**Unit-V**

**Digital Signal Processors**

**Session-1 Date: 03.10.13 , 1st hour, Time: 9.15 am- 10.05 am**

**Recap: Digital design using impulse invariant**

**Suggested Activity: Quiz**

1. **Warping Effect:** Distortion is introduced in the frequency scale of the digital filter relative to that of the analog filter.
2. **Digital frequency,** ω = 2/T tan ( ΩT/2)

**Content: Introduction to Digital Signal Processors**

**Suggested Activity: Board Activity**

**Central Processing Unit (CPU)**

The CPU of the ’54x devices contains:

\_ A 40-bit arithmetic logic unit (ALU)

\_ Two 40-bit accumulators

\_ A barrel shifter

\_ A 17 × 17-bit multiplier/adder

\_ A compare, select, and store unit (CSSU)

**Arithmetic Logic Unit (ALU)**

The ’54x devices perform 2s-complement arithmetic using a 40-bit ALU and

two 40-bit accumulators (ACCA and ACCB). The ALU also can perform

Boolean operations.

The ALU can function as two 16-bit ALUs and perform two 16-bit operations

simultaneously when the C16 bit in status register 1 (ST1) is set.

**Conclusion: Introduction to Digital Signal Processors**

**Suggested Activity:** Show & Tell Activity

**Accumulators**

The accumulators, ACCA and ACCB, store the output from the ALU or the multiplier / adder block; the accumulators can also provide a second input to the ALU or the multiplier / adder. The bits in each accumulator is grouped as follows:Guard bits (bits 32–39), A high-order word (bits 16–31), A low-order word (bits 0–15) Instructions are provided for storing the guard bits, the high-order and the low-order accumulator words in data memory, and for manipulating 32-bit accumulator words in or out of data memory. Also, any of the accumulators can be used as temporary storage for the other.

**Ref:** <http://www.ti.com/lsds/ti/dsp/multicore/c647x/products.page?paramCriteria=no>

**Session-2 Date: 04.10.13 , 2nd hour, Time: 10.05 am- 10.55 am**

**Recap: Introduction to Digital Signal Processors**

**Suggested Activity: Quiz**

1. The ’54x device architecture is built around eight major 16-bit buses:
2. One program-read bus (PB) which carries the instruction code and immediate operands from program memory
3. Two data-read buses (CB, DB) and one data-write bus (EB), which interconnect to various elements, such as the CPU, data-address generation logic (DAGEN), program-address generation logic (PAGEN), on-chip peripherals, and data memory
4. The CB and DB carry the operands read from data memory.
5. The EB carries the data to be written to memory.
6. Four address buses (PAB, CAB, DAB, and EAB), which carry the addresses needed for instruction execution.

**Content: Architecture**

**Suggested Activity:** Brain Storming

1. The ’54x DSPs use an advanced, modified Harvard architecture
2. Maximizes processing power by maintaining one program memory bus and

three data memory buses.

1. These processors also provide an arithmetic logic unit (ALU) that has a high degree of parallelism, application-specific hardware logic, on-chip memory, and additional on-chip peripherals.
2. These DSP families also provide a highly specialized instruction set, which is the basis of

the operational flexibility and speed of these DSPs.

**Conclusion: Architecture**

**Suggested Activity:** Questions & Answers

1. **Compare, Select, and Store Unit (CSSU)**

The CSSU also accelerates Viterbi-type butterfly computation with optimized on-chip hardware.

1. **Program Control**

Program control is provided by several hardware and software mechanisms:

* The program controller decodes instructions, manages the pipeline, stores the status of operations, and decodes conditional operations.
* Some of the hardware elements included in the program controller are the program counter, the status and control register, the stack, and the address-generation logic.
* Some of the software mechanisms used for program control include branches, calls, conditional instructions, a repeat instruction, reset, and interrupts.

