetic.

The CALU consists of the following elements:

1. 16-bit × 16-bit multiplier
2. 32-bit Arithmetic Logic Unit (ALU)
3. 32-bit accumulator (ACC)
4. 32-bit accumulator buffer (ACCB)
5. Additional shifters at the outputs of the accumulator and product register (PREG)

PREG (Product Register): A register that holds the output from the multiplier. The high and low words of the PREG can be accessed individually.

Parallel Logic Unit (PLU)

- The CPU includes an independent PLU, which operates separately from, but in parallel with, the ALU.
- The PLU performs Boolean operations and bit manipulations required for high-speed control operations.
- The PLU can set, clear, test, or toggle bits in a status register, control register, or any data-memory location.
- The results of a PLU operation are written back to the original data-memory location without affecting the contents of the ACC or PREG.

4.Auxiliary Register Arithmetic Unit (ARAU)

The CPU includes an unsigned 16-bit arithmetic unit that calculates indirect addresses using inputs from the auxiliary registers (ARs), index register (INDX), and auxiliary register compare register (ARCR).

Auxiliary Registers (AR0–AR7)

- The auxiliary register file consists of eight 16-bit registers.
- These registers are used for temporary data storage and indirect addressing of data memory.
- The address of the required memory location is placed in the selected auxiliary register for indirect data-memory addressing.
- The registers are referenced using a 3-bit Auxiliary Register Pointer (ARP), which selects AR0 through AR7 using values from 0 to 7.
- The contents of these registers and the ARP can also be stored in data memory or used as inputs to the CALU.

5.Index Register (INDX)

The index register is a memory-mapped register that specifies increment sizes greater than 1 for indirect-address updates.

Auxiliary Register

Compare Register (ARCR)

- The 16-bit ARCR is used for address-boundary comparison.
- The CMPR instruction compares the ARCR with the selected auxiliary register and places the comparison result in the TC bit of status register ST1.
- TC = 0 if the comparison result is false; otherwise, TC = 1.

Functions of the ARAU

- The auxiliary registers and ARP can be loaded from data memory, the ACC, the product register, or an immediate operand specified in an instruction.
- The auxiliary register file (AR0–AR7) is connected to the ARAU.
- The ARAU can automatically increment or decrement the current auxiliary register while addressing data memory. The increment or decrement can be ±1 or the value stored in INDX.
- Therefore, data access does not require the CALU for address manipulation, allowing the CALU to perform other operations in parallel.

Memory-Mapped Registers (MMR)

- The ’C5x has 96 registers mapped into page 0 of the data-memory space.
- All ’C5x DSPs have 28 core CPU registers and 16 input/output (I/O) port registers. They differ in the number of peripheral and reserved registers; generally, there are 17 peripheral registers and 35 reserved registers.
- Since memory-mapped registers are part of the data-memory space, they can be read and written in the same way as other data-memory locations.
- These registers are used for indirect data-address pointers, temporary storage, CPU status and control, and integer arithmetic processing through the ARAU.

6.Program Controller

The program controller contains logic circuitry that decodes instructions, manages the CPU pipeline, stores the status of CPU operations, and handles conditional operations.

The program controller consists of:

1. 16-bit Program Counter (PC)
2. 16-bit status registers ST0 and ST1
3. Processor Mode Status Register (PMST)
4. Circular Buffer Control Register (CBCR)
5. Hardware stack of \(8 \times 16\) bits
6. Address-generation logic
7. Instruction register
8. Interrupt Flag Register and Interrupt Mask Register

Parallelism of the Program Controller

The parallel architecture allows the ’C5x to perform three concurrent memory operations in a machine cycle when using Dual-Access RAM (DARAM):

1. Fetch an instruction.
2. Read an operand.
3. Write an operand.

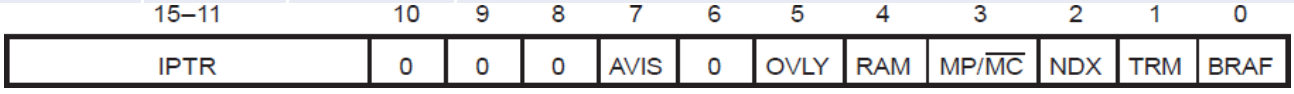
This increases processing speed by allowing multiple memory operations to take place in parallel.

