SEMICONDUCTORS

MOTOROLA

3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721

MICROPROCESSOR WITH CLOCK AND OPTIONAL RAM

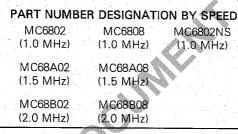
The MC6802 is a monolithic 8-bit microprocessor that contains all the registers and accumulators of the present MC6800 plus an internal clock oscillator and driver on the same chip. In addition, the MC6802 has 128 bytes of on-board RAM located at hex addresses \$0000 to \$007F. The first 32 bytes of RAM, at hex addresses \$0000 to \$001F, may be retained in a low power mode by utilizing VCC standby; thus, facilitating memory retention during a power-down situation.

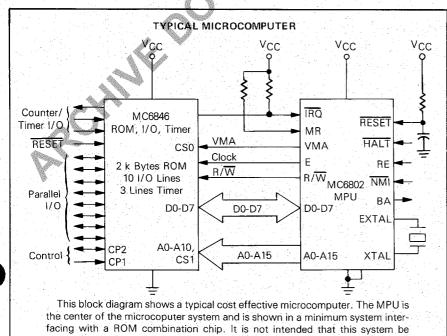
The MC6802 is completely software compatible with the MC6800 as well as the entire M6800 family of parts. Hence, the MC6802 is expandable to 64K words.

The MC6802NS is identical to the MC6802 without standby RAM feature. The MC6808 is identical to the MC6802 without on-board RAM.

- On-Chip Clock Circuit
- 128×8 Bit On-Chip RAM
- 32 Bytes of RAM are Retainable
- Software-Compatible with the MC6800
- Expandable to 64K Words
- Standard TTL-Compatible Inputs and Outputs
- 8-Bit Word Size
- 16-Bit Memory Addressing
- Interrupt Capability

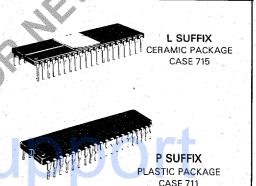
Microcomputer family.





limited to this function but that it be expandable with other parts in the M6800

MC6802 MC6808 MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS MC6802NS



V _{SS}	40 D RESET
HALT 2	39 EXTAL
MR [3	38 XTAL
IRQ 4	37 1 E
VMA L 5	36] RE**
	35 D V _{CC} Standby*
BA [7	34 🛛 R/W
V _{CC} I 8	33 D D0
A0 [9	32] D1
A1 [] 10	31] D2
A2 [11	30 D 3
A3 [12	29 D 4
A4 [13	28] D5
A5 [14	27 D 6
A6 [15	26 D 7
A7 [16	25 D A15
A8 [17	24 🗖 A14
A9 [18	23 🛛 A13
A10 [19	22 🗖 A12
A11 D 20	21 VSS

©MOTOROLA INC., 1981

Rating	Symbol	Value	Unit
Supply Voltage	Vcc	-0.3 to +7.0	V
Input Voltage	Vin	-0.3 to +7.0	V
Operating Temperature Range	ТА	0 to +70	°C
Storage Temperature Range	Tstg	- 55 to + 150	°C

This input contains circuitry to protect the inputs against damage due to high static voltages or electric fields; however, it is advised that normal precautions be taken to avoid application of any voltage higher than maximum rated voltages to this high-impedance circuit. Reliability of operation is enhanced if unused inputs are tied to an appropriate logic voltage level (e.g., either VSS or VCC).

(1)

(2)

(3)

THERMAL CHARACTERISTICS

1	Characteristic	Symbol Value	Unit
	Average Thermal Resistance (Junction to Ambient)		
2	Plastic	100	<u></u>
1. No.	Ceramic	[#] JA 50	C/W

POWER CONSIDERATIONS

The average chip-junction temperature, Tj, in °C can be obtained from:

 $\mathsf{T}_\mathsf{J} = \mathsf{T}_\mathsf{A} + (\mathsf{P}_\mathsf{D} \bullet \theta_\mathsf{J}_\mathsf{A})$

Where:

T_A = Ambient Temperature, °C

 $\theta_{JA} \equiv Package$. Thermal Resistance, Junction-to-Ambient, °C/W

PD≡PINT+PPORT

PINT≡ICC×VCC, Watts – Chip Internal Power

PPORT = Port Power Dissipation, Watts - User Determined

An approximate relationship between PD and TJ (if PPORT is neglected) is:

 $P_{D} = K + (T_{J} + 273 °C)$

Solving equations 1 and 2 for K gives:

 $K = P_{D} \bullet (T_{A} + 273 \,^{\circ}C) + \theta_{JA} \bullet P_{D}^{2}$

Where K is a constant pertaining to the particular part. K can be determined from equation 3 by measuring PD (at equilibrium) for a known T_A. Using this value of K the values of P_D and T_J can be obtained by solving equations (1) and (2) iteratively for any value of T_A.







OPERATING TEMPERATURE RANGE

Device	Speed	Symbol	Value Uni
MC6802P,L	(1.0 MHz)	TA	0 to + 70
MC6802CP,CL	(1.0 MHz)		- 40 to + 85
MC68A02P,L	(1.5 MHz)	Тд	0 to +70
MC68A02CP,CL	(1.5 MHz)		-40 to +85 °C
MC68B02P,L	(2.0 MHz)	ТА	0 to +70
MC68B02CP,CL	(2.0 MHz)		-40 to +85
MC6802NSP,L	(1.0 MHz)	TA	0 to + 70 •C
MC6808P,L MC68A08P,L MC68B08P,L	(1.0 MHz) (1.5 MHz) (2.0 MHz)	Тд	0 to +70 °C

DC ELECTRICAL CHARACTERISTICS (V_{CC}=5.0 Vdc ±5%, V_{SS}=0, T_A=0 to 70°C, unless otherwise noted)

Characteristic	Symbol	🕨 Min	Тур	Max	Unit
Input High Voltage Logic, EXT	V3	V _{SS} +2.0 V _{SS} +4.0		Vcc * Vcc	V
Input Low Voltage Logic, EXTAL, RES	SET VIL	V _{SS} -0.3		V _{SS} +0.8	V
Input Leakage Current (Vin = 0 to 5.25 V, V _{CC} = max)	ogic lin	_	1.0	2.5	μA
$(I_{Load} = -145 \mu\text{A}, V_{CC} = \text{min})$ A0-A15, R/W, VMA	-D7 A, E BA	V _{SS} +2.4 V _{SS} +2.4 V _{SS} +2.4			V
Output Low Voltage (ILoad = 1.6 mA, VCC = min)	VOL		-	V _{SS} +0.4	V
Internal Power Dissipation (Measured at TA=0°C)	PINT	·	0.600	1.0	W
V _{CC} Standby Power Do	1 000	4.0 4.75	-	5.25 5.25	V
Standby Current	ISBB		_	8.0	mΑ
Logic Inputs, EXT			10 6.5	12.5 10	рF
A0-A15, R/W, V	MA Cout		-	12	рF

*In power-down mode, maximum power dissipation is less than 42 mW. #Capacitances are periodically sampled rather than 100% tested.

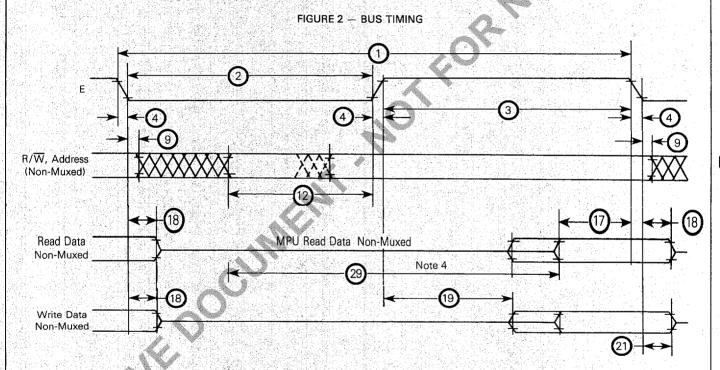
CONTROL TIMING $W_{CC} = 5.0 \text{ V} \pm 5\%$, $V_{SS} = 0$, $T_A = T_L$ to T_H , unless otherwise noted)

Characteristics	Symbol	MC68 MC6			8A02 8A08	MC6 MC6	8B02 8B08	Unit
		Min	Max	Min	Max	Min	Max	
Frequency of Operation	fo	0.1	1.0	0.1	1.5	0.1	2.0	MHz
Crystal Frequency	^f XTAL	1.0	4.0	1.0	6.0	1.0	8.0	MHz
External Oscillator Frequency	4xfo	0.4	4.0	0.4	6.0	0.4	8.0	MHz
Crystal Oscillator Start Up Time	trc	100		100	_	100	-	ms
Processor Controls (HALT, MR, RE, RESET, IRQ NMI)		· · ·						
Processor Control Setup Time	^t PCS	200		140	-	110	-	ns
Processor Control Rise and Fall Time (Does Not Apply to RESET)	^t PCr, tPCf	_	100	_	100	-	100	ns





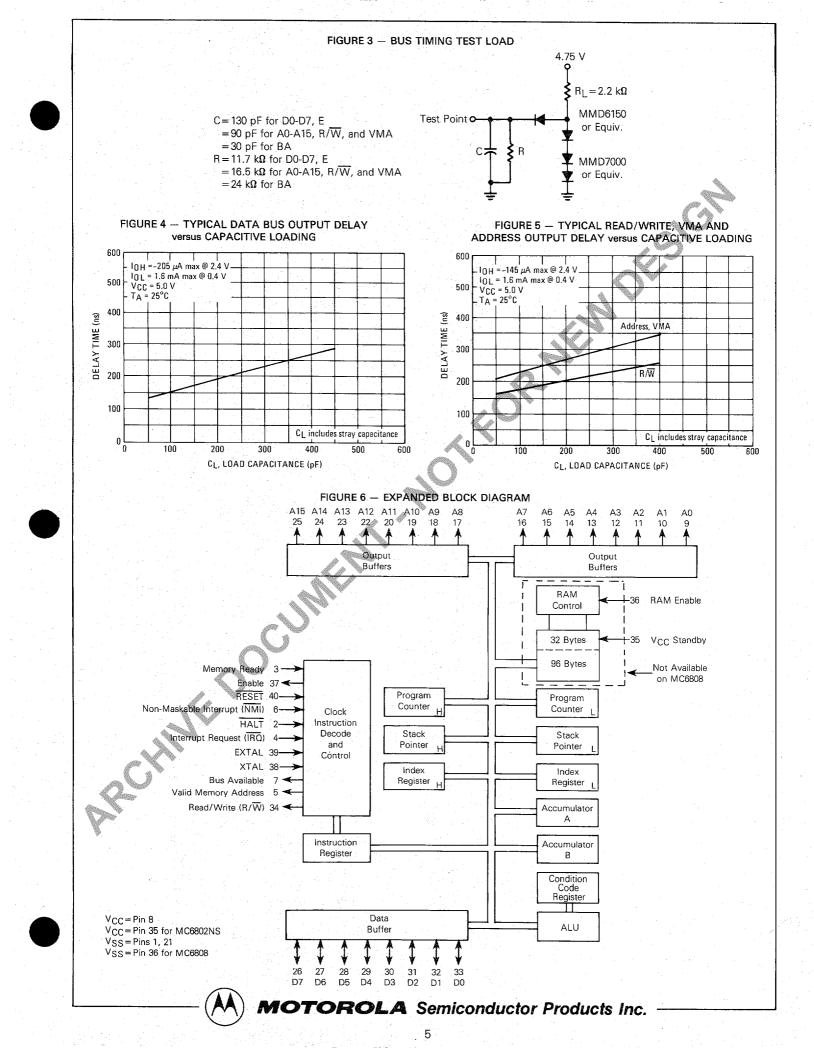
ldent. Number	Characteristic	Symbol	MC	302NS 6802 6808	1460167	8A02 8A08		8B02 8B08	Unit
			Min	Max	Min	Max	Min	Max	
1	Cycle Time	tcvc	1.0	10	0.667	10	0.5	10	μS
2	Pulse Width, E Low	PWEL	450	5000	280	5000	210	5000	ns
3	Pulse Width, E High	PWEH	450	9500	280	9700	220	9700	ns
4	Clock Rise and Fall Time	t _r , t _f		25		25	왕수 등	20	ns
9	Address Hold Time	tAH	20		- 20	1. 	20	_	ns
12	Non-Muxed Address Valid Time to E (See Note 5)	t _{AV1}	160 —	 270	100 —		50 —		ns
17	Read Data Setup Time	tDSR	100		70		60		ns
18	Read Data Hold Time	tDHB	10		10	/	10	-	ns
19	Write Data Delay Time	tDDW	-	225		170	4	160	ns,
21	Write Data Hold Time	tDHW	30	-	20 <		20	-	ns
29	Usable Access Time (See Note 4)	tACC	605	3 <u>14</u> (5	310	×	235		ns



NOTES:

- 1. Voltage levels shown are VL \leq 0.4 V, VH \geq 2.4 V, unless otherwise specified.
- 2. Measurement points shown are 0.8 V and 2.0 V, unless otherwise noted.
- 3. All electricals shown for the MC6802 apply to the MC6802NS and MC6808, unless otherwise noted.
- 4. Usable access time is computed by: 12+3+4-17.
- 5. If programs are not executed from on-board RAM, TAV1 applies. If programs are to be stored and executed from on-board RAM, TAV2 applies. For normal data storage in the on-board RAM, this extended delay does not apply. Programs cannot be executed from on-board RAM when using A and B parts (MC68A02, MC68A08, MC68B02, MC68B08). On-board RAM can be used for data storage with all parts.

MOTOROLA Semiconductor Products Inc.



MPU REGISTERS

A general block diagram of the MC6802 is shown in Figure 6. As shown, the number and configuration of the registers are the same as for the MC6800. The 128×8 -bit RAM* has been added to the basic MPU. The first 32 bytes can be retained during power-up and power-down conditions via the RE signal.