**Ref :** <http://www.ti.com/lsds/ti/dsp/embedded_processor.page>

**Session-3 Date: 07.10.13 , 7th hour, Time: 3.20 pm- 4.10 pm**

**Recap: Architecture**

**Suggested Activity: Quiz**

1. **Status Registers (ST0, ST1)**

The status registers, ST0 and ST1, contain the status of the various conditions and modes for the

’54x devices. ST0 contains the flags (OV, C, and TC) produced by arithmetic operations and bit

manipulations in addition to the data page pointer (DP) and the auxiliary register pointer (ARP)

1. **Auxiliary Registers (AR0–AR7)**

The eight 16-bit auxiliary registers (AR0–AR7) can be accessed by the central airthmetic logic unit (CALU) and modified by the auxiliary register arithmetic units (ARAUs).

**Content: Features**

**Suggested Activity:** Group Activity

The entire class is divided into totally five groups. Each group is assigned a specific topic and asked to discuss about various points involved in that topic.

**Group-1:** The first group is asked to discuss about **Temporary Register (TREG)**

The TREG is used to hold one of the multiplicands for multiply and multiply/accumulate instructions. It can hold a dynamic (execution-time programmable) shift count for instructions with a shift operation such as ADD, LD, and SUB. It also can hold a dynamic bit address for the BITT instruction.

**Group-2:** The second group is asked to discuss about **Transition Register (TRN)**

The TRN is a 16-bit register that is used to hold the transition decision for the path to new metrics to perform the Viterbi algorithm. The CMPS (compare, select, max, and store) instruction updates the contents of the TRN based on the comparison between the accumulator high word and the accumulator low word.

**Conclusion: Feature**

**Suggested Activity:** Recall by keywords

1. **Stack-Pointer Register (SP):** The SP is a 16-bit register that contains the address at the top of the system stack.
2. **Circular-Buffer-Size Register (BK) :** The 16-bit BK is used by the ARAUs in circular addressing to specify the data block size

**Ref :** <http://www.dspguide.com/ch28.htm>

**Ref :**  <http://www.futureelectronics.com/en/Microprocessors/digital-signal-processors.aspx>

**Session-4 Date: 08.10.13 , 1st hour, Time: 9.15 am- 10.05 am**

**Recap: Features**

**Suggested Activity: Quiz**

1. **Interrupt Registers (IMR, IFR) :** The interrupt-mask register (IMR) is used to mask off specific interrupts individually at required times. The interrupt-flag register (IFR) indicates the

current status of the interrupts.

1. **Processor-Mode Status Register (PMST):** The processor-mode status register (PMST) controls memory configurations of the ’54x devices.

**Content: Addressing Formats**

**Suggested Activity:** Odd man out

**Introduction to Addressing Modes**

•**Addressing Mode (definition):**

• **The strategy to identify WHERE the operand is located.**

• **Note1: each addressing mode has a unique formula to locate the**

**operand.**

• **Note2: The operand is in memory in all (but 2) addressing modes**

•**Effective Address (definition):**

• **Actual location of operand to be used by the instruction:**

•**How does each type of instruction use effective addr?**

• **“load type” instr: Rdest** **Mem[eff.addr]**

• **“store type” instr: Mem[eff.addr]** **Rsrc**

• **“jump type” instr: PC** **“effective address”**

• **“copy eff. addr type” instr: Rdest** **“effective address”**

• **“ALU type” instr (Intel ISA) Rdest** **Rsrc + Mem[eff.addr]**

**Conclusion: Addressing Formats**

**Suggested Activity:** Questions & Answers

**“(Direct) Register” Addressing Mode**

•**Memory Addressing Mode?** No

• **“Effective Address”:** operand is located in specified reg #

• **LC-3 syntax:** Rn, where 0 ! n ! 7

• **Generic RISC syntax:** Rn, where 0 ! n ! 31

• **Intel syntax:** EAX, EBX, ….