Processor Mode Status Register (PMST)

- PMST is a memory-mapped register that contains processor status and control bits.
- It resides in the memory-mapped register space of data-memory page 0.
- The PMST can be operated on directly by the CALU and PLU.

15 — 11	**IP TR**	**0000 0**	Interruptvectorpointerbits.Thesebitsselectanyof322K-word pages where the interrupt vectors reside.

7	**AVIS**	**0**	Addressvisibilitybit.Thisbitenables/disablestheinternalprogram address to be visible at the address pins.
5	**OVLY**	**0**	Thisbitenables/disablestheon-chipsingle-accessRAM(SARAM)to be addressable in data memory space
4	**RAM**	**0**	Thisbitenables/disablestheon-chipsingle-accessRAM(SARAM)to be addressable in program memory space.
3	**MP/MC**		Thisbitenables/disablestheon-chipROMtobeaddressablein program memory space.
2	**NDX**	**0**	NDX=0Any'C2x-compatibleinstructionthatmodifies orwritesAR0alsomodifiesorwritestheINDXandARCR NDX=1reverseofpreviousstatement
1	**TRM**	**0**	TRM=0Any'C2x-compatibleinstructionthatloadsTREG0alsoloads TREG1 and TREG2 TRM=1reverse ofpreviousstatement
0	**BRAF**	**0**	Blockrepeatactiveflagbit.Thisbitindicatesthatablockrepeat is currently active/de-active.



PMST (Processor Mode Status Register)

- The PMST has an associated one-level-deep shadow register for automatic context saving when an interrupt trap occurs.

CBCR (Circular Buffer Control Register)

- The CBCR is a memory-mapped register (MMR) that enables or disables circular buffers using the CENB1 and CENB2 bits.
- It defines which auxiliary registers are mapped to the circular buffers:
    - CAR1: Selects the auxiliary register for circular buffer 1.
    - CAR2: Selects the auxiliary register for circular buffer 2.

Hardware Stack

- The hardware stack is 16 bits wide and eight levels deep. It is accessible through the PUSH and POP instructions.
- Whenever the contents of the program counter (PC) are pushed onto the top of the stack (TOS), the previous contents of each level are pushed down, and the contents of the bottom (eighth) location are lost.
- Therefore, data is lost if more than eight successive PUSH operations occur before a POP operation.

Address Generation Logic (Program and Data)

- This logic generates addresses for program memory and data memory read and write operations.
- It can generate one address per machine cycle.

Interrupt Flag Register (IFR)

- The IFR is a memory-mapped register that indicates pending interrupts.

Interrupt Mask Register (IMR)

- The IMR is a memory-mapped register used to mask external and internal interrupts.
- Writing 1 to any IMR bit position enables the corresponding interrupt when the INTM (Interrupt Mode) bit = 0.

Some Flags in the Status Registers

- The status registers can be stored in data memory and loaded from data memory, allowing the ’C5x status to be saved and restored for subroutines.
- Both ST0 and ST1 have an associated one-level-deep shadow register for automatic context saving when an interrupt trap occurs.
- A TRAP is a software interrupt that transfers control to program memory location 22h.
- The shadow registers are restored when a return-from-interrupt instruction is executed.

Fig:Status register0 (ST0) bit assignment

ARP	OV	OVM	1	INTM	DP

Significance of Various Bits of ST0 and ST1

1. Status Register 0 (ST0)

a) ARP (Auxiliary Register Pointer)

- These bits select the auxiliary register (AR) to be used in indirect addressing.
- When the ARP is loaded, the previous ARP value is copied to the Auxiliary Register Buffer (ARB) in ST1.

b) OVM (Overflow Mode) Bit

- The OVM bit enables or disables accumulator overflow saturation mode in the ALU.

