NEC
NEC Electronics Inc.

μPD78PG11
HIGH-END, 8-BIT, SINGLE-CHIP NMOS MICROCOMPUTER WITH PIGGYBACK EPROM

Description

The NEC μPD78PG11 is a prototyping device used to emulate the masked-ROM μPD7811. The user can insert a standard EPROM (2732A or 2764) into the terminals on top of the μPD78PG11. The program will be executed from the EPROM just as it would be executed from the masked-ROM on the μPD7811.

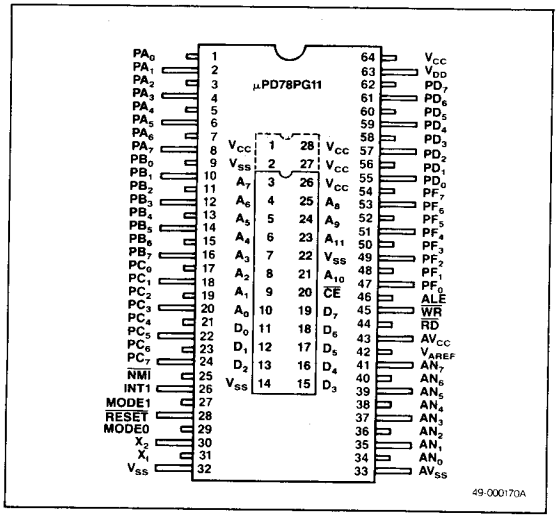
Features

- ☐ NMOS silicon gate technology requiring +5V power supply
- ☐ Complete single-chip microcomputer
  - — 16-bit ALU
  - — 4K-EPROM (via piggyback socket)
  - — 256-byte RAM
- ☐ 44 I/O lines
- ☐ Two zero-cross detect inputs
- ☐ Two 8-bit timers
- ☐ Multifunction 16-bit timer/event counter
- ☐ Expansion capabilities
  - — 8085A bus-compatible
  - — 60K-byte external memory address range
- ☐ 8-channel, 8-bit A/D converter
  - — Autoscan mode
  - — Channel select mode
- ☐ Full duplex USART
  - — Synchronous and asynchronous
- ☐ 153 instruction set
  - — 16-bit arithmetic, multiply, and divide
- ☐ 1μs instruction cycle time (12MHz operation)
- ☐ Prioritized interrupt structure
  - — 2 external
  - — 9 internal
- ☐ Standby function
- ☐ On-chip clock generator

Ordering Information

Part Number	Package Type	Max Frequency of Operation
μPD78PG11E	64-pin ceramic piggyback QUIP	12MHz

Pin Configuration



Terminal Identification (Note 1)

Pin No.	Symbol	Function
1	V_{CC}	Provides V_{PP} pin of μPD2764 with 5V.
2	V_{SS}	Maintains 0V on A_{12} address pin of μPD2764 forcing all instruction fetches from lower 4K of EPROM.
3 — 10, 21 23 — 25	A_0 — A_{11}	Address Bus. Outputs lower order 12 bits of the program counter which will be used as an EPROM address signal.
11 — 13, 15 — 19	D_0 — D_7	Data Bus. Inputs data read from EPROM.
14	V_{SS}	Connects to the GND terminal of EPROM.
20	$\overline{CE}$	Chip Enable. Outputs an EPROM chip enable signal.
22	V_{SS}	Ties EPROM $\overline{OE}$ signal to V_{SS}.
26	V_{CC}	Provides μPD2732A with +5V V_{CC} power supply.
27	V_{CC}	Maintains +5V on $\overline{PGM}$ pin of μPD2764.
28	V_{CC}	Provides μPD2764 with +5V V_{CC} power supply.

Note:

(1) Connections from μPD78PG11 to EPROM.