•**Usage:**

• For permanent location of scalar variables

• For temporary calculations of any data or addresses

•**Popularity:** In ISAs that are not a RISC “Load/Store” ISA, this addressing mode is used for 50% of operand references

**Ref:** <http://www.futureelectronics.com/en/Microprocessors/digital-signal-processors.aspx>

**Session-5 Date: 09.10.13 , 1st hour, Time: 9.15 am- 10.05 am**

**Recap: Addressing Formats**

**Suggested Activity: Brainstorming**

**Immediate Addressing Mode**

•**Memory Addressing Mode?** No

• **“Effective Address”:** operand located in the INSTRUCTION

• **LC-3 syntax:** #<decimal digits>, x<hex digits>, b<bits>

• **Generic syntax:** #<digits> (or symbolic name of a constant)

• **Intel syntax:** <decimal digits> for 8/16/32bit constants

•**Usage:**

• For any compile-time or assembly-time constants

• Modern ISAs have a limited # of bits for the operand, e.g. 10-16

•**Popularity: (3rd)** In ISAs that are not a RISC “Load/Store” ISA, this addressing mode is used for (9-20%) of operand references.

**Content: Functional modes**

**Suggested Activity: Board Activity**

The math processing is broken into three sections, a **multiplier**, an **arithmetic logic unit (ALU)**, and a **barrel shifter**. The multiplier takes the values from two registers, multiplies them, and places the result into another register. The ALU performs addition, subtraction, absolute value, logical operations (AND, OR, XOR, NOT), conversion between fixed and floating point formats, and similar functions.\

1. Family Functional Overview (such as this document)
2. Device-specific data sheets
3. Complete User Guides
4. Development support tools
5. Hardware and software application reports

**Conclusion: Functional modes**

**Suggested Activity:** Rapid fire

**Register Indirect Addressing Mode**

•**Memory Addressing Mode?** Yes

• **Effective Address:** contents(Register <n>)

• **LC-3 syntax:** can be modelled with Base + “offset of zero”

plus: JMP R7, JSRR R3

• **Generic syntax:** (R<n>)

•**Usage: accessing “a pointer” or a computed address**

• Most basic memory addressing mode to access any type of data

• Particularly important for locations that must be calculated at run-time

.Ref:<http://www.google.co.in/imgres?newwindow=1&sa=X&tbm=isch&tbnid=0u6nzxw5ktz6xM:&imgrefurl>

**Session-6 Date: 09.10.13 , 1st hour, Time: 10.05 am- 10.55 am**

**Recap: Functional modes**

**Suggested Activity:** Group Discussion

**On-Chip Peripherals**

All the ’54x devices have the same CPU structure; however, they have

different on-chip peripherals connected to their CPUs. The on-chip peripheral

options provided are:

1. Software-programmable wait-state generator
2. Programmable bank-switching
3. Parallel I /O ports
4. DMA controller
5. Host-port interface (standard 8-bit, enhanced 8-bit, and 16-bit)
6. Serial ports (standard, TDM, BSP, and McBSP)
7. General-purpose I/O pins
8. 16-bit timer with 4-bit prescaler
9. Phase-locked loop (PLL) clock generator

**Content: Introduction to Commercial Processors**

**Suggested Activity:** Brainstorming

The students were asked to discuss about the linear phase characteristics of various filters.

A typical RISC characteristic is a small and simple set of addressing modes

• ARM departs somewhat from this convention with a relatively rich set of addressing modes



**Conclusion: Introduction to Commercial Processors**

**Suggested Activity:** Brainstorming

* Digital Signal Processing can be divided into two categories, **fixed point** and **floating point**. These refer to the format used to store and manipulate numbers within the devices. Fixed point DSPs usually represent each number with a minimum of 16 bits, although a different length can be used.
* With **unsigned fraction** notation, the 65,536 levels are spread uniformly between 0 and 1. Lastly, the **signed fraction** format allows negative numbers, equally spaced between -1 and 1.