The MC6802NS is identical to the MC6802 except for the standby feature on the first 32 bytes of RAM. The standby feature does not exist on the MC6802NS and thus pin 35 must be tied to 5 V.

The MC6808 is identical to the MC6802 except for onboard RAM. Since the MC6808 does not have on-board RAM pin 36 must be tied to ground allowing the processor to utilize up to 64K bytes of external memory.

The MPU has three 16-bit registers and three 8-bit registers available for use by the programmer (Figure 7).

PROGRAM COUNTER

The program counter is a two byte (16-bit) register that points to the current program address.

STACK POINTER

The stack pointer is a two byte register that contains the address of the next available location in an external pushdown/pop-up stack. This stack is normally a random access read/write memory that may have any location (address) that is convenient. In those applications that require storage of information in the stack when power is lost, the stack must be non-volatile.

INDEX REGISTER

The index register is a two byte register that is used to store data or a 16-bit memory address for the indexed mode of memory addressing.

ACCUMULATORS

The MPU contains two 8-bit accumulators that are used to hold operands and results from an arithmetic logic unit (ALU).

CONDITION CODE REGISTER

The condition code register indicates the results of an Arithmetic Logic Unit operation: Negative (N), Zero (Z), Overflow (V), Carry from bit 7 (C), and Half Carry from bit 3 (H). These bits of the Condition Code Register are used as testable conditions for the conditional branch instructions. Bit 4 is the interrupt mask bit (I). The unused bits of the Condition Code Register (b6 and b7) are ones.

Figure 8 shows the order of saving the microprocessor status within the stack.

*If programs are not executed from on-board RAM, TAV1 applies. If programs are to be stored and executed from on-board RAM, TAV2 applies. For normal data storage in the on-board RAM, this extended delay does not apply. Programs cannot be executed from on-board RAM when using A and B parts (MC68A02, MC68A08, MC68B02, and MC68B08). On-board RAM can be used for data storage with all parts.

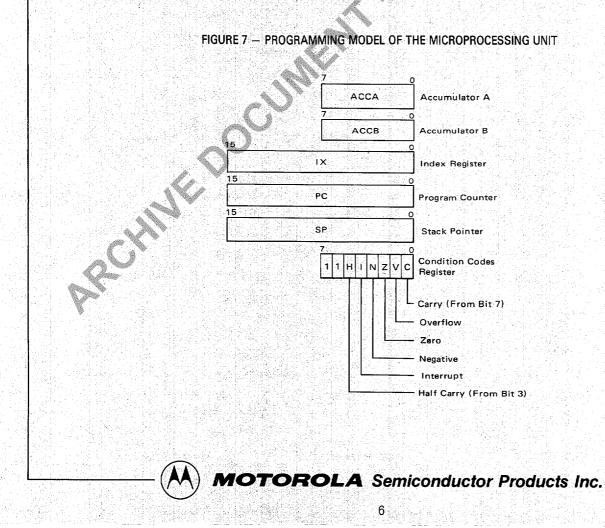
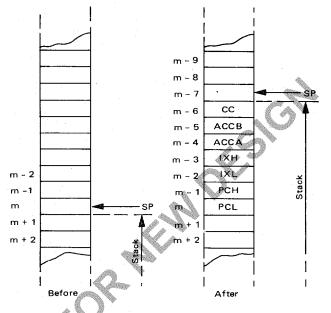






FIGURE 8 - SAVING THE STATUS OF THE MICROPROCESSOR IN THE STACK

SP = Stack Pointer CC = Condition Codes (Also called the Processor Status Byte) ACCB = Accumulator B ACCA = Accumulator A IXH = Index Register, Higher Order 8 Bits IXL = Index Register, Lower Order 8 Bits PCH = Program Counter, Higher Order 8 Bits PCL = Program Counter, Lower Order 8 Bits



MPU SIGNAL DESCRIPTION

Proper operation of the MPU requires that certain control and timing signals be provided to accomplish specific functions and that other signal lines be monitored to determine the state of the processor. These control and timing signals are similar to those of the MC6800 except that TSC, DBE, ϕ 1, ϕ 2 input, and two unused pins have been eliminated, and the following signal and timing lines have been added:

RAM Enable (RE) Crystal Connections EXTAL and XTA

Memory Ready (MR)

V_{CC} Standby

Enable ¢2 Output (E)

The following is a summary of the MPU signals:

ADDRESS BUS (A0-A15)

Sixteen pins are used for the address bus. The outputs are capable of driving one standard TTL load and 90 pF. These lines do not have three-state capability.

DATA BUS (D0-D7)

Eight pins are used for the data bus. It is bidirectional, transferring data to and from the memory and peripheral devices. It also has three-state output buffers capable of driving one standard TTL load and 130 pF.

Data bus will be in the output mode when the internal RAM is accessed and RE will be high. This prohibits external data entering the MPU. It should be noted that the internal RAM is fully decoded from \$0000 to \$007F. External RAM at \$0000 to \$007F must be disabled when internal RAM is accessed.

HALT

When this input is in the low state, all activity in the machine will be halted. This input is level sensitive. In the HALT mode, the machine will stop at the end of an instruc-

tion, bus available will be at a high state, valid memory address will be at a low state. The address bus will display the address of the next instruction.

To ensure single instruction operation, transition of the \overrightarrow{HALT} line must occur tPCS before the falling edge of E and the \overrightarrow{HALT} line must go high for one clock cycle.

HALT should be tied high if not used. This is good engineering design practice in general and necessary to ensure proper operation of the part.

READ/WRITE (R/W)

This TTL-compatible output signals the peripherals and memory devices whether the MPU is in a read (high) or write (low) state. The normal standby state of this signal is read

(high). When the processor is halted, it will be in the read state. This output is capable of driving one standard TTL load and 90 pF.

VALID MEMORY ADDRESS (VMA)

This output indicates to peripheral devices that there is a valid address on the address bus. In normal operation, this signal should be utilized for enabling peripheral interfaces such as the PIA and ACIA. This signal is not three-state. One standard TTL load and 90 pF may be directly driven by this active high signal.

BUS AVAILABLE (BA) — The bus available signal will normally be in the low state; when activated, it will go to the high state indicating that the microprocessor has stopped and that the address bus is available (but not in a three-state condition). This will occur if the HALT line is in the low state or the processor is in the WAIT state as a result of the execution of a WAIT instruction. At such time, all three-state output drivers will go to their off-state and other outputs to their normally inactive level. The processor is removed from the



WAIT state by the occurrence of a maskable (mask bit I = 0) or nonmaskable interrupt. This output is capable of driving one standard TTL load and 30 pF.

INTERRUPT REQUEST (IRQ)

A low level on this input requests that an interrupt sequence be generated within the machine. The processor will wait until it completes the current instruction that is being excuted before it recognizes the request. At that time, if the interrupt mask bit in the condition code register is not set, the machine will begin an interrupt sequence. The index register, program counter, accumulators, and condition code register are stored away on the stack. Next the MPU will respond to the interrupt request by setting the interrupt mask bit high so that no further interrupts may occur. At the end of the cycle, a 16-bit vectoring address which is located in memory locations \$FFF8 and \$FFF9 is loaded which causes the MPU to branch to an interrupt routine in memory.

The HALT line must be in the high state for interrupts to be serviced. Interrupts will be latched internally while HALT is low.

A nominal 3 k Ω pullup resistor to V_{CC} should be used for wire-OR and optimum control of interrupts. IRO may be tied directly to V_{CC} if not used.

RESET

This input is used to reset and start the MPU from a power-down condition, resulting from a power failure or an initial start-up of the processor. When this line is low, the MPU is inactive and the information in the registers will be lost. If a high level is detected on the input, this will signal the MPU to begin the restart sequence. This will start execu-

tion of a routine to initialize the processor from its reset condition. All the higher order address lines will be forced high. For the restart, the last two (\$FFFE, \$FFFF) locations in memory will be used to load the program that is addressed by the program counter. During the restart routine, the interrupt mask bit is set and must be reset before the MPU can be interrupted by IRO. Power-up and reset timing and powerdown sequences are shown in Figures 9 and 10, respectively.

RESET, when brought low, must be held low at least three clock cycles. This allows adequate time to respond internally to the reset. This is independent of the t_{rc} power-up reset that is required.

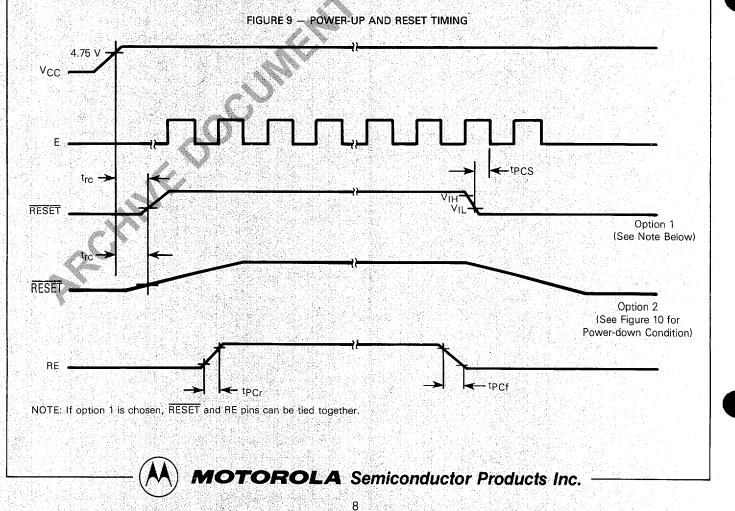
When RESET is released it *must* go through the low-tohigh threshold without bouncing, oscillating, or otherwise causing an erroneous reset (less than three clock cycles). This may cause improper MPU operation until the next valid reset.

NON-MASKABLE INTERRUPT (NMI)

A low-going edge on this input requests that a nonmaskable interrupt sequence be generated within the processor. As with the interrupt request signal, the processor will complete the current instruction that is being executed before it recognizes the NMI signal. The interrupt mask bit in the condition code register has no effect on NMI.

The index register, program counter, accumulators, and condition code registers are stored away on the stack. At the end of the cycle, a 16-bit vectoring address which is located in memory locations \$FFFC and \$FFFD is loaded causing the MPU to branch to an interrupt service routine in memory.

A nominal 3 kΩ pullup resistor to V_{CC} should be used for wire-OR and optimum control of interrupts. NMI may be tied





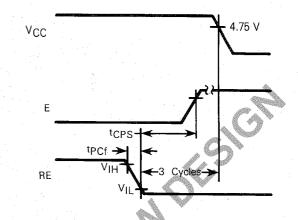
directly to V_{CC} if not used. Inputs \overline{IRQ} and \overline{NMI} are hardware interrupt lines that are sampled when E is high and will start the interrupt routine on a low E following the completion of an instruction.

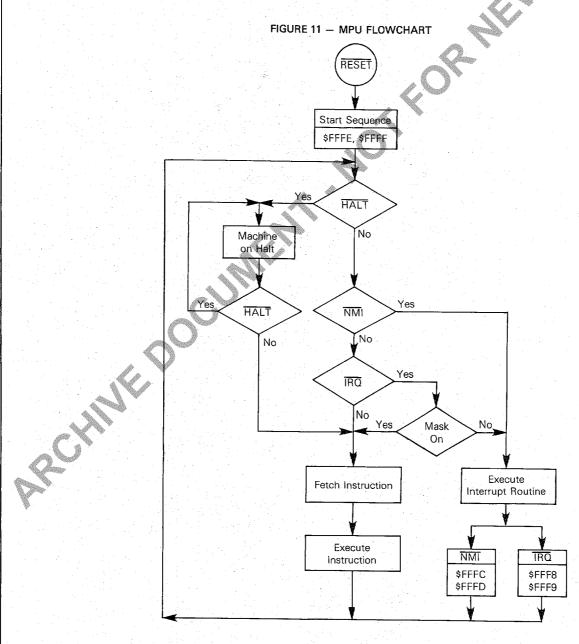
Figure 11 is a flowchart describing the major decision paths and interrupt vectors of the microprocessor. Table 1 gives the memory map for interrupt vectors.

TABLE 1 - MEMORY MAP FOR INTERRUPT VECTORS

Vec	ctor	0
MS	LS	Description
\$FFFE	\$FFFF	Restart
\$FFFC	\$FFFD	Non-Maskable Interrupt
\$FFFA	\$FFFB	Software Interrupt
\$FFF8	\$FFF9	Interrupt Request

FIGURE 10 - POWER-DOWN SEQUENCE





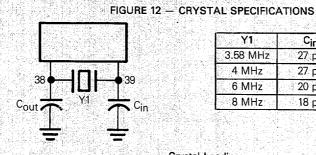






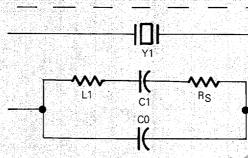
MOTOROLA Semiconductor Products Inc.