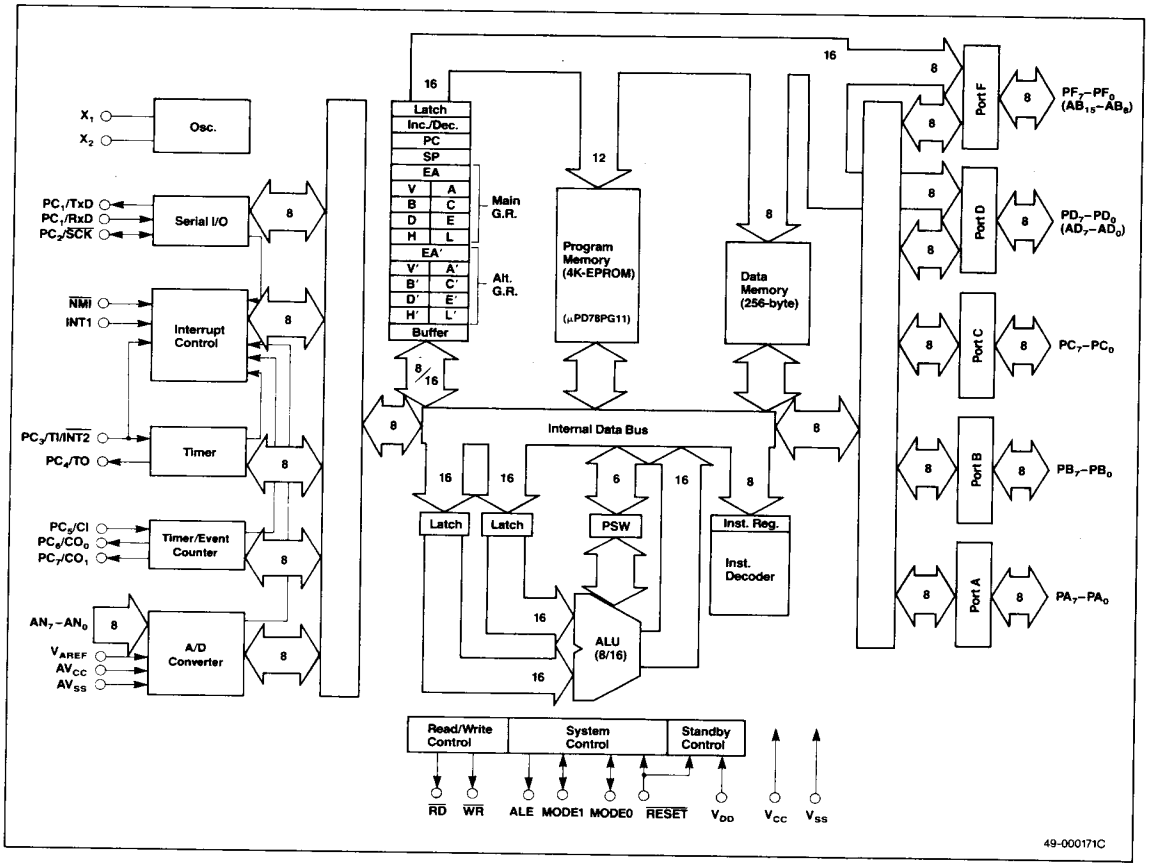
4

Pin Identification

Pin No.	Pin Symbol	Function
1—8	PA_0—PA_7	Port A: (Three-state input/output) 8-bit programmable I/O port. Each line independently programmable as an input or output. Reset places all lines of Port A in input mode.
9—16	PB_0—PB_7	Port B: (Three-state input/output) 8-bit programmable I/O port. Each line independently programmable as an input or output. Reset places all lines of Port B in input mode.
17—24	PC_0—PC_7	Port C: (Three-state input/output) 8-bit programmable I/O port. Each line independently programmable as an input or output. Alternatively, Port C may be used as control lines for USART and timer. Reset puts Port C in port mode and all lines in input mode.
17	PC_0	Transmit Data (T x D): Serial data output terminal.
18	PC_1	Receive Data (R x D): Serial data input terminal.
19	PC_2	Serial Clock ($\overline{SCK}$): Serial clock input/output terminal. When internal clock is used, the output can be selected: when an external clock is used, the input can be selected.
20	PC_3	Timer Input (TI)/interrupt request input ($\overline{INT2}$): Timer clock input terminal; can also be used as falling edge, maskable-interrupt input terminal and AC input zero-cross detection terminal.
21	PC_4	Timer Output (TO): This output signal is a square wave whose frequency is determined by the timer/counter.
22	PC_5	Counter Input (CI): External pulse input terminal to the timer/event counter.
23, 24	PC_6, PC_7	Counter Outputs 0, 1 (CO_0—CO_1): Programmable rectangular wave output terminal based on timer/event counter.
25	$\overline{NMI}$	Falling-edge, nonmaskable interrupt ($\overline{NMI}$) input.
26	INT1	A rising-edge, maskable interrupt input. Alternatively, can be used for a zero-cross detection AC input.
27	MODE1	Used as input in conjunction with MODE0 to select appropriate memory expansion mode. Also outputs M1 signal during each opcode fetch.
28	$\overline{RESET}$	(Input, active low). $\overline{RESET}$ initializes the μPD78PG11.
29	MODE0	Used as input in conjunction with MODE1 to select appropriate memory expansion mode. Also used to output IO/M.
30—31	X_2, X_1 (crystal)	A crystal connection terminal for system clock osciliation. When an external clock is supplied X_1 is the input.
32	V_{SS}	Power supply ground potential.
