

JET PROPULSION LABORATORY

INTEROFFICE MEMO

MJS: 2.64A

TO: MJS FDS Distribution 7 October 1974  
FROM: J. Wooddell  
SUBJECT: MJS FDS Processor Architecture and Instruction Set  
SUMMARY:

The architecture and instruction set of the MJS FDS Processor are described in the following pages.

I. PROCESSOR ARCHITECTURE (Figure 1)

A. General

The FDS processor is a general purpose type machine, but is functionally oriented toward FDS tasks. The major differences from other general purpose machines are:

1. There is no indirect addressing or address indexing in the usual sense. The autoindexed memory reference instructions do provide a similar function.
2. There are more special purpose registers (128) than one would expect in a small general purpose computer. The large number of registers allows the large number of relatively independent subroutines to be cycled with a minimal amount of moving data back and forth between registers and general memory.
3. The shift and rotate instructions are more powerful than those normally found in small computers. This is because one of the most time consuming FDS functions is that of packing and formatting data.

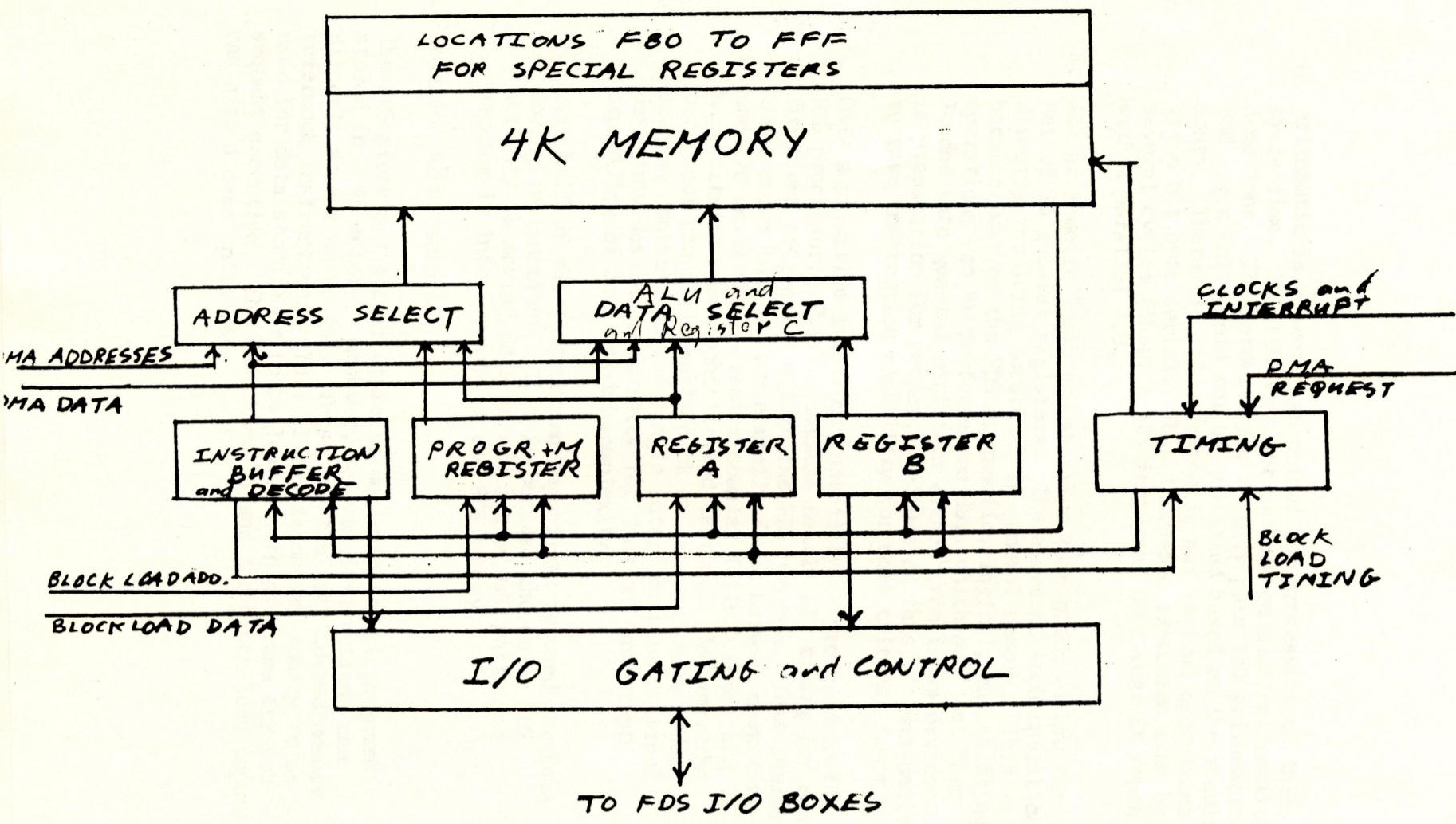


FIGURE 1  
FDS PROCESSOR BLOCK DIAGRAM

4. Arithmetic is slower than normal for processors of this cycle time. The FDS does not perform complex mathematical functions. The hardware for a fully parallel arithmetic and logic unit could not be justified based on the requirements. Therefore, all arithmetic and logical operations are 4 bit byte serial. This increased execution time by several cycles (about a 50% increase) over what it would be with a parallel ALU.
5. All arithmetic logic and shifting take place within the set of 16 general registers. There are no such operations directly involving locations in general memory. This is because most of the FDS arithmetic, logical, and shifting operations can be performed on data which has just been loaded into general registers either from input devices or in preparation for output. Because of this, it was decided to save instruction capability for more critical functions.
