CMR INSTITUTE OF TECHNOLOGY

USN | | | | | | | | | | |

Internal Assesment Test - II

Sub:	DSP Algorithms & Architecture							Code:	10EC751/10TE74
Date:	04 / 11 /2016	Duration:	90 mins	Max Marks:	50	Sem:	VII	Branch:	ECE "B" & "D" Sec TCE "A" & "B" Sec

Answer Any FIVE FULL Questions

		Marks	OBE CO	OBE RBT
1.	List and explain any four Data Addressing modes of TMS320C54XX DSP Processor with examples.	[10]	CO3	L1

2(a)	Show the Pipeline Operation of the following sequence of instructions if the initial value of AR3 is 80 and the values stored in the memory locations 80, 81, 82 are 1, 2, and 3 respectively.
	LD *AR3+,A
	ADD #1000h,A
	STL A,*AR3+
		[06]	CO3	L3

(b)	Explain the operations of the following instructions with respect to TMS320C54XX Processor: (i) MAS *AR3-,*AR4+,B,A (ii) MPY #01234,A (iii) RPT	[04]	CO3	L2
3.	Demonstrate an assembly language program of TMS320C54XX Processor to compute the sum of three product terms given by the equation $y(n) = h_0 x(n) + h_1 x(n-1) + h_2 x(n-2)$ where x (n), x (n-1) and x (n-2) are data samples stored at three successive data memory locations and h_0, h_1, h_2 are constants stored at three other successive locations in the data memory. The result y (n) is to be stored in the data memory. Use Indirect addressing mode and MAC Instruction.	[10]	CO3	L3
4(a)	Differentiate between MAC and MACD instructions with examples.	[4]	CO3	L2
(b)	Calculate the following for a 512 point FFT Computation: (i) Number of stages (ii) Number of butterflies in each stage (iii) Number of butterflies needed for the entire computation (iv)Number of butterflies that need no twiddle factors (v) Number of butterflies that require real twiddle factors (vi) Number of butterflies that require complex twiddle factors.	[6]	CO4	L3
5(a)	Describe DIT-FFT Algorithm and derive the equation to implement a butterfly structure.	[6]	CO4	L2
(b)	Illustrate a subroutine program to find the spectrum of the transformed data using TMS320C54XX DSP Processor.	[4]	CO4	L3
6	Discuss briefly about scaling and derive the expression for optimum scaling factor for DIT-FFT Butterfly Algorithm.	[10]	CO4	L1
7.	Explain briefly about Memory Interface and design a data memory system with address range 7FF800 – 7FFFFFh for a C5416 Processor. Use 2K x 8 SRAM memory chip.	[10]	CO5	L2,L4

Course Outcomes		PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1:	Apply basic signal processing concepts in sampling, discrete sequences, DFT, FFT and digital filters. Explain basic DSP architecture and computational blocks.	3	0	3	0	0	0	0	0	0	0	0	0
CO2:	Explain bus architecture digital signal processors, memory and data addressing modes, individual architectural blocks and features for external interfacing.	2	0	3	0	0	0	0	0	0	0	0	0
CO3:	Explain the differences in commercial DSP devices, addressing modes and memory space specifically for TMS320c54xx, program control, instructions, programming and pipeline operations	3	0	3	0	0	0	0	0	0	0	0	0
CO4:	Use the instruction and code basic algorithms for FIR, IIR filters and FFT. Explain and apply Q-notation. Apply algorithms for overflow and scaling and bit-reversed indexing on TMS320c54xx	3	2	3	2	0	0	0	0	0	0	0	0
CO5:	Develop algorithms for accessing interrupts interfacing peripherals, external buses.	3	2	3	3	0	0	0	0	0	0	1	0
CO6:	Develop and analyze algorithms for DSP bio-telemetry receivers, speech processing and image processing.	3	2	3	3	0	0	0	0	0	0	1	0

Cognitive level	KEYWORDS
L1	List, define, tell, describe, identify, show, label, collect, examine, tabulate, quote, name, who, when, where, etc.
L2	summarize, describe, interpret, contrast, predict, associate, distinguish, estimate, differentiate, discuss, extend
L3	Apply, demonstrate, calculate, complete, illustrate, show, solve, examine, modify, relate, change, classify, experiment, discover.
L4	Analyze, separate, order, explain, connect, classify, arrange, divide, compare, select, explain, infer.
L5	Assess, decide, rank, grade, test, measure, recommend, convince, select, judge, explain, discriminate, support, conclude, compare, summarize.