33	AV_{SS}	A/D converter power supply ground potential. Sets conversion range lower limit.
34—41	AN_0—AN_7	Eight analog inputs to the A/D converter. AN_7—AN_4 can also be used as a digital input port for falling edge detection.
42	V_{AREF}	Reference voltage for A/D converter. Sets conversion range upper limit.
43	AV_{CC}	Power supply voltage for A/D converter.
44	$\overline{RD}$	(Three-state output, active low) $\overline{RD}$ is used as a strobe to gate data from external devices onto the data bus. $\overline{RD}$ goes high during Reset.
45	$\overline{WR}$	(Three-state output, active low) $\overline{WR}$, when active, indicates that the data bus holds valid data. Used as a strobe signal for external memory or I/O write operations. $\overline{WR}$ goes high during Reset.
46	ALE	The strobe signal is for latching the address signal to the output from PD_7—PD_0 when accessing external expansion memory.
47—54	PF_0—PF_7	Port F: (Three-state input/output) 8-bit programmable I/O port. Each line configurable independently as an input or output. Address Bus: When external expansion memory is used, multiplexed address/data bus can be selected.
55—62	PD_0—PD_7	Port D: 8-bit programmable I/O port. This byte can be designated as either input or output. Address Bus: When external expansion memory is used, multiplexed address/data bus can be selected.
63	V_{DD}	Backup power terminal for on-chip RAM.
64	V_{CC}	+5V power supply.

Notes: 1 clock cycle = 1 CL = 3/f.
1 machine cycle = 3 or 4 clock cycles.
1 instruction cycle = 1 to 19 machine cycles.
f: System clock frequency (MHz).

Block Diagram



4

Absolute Maximum Ratings

T_A = 25°C

Power Supply Voltage V_{CC}, V_{DD}, AV_{CC}	−0.5V to + 7.0V
Input Voltage, V_I	−0.5V to + 7.0V
Output Voltage, V_O	−0.5V to + 7.0V
Reference Input Voltage, V_{AREF}	−0.5V to + V_{CC}
Operating Temperature, T_{ORP}	
10MHz<f_{XTAL}≤12MHz	−10°C to + 70°C
f_{XTAL}≤10MHz	−40°C to + 85°C
Storage Temperature, T_{STG}	−65°C to + 150°C

Comment: Exposing the device to stresses above those listed in Absolute Maximum Ratings could cause permanent damage. The device is not meant to be operated under conditions outside the limits described in the operational sections of this specification. Exposure to absolute maximum rating conditions for extended periods may affect device reliability.