&C)



C _{in} 27.pF	Cout 27 pF		
	1996-1975-19 6 3 - 16 - 16 -		
27 pF	27 pF		
20 pF	20 pF		
18 pF	18 pF		
	한쪽을 걸려 있는다.		
	한 그렇는 말을 했다.		(Δ^{\ast})
isk para Sin Sina ang Angana ang ang ang ang ang ∎ang ang ang ang ang ang ang ang ang ang			
		la 🕬	, i
	- 2018 - 일종이		
	_ < </td <td></td> <td></td>		
1			

Crystal Loading



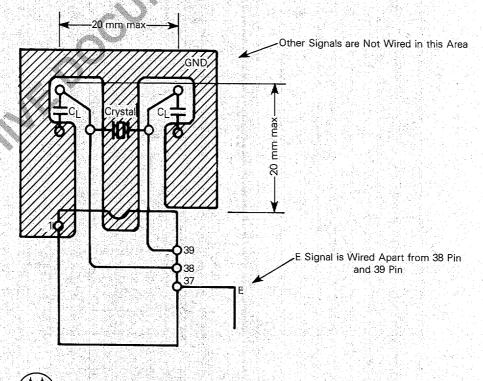
Nominal Crystal Parameters

		3.58 MHz	4.0 MHz 6.0 MHz 8.0 MH	lz
	RS	60 Ω	50 Ω 30-50 Ω 20-40 9	Ω
•	C0	3.5 pF	6.5 pF 4-6 pF 4-6 pF	-
	C1	0.015 pF	0.025 pF 0.01-0.02 pF 0.01-0.02	2 pF
1.1	Q	>40K	>30K >20K >20K	

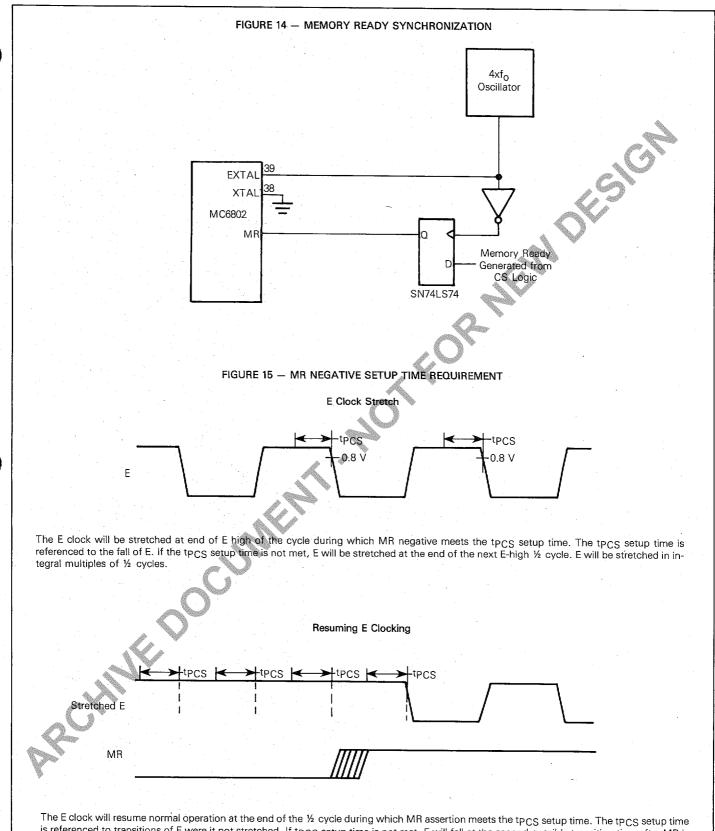
*These are representative AT-cut parallel resonance crystal parameters only. Crystals of other types of cuts may also be used.

Figure 13 - SUGGESTED PC BOARD LAYOUT

Example of Board Design Using the Crystal Oscillator



MOTOROLA Semiconductor Products Inc.



is referenced to transitions of E were it not stretched. If tPCS setup time is not met, E will fall at the second possible transition time after MR is asserted. There is no direct means of determining when the tPCS references occur, unless the synchronizing circuit of Figure 14 is used.



MOTOROLA Semiconductor Products Inc.

RAM ENABLE (RE - MC6802+MC6802NS ONLY)

A TTL-compatible RAM enable input controls the on-chip RAM of the MC6802. When placed in the high state, the onchip memory is enabled to respond to the MPU controls. In the low state, RAM is disabled. This pin may also be utilized to disable reading and writing the on-chip RAM during a power-down situation. RAM Enable must be low three cycles before V_{CC} goes below 4.75 V during power-down. RAM enable must be tied low on the MC6808. RE should be tied to the correct high or low state if not used.

EXTAL AND XTAL

These inputs are used for the internal oscillator that may be crystal controlled. These connections are for a parallel resonant fundamental crystal (see Figure 12). (AT-cut.) A divide-by-four circuit has been added so a 4 MHz crystal may be used in lieu of a 1 MHz crystal for a more cost-effective system. An example of the crystal circuit layout is shown in Figure 13. Pin 39 may be driven externally by a TTL input signal four times the required E clock frequency. Pin 38 is to be grounded.

An RC network is not directly usable as a frequency source on pins 38 and 39. An RC network type TTL or CMOS oscillator will work well as long as the TTL or CMOS output drives the on-chip oscillator.

LC networks are not recommended to be used in place of the crystal.

If an external clock is used, it may not be halted for more than $t_PW\phi_L$. The MC6802, MC6808 and MC6802NS are dynamic parts except for the internal RAM, and require the external clock to retain information.

MEMORY READY (MR)

MR is a TTL-compatible input signal controlling the stretching of E. Use of MR requires synchronization with the 4xf₀ signal, as shown in Figure 14. When MR is high, E will be in normal operation. When MR is low, E will be stretched integral numbers of half periods, thus allowing interface to slow memories. Memory Ready timing is shown in Figure 15.

MR should be tied high (connected directly to V_{CC}) if not used. This is necessary to ensure proper operation of the part. A maximum stretch is t_{CVC} .

ENABLE (E)

This pin supplies the clock for the MPU and the rest of the system. This is a single-phase, TTL-compatible clock, This clock may be conditioned by a memory read signal. This is equivalent to $\phi 2$ on the MC6800. This output is capable of driving one standard TTL load and 130 pF.

VCC STANDBY (MC6802 ONLY)

This pin supplies the dc voltage to the first 32 bytes of RAM as well as the RAM Enable (RE) control logic. Thus, retention of data in this portion of the RAM on a power-up, power-down, or standby condition is guaranteed. Maximum current drain at VSB maximum is ISBB. For the MC6802NS this pin must be connected to V_{CC}.

MPU INSTRUCTION SET

The instruction set has 72 different instructions. Included are binary and decimal arithmetic, logical, shift, rotate, load, store, conditional or unconditional branch, interrupt and stack manipulation instructions (Tables 2 through 6). The instruction set is the same as that for the MC6800.

MPU ADDRESSING MODES

There are seven address modes that can be used by a programmer, with the addressing mode a function of both the type of instruction and the coding within the instruction. A summary of the addressing modes for a particular instruction can be found in Table 7 along with the associated instruction execution time that is given in machine cycles. With a bus frequency of 1 MHz, these times would be microseconds.

ACCUMULATOR (ACCX) ADDRESSING

In accumulator only addressing, either accumulator A or accumulator B is specified. These are one-byte instructions.

IMMEDIATE ADDRESSING

In immediate addressing, the operand is contained in the second byte of the instruction except LDS and LDX which have the operand in the second and third bytes of the instruction. The MPU addresses this location when it fetches the immediate instruction for execution. These are two- or three-byte instructions.

DIRECT ADDRESSING

In direct addressing, the address of the operand is contained in the second byte of the instruction. Direct addressing allows the user to directly address the lowest 256 bytes in the machine, i.e., locations zero through 255. Enhanced execution times are achieved by storing data in these locations. In most configurations, it should be a random-access memory. These are two-byte instructions.

EXTENDED ADDRESSING

In extended addressing, the address contained in the second byte of the instruction is used as the higher eight bits of the address of the operand. The third byte of the instruction is used as the lower eight bits of the address for the operand. This is an absolute address in memory. These are three-byte instructions.

INDEXED ADDRESSING

In indexed addressing, the address contained in the second byte of the instruction is added to the index register's lowest eight bits in the MPU. The carry is then added to the higher order eight bits of the index register. This result is then used to address memory. The modified address is held in a temporary address register so there is no change to the index register. These are two-byte instructions.

MOTOROLA Semiconductor Products Inc.

IMPLIED ADDRESSING

In the implied addressing mode, the instruction gives the address (i.e., stack pointer, index register, etc.). These are one-byte instructions.

RELATIVE ADDRESSING

In relative addressing, the address contained in the second

byte of the instruction is added to the program counter's lowest eight bits plus two. The carry or borrow is then added to the high eight bits. This allows the user to address data within a range of - 125 to + 129 bytes of the present instruction. These are two-byte instructions.

TABLE 2 - MICROPROCESSOR INSTRUCTION SET - ALPHABETIC SEQUENCE

Add with Carry Add Logical And Arithmetic Shift Left Arithmetic Shift Right	CLF CLV CMI CON CPX
Branch if Carry Clear Branch if Carry Set Branch if Equal to Zero Branch if Greater or Equal Zero Branch if Greater than Zero Branch if Higher Bit Test Branch if Less or Equal Branch if Less than Zero Branch if Minus Branch if Minus Branch if Not Equal to Zero Branch if Plus Branch Always Branch to Subroutine Branch if Overflow Clear Branch if Overflow Set	DEC DES DEX EOF INC INS INS JSR LDA LDS LDS LDS LDS NEC NOI
Compare Accumulators Clear Carry Clear Interrupt Mask	OR/ PSF
CHINE DOC	
	Add Logical And Arithmetic Shift Left Arithmetic Shift Right Branch if Carry Clear Branch if Carry Set Branch if Equal to Zero Branch if Greater or Equal Zero Branch if Greater than Zero Branch if Higher Bit Test Branch if Less or Equal Branch if Less than Zero Branch if Minus Branch if Minus Branch if Minus Branch if Not Equal to Zero Branch if Plus Branch always Branch to Subroutine Branch if Overflow Clear Branch if Overflow Set Compare Accumulators Clear Carry

CLR	Clear
CLV	Clear Overflow
CMP	Compare
COM	Complement
CPX	Compare Index Register
DAA	Decimal Adjust
DEC	Decrement
DES	Decrement Stack Pointer
DEX	Decrement Index Register
EOR	Exclusive OR
INC	Increment
INS	Increment Stack Pointer
INX	Increment Index Register
JMP	Jump
JSR	Jump to Subroutine
LDA	Load Accumulator
LDS	Load Stack Pointer
LDX	Load Index Register
LSR	Logical Shift Right
	Negate No Operation
ORA	Inclusive OR Accumulator
PSH	Push Data

PUL	Pull Data
ROL	Rotate Left
ROR	Rotate Right
RTI	Return from Interrupt
RTS	Return from Subroutine
SBA 🔊	Subtract Accumulators
SBC	Subtract with Carry
SEC	Set Carry
SEI	Set Interrupt Mask
SEV	Set Overflow
STA	Store Accumulator
STS	Store Stack Register
STX	Store Index Register
SUB	Subtract
SWI	Software Interrupt
TAB	Transfer Accumulators
TAP	Transfer Accumulators to Condition Code Reg
TBA	Transfer Accumulators
TPA	Transfer Condition Code Reg. to Accumulator
TST	Test
TSX	Transfer Stack Pointer to Index Register
TXS	Transfer Index Register to Stack Pointer
WAI	Wait for Interrupt