Ref:[http://www.ti.com/general/docs/lit/getliterature.tsp?baseLiteratureNumber=sprt580&fileType= pdf](http://www.ti.com/general/docs/lit/getliterature.tsp?baseLiteratureNumber=sprt580&fileType=%20pdf)

**Session-7 Date: 10.10.13 , 1st hour, Time: 9.15 am- 10.05 am**

**Recap: Introduction to Commercial Processors**

**Suggested Activity: Quiz**

**Data Memory**

1. The data memory space on the ’54x device addresses 64K of 16-bit words.
2. The device automatically accesses the on-chip RAM when addressing within its bounds. When an address is generated outside the RAM bounds, the device automatically generates an external access.
3. The advantages of operating from on-chip memory are as follows:

* Higher performance because no wait states are required
* Higher performance because of better flow within the pipeline of the CALU
* Lower cost than external memory
* Lower power than external memory

1. In addition to general-purpose data memory, the CPU maintains a set of
2. memory-mapped registers in data memory for processor configuration and configuration/communication with the device peripherals.
3. For detailed information on the implementation of the memory-mapped CPU and

peripheral control registers, see the device-specific data sheets.

**Content:**  **Arithmetic Logic Unit**

**Suggested Activity:** Group Discussion

The entire class is divided into totally two groups. Each group is assigned a specific topic and asked to discuss about various points involved in that topic.

**Arithmetic Logic Unit (ALU)**

The ’54x devices perform 2s-complement arithmetic using a 40-bit ALU and two 40-bit accumulators (ACCA and ACCB). The ALU also can perform Boolean operations. The ALU can function as two 16-bit ALUs and perform two 16-bit operations simultaneously when the C16 bit in status register 1 (ST1) is set.

**1.2.3 Accumulators**

The accumulators, ACCA and ACCB, store the output from the ALU or the multiplier / adder block; the accumulators can also provide a second input to the ALU or the multiplier / adder. The bits in each accumulator is grouped as follows:

* Guard bits (bits 32–39)
* A high-order word (bits 16–31)
* A low-order word (bits 0–15)

Instructions are provided for storing the guard bits, the high-order and the low-order accumulator words in data memory, and for manipulating 32-bit accumulator words in or out of data memory. Also, any of the accumulators can be used as temporary storage for the other.

**Conclusion: Arithmetic Logic Unit**

**Suggested Activity:** One word Answer

**Barrel Shifter**

The ’54x’s barrel shifter has a 40-bit input connected to the accumulator or data memory (CB, DB) and a 40-bit output connected to the ALU or data memory (EB). The barrel shifter produces a left shift of 0 to 31 bits and a right shift of 0 to 16 bits on the input data.

Ref: <http://www.ehow.com/list_7332165_types-addressing-modes-computers.html>

**Session-8 Date: 11.10.13 , 1st hour, Time: 9.15 am- 10.05 am**

**Recap: Arithmetic Logic Unit**

**Suggested Activity: Quiz**

Time-division multiplexed (TDM) serial port

1. Buffered serial port (BSP)
2. Multichannel buffered serial port (McBSP)
3. Direct memory access (DMA) controller
4. 8-bit parallel host-port interface (HPI)
5. Enhanced 8-bit parallel host-port interface (HPI8)
6. 16-bit parallel host-port interface (HPI16)
7. 16-bit timer with 4-bit prescaler
8. Interprocessor first-in first-out (FIFO) unit (on multiple CPU devices)

**Content: Accumulator Register**

**Suggested Activity:** Board Activity

1. Power conservation features
2. Software power consumption control with IDLE1, IDLE2, and IDLE3 power-down modes
3. Ability to disable external address bus, data bus, and control bus signals under software control
4. Ability to disable CLKOUT under software control
5. Low-voltage device options to reduce power consumption without compromising performance