Operating Conditions

Oscillator Frequency	T_A	V_{CC}, AV_{CC}
10MHz<f_{XTAL}≤12MHz	−10°C to + 70°C	+ 5.0V ± 5%
f_{XTAL}≤10MHz	−40°C to + 85°C	± 5.0V + 10%

Capacitance

T_A = 25°C; V_{CC} = V_{DD} = V_{SS} = 0V

		Limits				
Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Capacitance	C_I			10	pF	All Af_C = 1MHz Unmeasured pin returned to 0V.
Output Capacitance	C_O			20	pF	
I/O Capacitance	C_{IO}			20	pF	

DC Characteristics

T_A = −10°C to +70°C; V_{CC} = −5.0 V ±5%; V_{SS} = 0 V;
V_{CC} −0.8 V ≤ V_{DD} ≤ V_{CC}

		Limits				
Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Input Low Voltage	V_{IL}	0		0.8	V	
Input High Voltage	V_{IH1}	2.0		V_{CC}	V	All except SCK, RESET and X_1
	V_{IH2}	0.8V_{CC}		V_{CC}	V	SCK, X_1
	V_{IH3}	0.8V_{DD}		V_{CC}	V	RESET
Output Low Voltage	V_{OL}			0.45	V	I_{OL} = 2.0mA
Output High Voltage	V_{OH}	2.4			V	I_{OH} = −200μA
Input Current	I_I			± 200	μA	INT1, TI (PC_3); + 0.45V ≤ V_{IN} < V_{CC}
Input Leakage Current	I_{LI}			± 10	μA	All except INT1, TI(PC_3) 0V ≤ V_{IN} ≤ V_{CC}
Output Leakage Current	I_{LO}			± 10	μA	+ 0.45V ≤ V_O ≤ V_{CC}
V_{DD} Supply Current	I_{DD}		1.5	3.5	mA	T_A = −40°C to + 85°C
V_{CC} Supply Current	I_{CC}		140[1]	250	mA	T_A = −40°C to + 85°C

Note: 1. T_A = 25°C; V_{CC} = V_{DD} = +5.0 V

Zero-cross Characteristics

		Limits				
Parameter	Symbol	Min	Typ	Max	Unit	Test Conditions
Zero-cross Detection Input	V_{ZX}	1		3	VAC_{P-P}	AC Coupled
Zero-cross Accuracy	A_{ZX}			± 135	mV	60Hz Sine Wave
Zero-cross Detection Input Frequency	f_{ZX}	0.06		1	kHz	

Serial Operation

Parameter	Symbol	Limits Min	Typ	Max	Unit	Test Conditions
$\overline{SCK}$ Cycle Time	t_{CYK}	1			μs	$\overline{SCK}$ Input 1, 2
		500			ns	
		2			μs	$\overline{SCK}$ Output
$\overline{SCK}$ Width Low	t_{KKL}	400			ns	$\overline{SCK}$ Input 1, 2
		200			ns	
		900			ns	$\overline{SCK}$ Output
$\overline{SCK}$ Width High	t_{KKH}	400			ns	$\overline{SCK}$ Input 1, 2
		200			ns	
		900			ns	$\overline{SCK}$ Output
RxD Set-up Time to $\overline{SCK}$ ↑	t_{RXK}	80			ns	Note 1
RxD Hold Time After $\overline{SCK}$ ↑	t_{KRS}	80			ns	Note 1
$\overline{SCK}$ ↓ TxD Delay Time	t_{KTX}			210	ns	Note 1

Notes: 1. 1x Baud rate in Asynchronous, Synchronous, or I/O interface mode.
2. 16x Baud rate or 64x Baud rate in Asynchronous mode.

A/D Converter Characteristics

$T_A = -10°C$ to $+70°C$; $V_{CC} = AV_{CC} = +5.0\,V \pm 5\%$; $V_{SS} = AV_{SS} = 0\,V$; $AV_{CC} - 0.5\,V \leq V_{AREF} \leq AV_{CC}$

Parameter	Symbol	Limits Min	Typ	Max	Unit	Test Conditions
Resolution		8			Bits	
Absolute Accuracy			0.4% ± ½		LSB	$T_A = -10°C$ to $+50°C$
			0.6% ± ½		LSB	$T_A = -10°C$ to $+70°C$ Note 1
Conversion Time	t_{CONV}	576			t_{CYC}	$83ns \leq t_{CYC} \leq 110ns$
		432			t_{CYC}	$110ns \leq t_{CYC} \leq 170ns$
Sampling Time	t_{SAMP}	96			t_{CYC}	$83ns \leq t_{CYC} \leq 110ns$
		72			t_{CYC}	$110ns \leq t_{CYC} \leq 170ns$
Analog Input Voltage	V_{IA}	0		V_{AREF}	V	

Note: 1. In case of $f_{XTAL} \leq 10$ MHz, $T_A = -40°C$ to $+85°C$.