- OVM = 0: An overflowed result is loaded into the accumulator without modification.
- OVM = 1: An overflowed result is loaded into the accumulator with either the most positive value (00 7FFF FFFFh) or the most negative value (FF 8000 0000h).

c) OV (Overflow) Flag Bit

- The OV flag indicates arithmetic-operation overflow in the ALU.

d) INTM (Interrupt Mode) Bit

- The INTM bit globally masks or enables all maskable interrupts.
- The INTM bit has no effect on the non-maskable RS and NMI interrupts.

e) DP (Data Memory Page Pointer) Bits

- The DP bits specify the address of the current data memory page.
- The DP bits are concatenated with the 7 least significant bits (LSBs) of an instruction word to form a 16-bit direct memory address.

2. Status Register 1 (ST1)

a) ARB (Auxiliary Register Buffer)

- This three-bit field holds the previous value contained in the ARP in ST0.
- Whenever the ARP is loaded, the previous ARP value is copied to the ARB, except when using the LST #0 instruction.
- LST: Loads a data memory value into Status Register 0 or Status Register 1.
- When the ARB is loaded using the LST #1 instruction, the same value is also copied to the ARP.
- The ARB is useful when restoring context in a subroutine that modifies the current ARP without using automatic context saving.

b) CNF (On-Chip RAM Configuration Control Bit)

- This one-bit field enables on-chip dual-access RAM block B0 (DARAM B0) to be addressed in either data memory space or program memory space.
- The CNF bit can be modified using the LST #1 instruction.
- DARAM: Memory that can be read from and written to within the same clock cycle.

- CNF = 0: On-chip DARAM block B0 is mapped into data memory space (CLRC CNF).
- CNF = 1: On-chip DARAM block B0 is mapped into program memory space (SETC CNF).

c) TC (Test/Control Flag Bit)

- This one-bit flag stores the result of ALU or Parallel Logic Unit (PLU) test-bit operations.
- The status of TC determines whether conditional branch, call, and return instructions are executed.

d) SXM (Sign-Extension Mode Bit)

- This one-bit field enables or disables sign extension during arithmetic operations.
- The SXM bit affects the operation of the Shift Accumulator Right (SFR) instruction.
- SXM = 1: SFR performs an arithmetic right shift, maintaining the sign of the accumulator (ACC) data.
- SXM = 0: SFR performs a logical right shift, shifting out the LSBs and shifting in zeros at the MSB.

e) C (Carry Bit)

- This one-bit field indicates an arithmetic carry or borrow in the ALU.
- Shift or rotate operations can also affect the C bit.

f) HM (Hold Mode Bit)

- This one-bit field determines whether the CPU stops or continues execution when acknowledging an active HOLD signal from DMA.

g) XF (External Flag Pin Status Bit)

- This one-bit field determines the logic level of the external flag (XF) output pin.
- The XF pin signals to external devices through software.
- The XF pin is set high when the device is reset.

h) PM (Product Shift Mode Bits)

- This two-bit field determines the product shifter (P-SCALER) mode and the shift value applied to the PREG (product register) output before it enters the ALU.

PMbits		Function
B1	B0	P-SCALERmodeforPREGoutput
0	0	Noshift
0	1	Left-shifted1bit;LSBzerofilled
1	0	Left-shifted4bits;4LSBzerofilled
1	1	Right-shifted6bits;6LSBlost.Productalwayssignextended

On-Chip Memory

- The ’C5x architecture contains a considerable amount of on-chip memory to improve system performance and integration.

The on-chip memory consists of:

1. Program Read-Only Memory (ROM)
2. Data/Program Dual-Access RAM (DARAM)
3. Data/Program Single-Access RAM (SARAM)

- The ’C5x has a total address range of 224K words × 16 bits.
- The memory space is divided into four individually selectable memory segments:

1. 64K-word program memory space
2. 64K-word local data memory space
3. 64K-word input/output (I/O) ports
4. 32K-word global data memory space

PLL Clock Options ('LC56, 'C57S, and 'LC57 only)

CLKMD1	CLKMD2	CLKMD3	Clock Mode
0	0	0	PLL multiply-by-3 option
0	0	1	External divide-by-2 option with internal oscillator disabled
0	1	0	PLL multiply-by-4 option
0	1	1	PLL multiply-by-2 option
1	0	0	PLL multiply-by-5 option
1	0	1	PLL multiply-by-1 option
1	1	0	PLL multiply-by-9 option
1	1	1	External divide-by-2 option or internal divide-by-2 option with an internal oscillator enabled