ESIGN



TABLE 3 - ACCUMULATOR AND MEMORY INSTRUCTIONS

	여 시 알 갑 같다.		te S		100.00	12.	-	10.14	1.22.2	6 M O				1. 		BOOLEAN/ARITHMETIC OPERATION	· —	t			-
PERATIONS	MNEMONIC	OP	MME	D =	0 OP	IREC	60 Y.	OP	NDE)	1.11	10.00	<u>XTN</u>		1.62.1	PLIED	(All register labels refer to contents)	5 H		3 N	2 Z	1
dd	ADDA	38	2	2	9B	3	= 2	AB	5	=	OP BB	4	= 3	OP	<u>~</u> ≓. ⊡			-		-	-
	ADDB	CB	2	2	DB	3	2	EB	5	2	FB	4	3			A + M → A B + M → B			. [,] •	. 1	1
dd Acmitrs	ABA					1.95							័	1B	2 1	A + B → A				4	1
dd with Carry	ADCA	89	2	2	99	3	2	A9	5	2	B9	4	3	neie i		$A + M + C \rightarrow A$	T I			t	
김 이렇는 잘 깨끗한	ADCB	C9	2	2	D9	3	2	E9	• 5	2	F9	4	3			B + M + C - + B				+	1
nd	ANDA	84	2	2	94	3	2	A4	5	2	B4 :	4	3			A · M - A				+	R
일 같은 것 같은 것	ANDB	C4	2	2	D4	3	2	E4	5	2	F4	4	3			B • M • B					R
t Test	BITA	. 85	2	2	95	3	2	A5	5	2	B5	4	3	11		A · M			f		R
	BITB	C5 .	2	2	D5	3	2	E5	5	2	F5	4	3	1 1911		B • M		•	1 I	- L	R
ear	CLR					6.26		6F	7	2	7F.	6	3			00 → M	. é	•	R		Ĥ.
	CLRA													4F	2 1	00 → A		•	R	S	A
	CLAB			3÷			50				130		31.5	5F	2 1	00 → B			R	S	-
mpare	CMPA	81	2	2	91	3	2	A1	5	2	B1	4	3			A – M			Ŧ	10	×1
20년 문화 같은	СМРВ	C1	2	2	D1	3	2	E1	5	2	.F1	4	3			B – M	•		The second	1	t
inpare Acmitrs	CBA									L (9Ê.			11	2 1	A - B	•		1	1	1
mplement, 1's	COM			-		60		63	7	2	73	6	3		에 가지 않다. 1971년 1981	M → M		j#	4.	1	R
영양 물건 승규는	COMA		1										201	43	2 1	Ā→A	. •	•	Ŧ	.t	R
	COMB		172 L.	30 S			2			g de			. 11	53	2 1	B→B	: •	•	1	t	R
nplement, 2's	NEG						20	60	7	2	70	6	3			00 – M ··· M		•	't	tlo	ി
gate)	NEGA	汤净									図魚			40	2 1	00 – A → A		•			ŏ
동일 전 영화	NEGB	<u></u>							변환	9. a.C	1993			50	2 1	00 - B -+ B			t	- 1	Ť
imal Adjust, A	DAA					838					12, 24 12, 24			19	2 1	Converts Binary Add. of BCD Characters			i		T
				19						62.						into BCD Format	17		+	1	4
rement	DEC			57.		. 39		6A	7	2	7A	6	3					1		. 1	~
	DECA		연단			영화		<u>с</u>			1	U	്	Sec. 2		M − 1 → M	· •	•	1		4
	DECH						94		이 있다. 14 43 4	<u>1</u> 07	100				2 1	A - 1 - A	•	•	1		4
lusive OR	EORA				-									5A	2 1	8 – 1 – B	` • •	٠	1		4
unsine ou		88	2	2	98	3	2	A8	5	2	B8	4	3	0 - 10	- M	A⊕M ≁A	•	•	1. I.	- I	Ŗ
	EORB	C8	2	. 2	D8	3	2	E8	5	2	F8	4	3		·	B⊕M→B	. •	۰	1		R
rement	INC							6C	7	2	70	6	3		s. 19	M+1→M	•	•	1		5
일을 혼 가 갔다. 말	INCA		1.00			- 12 - 12			201					4C	2 1	A + 1 → A	: •	•	1	10	5
ad Acmitr	INCB	0.0			0.0									SC	2 1	B+1B	•	•	1	10	5
au Acimiti	LDAA	86	2	2	96	3	2	A6	5	2	86	୍ 4	3			M → A	•	•	1	1	R
	LDAB	C6	2	2	D6	3	2	E6	5	2	F6	4	3			M • B	•	•	1	1	R
Inclusive	ORAA	8A	2	2	9A	3	2	AA	5	2	BA	4	3	ant -	13. TP	$A + M \rightarrow A$	•	•	.1	1	R
	ORAB	CA	- 2	2	DA	3	2	EA	5	2	FA	-4	3			B + M → B	•	•	1	1	R
h Data	PSHA	1.64			1.1						고소	16		36	4 1	A → M _{SP} , SP - 1 → SP		•		•	٠
	PSHB		19.89 19.89	12		14	175			- 6- L	100			37	4 1	B · · MSP, SP - 1 · · SP	•	•	٠	•	٠
Data	PULA		24			요솔	84		đ		÷.		21	32	4 1	SP + 1 - SP, M _{SP} - A	•	٠	٠	•	٠
일을 가지 않는 것을 알았다. 같은 것은 것은 방법을 통했다.	PULB								A.		분화			33	4 1	SP + 1 - SP, M _{SP} - B		۲	۲	•	•
ate Left	ROL		4		196			69	7	2	79	6	3	19		[M]	•	•	1	I (6
이 아파 가 있는 것	ROLA				80		đ	and the second s	R.	.96	소문	19 Q	19	49	2 1				1		č
	ROLB	1976 I.		秘入		Å	1. N					,et fi		59	2 1	B C b7 - b0	1.		1		č
tate Right	ROR		12			Æ.	dit .	66	7	2	76	6	3			M)		•	t		č
문 수준 정말로	RORA			her.			S.A	Ø		22				46	2 1		•	•	t		Ğ
	RORB			ેં	1	A.			ysi.		220			56	2 1	вј С b7 — b0		•	t		č
ft Left, Arithmetic	ASL		630	- 19 Ab	5			68	70	2	78	6	3			M)	•	•	t		č
	ASLA	189		1		1988 1			343 -	Şa.	슬쁥	1	I	48	2 1	A } □ → □ □ → 0		•	t		ē
4 D-1 4	ASLB		<u>_</u>	20 20					29	1				58	2 1	B C b7 b0	•	•	1	10	6
ft Right, Arithmetic	ASR		W	Q _{ER}	Ø			67	7	2	77	6	3		112	M	•	•	T.		Ē
1799년 전달(M	ASRA						<u>_</u>		er bi	<u>e</u>		18	영향	47	2 1			•	t		ē
	ASRB	10		erity Algo		661				÷.,	1.40			57	2 1	B b7 b0 C	•	•	t	10	Õ
ft Right, Logic	LSR							64	7	2	74	6	3			[M] → →			R	2 .	č
	LSRA			3. Î.	11 E.					(a			÷.	44	2 1	A } 0		1.1		- 13	-
	LSRB	02	88	dia.		문화					堂静				- I	ь) b7 b0 С	•	•	R	. 12	6
co Λemlie	1900 Mill		હેતુના						1			44	$\delta(z)$	54	2	B) D/ NJ L	. •	•	R		6
re Acmiir.	STAA		<u> </u>		97	4	2	A7	6	2	B7	5	3			A - M	•	•	1		Ŗ
	STAB				D7	4	2	E7	6	2	F7		3			B • M	•	٠	1	1	R
otract	SUBA	80	2	2	90	3	2	AO	5	2	80	4	3			A M A	•	•	1	1	1
	SUB8	CO	2	2	DO	3	2	80	5	2	FO	4	3			B – M – B	•	•	1		ŧ,
utract Acmilirs.	SBA										19	\$ C	in 1	10	2 1	A - B • A		•	1		ī
otr. with Carry	SBCA	82	2	2	92	3	2 2	A2		2	B2	4	3			A - M - C + A		•	1		ţ
	SBCB	C2	2	2	D2	3	2	E2	5	2	F2	4	3		한관련	$\mathbf{B} - \mathbf{M} - \mathbf{C} \cdot \mathbf{B}$	•	•	1		i
nsfer Acmltrs	TAB	12								6.54	日常	хe.			2 1	А В	•	•	1		R
	TBA		10. di				14		승규는		124		올림	17	2 1	B→A	•	•	1		R
st, Zero or Minus	TST		taria). Canar			と含		60	7	2	70	6	3	金田		M - 00	•	٠	1	1	Ŕ
	TSTA	155								2					2 1	A - 00			1		R
	TSTB		생산	99	$x^{-1} x^{-1}$	135	0.2		<u>en -</u> -					5D	2 1	B - 00	•	•	ţ.		R
 A second sec second second sec	and name of	04.111	62.054																-+		

LEGEND:

- Operation Code (Hexadecimal); OP
- Number of MPU Cycles;
- Number of Program Bytes; <u>ن</u>
- Arithmetic Plus; Arithmetic Minus;
- Boolean AND;
- MSP Contents of memory location pointed to be Stack Pointer;

Note - Accumulator addressing mode instructions are included in the column for IMPLIED addressing

CONDITION CODE SYMBOLS:

- Half-carry from bit 3; Н
- Interrupt mask 4.
- Negative (sign bit) N
- Zero (byte) Ż
- ٧ Overflow, 2's complement
- Carry from bit 7 C
- Reset Always R
- S Set Always

١.

- 1
- Test and set if true, cleared otherwise Not Affected



14

Boolean Inclusive DR;

Boolean Exclusive OR;

Complement of M;

Transfer Into;

Bit = Zero;

Byte = Zero;

 \odot

M

<u>.</u>

٥.

TABLE 4 - INDEX REGISTER AND STACK MANIPULATION INSTRUCTIONS

		L II	MME	D	D	IREC	т	1	NDE	х	E	XTN	D	IN	IPLIE	D		5	4	3 2	! 1	
DINTER OPERATIONS	MNEMONIC	OP	~	#	OP	~	#	OP	~	=	OP	~	=	OP	~	n	BOOLEAN/ARITHMETIC OPERATION	н	1	NZ	<u>'</u> v	1
ompare Index Reg.	CPX	8C	3	3	9C	4	2	AC	6	2	BC	5	3				$X_{H} - M, X_{L} - (M + 1)$	•		কা	: (8)	st
ecrement Index Reg	DEX						- s.							09	4	1	$X - 1 \rightarrow X$	•				5
ecrement Stack Potr	DES													34	4	1	$SP - 1 \rightarrow SP$	•				
crement Index Reg	INX										1			08	4	1	X + 1 → X .	•		<u>م</u> ر	. •	j
crement Stack Potr	INS		1			. •							· .	31	4	1	$SP + 1 \rightarrow SP$			ø.	• •	
oad Index Reg	LDX	CE	3	3	DE	4	2	EE	6	2	FE	5	3				M → X _H , (M + 1) → X ₁		• (9	1 R	1
oad Stack Pntr	LDS	8E	3	3.	9E	4	2	AE	6	2	BE	5	3				M → SPH, (M + 1) → SPI) ک	ดิ	I R	ł
ore Index Reg	STX -				DF	5	2	EF	7	2	FF	6	3		-	÷	XH M, XL (M + 1)			Ď	t R	- 1
ore Stack Pntr	STS			1.11	9F	5	2	AF	7	2	BF	6	3	· · .			$SP_H \rightarrow M, SP_L \rightarrow (M + 1)$))	t R	
dx Reg → Stack Pritr	TXS										-	-		35	. 4	1	X - 1 → SP					,
ack Pntr → Indx Reg	⊤sx		6 T.		·			l .	ŀ .					30	4	1	SP + 1 → X		•			,
		- - - - -		-													ONS					

		RE	LATI	IVE	1	NDE:	x	F	XTN	D	IN	PLIE	D		5	4	3	2	Τ.
OPERATIONS	MNEMONIC	OP		#	OP	~	#	OP	r	#	OP	~	#	BRANCH TEST	Н	4	N	Z	1
Branch Always	BRA	20	4	2									Ó	None	•	•		•	+
Branch If Carry Clear	всс	24	4	2			·					do -		C = 0					
Branch If Carry Set	BCS	25	4	2				1				K		C = 1					1.
Branch If = Zero	BEQ	27	4	2						1		1		Z = 1					
Branch If ≥ Zero	BGE	2C	4	2						. 🕷				N ⊕ V = 0					
Branch If >Zero	BGT	2E	4	2	·				4		ter f			Z + (N ⊕ V) = 0					ľ
Branch If Higher	вні	22	4	2]									C + Z = 0					
Branch If ≤Zero	BLE	2F	4	2					. 4	Q				Z + (N ⊕ V) = 1					
Branch If Lower Or Same	8 LS	23	4	2			A	4	91°					C + Z = 1					
Branch If < Zero	BLT	20	4	2	1	4	A.							N ⊕ V = 1					1
Branch If Minus	BMI	2B	4	2	· ·			\$			İ.			N = 1					
Branch If Not Equal Zero	BNE	26	4	2							ļ			Z = 0		•			
Branch If Overflow Clear	BVC	28	: 4	2	1									V = 0					
Branch If Overflow Set	BVS	29	4	2	K.	AND .								V = 1				•	
Branch If Plus	BPL	2A	A	2						1		1.1	·	N = 0					
Branch To Subroutine	BSR	: 8D	8	2						-						•	9	•	
Jump	JMP	4		1900 - C	6E	• 4	2	7E	3	3			ł	See Special Operations	•	۰	•		
Jump To Subroutine	JSR	Ø "	MESTER.		AD	8	2	BD	9	3) (Figure 16)	•	8	•	•	
No Operation Return From Interrupt	NOP								1.		01	2	1	Advances Prog. Cntr. Only	•	9	•	•	ļ
Return From Interrupt Return From Subroutine	RTI RTS										3B	10	1	· · · · · ·			- (10 -	,
Software Interrupt	SWI				1						39	5			•	8	•	•	
Wait for Interrupt	WAI										3F 3E	12 9	1	 See Special Operations (Figure 16) 	•	l în			
											ЪЕ	9	') (Figure To)	•	ĮΨ			⊥
	¢.																		
	da i i																		
	r																		
														and the second					
and the second																			
Charles in the second s													•						
No.																			
80m. 5989-																			



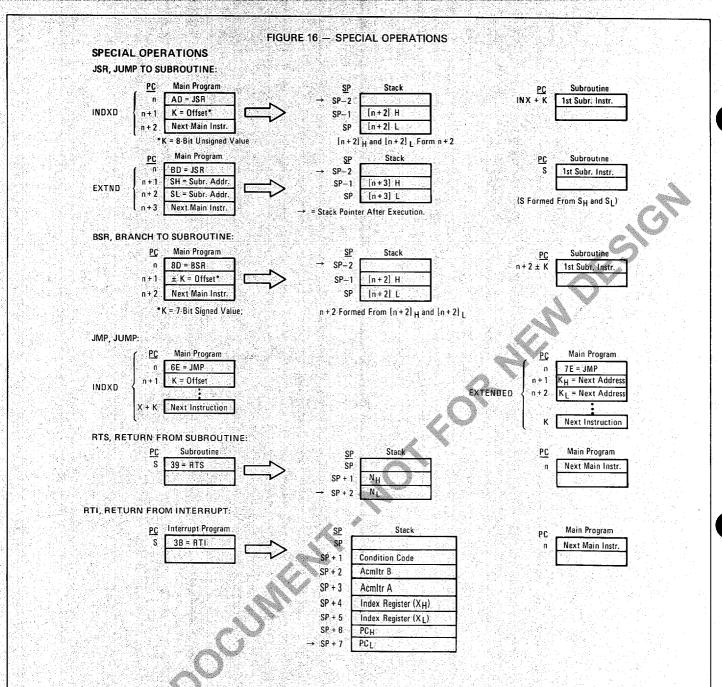


TABLE 6 - CONDITION CODE REGISTER MANIPULATION INSTRUCTIONS

		2					CON	D. C(DDE	REG	
		IN	IPLIE	D	[일종] : 1993년 1 1993년 1월 1993년 br>1993년 1993년 199	5	4	3	2	1	٥
OPERATIONS	MNEMONIC	OP	\sim	=	BOOLEAN OPERATION	Н	1	N	z	۷	C
Clear Carry	CLC	00	2	1	0 → C			•	•		F
Clear Interrupt Mask	CLI	0E	2	1	0→1		R	•	•		
Clear Overflow	CLV	0A	2	1	0 → V	•	•	•	•	R	
Set Carry	SEC	0D	2	1	1 → C	۲			•		
Set Interrupt Mask	SEI	OF	2	1	listen an i →instation	٠	S		•	•	
Set Overflow	SEV	OB	2	1	1 → V	•	•	•	•	S	
Acmltr A → CCR	TAP	06	2	1	A → CCR			-6	2)-		1
CCR → Acmltr A	TPA	07	2	ាំ	CCR → A			•		<u>.</u>	

CONDITION CODE REGISTER NOTES: (Bit set if test is true and cleared otherwise)

Test: Decimal value of most significant BCD Character greater than nine?