Bus Timing Depending on t_{CYC}

Symbol	Calculating Expression	Min/Max
t_{RP}	60T	Min
t_T	6T	Min
t_{CI}[2]	6T	Min
t_{CI}[3]	48T	Min
t_{IP}	36T	Min
t_{AL}	2T - 100	Min
t_{LA}	T - 30	Min
t_{AR}	3T - 100	Min
t_{AD}	7T - 220	Max
t_{LDR}	5T - 200	Max
t_{RD}	4T - 150	Max
t_{LR}	T - 50	Min
t_{RL}	2T - 50	Min
t_{RR}	4T - 50 (Data Read)	Min
	7T - 50 (Opcode Fetch)	
t_{LL}	2T - 40	Min
t_{AW}	3T - 100	Min
t_{LDW}	T + 110	Max
t_{LW}	T - 50	Min
t_{DW}	4T - 100	Min
t_{WDH}	2T - 70	Min
t_{WL}	2T - 50	Min
t_{WW}	4T - 50	Min
t_{CYK}	12T ($\overline{SCK}$ Input)[1]	Min
	24T ($\overline{SCK}$ Output)	
t_{KKL}	6T - 100 ($\overline{SCK}$ Input)[1]	Min
	12T - 100 ($\overline{SCK}$ Output)	
t_{KKH}	6T - 100 ($\overline{SCK}$ Input)[1]	Min
	12T - 100 ($\overline{SCK}$ Output)	

Notes: 1. 1x Baud rate in Asynchronous, Synchronous, or I/O interface mode. $T = t_{CYC} = 1\,f_{XTAL}$. The items not included in this list are independent of oscillator frequency (f_{XTAL}).
2. Event Counter mode.
3. Pulse Width Measurement mode.

4

AC Characteristics

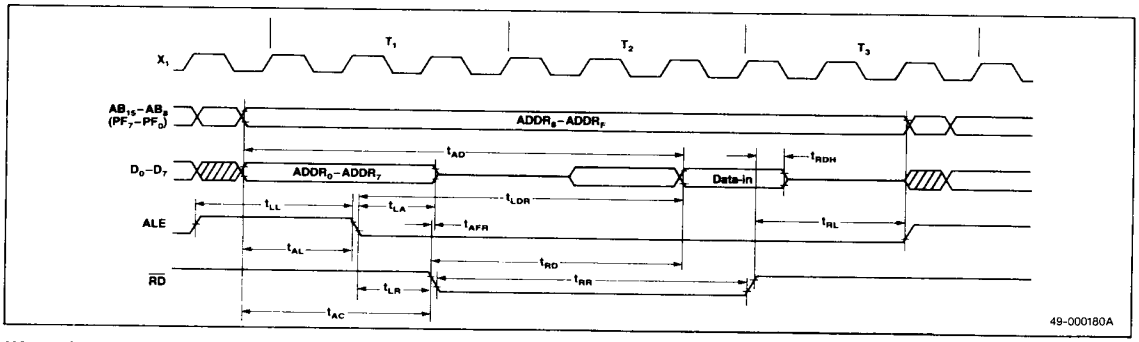
$T_A = 10°C$ to $+70°C$; $V_{CC} = +5.0\ V \pm 5\%$; $V_{SS} = 0\ V$; $V_{CC} - 0.8\ V \leq V_{DD} \leq V_{CC}$