**Conclusion: Accumulator Register**

**Suggested Activity:** Pick & Answer

1. On-chip scan-based emulation capability IEEE 1149.1† (JTAG) boundary scan test capability
2. 5.0-V power supply devices with speeds up to 40 million instructions per second (MIPS) (25-ns instruction cycle time)
3. 3.3-V power supply devices with speeds up to 80 MIPS (12.5-ns instruction cycle time)
4. 2.5-V power supply devices with speeds up to 100 MIPS (10-ns instruction cycle time)
5. 1.8-V power supply devices with speeds up to 200 MIPS (10-ns instruction cycle time per CPU core)
6. 1.5-V power supply devices with speeds up to 532 MIPS (7.5-ns instruction cycle time per CPU core)

**Ref:** <http://www.search.ask.com/web?o=APN10152&tpr=5&q=Types+of+Addressing+Modes&tpr=6>

**Session-9 Date: 11.10.13 , 2nd hour, Time: 10.05 am- 10.55 am**

**Recap: Accumulator Register**

**Suggested Activity:** Quiz

The accumulators, ACCA and ACCB, store the output from the ALU or the multiplier / adder block; the accumulators can also provide a second input to the ALU or the multiplier / adder. The bits in each accumulator is grouped as follows:

1. Guard bits (bits 32–39)
2. A high-order word (bits 16–31)
3. A low-order word (bits 0–15)

**Content: Design of Arithmetic Logic Unit**

**Suggested Activity:** Board Activity

1. **Central Processing Unit (CPU)**

* The CPU of the ’54x devices contains:
* A 40-bit arithmetic logic unit (ALU)
* Two 40-bit accumulators
* A barrel shifter
* A 17 17-bit multiplier/adder
* A compare, select, and store unit (CSSU)

1. **Arithmetic Logic Unit (ALU)**

The ’54x devices perform 2s-complement arithmetic using a 40-bit ALU and two 40-bit

accumulators (ACCA and ACCB). The ALU also can perform Boolean operations.

The ALU can function as two 16-bit ALUs and perform two 16-bit operations simultaneously when

the C16 bit in status register 1 (ST1) is set.

**Conclusion: Design of Arithmetic Logic Unit**

**Suggested Activity:** Questions & Answers

The multiplier / adder performs 17 × 17-bit 2s-complement multiplication with a 40-bit accumulation in a single instruction cycle. The multiplier / adder block consists of several elements: a multiplier, adder, signed/unsigned input control, fractional control, a zero detector, a rounder (2s-complement),

overflow/saturation logic, and TREG. The multiplier has two inputs: one input

is selected from the TREG, a data-memory operand, or an accumulator; the

other is selected from the program memory, the data memory, an accumulator,

or an immediate value. The fast on-chip multiplier allows the ’54x to perform

operations such as convolution, correlation, and filtering efficiently.

**Ref:** <http://www.search.ask.com/web?o=APN10152&tpr=5&q=Addressing+Modes+Tutorial>

**Session-10 Date: 12.10.13 , 5th hour, Time: 1.30 pm- 2.20 pm**

**Content: Tutorial: Design of Accumulator**

**Suggested Activity:** Board Activity- Problem Solving

**Ref:**<http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=1458134&url=http%3A%2F%2Fieeexplore.ieee.org%2Fiel5%2F5%2F31377%2F01458134>

**Session-11 Date: 12.10.13 , 6th hour, Time: 2.20 pm-3.10 pm**

**Content: Tutorial: Status Register**

**Suggested Activity:** Board Activity- Problem Solving

**Ref:** <http://www.ti.com/general/docs/lit/getliterature.tsp?baseLiteratureNumber=SPRU307&track=no>

**Session-12 Date: 17.10.13 , 4th hour, Time: 12.10 pm-12.50 pm**

**Content: Tutorial: Processor Unit**

**Suggested Activity:** Board Activity- Problem Solving

**Ref:**<http://www.search.ask.com/web?q=functional+modes+of+digital+signal+processors+&o=APN10152&tpr=2>