PO1 - *Engineering knowledge*; PO2 - *Problem analysis*; PO3 - *Design/development of solutions*; PO4 - *Conduct investigations of complex problems*; PO5 - *Modern tool usage*; PO6 - *The Engineer and society*; PO7- *Environment and sustainability*; PO8 – *Ethics*; PO9 - *Individual and team work*; PO10 - *Communication*; PO11 - *Project management and finance*; PO12 - *Life-long learning*

CMR INSTITUTE OF TECHNOLOGY
DEPARTMENT OF ECE/TCE
10EC751/10TE74-DSP ALGORITHMS & ARCHITECTURE
SCHEME AND SOLUTION –Internal Assessment Test 2

1. Addressing Modes of TMS320C54XX Processor:
   Any four Addressing Modes (Each 2.5M x 4= 10M)

   The 54XX devices offer seven basic addressing modes
   1. Immediate addressing.
   2. Absolute addressing.
   3. Accumulator addressing.
   4. Direct addressing.
   5. Indirect addressing.
   6. Memory mapped addressing
   7. Stack addressing

Immediate addressing:

The instruction contains the specific value of the operand. The operand can be short (3, 5, 8 or 9 bit in length) or long (16 bits in length). The instruction syntax for short operands occupies one memory location,
Example: LD #20, DP.
RPT #0FFFFh.

Absolute Addressing:

The instruction contains a specified address in the operand.

1. Dmad addressing.
   MVDK Smem,dmad MVDM dmad,MMR
2. Pmad addressing.
   MVDP Smem,pmad MVPD pmem,Smad
3. PA addressing. PORTR PA, Smem,
4. *(lk) addressing

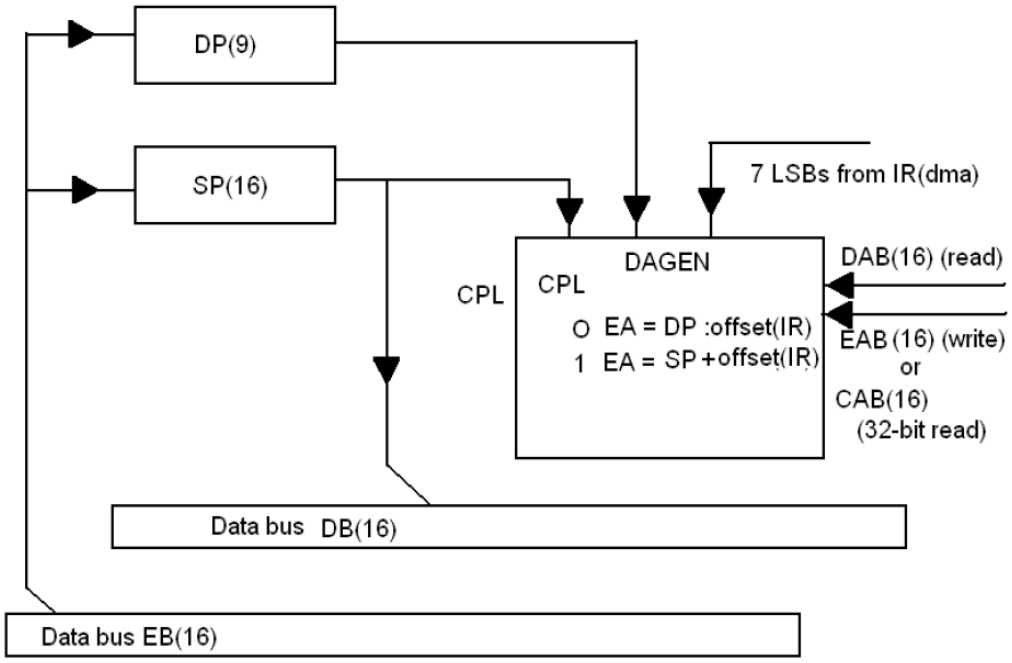
Example: MVKP 1000h, *AR5; 1000 H *AR5 (dmad addressing)
MVPD 1000h, *AR7; 1000h *AR7 (pmad addressing)
PORTR 05h, *AR3; 05h *AR3 (PA addressing)
LD *(1000h), A; *(1000h) A (*(lk) addressing)