Read/Write Operation		Limits					Test Conditions
		f_{XTAL} = 10 MHz		f_{XTAL} = 12 MHz			
Parameter	Symbol	Min	Max	Min	Max	Unit	1
$\overline{\text{Reset}}$ Pulse Width	t_{RP}	6.0		5.0		μs	
Interrupt Pulse Width	t_{IP}	3.6		3.0		μs	
Counter Input Pulse Width	t_{CI} 2	600		500		ns	
	t_{CI} 3	4.8		4.0		μs	
Timer Input Pulse Width	t_{TI}	600		500		ns	
X_1 Input Cycle Time	t_{CYC}	100		83		ns	
Address Set-Up to ALE ↓	t_{AL}	100		65		ns	
Address Hold after ALE ↓	t_{LA}	70		50		ns	
Address to $\overline{\text{RD}}$ ↓ Delay Time	t_{AR}	200		150		ns	
$\overline{\text{RD}}$ ↓ to Address Floating	t_{AFR}		20		20	ns	
Address to Data Input	t_{AD}		480		360	ns	
ALE ↓ to Data Input	t_{LDR}		300		215	ns	
$\overline{\text{RD}}$ ↓ to Data Input	t_{RD}		250		180	ns	
ALE ↓ to $\overline{\text{RD}}$ ↓ Delay Time	t_{LR}	50		35		ns	
Data Hold Time to $\overline{\text{RD}}$ ↑	t_{RDH}	0		0		ns	
$\overline{\text{RD}}$ ↑ to ALE ↑ Delay Time	t_{RL}	150		115		ns	
$\overline{\text{RD}}$ Width Low	t_{RR}	350		280		ns	Data Read
		650		530		ns	Opcode Fetch
ALE Width High	t_{LL}	160		125		ns	
Address to $\overline{\text{WR}}$ ↓ Delay	t_{AW}	200		150		ns	
ALE ↓ to Data Output	t_{LDW}		210		195	ns	
$\overline{\text{WR}}$ ↓ to Data Output	t_{WD}	130		100		ns	
ALE ↓ to $\overline{\text{WR}}$ ↓ Delay	t_{LW}	50		35		ns	
Data Set-up Time to $\overline{\text{WR}}$ ↑	t_{DW}	300		230		ns	
Data Hold Time to $\overline{\text{WR}}$ ↑	t_{WDH}	130		95		ns	
$\overline{\text{WR}}$ ↑ to ALE ↑ Delay Time	t_{WL}	150		115		ns	
$\overline{\text{WR}}$ Width Low	t_{WW}	350		280		ns	

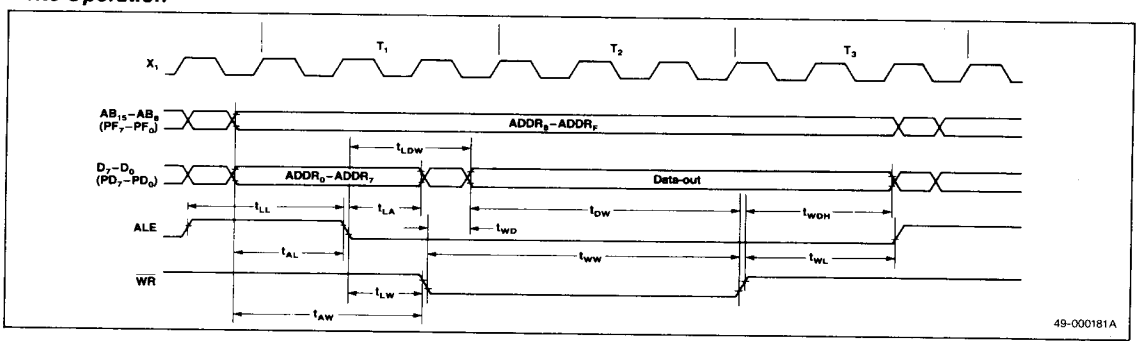
Notes: 1. Load Capacitance: $C_L = 150$ pF. 2. Event counter mode. 3. Pulse width measurement mode.