Program ROM

- All 'C5x DSPs contain a 16-bit on-chip mask-programmable ROM.
- The 'C50 and 'C57S DSPs have boot loader code resident in the on-chip ROM. The main function of the boot loader is to transfer code from an external source to program memory at power-up. Other 'C5x DSPs offer the boot loader code as an option.
- This memory is used for booting program code from slower external ROM or EPROM to fast on-chip or external RAM.
- Once the custom program has been booted into RAM, the boot ROM space can be removed from program memory space by setting the MP/MC bit in the Processor Mode Status Register (PMST).
- The on-chip ROM is selected at reset by driving the MP/MC pin low.
- If the on-chip ROM is not selected, the 'C5x devices start execution from off-chip memory.
- The on-chip ROM may be configured with or without boot loader code.
- The on-chip ROM is intended for your specific program. Once the program is in its final form, you can submit the ROM code to Texas Instruments for implementation into your device.

- The microprocessor/microcomputer (MP/MC) mode is available on all ROM-coded TMS320 DSP devices when accesses to either on-chip or off-chip memory are required.
- The microprocessor mode is used to develop, test, and refine a system application.
- In this mode, the TMS320 acts as a standard microprocessor by using external program memory.
- When the algorithm has been finalized, the code can be submitted to Texas Instruments for masking into the on-chip program ROM.
- At that time, the TMS320 becomes a microcomputer that executes customized programs from the on-chip ROM.
- If the code needs to be changed or upgraded, the TMS320 can once again be used in microprocessor mode.

Data/Program Dual-Access RAM (DARAM)

- All ’C5x DSPs contain 1056 words × 16 bits of on-chip dual-access RAM (DARAM).
- The DARAM is divided into three individually selectable memory blocks:
  - B0: 512-word data or program DARAM block.
  - B1: 512-word data DARAM block.
  - B2: 32-word data DARAM block.
- DARAM is primarily intended to store data values but can also be used to store programs when needed.
- DARAM blocks B1 and B2 are always configured as data memory. However, DARAM block B0 can be configured by software as either data memory or program memory.
- DARAM improves the operational speed of the ’C5x CPU.
- The CPU operates with a four-stage pipeline: Fetch, Decode, Execute, and Write Back.
- In this pipeline, the CPU reads data in the third stage and writes data in the fourth stage.
- Therefore, in a given instruction sequence, the second instruction can read data while the first instruction is writing data.
- The dual data buses (DB and DAB) allow the CPU to read from and write to DARAM in the same machine cycle.

Data/Program Single-Access RAM (SARAM)

- All ’C5x DSPs except the ’C52 contain 16-bit on-chip single-access RAM (SARAM) of various sizes.

- Code can be booted from an off-chip ROM and then executed at full speed once it is loaded into the on-chip SARAM.
- SARAM can be configured by software in one of three ways:
    1. All SARAM configured as data memory.
    2. All SARAM configured as program memory.
    3. SARAM configured as both data memory and program memory.
- SARAM is divided into 1K-word and/or 2K-word blocks that are contiguous in address memory space.
- All ’C5x CPUs support parallel access to these SARAM blocks.
- However, one SARAM block can be accessed only once per machine cycle.
- In other words, the CPU can read from or write to one SARAM block while accessing another SARAM block.
- SARAM supports more flexible address mapping than DARAM because it can be mapped to both program and data memory spaces simultaneously.
- However, because of simultaneous program and data mapping, an instruction fetch and a data fetch that could be performed in one machine cycle with DARAM (using PB and DB) may take two machine cycles with SARAM.

On-Chip Peripherals

- All ’C5x DSPs have the same CPU structure. However, they have different on-chip peripherals connected to their CPUs.
- The available ’C5x DSP on-chip peripherals are:
    1. Clock generator
    2. Hardware timer
    3. Software-programmable wait-state generators
    4. Parallel I/O ports
    5. Host Port Interface (HPI)
    6. Serial port
    7. Buffered Serial Port (BSP)
    8. Time-Division Multiplexed (TDM) serial port
    9. User-maskable interrupts

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 resonator circuit or externally by a clock source.