6. Only a primitive interrupt capability is provided in the FDS processor. This is because nearly all timing for control functions originates within the FDS program. This avoids the complex interaction normally found between computers and I/O units which creates complex failure modes and necessitates more sophisticated testing. However, the processor can efficiently pick up external data under program control, so the single interrupt plus external data provides the capability for "add on" interrupt capability of any desired complexity.
7. Over half of the instruction capability is used for data moving instructions. This is because the major FDS activity is moving data in and out of I/O units and packing it into formats in the FDS memory.

#### B. Detailed Architecture

The FDS processor architecture is designed to run programs stored in a 4K memory. However, several instructions can directly access up to 16K. These are the autoindexed memory reference instructions. This allows secondary memory to be used for data storage or to be loaded with programs for subsequent execution. The DMA channels and the block load inputs can also access up to 16K.

The processor consists of:

1. A 16-bit instruction buffer register (IB).
2. A 12-bit program address register (PR).
3. Two 16-bit working registers ( $R_A$  and  $R_B$ ).
4. Memory input data selection gating (16 bits wide).
5. Memory address selection gating (14 bits wide).
6. Provision for block loading memory from CCS.
7. Logic to perform arithmetic, logical and shifting operations.
8. I/O: Provision is made for input and output from a 4 channel direct memory access module, 32 serial data inputs, 16 serial data outputs, 8 parallel data inputs, 10 parallel data outputs, and 16 discrete outputs to be associated with a 5-bit subcode for 512 possible discretes.
9. Timing: An 806.4 kHz clock is used for all processor sequencing. 403.2 kHz is the maximum memory access rate. All instruction times are listed in cycles of 403.2 kHz (2.48 microseconds).
10. Interrupt: A provision is made for forcing the program register to an all zero state with an external signal. This provides a means for periodic re-initialization of programs for error recovery and allows some relaxation of fine timing requirements for the programs.
11. Special purpose registers: There are 16 general registers (GR1-GR16), 32 memory pointers (MP1-MP32), 8 index registers (IR1-IR8), and a line count register. GR1, MP1, and IR1 refer to the same register. All others are distinct. An additional 73 registers are available for use as counters. These registers reside in the top 128 memory locations.