Timing Waveforms

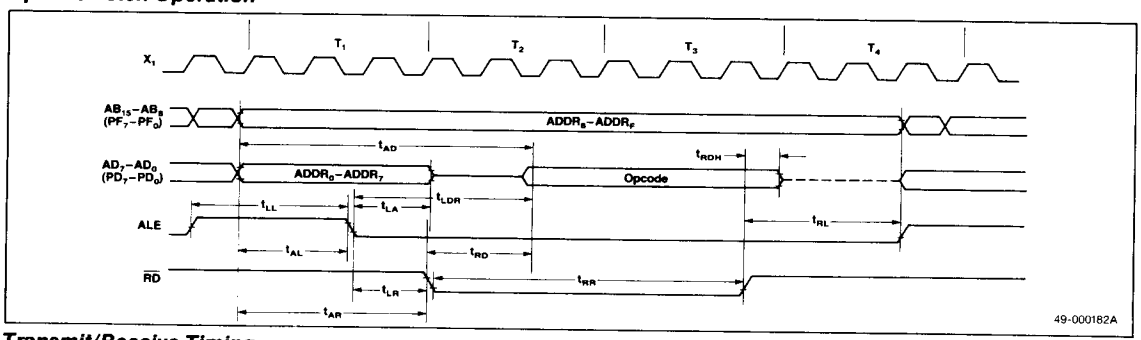
Read Operation



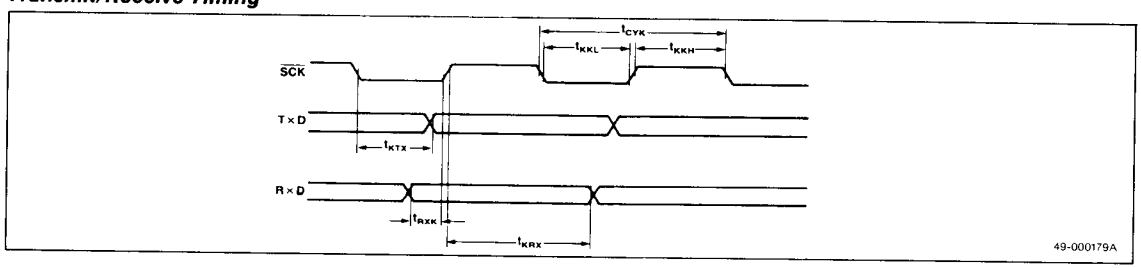
Write Operation



Opcode Fetch Operation



Transmit/Receive Timing



Remarks

1. sr—sr4 (special register)

PA	= Port A	ECNT	= Timer/Event Counter Upcounter
PB	= Port B	ECPT	= Timer/Event Counter Capture
PC	= Port C	ETMM	= Timer/Event Counter Mode
PD	= Port D	EOM	= Timer/Event Counter Output Mode
PF	= Port F	ANM	= A/D Channel Mode
MA	= Mode A	CR_0 to CR_3	= A/D Conversion Results 0—3
MB	= Mode B	TxB	= TxBuffer
MC	= Mode C	RxB	= RxBuffer
MCC	= Mode Control C	SMH	= Serial Mode High
MF	= Mode F	SML	= Serial Mode Low
MM	= Memory Mapping	MKH	= Mask High
TM_0	= Timer Register 0	MKL	= Mask Low
TM_1	= Timer Register 1		
TMM	= Timer Mode		
ETM_0	= Timer Event Counter Register 0		
ETM_1	= Timer Event Counter Register 1		

2. rp—rp3 (register pair)

SP	= Stack Pointer	H	= HL
B	= BC	V	= VA
D	= DE	EA	= Extended Accumulator

3. rpa—rpa3 (rp addressing)

B	= (BC)	D++	= (DE)++
D	= (DE)	H++	= (HL)++
H	= (HL)	D+byte	= (DE+byte)
D+	= (DE)+	H+A	= (HL+A)
H+	= (HL)+	H+B	= (HL+B)
D-	= (DE)-	H+EA	= (HL+EA)
H-	= (HL)-	H+byte	= (HL+byte)

4. (flag)

CY = Carry	HC = Half Carry	Z = Zero

5. irf (interrupt flag)

FNMI	= INTFNMI	FSR	= INTFSR
FT0	= INTFT0	FST	= INTFST
FT1	= INTFT1	ER	= Error
F1	= INTF1	OV	= Overflow
F2	= INTF2	AN_4 to AN_4	= Analog Input 4—7
FE0	= INTFE0	SB	= Standby
FE1	= INTFE1		
FEIN	= INTFEIN		
FAD	= INTFAD		

Instruction Set Symbol Definitions

Symbol	Description
←	Transfer direction, result
∧	Logical product (logical AND)
∨	Logical sum (logical OR)
∀	Exclusive OR
—	Complement
•	Concatenation

