INTERSIL

IM6100 CMOS MICROPROCESSOR MEMORY EXTENSION

INTERSIL CMOS/LSI

APPLICATION BULLETIN
M007

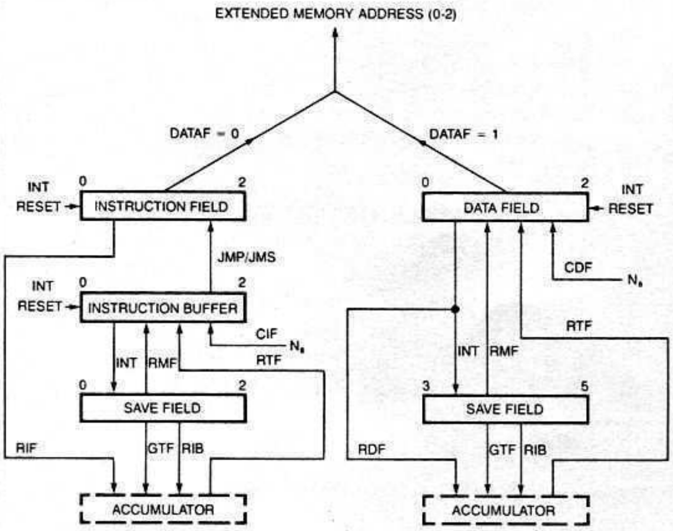
INTRODUCTION

The IM6100, being a 12-bit machine, has a basic addressing capacity of 4096 words of memory. The addressing space can be extended by external logic. This paper discusses a memory extension controller which is program compatible with the DEC PDP-8/E KM8-E Memory Extension option. The controller extends the addressing capability of the IM6100 from 4096 words to 32768 words. The module generates a 3-bit extended address which is decoded to select 1 of 8 memory modules, each of which provides 4096 words of memory. The main memory is divided into 4K fields, starting with field 0, for the basic 4K memory, up to field 7, when 32K of memory is used. The three memory extension bits augment the 12-bit addresses generated by the microprocessor.

MEMORY EXTENSION CONTROLLER

A simplified block diagram of the Memory Extension Controller is represented in Figure 1. The diagram shows two 3-bit field registers: the Instruction Field, which acts as an extension to the instruction and directly obtained operand addresses and the Data Field, which augments indirectly obtained operand addresses. The program can, therefore, use one field for instructions and address pointers and another field for data. The selection between Instruction and Data Fields is controlled by the DATAF signal generated by the IM6100. A discussion of the various register functions follows.

FIGURE 1



INSTRUCTION FIELD REGISTER (IF)

The IF is a 3-bit register that serves as an extension of the Program Counter (PC). The IF, however, is not incremented when the PC goes from 7777_8 to 0000_8. The contents of the IF determine the field from which all instructions are taken. Operands for all directly addressed memory reference instructions also come from the Instruction Field. The indirect pointer for all indirectly addressed memory reference instructions reside in the Instruction Field.

DATA FIELD REGISTER (DF)

The DF is a 3-bit register which determines the memory field from which operands are fetched in indirectly addressed AND, TAD, ISZ or DCA instructions. However, the branch address for indirectly addressed JMS or JMP instructions is obtained from the Instruction Field. The Data Field register may be modified under program control.

INSTRUCTION BUFFER REGISTER (IB)

The IB is a 3-bit register which serves as an input buffer for the Instruction Field (IF) register. All programmed modifications of the IF register are made through the IB register. The transfer from IB to IF takes place after the "next" JMP or JMS instruction. Using this feature, a program segment can execute an instruction to modify the IF and then "exit" the program segment before the actual modification of the IF takes place. If instructions could change the IF directly, it would be impossible to execute the "next" sequential instruction, following a Change IF instruction.

SAVE FIELD REGISTER (SF)

The SF is a 6-bit register in which the IB and DF registers are saved during an Interrupt Grant. When an Interrupt occurs, the contents of IB and DF are automatically stored in SF (0-2) and SF (3-5), respectively, and the IF, IB and DF registers are cleared. The INTGNT (Interrupt Grant) cycle stores the "current" Program Counter (PC) in location 0000_8 of Memory Field 0_8 and the CPU resumes operation in location 0001_8 of Memory Field 0_8. The Instruction Field and Data Field of the program segment, being executed by the CPU before the interrupt was acknowledged, are available in the SF register.