RCIN

(Bit V)

(Bit C)

(Bit C)

(Bir V)

(Bit V)

1

2

- 3

4

5

6

Test: Result = 10000000?

Test: Result # 00000000?

(Not cleared if previously set.)

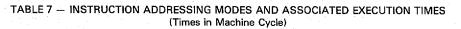
Test: Operand = 10000000 prior to execution?

Test: Operand = 01111111 prior to execution?

(Bit V) Test: Set equal to result of N⊕C after shift has occurred.

- 7 (Bit N) Test: Sign bit of most significant (MS) byte = 1?
 - (Bit V) Test: 2's complement overflow from subtraction of MS bytes? 8
 - 9
 - (Bit N) Test: Result less than zero? (Bit 15 = 1) 10 (AII)
 - Load Condition Code Register from Stack. (See Special Operations)
 - 11 (Bit I) Set when interrupt occurs. If previously set, a Non-Maskable
 - Interrupt is required to exit the wait state. 12 (AII) Set according to the contents of Accumulator A

NOTOROLA Semiconductor Products Inc.



	Ð			Ð		
	(Dual Operand) ACC X	diate ded ed	₽	(Dual Operand) ACC X	diate t ded	pa pa
			Relative			Indexed
A	BA • DC x • DD x •	2 3 4 5 2 3 4 5	2 •	INC 2 INS • INX •		7 • 4 • 4
A: A:	ND X • SL 2 SR 2	2 3 4 5 • 6 7 • 6 7	• • a • • •	JMP • JSR • LDA x •	• • 9 2 3 4	4 • 8 • 5 •
B	CC • CS • EA •	• • • • • • • •	• 4 • 4 • 4	LDS • LDX • LSR 2		6 6 7
B	GE • GT • HI •	• • • • • • • • • • • •	• 4 • 4 • 4	NEG 2 NOP • ORA x •	• • 6 • • • 2 3 4	7 • • 2 5 •
B	LE • LS •	2 3 4 5 • • • • • • • •	• • • • • • • • • • • • • • • • • • •	PSH • PUL • ROL 2		• 4 • 4 7 •
B	LT • MI • NE •	• • • • • • • • • • • •	• 4 • 4 • 4	ROR RTI RTS	• • 6	7 • • 10 • 5
B	PL • RA • SR •	• • • • • • • • • • • •		SBA SBC SEC x	• • • 2 3 4 • • •	• 2 5 • • 2
B C	VC • VS • BA •	• • • • • • • • • • • •	• 4 • 4 • • 4	SEI SEV STA x •	• • • • • • • 4 5	• 2 • 2 6 •
C C	LC • LI • LR 2	• • • • • • 6 7	2 • •	STS STX SUB x	• 5 6 • 5 6 2 3 4	7 • 7 • 5 •
C C	LV • MP x • OM 2	2 3 4 5 • 6 7	2 •	SWI • TAB • TAP •	6 6 9 6 7 6 8 6	 12 2 2
D	PX • AA • EC 2	3 4 5 6 • • • • • 6 7	• • 2 • • •	TBA • TPA • TST 2	• • • • • • • • 6	• 2 • 2 7 •
D	ES • EX • OR x •	2 3 4 5	4 • 4 • • •	TSX • TSX • WAI •	8 0 9 • • 0 • • 0	• 4 • 4 • 9
N		time is 12 cycles fro				
		ction being executed truction. Then it is 4		9	1. 1	
Q2						
	- (M) n	OTORC	LA Sem	niconductor	Products	Inc



SUMMARY OF CYCLE-BY-CYCLE OPERATION

Table 8 provides a detailed description of the information present on the address bus, data bus, valid memory address line (VMA), and the read/write line (R/\overline{W}) during each cycle for each instruction:

This information is useful in comparing actual with expected results during debug of both software and hardware

as the control program is executed. The information is categorized in groups according to addressing modes and number of cycles per instruction. (In general, instructions with the same addressing mode and number of cycles execute in the same manner; exceptions are indicated in the table.)

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
IMMEDIATE	1893.5					
ADC EOR		1	1	Op Code Address	1	Op Code
ADD LDA AND ORA	2	2	1	Op Code Address + 1	1	Operand Data
BIT SBC						
CMP SUB						
CPX LDS	3	1	1	Op Code Address	1	Op Code
LDX	്	2	1	Op Code Address + 1		Operand Data (High Order Byte)
DIRECT		3		Op Code Address + 2		Operand Data (Low Order Byte)
ADC EOR			<u>.</u>			
ADD LDA				Op Code Address		Op Code
	3	2		Op Code Address + 1	1	Address of Operand
BIT SBC CMP SUB		3	1	Address of Operand	1	Operand Data
CPX		1	1	Op Code Address	1	Op Code
LDS LDX	4	2	1	Op Code Address + 1	1	Address of Operand
		3	ો	Address of Operand	1	Operand Data (High Order Byte)
		4	1	Operand Address + 1	1	Operand Data (Low Order Byte)
STA		1	<u>_1</u>	Op Code Address	1	Op Code
	4	2	1 1	Op Code Address + 1		Destination Address
		3	0	Destination Address	1	Irrelevant Data (Note 1)
		4	1	Destination Address	0	Data from Accumulator
STS		1	1	Op Code Address	 1	Op Code
STX		2	্	Op Code Address + 1	1	Address of Operand
	5	3	0	Address of Operand	1	Irrelevant Data (Note 1)
1997년 - 1991년 1991년 1991년 1991년 1991년 - 1991년 br>1991년 - 1991년 1		4		Address of Operand	0	Register Data (High Order Byte)
		5	1	Address of Operand + 1	0	Register Data (Low Order Byte)
INDEXED		19. Z.).				
JMP		1	1	Op Code Address	1	Op Code
		2	62	Op Code Address + 1	1	Offset
	4	3	0	Index Register		Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
ADC' EOR	는 이번 이상 제품은 사람들이		a a standing Alight	Op Code Address	1	Op Code
ADD LDA		2		Op Code Address + 1	1	Offset
AND ORA BIT SBC	5	3	0	Index Register	1	Irrelevant Data (Note 1)
CMP SUB		4	o	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data
CPX		1	1	Op Code Address	1	Op Code
LDS LDX		2	1	Op Code Address + 1	1	Offset
	6	3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	1	Index Register Plus Offset	1	Operand Data (High Order Byte)
		6	1	Index Register Plus Offset + 1	1	Operand Data (Low Order Byte)

TABLE 8 - OPERATIONS SUMMARY

IOTOROLA Semiconductor Products Inc.

Address Mode and Instructions	Cycles	Cycle #	VMA Line	Address Bus	R/W Line	Data Bus
INHERENT (Continued)					1999. 1997	
WAI		1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Op Code of Next Instruction
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
	9	5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
		7	1	Stack Pointer – 4	0	Contents of Accumulator A
an an Albert an Albert	a la sura d	8	1	Stack Pointer – 5	0	Contents of Accumulator B
	and she	9	1	Stack Pointer – 6	1	Contents of Cond. Code Register
RTI		1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	and the second
		5 S. 4				Irrelevant Data (Note 2)
		3	0	Stack Pointer	1	Irrelevant Data (Note 1)
	10	4	1	Stack Pointer + 1	1	Contents of Cond. Code Register from Stack
	10	5	1	Stack Pointer + 2	1	Contents of Accumulator B from Stack
		6	1	Stack Pointer + 3	1	Contents of Accumulator A from Stac
		7	1	Stack Pointer + 4	1	Index Register from Stack (High Order Byte)
		8	1	Stack Pointer + 5	1	Index Register from Stack (Low Order Byte)
		9	1	Stack Pointer + 6	ୀ	Next Instruction Address from Stack (High Order Byte)
		10	1	Stack Pointer + 7	1	Next Instruction Address from Stack (Low Order Byte)
SWI		1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Irrelevant Data (Note 1)
		3	1	Stack Pointer	0	Return Address (Low Order Byte)
		4	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		5	1	Stack Pointer – 2	0	Index Register (Low Order Byte)
		6	1	Stack Pointer – 3	0	Index Register (High Order Byte)
	12	7	Litter	Stack Pointer – 4	o	Contents of Accumulator A
		8		Stack Pointer – 5		
					0	Contents of Accumulator B
		9		Stack Pointer – 6	0	Contents of Cond. Code Register
		10	0	Stack Pointer — 7	1	Irrelevant Data (Note 1)
		\bigcirc	1	Vector Address FFFA (Hex)	1	Address of Subroutine (High Order Byte)
	<u>by</u>	12	1	Vector Address FFFB (Hex)	1	Address of Subroutine (Low Order Byte)
RELATIVE						
BCC BHI BNE		1	1	Op Code Address	1	Op Code
BCS BLE BPL BEQ BLS BRA	4	2	1	Op Code Address + 1	1	Branch Offset
BGE BLT BVC		3	0	Op Code Address + 2	1	Irrelevant Data (Note 1)
BGT BMI BVS		4	0	Branch Address	1	Irrelevant Data (Note 1)
BSR		1	1	Op Code Address	1	Op Code
		2	1	Op Code Address + 1	1	Branch Offset
() Year de biel		3	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		4	1	Stack Pointer		
	8	1. S.			0	Return Address (Low Order Byte)
		5	1	Stack Pointer – 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer – 2	1	Irrelevant Data (Note 1)
		7	0	Return Address of Main Program	1	Irrelevant Data (Note 1)
		8	0	Subroutine Address (Note 4)	1	Irrelevant Data (Note 1)

Depending on bus capacitance, data from the previous cycle may be retained on the Data Bus. 2. Data is ignored by the MPU.

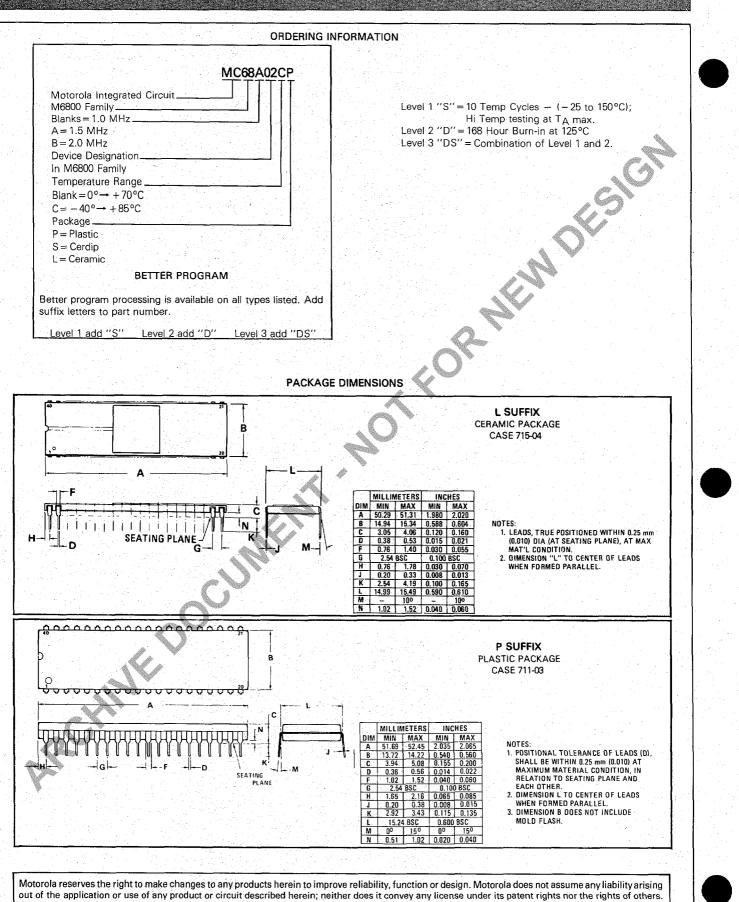
But IS ignored by the integral
 For TST, VMA=0 and Operand data does not change.
 MS Byte of Address Bus=MS Byte of Address of BSR instruction and LS Byte of Address Bus=LS Byte of Sub-Routine Address.