J. Wooddell

JLW:cj

JP →

PRESENT EXECUTION TIME REQUIREMENT SHOWING NEED FOR DMA

- SIGNALS TO INSTRUMENTS AND DATA MANIPULATION       $\approx 48,000$  INSTRUCTIONS PER SECOND WITH NO MARGIN
- DATA I/O THROUGH DMA       $\approx 19,800$  WORDS PER SECOND      *316,800 bps*
- PRESENT EXECUTION RATE       $\approx 80,000$  INSTRUCTIONS PER SECOND
- REQUIRED RATE IF EACH DATA I/O WORD TAKES TWO INSTRUCTIONS       $\approx 87,600$  INSTRUCTIONS PER SECOND

JW  
11/19/74

jp →

CMOS PROCESSOR #1  
(FIRST BREADBOARD)

TECHNOLOGY: CMOS LOGIC  
PLATED WIRE MEMORY (CCS?)

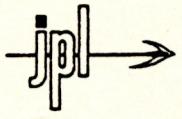
INSTRUCTION SET: SAME AS FOR 54L MODEL

EXECUTION TIME:  
(IN MICROSECONDS)      ADD      5.78  
                            JUMP      5.78  
                            LONG SHIFT 11.57

ARCHITECTURE: SAME AS FOR 54L MODEL PLUS  
HARDWARE INTERRUPT

MAXIMUM CLOCK RATE ≈ 1200 KHZ

JW  
11-19-74



CMOS PROCESSOR #2

TECHNOLOGY: CMOS LOGIC  
CMOS MEMORY

INSTRUCTION SET: 36 INSTRUCTIONS = THE ORIGINAL  
SET PLUS BRANCHING CAPABILITY

EXECUTION TIME:  
(IN MICROSECONDS)