Accumulator Addressing:

Accumulator content is used as address to transfer data between Program and Data memory.
Ex: READA *AR2

Direct Addressing:

Base address + 7 bits of value contained in instruction = 16 bit address. A page of 128 locations can be accessed without change in DP or SP.Compiler mode bit (CPL) in ST1 register is used.
If CPL =0 selects DP
CPL = 1 selects SP
It should be remembered that when SP is used instead of DP, the effective address is computed by adding the 7-bit offset to SP.



2 (a) Pipeline Operation: (6 Marks)

Cycle	Pre-fetch	Fetch	Decode	Access	Read	Exec & Write	AR3	A
1	LD						80	X
2	ADD	LD					80	X
3	STL	ADD	LD				80	X
4		STL	ADD	LD			81	X
5			STL	ADD	LD		81	1
6				STL		LD	82	0001h
7					STL	ADD	82	1001h
8						STL	82	1001h

(b)

(i) MAS *AR3-,*AR4+, B, A (2 marks)

This instruction multiplies the contents of the data memory pointed by AR3 by the Contents of the data memory pointed by AR4.The product is subtracted from the Contents of the accumulator B and the result is placed in the accumulator A.During This instruction, register T is loaded with the contents of the same data-memory location Pointed by AR3.AR3 is then decremented by 1 and AR4 incremented by 1.

(ii)MPY #01234, A (2 marks)

This instruction multiplies the current contents of the T register by the constant 1234 and Places the result in the accumulator A.

(iii)RPT (2 marks)

The format of this instruction is

RPT Smem

Or RPT #k

The instruction loads the operand in the repeat counter, RC.The instruction following the RPT Instruction is repeated k+1 times, where k is the initial value of the RC.

3. Program to compute y (n) = h_0 x (n) + h_1x (n-1) + h_2 x (n-2) using MAC instruction

Initialization-(2 marks)
Program- (8 marks)

```
.global _c_int00
.data
.bss  x, 3
.bss  y, 2

h       .int 10, 20, 30

        .text

_c_int00:
        SSBX  SXM          ; Select sign extension mode
        STM   #x, AR2              ; Initialize AR2 to point to x(n)
```

```
STM   #h, AR3          ; Initialize AR3 to point to h(0)
LD    #0H,   A         ; Initialize result in A = 0

RPT   #2               ; Repeat the next operation 3 times
MAC   *AR2+,*AR3+,A    ; y(n) computed

STM   #y, AR2          ; Select the page for y(n)
STL   A,  *AR2+        ; Save the low part of y(n)
STL   A,  *AR2+        ; Save the high part of y(n)

NOP                    ; No operation
.end
```

4. (a) MAC Instruction: (2 marks)

One of the formats of this instruction is

MAC Xmem, Ymem,src,dst

Xmem and Ymem are dual data memory operands and src and dst are accumulators A and B.

The instruction multiplies a data memory value by another data-memory value and adds the Product to the contents of the source, which may be either of the two accumulators A and B.The Result is stored in the other accumulator. The register T is loaded with the Xmemvalue. Similar tothe MPY instruction, this instruction can modify the contents of auxiliary registers used in Indirect addressing.