21

MOTOROLA Semiconductor Products Inc.

and Instructions Cycles # Line Date Bus STENDED (Continued)	Address Mode		Cycle	VMA		R/W	
TS 1 0p Code Address 1 Op Code Address 1 Address of Operand (High Ord Address of Operand (High Ord Address of Operand 1 8 3 1 Op Code Address 1 Address of Operand 1 1 Indexes of Operand 1 1 Indexes of Operand 1 0 Operand Data (High Orde Byt Operand 1 0 Operand Data (High Orde Byt Operand 1 1 Op Code Address + 1 1 Address of Operand + 1 0 Operand Data (High Orde Byt Operand 1 0 Operand Data (High Orde Byt Operand 1 1 Op Code Address + 1 1 Address of Subroutine High Orde Byt Operand Data (How Order Byt Ope Code Address + 1 1 Address of Next Instruction Operand Data (How Order Byt Ope Code Address + 2 1 Address of Next Instruction Operand Data (How Order Byt Ope Code Address + 2 1 Address of Next Instruction Ope Code Address + 2 1 Interevent Data (Note 1) Interevent Data (Note 1) Note Next Instruction Ope Code Address + 2 1 Address of Subroutine (How Order Byt Ope Code Address + 2 1 Interevent Data (Note 1) Address of Subroutine (How Order Byt Ope Code Address + 2 1 Interevent Data (Note 1) Interevent Data (Note 1) Interevent Data (Note 1) Interevent Data (Note 1) Interevent Data (Note 1)<	and Instructions		#		Address Bus		Data Bus
STX 2 1 Op Code Address + 1 1 Address of Operand (Low Order By Address of Operand 1 6 3 1 Address of Operand 1 1 Irrelevant Dats (Note 1) 1 Address of Operand 1 1 Irrelevant Dats (Note 1) Operand Dats (High Order Byt Operand Dats (High Order Byt Operand Dats (Low Order Byt Operand Dats (Low Order Byt Operand Dats (Low Order Byt Address of Operand 1 1SR 1 1 Op Code Address + 1 1 Address of Subroutine High Order 3 1 Address of Operand 1 Address of Subroutine High Order 4 1 Address of Operand 1 Address of Subroutine High Order 4 1 Address of Operand 1 Address of Subroutine High Order 1 Address of Operand 1 Address of Operand 1 Address of Subroutine High Order 4 1 Address of Operand 1 Address of Subroutine High Order 1 Address of Subroutine High Order 4 1 Address of Subroutine High Order 4 1 Address of Subroutine High Order 4 1 Intervent Address High Order 4 1 Address of Subroutine High Order 4 1 Intervent Address High Order 4			a Calla Taria est				
2 1 Address of Dparad (Higo Ord) 6 3 4 0 Op Gode Address + 2 1 Address of Dparad Data (Higo Ord) JSR 1 1 1 Op Code Address + 1 1 Operand Data (Higo Ord) JSR 1 1 1 Op Code Address + 1 1 Op Code Address + 1 JSR 1 1 Op Code Address + 1 1 Address of Subroutine(Higb Ord) JSR 1 1 Op Code Address + 2 1 Address of Subroutine(Higb Ord) 3 1 Op Code Address + 2 1 Address of Subroutine(Higb Ord) 9 5 T Stack Pointer 0 Return Address (Higb Ord) 9 6 1 Stack Pointer 1 Irrelevant Data (Note 1) 1 Irrelevant Data (Data (D			1			요즘 동안 가 좋지 않.	
9 4 0 Address of Operand : 1 Irrelevant Data (Note 1) JSR 6 1 Address of Operand 1 0 Operand Data (Low Order Byt Operand Data (Low Order Byt Operand Data (Low Order Byt Op Code Address + 1 1 Op Code Address of Subroutine (Low Order Byt Op Code Address + 2 JSR 2 1 Op Code Address + 1 1 Address of Subroutine (Low Order Byt Op Code of Next Instruction Return Address (High Order Byt Op Code of Next Instruction Return Address (High Order Byt Op Code Address + 2 1 Address of Subroutine (Low Od Return Address (High Order Byt Op Code Address + 2 1 Irrelevant Data (Note 1) NHERENT 3 1 Op Code Address + 2 1 Irrelevant Data (Note 1) NHERENT 2 1 1 Op Code Address + 2 1 Irrelevant Data (Note 1) NMERENT 2 1 1 Op Code Address + 1 1 Address of Subroutine (Low Od PCode Address + 1 1 Op Code CCL NG FAP CL 2 1 1 Op Code Address + 1 1 Op Code O Next Instruction Op Code of Next Instruction Op Code of Next Instruction Op Code of Next Instruction PUL 4 1 1				친구, 김요.		승규는 물건을 많다.	Address of Operand (High Order Byte)
SR 5 1. Address of Operand + 1 0 Operand Date (High Order Byt Operand Date (Low Order Byt Op Code ISR 1 1 0 Op Code Address + 1 1 Address of Subroutine (High Order Byt Op Code ISR 2 1 Op Code Address + 2 1 Address of Subroutine (High Order Byt Op Code ISR 3 1 Op Code Address + 2 1 Address of Subroutine (High Order Byt Op Code Order Byt Op Code Address + 2 1 Address of Subroutine (Low Order Byt Op Code Order Byt Op Code Address + 2 1 Address of Subroutine (Low O Op Code Order Byt Op Code Address + 2 1 Irrelevant Date (Note 1) INHERENT 8 0 Op Code Address + 2 1 Address of Subroutine (Low O Op Code Of Next Instruction Op Code Of Next Instruction Op Code Of Next Instruction Op Code Of Next Instruction Op Code Of Next Instruction Irrelevant Date (Note 1) INK 4 0 New Register Contents 1 Op Code Of Next Instruction Irrelevant Date (Note 1) INX 4 0 New Register Contents 1 Op Code Of Next Instruction Irrelevant Date (Note 1) INX 4 0 New Register Contents 1 1	는 것은 것은 가격한 것은 것은 것이다. 2013년 - 1011년 전 1913년	6			[2] 2] 전 22 22 23 24 24 24 24 24 24 24 24 24 24 24 24 24	아파말 소리 가슴?	Address of Operand (Low Order Byte)
JSR I Address of Operand + 1 O Operand Data (Low Order Byts) JSR 1 1 Op Code Address + 1 1 Op Code Address + 2 1 Address of Subroutine (High) Ocde JSR 3 1 Op Code Address + 2 1 Address of Subroutine (High) Ocde JSR 4 1 Subroutine Starting Address 1 Op Code Address + 2 1 JSR 6 1 Stack Pointer 0 Return Address (Hold Order Byts) JSR 0 Stack Pointer - 1 0 Return Address (Hold Order Byts) JSR 0 Do Code Address + 2 1 Irrelevan Data (Note 1) Intereavem Data (Note 1) 0 Op Code Address + 2 1 Address of Subroutine (Low O INHERENT 2 1 0 Op Code Address + 1 1 Op Code of Next Instruction CL NG TAB 0 Op Code Address + 1 1 0p Code of Next Instruction INN 4 2 1 0p Code Address + 1 1 0p Code of Next In			4			그는 말을 하는 것을 수가 없다. 이렇게 하는 것을 하는 것을 하는 것을 하는 것을 하는 것을 수가 없는 것을 하는 것을 수가 없는 것을 것을 수가 없는 것을 것을 것 같이 않는 것을 것 않는 것 같이 않는 것을 것 않는 것 같이 않는 것 같이 없다. 않는 것 않는 것 같이 없는 것 같이 없는 것 않는 것 같이 않는 것 않는	비 사람들은 동안을 걸 날 때 가 많은 것은 것 같아. 것 같아.
JSR 1 1 Op Code Address 1 Op Code 2 1 Op Code Address + 1 1 Address of Subroutine (High O Address of Subroutine (How O Previous Address + 2 1 Address of Subroutine (How O Address of Subroutine (How O Previous Address + 2 9 5 1 Stack Pointer 0 Return Address (How Order B Previous Address + 2 7 0 Stack Pointer - 1 0 Return Address (How Order B Previous Address + 2 8 0 Op Code Address + 2 1 Itrelevant Data (Note 1) 8 0 Op Code Address + 2 1 Itrelevant Data (Note 1) 8 0 Op Code Address + 2 1 Address of Subroutine (Low O INHERENT 7 0 Stack Address + 1 1 Op Code Address + 1 8 De Code 1 Op Code Address + 1 1 Op Code Address + 1 11 Now Register Contents 1 1 Op Code Address + 1 1 0p Code Address + 1 <t< td=""><td>가 있는 것은 것이 있는 것이 있었다. 같은 것은 것이 있는 것이 있는 것이 있는 것이 있는 것이 있는 것이 있는 것이 없다. 것이 있는 것이 없는 것이 없는 것이 없는 것이 없는 것이 없는 것이 있는 것이 없는 것이 없는 것이 없는 것이 같은 것은 것이 같은 것이 없는 것이 있</td><td></td><td>2 전 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</td><td></td><td></td><td>성영상 동안 것 같아요.</td><td>년 1월 /td></t<>	가 있는 것은 것이 있는 것이 있었다. 같은 것은 것이 있는 것이 있는 것이 있는 것이 있는 것이 있는 것이 있는 것이 없다. 것이 있는 것이 없는 것이 없는 것이 없는 것이 없는 것이 없는 것이 있는 것이 없는 것이 없는 것이 없는 것이 같은 것은 것이 같은 것이 없는 것이 있		2 전 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1			성영상 동안 것 같아요.	년 1월
2 1 Op Code Address + 1 1 Address of Subroutine (Hub O 3 1 Op Code Address + 2 1 Address of Subroutine (Low O 9 5 1 Stack Pointer - 1 0 De Code Address + 2 1 7 0 Stack Pointer - 2 1 1 Irrelevant Data (Note 1) 7 0 Stack Pointer - 2 1 1 Irrelevant Data (Note 1) 8 0 Op Code Address + 2 1 1 Irrelevant Data (Note 1) 8 0 Op Code Address + 2 1 1 Irrelevant Data (Note 1) 8 0 Op Code Address + 2 1 1 Dop Code Address + 2 1 Nov Faa 2 1 0p Code Address + 1 1 Op Code Code Address + 1 1 Nov Faa 2 1 0p Code Address + 1 1 Op Code O 1 Nov Faa 2 1 0p Code Address + 1 1 Op Code O 1 1 1 0p Code Address + 1			6	Martine at		1944 general og og	
3 1 Op Code Address + 2 1 Address of Subroutine Now O 9 5 1 Subroutine Starting Address 1 Op Code of Next Instruction 9 5 1 Stack Pointer 0 Return Address (High Order B) 7 0 Stack Pointer - 1 0 Return Address (High Order B) 9 1 Op Code Address + 2 1 Irrelevant Data (Note 1) Address Code Of Address + 2 1 Address of Subroutine (Low O 1 NHERENT 7 0 Stack Address + 2 1 ABA DAA SEC ASL DEC SEV ASE INS STAP 2 1 0p Code Address + 1 1 0 Op Code Address + 1 1 Op Code Inservation 0 CL NEG TAP CL NOP TBA CL NOP TST COM SBA 1 1 0p Code Address + 1 1 0p Code Inservation NS 4 2 1 0p Code Address + 1 1 0p Code Inservation NS 4 2 1 0p Code Address + 1 1 0p Code Inservation NSBA <td>JSR</td> <td></td> <td></td> <td></td> <td>동생·방법에서 문화되었다. 2013년 - 1913년 - 1 1913년 - 1913년 -</td> <td>상품 수준 다음 가슴을 가슴다.</td> <td>이 가 물었다. 영화 가 가 있는 것은 것은 것은 것을 했다. 이 것은 것을 했다.</td>	JSR				동생·방법에서 문화되었다. 2013년 - 1913년 - 1 1913년 - 1913년 -	상품 수준 다음 가슴을 가슴다.	이 가 물었다. 영화 가 가 있는 것은 것은 것은 것을 했다. 이 것은 것을 했다.
Jack 1 Subroutine Starting Address 1 Op Code of Next Instruction Return Address Low Order By Return Bat Note Low Order Address Low Order Sy Return Bat Note Low Order By Return Bat Note Low Order Address Low Order Data Note Low Order Return Bat Note Low Order Address Low Order Address Low Order Data Note Low Return Bat Note Low Now Regi			1월 17 1일		(1997) (1997) (1998) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997) (1997)		Address of Subroutine (High Order Byte
9 5 1 Stack Pointer 0 Return Address (Low Order By Return Address (High Order By Address of Subroutine (Low O Decide Address + 2 INHERENT - - 0 Op Code Address - - 1 Irrelevant Data (Note 1) Address of Subroutine (Low O Decide Address + 2 INHERENT - <td></td> <td></td> <td>自己の正常に</td> <td>[1991년 4일]</td> <td>[편안]: 56 (1.1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1</td> <td>다양時 나는 물값 난</td> <td>Address of Subroutine (Low Order Byte</td>			自己の正常に	[1991년 4일]	[편안]: 56 (1.1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	다양時 나는 물값 난	Address of Subroutine (Low Order Byte
6 1 Stack Pointer – 1 0 Return Address (High Order B) 7 0 Stack Pointer – 2 1 Irrelevant Data (Note 1) 8 0 0p Code Address + 2 1 Irrelevant Data (Note 1) 9 1 0p Code Address + 2 1 Address of Subroutine (Low O INHERENT - - 0p Code Address + 2 1 Address of Subroutine (Low O ASR INC SEV 2 1 1 Op Code Address + 1 1 Op Code ASR INC SEV 2 1 1 Op Code Address + 1 1 Op Code CL NGS TAB - 1 Op Code Address + 1 1 Op Code Op Code CL NGS TAB - 1 Op Code Address + 1 1 Op Code Next Instruction MA 4 2 1 Op Code Address + 1 1 1 Op Code Address + 1 1 0p Code Next Instruction NA 4 2 1 Op Code Address + 1				北京的公司	1993년 20년 전원은 대학교 1992년 - 1992년 1992년 - 1992년 -	计记忆时 化水气的复数	영상 수상 위치 방법에 걸려서 걸려가 있는 것이 같은 것이 많이 많이 했다.
7 0 Stack Pointer - 2 1 Irrelevant Data (Note 1) 1 0 Code Address + 2 1 1 Irrelevant Data (Note 1) INHERENT Address of Subroutine (Low O Address of Subroutine (Low O Address of Subroutine (Low O ABA DAA SEC 2 1 1 Op Code Address 1 Address of Subroutine (Low O ABA DAA SEC 2 1 1 Op Code Address 1 0 Op Code Address CL NG FAP -	승규는 것을 잡는 것]	9	1842		물건 영영 현실 전문에 많은 말까? 한 것은 것은 것이 집니다.	그 전문 문제 문제 문제	
B: O Op Code Address + 2 1 Irrelevan Data (Note 1) Address of Subroutine (Low O INHERENT ABA DAA SEC ASL DEC SE1 ASR INC SEV CASL DEC SE1 ASR INC SEV CASL DEC SE1 ASR INC SEV CASL DEC SE1 ASR INC SEV CL NEG TAP CL NOT TST 1 0p Code Address Op Code Address + 1 1 Op Code DP Code of Next Instruction DES DES DES DES DES DES DES DES DES DES					물수는 것을 많은 것이 같아요. 것은 것 같은 것 같아요. 것 같아요.	지수는 것은 상품이다.	
9 1 Op Code Address + 2 1 Address of Subroutine (Low O INHERENT ABA SEC 2 1 1 Op Code Address 0 ASL DEC SEL 2 1 0 Op Code Address 0 0 ASR INC SEV 2 1 0 Op Code Address 0 </td <td>승규는 문제로 한 것이</td> <td></td> <td></td> <td></td> <td></td> <td></td> <td>이는 것을 못 수는 것 같아요. 물건을 물건을 물건을 했다.</td>	승규는 문제로 한 것이						이는 것을 못 수는 것 같아요. 물건을 물건을 물건을 했다.
INHERENT ABA DAA SEC ASL 2 1 1 Op Code Address 0 Op Code Op Cod				0	[kk] 2.50 2.10 2.10 2.50 2.10 2.10 2.10 2.10 2.10 2.10 2.10 2.1		· 전문· 영상 · · · · · · · · · · · · · · · · · ·
ABA DAA SEC ASL DEC 2 EX 1 1 0p Code Address Op Code Address 1 0p Code Op Code of Next Instruction CASL CL NEG CL NEG CL NOP TBA CLT NOP TBA CLT NOP TBA CLT NOP TBA CLT ROL TPA CLT ROR TST 1 1 0p Code Address 1 0p Code of Next Instruction DES INS 4 2 1 0p Code Address 1 0p Code of Next Instruction DEX INS 4 2 1 0p Code Address 1 0p Code of Next Instruction INX 4 2 1 0p Code Address 1 0p Code of Next Instruction INX 4 2 1 0p Code Address 1 0p Code of Next Instruction INX 4 2 1 0p Code Address 1 0p Code Of Next Instruction New Register Contents 1 1 0p Code Address 1 0p Code Of Next Instruction PUL 1 1 0p Code Address 1 0p Code Of Next Instruction 4 2 1 0p Code Address 1 0p Code Of Next Instruction FX 1 1 0p Code Address 1		Site and	9	1	Op Code Address + 2		Address of Subroutine (Low Order Byte
ASL ASR INC. SEV CBA LSR TAB CLO NEG TAP CLI NOP TBA CLO NEG TAP CLI NOP TBA CLI NOP TO CLI NOP TBA CLI NOP TAB CLI NOP	INHERENT						
ASL ASR INC DEC SEV CBA LLSR TAB CLC REGUE NOP TBA CLC NEG TAP CLL NOP TBA CLC NEG TAP CLL NOP TBA CLC NEG TAP CLC NOP TAP CLC NOP COM CLC ADDRESS 1 1 OP Code Of Next Instruction Interviewant Data (Note 1) Index Register CLC NEG TAP CLC NOP Code CLC NOP Code CLC NOP Code CLC NOP Code CLC NOP CODE ADDRESS 1 OP Code Of Next Instruction Index Register CLC NOP Code CLC NOP CODE ADDRESS 1 OP Code OF Next Instruction Index Register CLC NOP Code CLC NOP CODE CLC NOP CODE ADDRESS CLC NOP CODE CLC NOP CODE ADDRESS CLC NOP CODE CLC NOP CODE CLC NOP CODE ADDRESS CLC NOP CODE CLC NOP CODE CLC NOP CODE CLC NOP CODE CLC NOP CODE CLC NOP CODE CLC NOP CLC NOP CODE CLC NOP CLC NOP CLC NOP CODE CLC NOP CLC	ABA DAA SEC		11	1	Op Code Address		Op Code
Ann inde sev BAC LSR TAB CLC NEG TAP CLC	이 제가 가지에 바다는 것이 되는 것이 하는 것이 나갔다. 영화 영화 영화	2	2	1124 Fi 1117	Op Code Address + 1	1	Op Code of Next Instruction
CLC. NEG TAP CLL NOP TBA CLF ROL TPA CLF ROL TPA CLF ROL TPA CLF ROL TPA COM SBA 1 0 Op Code Address 1 0p Code DES INS 4 2 1 0p Code Address 1 0p Code of Next Instruction Irrelevant Data (Note 1) INX 4 2 1 0p Code Address 1 0p Code of Next Instruction Irrelevant Data (Note 1) PSH 1 1 0p Code Address 1 0p Code of Next Instruction Irrelevant Data (Note 1) PSH 1 1 0p Code Address 1 0p Code of Next Instruction Accumulator Data PUL 1 1 0p Code Address 1 0p Code of Next Instruction Accumulator Data PUL 1 1 0p Code Address 1 0p Code of Next Instruction Accumulator Data PUL 1 1 0p Code Address 1 0p Code of Next Instruction Irrelevant Data (Note 1) TSX 1 1 0p Code Address + 1 1 0p Code of Next Instruction Irrelevant Data (Note 1) TXS 1 1 0p Code Address + 1 1 0p Code of Next Instruction Irrelevant Data (Note 1) TXS 1 1 0p Code Address + 1	그는 것 같은 것 같은 것 같은 것을 가지고 말했다. 말 것 같은 것 같은 것 같은 것 같이 있다.	的名称	S. S.				