INTERRUPT INHIBIT FLIP-FLOP

The INTREQ (Interrupt Request) line to the IM6100 must be "gated" by the Interrupt Inhibit Flip-Flop so that, when the Instruction Field is changed under program control, all interrupts are disabled until a JMP or JMS instruction is executed. Since the actual modification of the Instruction Field takes place only after the "next" JMP/JMS, this inhibition of the INTREQ's ensures that the program sequence resumes operation in the "new" memory field before an Interrupt Request is granted.

Since Interrupt Requests are asynchronous in nature, a situation may arise in which an INTREQ is generated when the IF and IB bits are different. The Interrupt Inhibit FF guarantees the structural integrity of the program segment.

OPERAND FETCHING

Instructions are accessed from the currently assigned Instruction Field. For indirect AND, TAD, ISZ or DCA instruction, the operand address refers first to the Instruction Field to obtain an Effective Address which in turn refers to a location in the currently addressed Data Field.

Thus, DF is active only in the Execute phase of an AND, TAD, ISZ or DCA when it is directly preceded by an Indirect phase.

ADDRESS MODE	IF	DF	AND, TAD, ISZ or DCA
Direct	m	n	Operand in field m
Indirect	m	n	Absolute address of operand in field m; operand in field n

INSTRUCTIONS

Table 1 lists the mnemonics for the Extended Memory Control, their OPCODE and the operations performed.

TABLE 1

MNEMONIC	OCTAL CODE	OPERATION	
GTF	6004_8	GET FLAGS	
		Operation:	AC (0) ← LINK AC (2) ← INTREQ Line AC (3) ← INT INHIBIT FF AC (4) ← INT ENABLE FF AC (6-11) ← SF (0-5)
		Description:	LINK, INTREQ and INT ENABLE FF are internal to the CPU. The INT INHIBIT FF and SF are in the Memory Extension Controller.
		Implementation:	The CPU accepts the device data available on DX (0-11) and then bits 0, 2 and 4 are modified by the respective internal flags before the data is loaded into the Accumlator (AC). The Memory Extension Controller must drive the C-lines (CO=L, C1=L). AC(1) and AC(5) are determined externally.
RTF	6005_8	RETURN FLAGS	

		Operation:	LINK ← AC (0)
			IB ← AC (6-8)
			DF ← AC (9-11)

		Description:	LINK is restored. All AC bits are available externally during IOTA T6 to restore other flag bits. The internal Interrupt System is enabled. However, the Interrupt Inhibit FF is made active until the "next" JMS/JMP. The IB is transferred to IF after the "next" JMS/JMP.
CDF	$62N1_8$	CHANGE DATA FIELD	
		Operation:	DF ← N_8
		Description:	Change DF register to N (0_8-7_8).
CIF	$62N2_8$	CHANGE INSTRUCTION FIELD	
		Operation:	IB ← N_8
		Description:	Change IB to N (0_8-7_8). Transfer IB to IF after the "next" JMP/JMS. The Interrupt Inhibit FF is active until the "next" JMP/JMS.
CDF, CIF	$62N3_8$	CHANGE DF, IF	

		Operation:	DF ← N_8
			IB ← N_8

		Description:	Combination of CDF and CIF.
RDF	6214_8	READ DATA FIELD	
		Operation:	AC (6-8) ← AC (6-8) V DF
		Description:	OR's the contents of DF into bits 6-8 of the AC. All other bits are unaffected.
		Implementation:	DataX (0-5) and DataX (9-11) must be 0's. Drive C1=L.

Table 1, Cont.

MNEMONIC	OCTAL CODE	OPERATION
RIF	6224_8	READ INSTRUCTION FIELD Operation: AC (6-8) ← AC (6-8) V IF Description: OR's the contents of IF into bits 6-8 of the AC. All other bits of the AC are unaffected. Implementation: Same as RDF
RIB	6234_8	READ INTERRUPT BUFFER READ SAVE FIELD Operation: AC (6-11) ← AC (6-11) V SF Description: OR's the contents of SF into bits 6-11 of the AC. All other bits of the AC are unaffected. Implementation: Same as RDF, RIF
RMF	6244_8	RESTORE MEMORY FIELD Operation: IB ← SF (0-2) DF ← SF (3-5) Description: The SF register saves the contents of the IB and DF when an interrupt occurs. This command is used to restore IB and DF when "exiting" from the interrupt service routine in another field. Transfer IB to IF after the next JMP/JMS. The Interrupt Inhibit FF is active until the next JMP/JMS.