MACD Instruction: (2 marks)

Multiply, Accumulate, and Delay instruction carries out all the functions of the MAC Instruction and in addition, copies the contents of the current data-memory address to the next higher data-memory address. This feature is equivalent to implementing the z^{-1} delay encountered in digital signal processing algorithms.

(b) N = 512 (6 marks)

Number of stages = $\log_2 N = \log_2 512 = 9$

Number of butterflies in each stage = $\frac{N}{2} = \frac{512}{2} = 256$

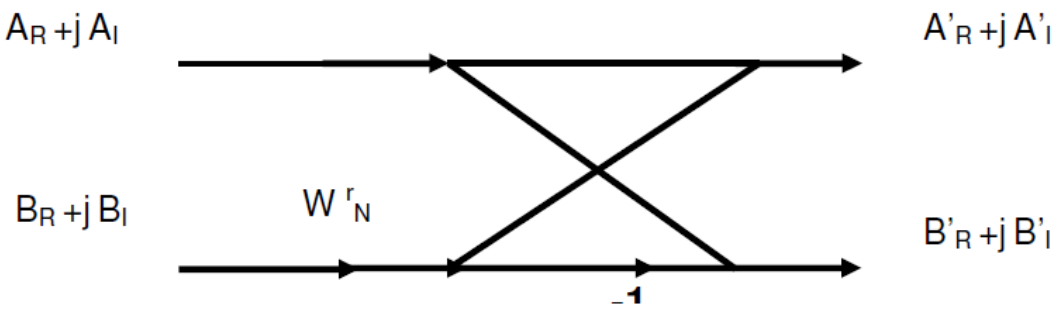
Number of butterflies needed for the entire computation= $\frac{N}{2}\log_2 N = 2304$

Number of butterflies that need no twiddle factors = N-1 = 511

Number of butterflies that need real twiddle factors= 0

Number of butterflies that need complex twiddle factors= 2304-511=1793

5. (a) DIT-FFT Butterfly Structure (6 marks)



Solution: Butterfly structure for DIFFFT:

The input / output relations are

$$A'_R + j\,A'_I = A_R + j\,A_I + B_R + j\,B_I$$
$$B'_R + j\,B'_I = (A_R + j\,A_I - (B_R + j\,B_I))(W_R^r + j\,W_I^r) \quad \text{P6.2.1}$$

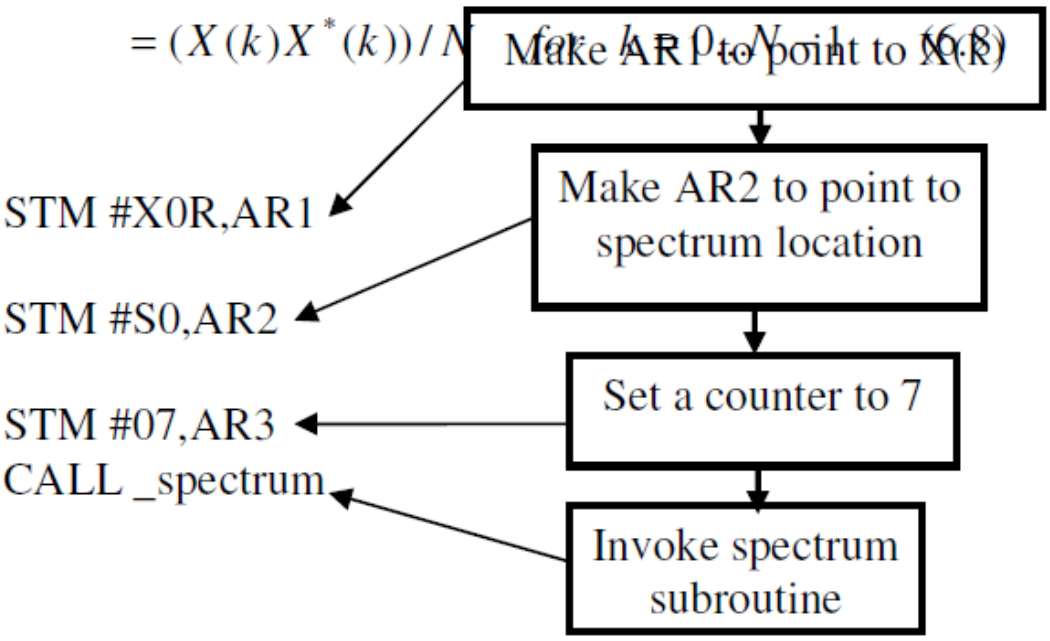
Separating the real and imaginary parts,