CLP ROL TPA COM SBA 1 1 Op Code Address 1 Op Code DES INS 1 1 1 Op Code Address 1 Op Code of Next Instruction INS 4 2 1 Op Code Address 1 1 Op Code of Next Instruction INS 4 2 1 Op Code Address 1 1 Irrelevant Data (Note 1) INX 4 0 New Register Contents 1 Irrelevant Data (Note 1) PSH 1 1 Op Code Address 1 Op Code of Next Instruction 4 0 Stack Pointer 0 Accumulator Data PUL 1 1 Op Code Address 1 Op Code of Next Instruction 4 0 Stack Pointer 1 Accumulator Data PUL 1 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 5 4 1	CLC NEG TAP						
COM SBA Image: SBA <thimage: sba<="" th=""> <thimage: sba<="" th=""></thimage:></thimage:>							한 알려난 생활을 했는 것은 것은 것 같아. 것이다. 사가 방법은 것 같은 것은 것은 것은 것이다. 것이다.
DES DEX INS INS111Op Code Address1Op Code Op Code of Next Instruction Irrelevant Data (Note 1) Irrelevant Data (Note 1)INS INX421Op Code Address + 11Irrelevant Data (Note 1) Irrelevant Data (Note 1)PSH111Op Code Address1Op Code4211Op Code Address1Op Code4211Op Code Address1Op Code431Stack Pointer0Accumulator DataPUL11Op Code Address1Op Code421Op Code Address1Op CodePUL11Op Code Address1Op Code430Stack Pointer - 11Op Code430Stack Pointer1Op Code421Op Code Address1Op Code430Stack Pointer1Op Code430Stack Pointer1Op Code741Op Code Address1Op Code430Stack Pointer1Irrelevant Data (Note 1)7410p Code Address1Op Code430Stack Pointer1Irrelevant Data (Note 1)7410p Code Address1Op Code430Index Register1Irrelevant							
DEX INS INS421Op Code Address + 11Op Code of Next Instruction Irrelevant Data (Note 1)PSH40New Register Contents1Irrelevant Data (Note 1)PSH411Op Code Address1Op Code421Op Code Address1Op Code of Next Instruction431Stack Pointer0Accumulator DataPUL40Stack Pointer1Op Code421Op Code Address1Op Code40Stack Pointer1Op Code421Op Code Address1Op CodePUL111Op Code Address1Op Code430Stack Pointer1Irrelevant Data (Note 1)PUL110p Code Address + 11Op Code421Op Code Address1Op Code430Stack Pointer1Irrelevant Data (Note 1)7XX11Op Code Address + 11Op Code30Stack Pointer1Irrelevant Data (Note 1)7XS110p Code Address + 11Op Code421Op Code Address + 11Op Code421Op Code Address + 11Op Code430Irrelevant DataIrrelevant Data7XS110p Code Addres			 	lander og som Det som som	On Code Address	<u>1977 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997 - 1997</u>	On Code
INS 4 3 0 Previous Register Contents 1 Irrelevant Data (Note 1) PSH 1 1 0p Code Address 1 Op Code 4 2 1 0p Code Address 1 Op Code 3 1 Stack Pointer 0 Accumulator Data PUL 1 1 0p Code Address 1 Op Code 4 0 Stack Pointer 0 Accumulator Data PUL 1 1 0p Code Address 1 Op Code 4 0 Stack Pointer 1 Accumulator Data PUL 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 9 0 Stack Pointer - 1 1 Op Code Of Next Instruction 4 2 1 Op Code Address 1 Op Code 4 1 Stack Pointer + 1 1 Op Code Op Code 4 3 0 Stack Pointer + 1 1 Op Code 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 7 1 1 Op Code Address 1 Op C	DEX				이 같은 것이 있는 것은 것이 있는 것이 있다. 이 것이 있는 것이 있다. 이 것이 있는 것이 있는 것이 있는 것이 있는 것 같은 것이 같은 것이 같은 것이 있는 것이 없는 것	2년 일을 청료를 들었으며	승규는 방법에 가장하는 것은 것이 같은 것이 있는 것이 같아요. 이 것이 없는 것이 같아요.
PSH111Op Code Address1Irrelevant Data (Note 1)PSH1110p Code Address10p Code4210p Code Address + 110p Code of Next Instruction31Stack Pointer0Accumulator Data40Stack Pointer - 11Accumulator DataPUL110p Code Address10p Code4210p Code Address10p Code430Stack Pointer - 11Op Code430Stack Pointer10p Code41Stack Pointer1Irrelevant Data (Note 1)41Stack Pointer + 11Op Code4210p Code Address1Op Code7XX110p Code Address1Op Code30Stack Pointer1Irrelevant Data (Note 1)41Op Code Address1Op Code30Stack Pointer1Irrelevant Data (Note 1)7XX110p Code Address1Op Code4210p Code Address1Op Code430Stack Pointer1Irrelevant Data (Note 1)7XX110p Code Address1Op Code40New Index Register1Irrelevant Data7XX110p Code Address1Op Code <td></td> <td>4</td> <td></td> <td>1.000</td> <td></td> <td></td> <td>[1] 이 같은 말했는 것이 같이 많이 많이 많이 있는 것이 많이 /td>		4		1.000			[1] 이 같은 말했는 것이 같이 많이 많이 많이 있는 것이 많이
PSH 1 1 1 0p Code Address 1 Op Code 4 2 1 Op Code Address 1 1 Op Code of Next Instruction 3 1 Stack Pointer 0 Accumulator Data 4 0 Stack Pointer 1 Accumulator Data PUL 1 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 3 0 Stack Pointer 1 Inrelevant Data (Note 1) 7SX 1 1 Op Code Address 1 Op Code of Next Instruction TSX 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 7SX 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 7XS 1 1 Op Code Address 1 Op Code of Next Instruction 1 1 Op Code Address 1 Op Code New Index Register					[17:16] 방송 20:16 - 17:16 - 27:17 - 28:26 - 28:26 - 28:26 - 28:26 - 28:26 - 28:26 - 28:26 - 28:26 - 28:26 - 28:2	말 아이지 않는 것이 같이 같이 같이 많이	[안영 생활값] 그 사람 그는 가슴에 가는 옷이 먹는 것이다.
421Op Code Address + 11Op Code of Next Instruction Accumulator Data Accumulator DataPUL40Stack Pointer - 11Accumulator Data Accumulator DataPUL411Op Code Address1Op Code421Op Code Address1Op Code of Next Instruction Irrelevant Data (Note 1)430Stack Pointer1Irrelevant Data (Note 1)7SX11Op Code Address1Op Code30Stack Pointer + 11Op Code30Stack Pointer + 11Op Code30Stack Pointer + 11Op Code421Op Code Address1Op Code7SX11Op Code Address1Op Code421Op Code Address1Op Code7XS421Op Code Address1Op Code421Op Code Address1Op Code7XS421Op Code Address1Op Code40New Index Register1Irrelevant DataIrrelevant Data7XS111Op Code Address1Op Code421Op Code Address1Op Code7XS111Op Code Address1Op Code7XS111Op Code Address1Irrelevant Data7	PSH		and the second sec	te ante Tell Asi			
4 3 1 Stack Pointer 0 Accumulator Data PUL 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 7SX 1 1 Op Code Address 1 Op Code 3 0 Stack Pointer + 1 1 Op Code 4 1 Op Code Address 1 Op Code 7SX 1 1 Op Code Address 1 Op Code 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 7XS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 7XS 1 1 Op Code Address 1 Op Code 4 3 0 Index Register 1 Irrelevant Data 7XS 1 1 Op Code Address 1 Op Code 4 0 New Stack Pointer 1 Irrelevant Data			2	1		한 김 사람이 같이 같이 같이 같이 같이 같이 같이 같이 않는 것이 같이 않는 것이 같이 많이 했다.	방법 방법 방법 가격에 가격을 가지 않는 것을 많이 하는 것을 했다.
PUL 1 Accumulator Data 4 0 Stack Pointer - 1 1 Op Code 4 2 1 Op Code Address 1 Op Code of Next Instruction 3 0 Stack Pointer 1 Op Code of Next Instruction Irrelevant Data (Note 1) 4 1 Stack Pointer 1 Op Code of Next Instruction Irrelevant Data (Note 1) 7SX 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 7XS 1 1 Op Code Address 1 Op Code 4 0 New Index Register 1 Irrelevant Data (Note 1) 7XS 1 1 Op Code Address + 1 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 3 0 Index Register 1 Irrelevant Data		4	한 방송을 잡다.	1	「作用を使い、ない」となってもなっては知られます。	0	전문 제품 물건적 것은 그 요즘 집안을 보는 것을 수요.
PUL 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 3 0 Stack Pointer 1 1 Op Code of Next Instruction TSX 1 1 Op Code Address 1 Op Code TSX 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 3 0 Stack Pointer + 1 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 1 1 1 Op Code Address 1 Op Code 4 0 New Index Register 1 Irrelevant Data (Note 1) TXS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 3 0 Index Register 1 Irrelevant Data 1 1 Op Code Address 1 Op Code 4 0 New Stack Pointer					· 영영화 영상 · 영화 · 영화 · 영화 · 영화 · 영화 · 영화 ·		
4 2 1 Op Code Address + 1 1 Op Code of Next Instruction Irrelevant Data (Note 1) TSX 1 1 Op Code Address 1 Op Code of Next Instruction TSX 1 1 Op Code Address 1 Op Code of Next Instruction TSX 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 7XS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 7XS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 2 <td< td=""><td>PIII</td><td>2 1.525 (* 5.97 2 5.9288 (* 3.</td><td></td><td>1.4</td><td></td><td></td><td>en en e</td></td<>	PIII	2 1.525 (* 5.97 2 5.9288 (* 3.		1.4			en e
4 3 0 Stack Pointer 1 Irrelevant Data (Note 1) TSX 1 1 Op Code Address 1 Op Code 4 1 0 Code Address 1 Op Code 2 1 Op Code Address 1 Op Code of Next Instruction 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 0 New Index Register 1 Irrelevant Data (Note 1) TXS 1 1 Op Code Address 1 Op Code 4 0 New Index Register 1 Irrelevant Data (Note 1) TXS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data 8 1 1 Op Code Address 1 Op Code 9 2 1 Op Code Address + 1 1 Irrelevant Data (Note 2)	IUL					이 같은 것을 많이 같다.	방법 수 있는 것을 많은 것이 가 바람을 많다. 것이 가 가지 않는 것이 같다.
4 1 Stack Pointer + 1 1 Operand Data from Stack TSX 1 1 1 Op Code Address 1 Op Code 2 1 0p Code Address + 1 1 0p Code of Next Instruction 3 0 Stack Pointer 1 Irrelevant Data (Note 1). 4 0 New Index Register 1 Irrelevant Data (Note 1). TXS 1 1 0p Code Address 1 Op Code 4 0 New Index Register 1 Irrelevant Data (Note 1). TXS 1 1 0p Code Address 1 Op Code 4 2 1 0p Code Address 1 Op Code 4 3 0 Index Register 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 2 1 0p Code Address 1 Op Code 2 1 0p Code Address 1 Irrelevant Data 8 2 1 0p Code Address 1 Irrelevant Data (Note 2) 5 </td <td></td> <td>4</td> <td></td> <td></td> <td>· · · · · · · · · · · · · · · · · · ·</td> <td></td> <td>전쟁 여기는 승규는 것은 것이 흔들었는 것이 가지 않는 것이다.</td>		4			· · · · · · · · · · · · · · · · · · ·		전쟁 여기는 승규는 것은 것이 흔들었는 것이 가지 않는 것이다.
TSX 1 1 0p Code Address 1 Op Code 2 1 Op Code Address + 1 1 Op Code of Next Instruction 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 0 New Index Register 1 Op Code TXS 1 1 Op Code Address 1 Op Code 4 0 New Index Register 1 Op Code Op Code TXS 1 1 0p Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code of Next Instruction TXS 1 1 0p Code Address 1 Op Code of Next Instruction TXS 1 1 0p Code Address 1 Op Code of Next Instruction RTS 1 1 0p Code Address 1 Op Code 2 1 0p Code Address 1 Op Code 2 1 0p Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) </td <td></td> <td></td> <td>dillon.</td> <td></td> <td>비행 중 사망 같은 것을 것을 것을 것 같아요. 것은 것을 것 같아요. 것은 것을 것을 것 같아요. 것은 것은 것을 것 같아요. 것은 것을 것 같아요. 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것을 것 같아요. 것은 것을 것 같아요. 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것이 같아요. 것은 것이 같아요. ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ?</td> <td></td> <td>- ' 방법에 있는 것 같은 것 같은 것이다. 이번 명이라는 것이다. 한 것이라 가지?</td>			dillon.		비행 중 사망 같은 것을 것을 것을 것 같아요. 것은 것을 것 같아요. 것은 것을 것을 것 같아요. 것은 것은 것을 것 같아요. 것은 것을 것 같아요. 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것을 것 같아요. 것은 것을 것 같아요. 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것은 것을 것 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것은 것은 것이 같아요. 것이 같아요. 것은 것이 같아요. ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ? ?		- ' 방법에 있는 것 같은 것 같은 것이다. 이번 명이라는 것이다. 한 것이라 가지?
4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 0 New Index Register 1 Irrelevant Data (Note 1) TXS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 2 1 Op Code Address 1 Op Code of Next Instruction 4 3 0 Index Register 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 5 3 0 Stack Pointer 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer 1 Irrelevant Data (Note 1) 1			4		* And Market and Annual br>Annual Annual br>Annual Annual br>Annual Annual br>Annual Annual br>Annual Annual Annua Annual Annual Annua Annual Annua		
4 -3 0 Stack Pointer 1 Irrelevant Data (Note 1) TXS 1 1 0 New Index Register 1 0 Op Code 4 2 1 0 Op Code Address 1 0p Code Op Code 4 2 1 0p Code Address 1 0p Code of Next Instruction 3 0 Index Register 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 2 1 1 0p Code Address 1 Op Code 8 1 1 0p Code Address 1 Op Code 2 1 0p Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer 1 Irrelevant Data (Note 1) 1 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herricevant Data (Note 1))	TSX				김해는 것을 중하는 것, 것과 모두 많아도 말했는 것을 얻는 것을 만들었다.	2011년 2013년	신경양 승규가 물려 온 것을 알고 있는 것이 있는 것이다.
A 0 New Index Register 1 Irrelevant Data (Note 1) TXS 1 1 0p Code Address 1 0p Code 4 2 1 0p Code Address 1 0p Code of Next Instruction 4 3 0 Index Register 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 8 1 1 0p Code Address 1 Op Code 8 1 1 0p Code Address 1 Op Code 8 1 1 0p Code Address 1 Op Code 6 3 0 Stack Pointer 1 Irrelevant Data (Note 2) 1 1 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer 1 Address of Next Instruction (Herrice)		4			. 방송가락 강성했는 것들은 것은 것이 한 것을 가져서 한 것을 했다.	지원은 감독 가지?	
TXS 1 1 Op Code Address 1 Op Code 4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 3 0 Index Register 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 5 3 0 Stack Pointer 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herein Code)					에 잘못 수많 수많은 것이 아니는 것은 것은 것이 가지 않는 것이 하는 것이다.		
4 2 1 Op Code Address + 1 1 Op Code of Next Instruction 3 0 Index Register 1 Irrelevant Data 4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 8 1 1 Op Code Address 1 Op Code 8 1 1 Op Code Address 1 Op Code 5 3 0 Stack Pointer 1 Irrelevant Data (Note 2) 1 1 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herricity)			te da estructura.			e Barro - Marcine Standa	
4 3 0 Index Register 1 Irrelevant Data RTS 1 1 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 0 Code Address 1 Op Code 2 1 0 Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herein Code)	TXS			的路底	에 상황 수상 수상 수상 전쟁이 가지 않는 것이 가지 않는 것이 않았다. 또 많은 것이다. 이 상태가 많은 것이 같은 것이 같은 것을 알았다. 것이 같은 것이 없는 것이 같은 것을 같은 것		
4 0 New Stack Pointer 1 Irrelevant Data RTS 1 1 Op Code Address 1 Op Code 2 1 Op Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Hereits)		4	이 한 것이 같아.	1.2.5	. 영화 방법 비행을 알았는 것이 못한 것 같이 있는 것 같아. 것 같아.	손에 다 물건을 생겼어?	[[문화] 문화 방송 신길에 가장 방송 가장 입니다. 그는 것이다.
RTS 1 1 Op Code Address 1 Op Code 2 1 Op Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Here)					[] MUE HEMELE 전 전 2017 11 11 11 11 11 11 11 11 11 11 11 11 1	이 집안에 다니지 않는다.	· 나는 나는 것 같은 것 같
2 1 Op Code Address + 1 1 Irrelevant Data (Note 2) 5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herein and the state)			21 - 12 - 12 - 12 - 12 - 12 - 12 - 12 -	6 - AND			
5 3 0 Stack Pointer 1 Irrelevant Data (Note 1) 4 1 Stack Pointer + 1 1 Address of Next Instruction (Herein and the state)	RTS					승규는 영화 가슴을 가슴을 가슴을 가슴을 다 다 나는 것이 가슴을 다 다 나는 것이 가슴을 다 다 나는 것이 가슴을 다 나는 것이 같이	情報 이번 방송 방송적인 것 같아. 이번 것은 것 같아. 가지 않는 것 같아. 가지 않는 것 같아.
4 1 Stack Pointer + 1 1 Address of Next Instruction (F	승규는 비 비행 관람을 얻는 것이 없는 것이 않는 것이 없는 것이 않이		2		[방황화] 관계하다 같이 걸 수밖에서 다 나라가 가지 않는다.	<u>े वि</u> 1 े	Irrelevant Data (Note 2)
4 1 Stack Pointer + 1 1 Address of Next Instruction (F		5	3	0	Stack Pointer	ं ।	Irrelevant Data (Note 1)
			4	1	Stack Pointer + 1	1	Address of Next Instruction (High
그는 사람이 있는 것 것 같아요. 2012년 2월 13월 2월							Order Byte)
5 1 Stack Pointer + 2 1 Address of Next Instruction (L Order Byte)	2011년 - 2012년 1월 19일 1943년 - 1921년 - 1921년 1월 19일		5	11	Stack Pointer + 2	1	Address of Next Instruction (Low