- The PLL circuit can generate an internal CPU clock by multiplying the clock source by a specific factor. Therefore, a clock source with a lower frequency than the CPU frequency can be used.
- The PLL has a maximum operating frequency of 28.6 MHz, according to the provided notes.

Phase-Locked Loop (PLL)

- A phase-locked loop (PLL) is a control system that generates an output signal whose phase is related to the phase of an input reference signal.
- It is an electronic circuit consisting of a variable-frequency oscillator and a phase detector.
- The circuit compares the phase of the input signal with the phase of the signal derived from its output oscillator and adjusts the oscillator frequency to keep the phases matched.
- The signal from the phase detector controls the oscillator through a feedback loop.
- Consequently, a PLL can track an input frequency or generate a frequency that is a multiple of the input frequency.
- The former property is used for demodulation, while the latter is used for indirect frequency synthesis.
- Phase-locked loops are widely used in radio, telecommunications, computers, and other electronic applications.

Hardware Timer

- A 16-bit hardware timer with a 4-bit prescaler is available.
- This programmable timer operates at a rate between 1/2 and 1/32 of the machine-cycle rate (CLKOUT1), depending on the timer's divide-down ratio. In the 8051 microcontroller, the timer frequency is 1/12 of the crystal frequency.
- CLKOUT1: This is the master clock output signal. It operates at the machine-cycle rate of the CPU. The frequency of CLKOUT1 is one-half of the crystal oscillation frequency or CLKIN rate (20–40 MHz). Sometimes, the CLKOUT rate equals the CLKIN rate.
- The internal machine cycle is bounded by the rising edges of the CLKOUT1 signal.
- The timer can be stopped, restarted, reset, or disabled using specific status bits.

Table 9–4. Timer Control Register (TCR) Bit Summary

Bit	Name	Reset value	Function
15–12	Reserved	—	These bits are reserved and are always read as 0.
11	Soft	0	This bit is used in conjunction with the Free bit to determine the state of the timer when a halt is encountered. When the Free bit is cleared, the Soft bit selects the emulation mode. Soft = 0 The timer stops immediately. Soft = 1 The timer stops after decrementing to zero.
10	Free	0	This bit is used in conjunction with the Soft bit to determine the state of the timer when a halt is encountered. When the Free bit is cleared, the Soft bit selects the emulation mode. Free = 0 The Soft bit selects the timer mode. Free = 1 The timer runs free regardless of the Soft bit.

9–6	PSC	—	Timer prescaler counter bits. These bits specify the count for the on-chip timer. When the PSC is decremented past 0 or the timer is reset, the PSC is loaded with the contents of the TDDR, and the TIM is decremented.
5	TRB	—	Timer reload bit. This bit resets the on-chip timer. When the TRB is set, the TIM is loaded with the value in the PRD and the PSC is loaded with the value in the TDDR. The TRB is always read as a 0.
4	TSS	0	Timer stop status bit. This bit stops or starts the on-chip timer. At reset, the TSS bit is cleared and the timer immediately starts timing.Note that due to timer logic implementation, two successive writes of one to the TSS bit are required to properly stop the timer. TSS = 0 The timer is started. TSS = 1 The timer is stopped.
3–0	TDDR	0000	Timer divide-down register bits. These bits specify the timer divide-down ratio (period) for the on-chip timer. When the PSC bits are decremented past 0, the PSC is loaded with the contents of the TDDR.

Software-Programmable Wait-State Generators

- Software-programmable wait-state logic is incorporated into ’C5x DSPs, allowing wait-state generation without any external hardware for interfacing 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 with different wait states for off-chip memory accesses.

Parallel I/O Ports

- Sixteen 16-bit parallel I/O ports are available. These ports are memory-mapped in data memory space, page 0.
- Each I/O port can be addressed using the IN or OUT instruction.
- The memory-mapped I/O ports can be accessed with any instruction that reads from or writes to data memory.
- The IS signal indicates a read or write operation through an I/O port.

- The ’C5x can easily interface with external I/O devices through the I/O ports while requiring minimal off-chip address-decoding circuitry.