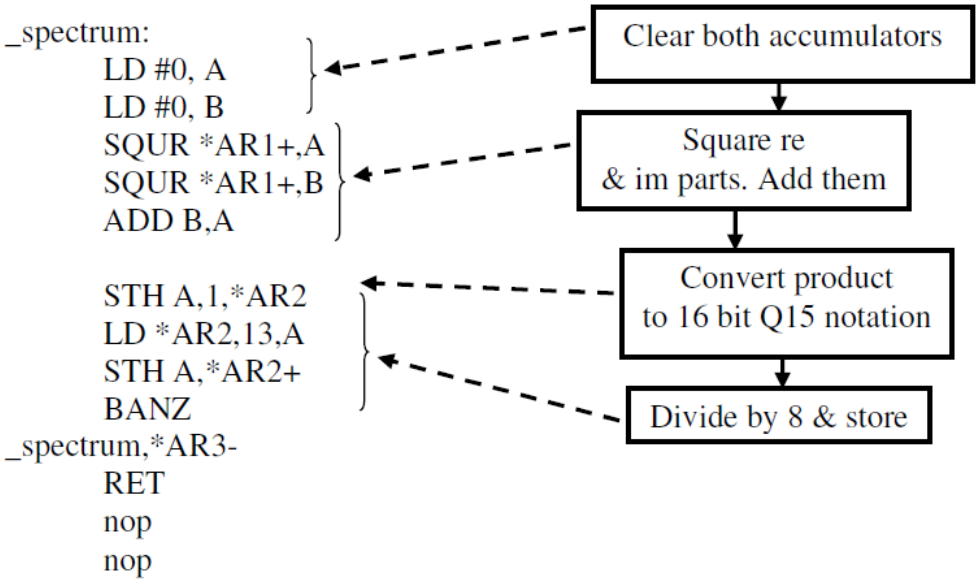
$$\therefore\ A'_R = A_R + B_R \quad \& \quad A'_I = A_I + B_I$$
$$B'_R = (A_R - B_R)W_R^r - (A_I - B_I)W_I^r$$
$$B'_I = (A_R - B_R)W_I^r + (A_I - B_I)W_R^r \quad \text{P6.2.2}$$

(b) Subroutine Program to find the Spectrum of the transformed data in DIT-FFT Algorithm: (4 marks)

```
; 8 locations for Spectrum
S0      .word  0        ;Frequency content at 0
S1      .word  0        ;Frequency content at fs/8
S2      .word  0        ;Frequency content at 2fs/8
S3      .word  0        ;Frequency content at 3fs/8
S4      .word  0        ;Frequency content at 4fs/
S5      .word  0        ;Frequency content at 5fs/8
S6      .word  0        ;Frequency content at 6fs/8
S7      .word  0        ;Frequency content at 7fs/8
```

$$S(k) = (X^2(k))/N$$
$$= (X(k)X^*(k))/N \quad \text{for } k = 0, N-1 \qquad (6.8)$$





6. Overflow and Scaling:

$$\sin\theta = \frac{(A_R - B_R)}{\sqrt{(A_R - B_R)^2 + (A_I - B_I)^2}}$$

$$\& \quad \cos\theta = \frac{(A_I - B_I)}{\sqrt{(A_R - B_R)^2 + (A_I - B_I)^2}}$$

$$\therefore \quad B'_{I,\max} = \sqrt{(A_R - B_R) + (A_I - B_I)}$$

$$= \sqrt{2}$$

P6.5.3

Thus scaling factor is 0.707. To achieve multiplication by right shift, it is chosen as 0.5.

7.