ADD	14.88
JUMP	4.96
LONG SHIFT	22.32

JW  
11-19-74

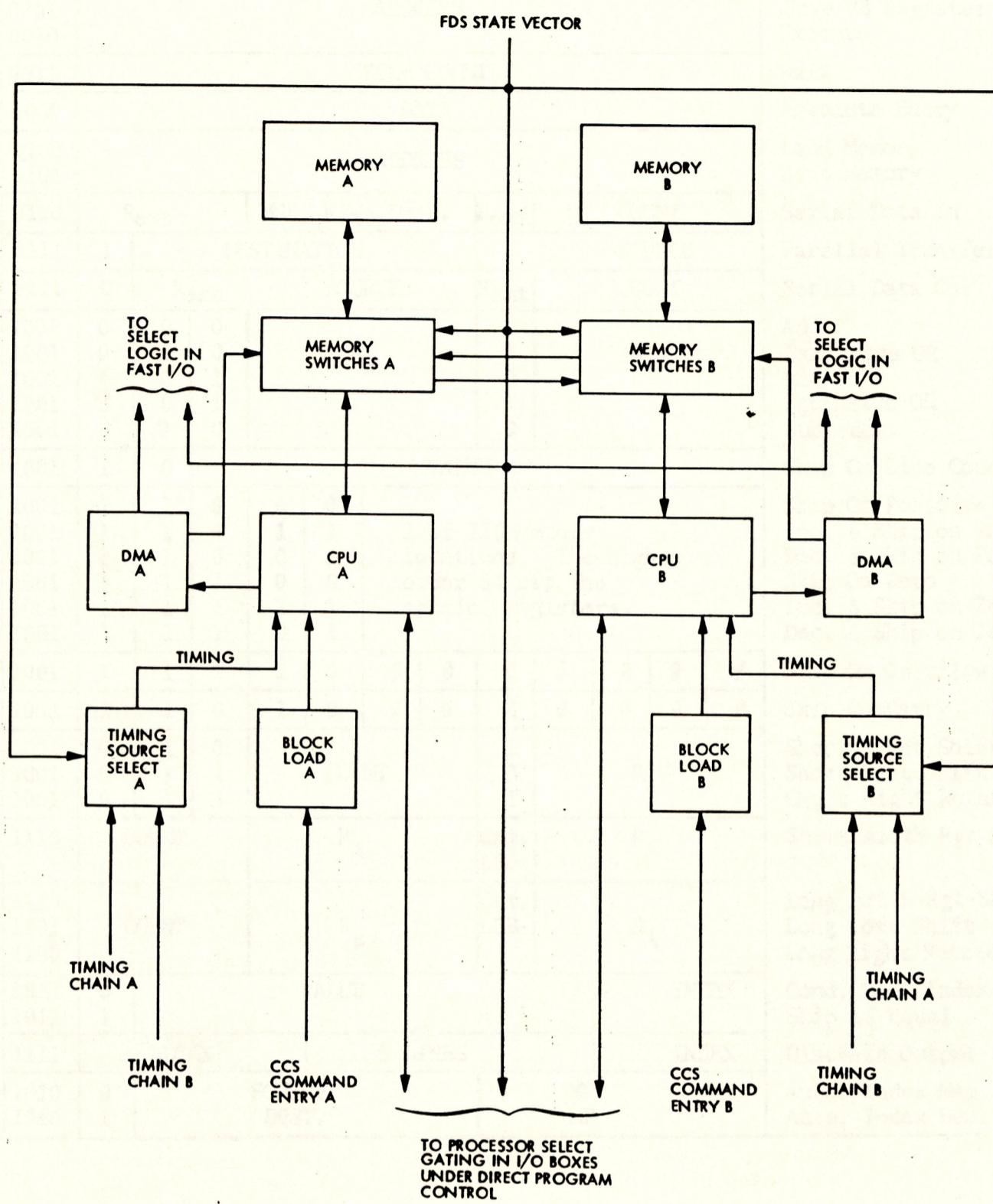


Figure 3. FDS Processor Functional Block Diagram

U-20-14  
FIGURE 2  
FDS INSTRUCTION SET

CLES	BITS 16-13													FUNCTION				
	12	11	10	9	8	7	6	5	4	3	2	1						
2	JMP	0000	ADDRESS															
3	SRB	0001																
2	EXC	0010																
4097	WAT	0011	CYCLE COUNT															
3	AES	1000	DATA															
4	MLD	0100	ADDRESS															
4	MRD	0101																
11	SWI	0110	Rext		MSB, Rext Dest.			Rext	COUNT									
4	PWD	0111	1	DESTINATION				SOURCE										
11	SWO	0111	0	Rext		SOURCE			Rext	COUNT								
6	ADD	1001	0	0	0	$R_B$				0	$R_A$							
6	LXR	1001	0	0	0					1	Add							
6	AND	1001	0	0	1					0	Exclusive OR							
6	LOR	1001	0	0	1					1	And							
6	SUB	1001	0	1	0					0	Inclusive OR							
or 6	SLC	1001	1	0	VALUE										Skip			
3	SKP	1001	1	1	0	0	0	0	1 of 128 Memory locations. The high order 64 are the special registers.						On Line Count			
4	ISP	1001	1	1	0	0	1	1							Skip On Positive			
4	DSP	1001	1	1	0	0	0	1							Inc. & Skip on Pos.			
3	SKZ	1001	1	1	1	0	0	0							Dec. & Skip on Pos.			
4	ISZ	1001	1	1	1	1	1	1							Skip On Zero			
4	DSZ	1001	1	1	1	0	1	1							Inc. & Skip on Zero			
3	SKO	1001	1	1	1	1	0	0	0	0	0	0	0	0	Skip On Zero			
3	SKC	1001	1	1	0	1	0	0	0	0	0	0	0	0	Skip On Carry			
6	SRS	1001	0	1	0	COUNT			1	$R_A$					Short Right Shift			
6	SLS	1001	0	1	1				0						Short Left Shift			
6	SRR	1001	0	1	1				1						Short Right Rotate			
6	ARS	1110	COUNT			$R_A$			Cnt. LSB	$R_A$					Short Arith Rgt Shift			
9	LRS	1110	COUNT			$R_B$			Cnt. LSB	$R_A$					Long Arith Rgt Shift			
9	LLS	1101													Long Left Shift			
9	LRR	1100													Long Right Rotate			
4	MCX	1011	0	VALUE								INDEX			Cond. Mod. Index			
3	SKE	1011	1												Skip if Equal			
3	OUT	1111	BLOCK			5 LINES			INDEX						Discrete Output			
6	AML	1010	0	SOURCE				MP				INDEX			Auto. Index Mem. Load			
6	AMR	1010	1	DEST.				MP							Auto. Index Mem. Read			

## FDS MEMORY MAP