Host Port Interface (HPI)

- The HPI available on the ’C57S and ’LC57 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.

Table 2–1. Number of Serial/Parallel Ports Available in Different ’C5x Package Types

TMS320 Device	Package ID†	High-Speed Serial Port	TDM Serial Port	Buffered Serial Port	Host Port (Parallel)
’C50/’LC50	PQ	1	1	–	–
’C51/’LC51	PQ/PZ	1	1	–	–
’C52/’LC52	PJ/PZ	1	–	–	–
’C53/’LC53	PQ	1	1	–	–
’C53S/’LC53S	PZ	2	–	–	–
’LC56	PZ	1	–	1	–
’C57S/’LC57S	PGE	1	–	1	1
’LC57	PBK	1	–	1	1

† PGE is a 20 × 20 × 1.4 mm thin quad flat-pack (TQFP) package
PJ is a 14 × 20 × 2.7 mm quad flat-pack (QFP) package
PQ is a 20 × 20 × 3.8 mm bumpered quad flat-pack (BQFP) package
PZ and PBK are a 14 × 14 × 1.4 mm thin quad flat-pack (TQFP) package

Serial Port

- 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 at least one general-purpose, high-speed, synchronous, full-duplex serial port interface that provides direct communication with serial devices such as codecs, serial analog-to-digital (A/D) converters, and other serial systems.
- The serial port is capable of operating at up to one-fourth of the machine cycle rate (CLKOUT1).
- The serial port transmitter and receiver are double-buffered and individually controlled by maskable external interrupt signals.
- Data is framed either as bytes or as words.

Buffered Serial Port (BSP)

- The BSP available on the ’C56 and ’C57 devices is a full-duplex, double-buffered serial port with an auto-buffering unit (ABU).
- The BSP provides flexibility in the data stream length.
- The ABU supports high-speed data transfer and reduces interrupt latency.

TDM Serial Port

- The TDM serial port available on the ’C50, ’C51, and ’C53 devices is a full-duplex serial port that can be configured by software for either synchronous operation or time-division multiplexed operation.
- The TDM serial port is commonly used in multiprocessor applications.

User-Maskable Interrupts

- Four external interrupt lines (INT1–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.
    - The contents of eleven specific CPU registers are automatically saved (shadowed) on a 1-level-deep stack (shadow registers).
- When a return-from-interrupt instruction is executed, the CPU register contents are restored.

Test/Emulation

- On the ’C50, ’LC50, ’C51, ’LC51, ’C53, ’LC53, ’C57S, and ’LC57S, an IEEE Standard 1149.1 (JTAG) interface with boundary-scan capability is used for emulation and testing.
- This logic provides boundary scanning to and from the interfacing devices.
- It can be used to test pin-to-pin continuity and perform operational tests on devices that are peripheral to the ’C5x.

Important Questions:

1. Explain the architecture of the TMS320C5x DSP processor with a neat block diagram.

2. Explain the bus structure of the TMS320C5x architecture and describe the functions of the different buses.

3. Explain the Central Arithmetic Logic Unit (CALU) and its functions in the TMS320C5x DSP processor.

4. Explain the Auxiliary Register ALU (ARAU) and describe how it supports indirect addressing.

5. Explain the functions of the Index Register and Block Move Address Register in the TMS320C5x architecture.

6. Explain the Parallel Logic Unit (PLU) and its role in bit manipulation and logical operations.

7. Describe the memory-mapped registers (MMRs) of the TMS320C5x processor and explain their functions.

8. Explain the program controller of the TMS320C5x architecture, including program sequencing, branching, and interrupt handling.

9. Explain the important flags and bits in the Status Registers ST0 and ST1, including ARP, OVM, OV, INTM, DP, ARB, CNF, TC, SXM, C, HM, XF, and PM.

10. Explain the on-chip memory organization of the TMS320C5x DSP, including Program ROM, DARAM, SARAM, and memory-space organization.

11. Explain the on-chip peripherals of the TMS320C5x DSP, including the clock generator, hardware timer, wait-state generators, parallel I/O ports, HPI, serial ports, interrupts, and JTAG interface.