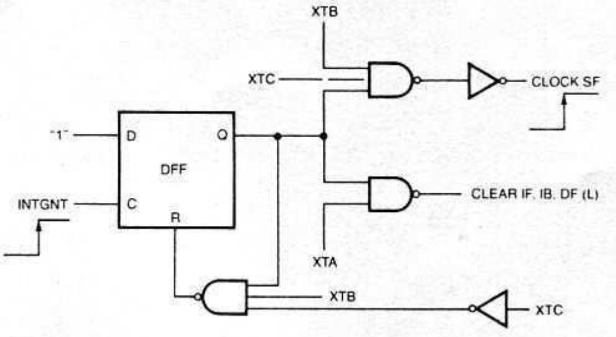
IM6100 SIGNALS

The IM6100 provides all of the signals required to implement the Memory Extension Controller.

INTGNT (Interrupt Grant)—Pin 39

If the interrupt system is enabled and there are no higher priority requests to the IM6100 pending, for example, RESET, CPREQ, RUN/HLT or DMAREQ, the IM6100 grants an INTREQ by activating the INTGNT signals at the end of the current instruction. The positive transition of this signal can be used to load the Save Field (SF) register with Instruction Buffer (IB) register and Data Field (DF) register and then to clear IF, IB and DF.

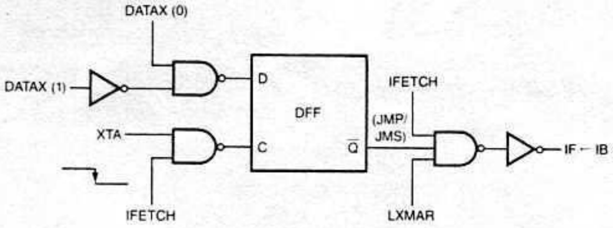
The INTGNT signal is reset by any IOT instruction.



DATAF (Data Field select)—Pin 40

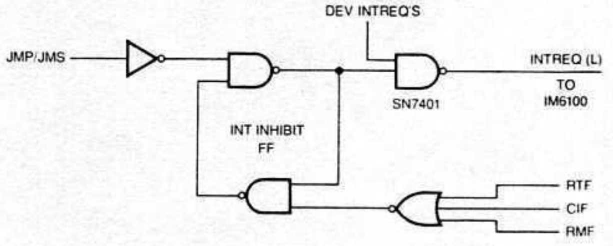
The DATAF is high in the Execute phase of an AND, TAD, ISZ or DCA when it is directly preceded by an Indirect phase.

When accessing memory for operands in indirectly addressed AND, TAD, ISZ or DCA, the DATAF selects the DF instead of the IF as the extended memory address.



IFETCH (Instruction Fetch phase)—Pin 36

The IFETCH signal is active high during an Instruction Fetch cycle. The Memory Extension Controller can monitor the DX (0-2) lines during IFETCH•XTA to decode JMP/JMS. JMP/JMS transfers IB to IF and resets the Interrupt Inhibit FF.



CONCLUSION

The memory extension controller that we have discussed in this bulletin shows three important design considerations involved in extending memory addressing space. The first is the concept of having separate instruction and data fields for program flexibility. The second is the importance of double buffering the instruction field register to maintain structural integrity of programs and the third is the provision for saving the current field status upon interrupts and disabling interrupts until a change of instruction field has been completely executed.

The length of the field registers is limited only by the PDP-8 instructions that manipulate the contents of the field registers. For example, the instruction CDF-$62N1_8$ (Change Data Field to N_8) provides for only 3 bits (N_8) for the Data Field. The user, of course, has the option to have any subset or superset of these features for specific implementations. However, it is recommended that the memory extension controller adhere to the KM8-E architecture to execute DEC software properly, especially, if the application does not require more than 32K of memory.