MOTOROLA Semiconductor Products Inc.



MOTOROLA Semiconductor Products Inc.

3501 ED BLUESTEIN BLVD., AUSTIN, TEXAS 78721 • A SUBSIDIARY OF MOTOROLA INC.

D59818-R2

A11138-7 PRINTED IN USA 9-81 IMPERIAL LITHO 98685 10,000

Address Mode		Cycle	VMA		R/W	
and Instructions	Cycles	#	Line	Address Bus	Line	Data Bus
NDEXED (Continued)					2011 - 101 101	
STA				Op Code Address	1	Op Code
		2		Op Code Address + 1	1	Offset
이 가지 않는 것을 가지 않는다. 이 가지 않는 것을 가지 않는다. 이 가지 않는다.	6	3	0	Index Register	1	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
		5	G 0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
		6	1.	Index Register Plus Offset	0	Operand Data
ASL LSR ASR NEG		1	91 1 년 1 년	Op Code Address	1	Op Code
CLR ROL		2	Ó	Op Code Address + 1		Offset
COM ROR DEC TST	7	- S - 4	0	Index Register		Irrelevant Data (Note 1)
INC		4 5	1	Index Register Plus Offset (w/o Carry)		Irrelevant Data (Note 1)
		6	0	Index Register Plus Offset Index Register Plus Offset		Current Operand Data
		7	1/0	Index Register Plus Offset	1	Irrelevant Data (Note 1)
			(Note	Index negister rius Onset	0	New Operand Data (Note 3)
			3)			
STS		1	が必要	Op Code Address	1	Op Code
STX		2		Op Code Address + 1	1	Offsøt
	7	3	0	Index Register	. ()	Irrelevant Data (Note 1)
		4	0	Index Register Plus Offset (w/o Carry) 🆼	1	Irrelevant Data (Note 1)
		5	0	Index Register Plus Offset	la p	Irrelevant Data (Note 1)
		6	1	Index Register Plus Offset	0	Operand Data (High Order Byte)
에는 이는 것에서 이가 가슴을 걸었다. 이는 이는 것이 가지 않는 것을 걸쳐졌다.		7	1	Index Register Plus Offset + 1	0	Operand Data (Low Order Byte)
JSR		1	$\gtrsim 1.1^{-10}$	Op Code Address	1	Op Code
		2	》。 1135	Op Code Address + 1	1	Offset
		3	0	Index Register	1	Irrelevant Data (Note 1)
사망 : 2011 : 2012 : 2014 - 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014 : 2014	8	4	1	Stack Pointer	0	Return Address (Low Order Byte)
		5	1	Stack Pointer — 1	0	Return Address (High Order Byte)
		6	0	Stack Pointer - 2	1	Irrelevant Data (Note 1)
		7	0. ₁₀ .	Index Register	1	Irrelevant Data (Note 1)
		8	0	Index Register Plus Offset (w/o Carry)	1	Irrelevant Data (Note 1)
EXTENDED		pro: 34				
JMP		1	1	Op Code Address	1	Op Code
	3	2		Op Code Address + 1	1	Jump Address (High Order Byte)
		3	1	Op Code Address + 2	1	Jump Address (Low Order Byte)
ADC EOR		1		Op Code Address	1	Op Code
ADD LDA		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte
AND ORA BIT SBC	4	3		Op Code Address + 1	1	Address of Operand (High Order Byte Address of Operand (Low Order Byte
CMP SUB		4	1	Address of Operand	1	Operand Data
CPX		1	1	Op Code Address	1	Op Code
LDS LDX		2	1	Op Code Address + 1	1	Address of Operand (High Order Byte
	5	3	1	Op Code Address + 2	1	Address of Operand (Low Order Byte
		4	18	Address of Operand	1	Operand Data (High Order Byte)
		5		Address of Operand + 1	1	Operand Data (Low Order Byte)
		1	1	Op Code Address	1	Op Code
STA B		2	1	Op Code Address + 1	1	Destination Address (High Order Byt
	5	3		Op Code Address + 2	1	Destination Address (Low Order Byte
		4	0	Operand Destination Address	1	Irrelevant Data (Note 1)
		5	1	Operand Destination Address	0	Data from Accumulator
ASL LSR		1	5.18	Op Code Address	1	Op Code
ASR NEG		2	広日135 1日1日	Op Code Address + 1	1	Address of Operand (High Order Byte
CLR ROL COM ROR		3	1	Op Code Address + 2	10.5 185	
DEC TST	6	4	1	Address of Operand		Address of Operand (Low Order Byte
INC			0	[K. 11] 2016년 1월 2017년 1월 2017	1	Current Operand Data
요즘		6	1/0	Address of Operand Address of Operand	1	Irrelevant Data (Note 1)
이 집에서 나는 이 것은 지갑을 수도로 했다.	네. 소설화전	1. O C C	r ://∪ :	Audress of Operand	0	New Operand Data (Note 3)

MOTOROLA Semiconductor Products Inc.