Instruction Groups

8-bit Data Transfer

Mnemonic		Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State[1]	Operation	Skip Condition
MOV		r1, A	$00011T_2T_1T_0$				4	(r1)←(A)	
		A, r1	$00001T_2T_1T_0$				4	(A)←(r1)	
	*	sr, A	01001101	$110S_4S_3S_2S_1S_0$			10	(sr)←(A)	
	*	A, sr1	01001100	$11S_5S_4S_3S_2S_1S_0$			10	(A)←(sr1)	
		r, word	01110000	$01101R_2R_1R_0$	Low Addr	High Addr	17	(r)←(word)	
		word, r	01110000	$01111R_2R_1R_0$	Low Addr	High Addr	17	(word)←(r)	
MVI	*	r, byte	$01101R_2R_1R_0$	Data			7	(r)←byte String skip, when r = A or L	
		sr2, byte	01100100	$S_30000S_2S_1S_0$	Data		14	(sr2)←byte	
MVIW	*	wa, byte	01110001	Offset	Data		13	((V), (wa))←byte	
MVIX	*	rpa1, byte	$010010A_1A_0$	Data			10	(rpa1)←byte	
STAW	*	wa	01100011	Offset			10	((V)(wa))←A	

Instruction Groups (cont)

Mnemonic		Operand	Opcode B1	B2	B3	B4	State [1]	Operation	Skip Condition
8-Bit Data Transfer (cont)									
LDAW	*	wa	00000001	Offset			10	(A)←((V), (wa))	
STAX	*	rpa2	$A_3$0111$A_2A_1A_0$	Data [2]			7/13 [3]	(rpa2)←(A)	
LDAX	*	rpa2	$A_3$0101$A_2A_1A_0$	Data [2]			7/13 [3]	(A)←((rpa2))	
EXX			00010001				8	(B)↔(B'), (C)↔(C'), (D)↔(D') (E)↔(E'), (H)↔(H'), (L)↔(L')	
EXA			00010000				8	(V)↔(V'), (A)↔(A'), (EA)↔(EA')	
EXH			01010000				8	(H)↔(H'), (L)↔(L')	
16-bit Data Transfer									
BLOCK			00110001				13 (C + 1)	((DE))←((HL)), (DE)←(DE) +), (HL)←(HL) + 1, (C)←(C) − 1 End if borrow	
DMOV		rp3, EA	101101P_1P_0				4	($rp3_L$)←(EAL), ($rp3_H$)←(EAH)	
		EA, rp3	101001P_1P_0				4	(EAL)←($rp3_L$), (EAH)←($rp3_H$)	
		sr3, EA	01001000	1101001U_0			14	(sr3)←(EA)	
		EA, sr4	↓	110000V_1V_0			14	(EA)←(sr4)	
SBCD		word	01110000	00011110	Low Addr	High Addr	20	(word)←(C), ((word) + 1) ←(B)	
SDED		word	↓	00101110	↓	↓	20	(word)←(E), ((word) + 1) ←(D)	
SHLD		word	↓	00111110	↓	↓	20	(word)←(L), ((word) + 1) ←(H)	
SSPD		word	↓	00001110	↓	↓	20	(word)←(SP_L), ((word + 1) ← (SP_H)	
STEAX		rpa3	01001000	1001$C_3C_2C_1C_0$	Data [4]		14/20 [3]	((rpa3))←(EAL), ((rpa3) + 1)← (EAH)	
LBCD		word	01110000	00011111	Low Addr	High Addr	20	(C)←(word), (B)←((word) + 1)	
LBCD		word	↓	00101111	↓	↓	20	(E)←(word), (D)←((word) + 1)	
LHLD		word	↓	00111111	↓	↓	20	(L)←(word), (H)←((word) + 1)	
LSPD		word	↓	00001111	↓	↓	20	(SP_L)←(word), (SP_H)←((word) + 1)	
LDEAX		rpa3	01001000	1000$C_3C_2C_1C_0$	Data [4]		14/20 [3]	(EAL)←((rpa3)), (EAH)←((rpa3) + 1)	
PUSH		rp1	10110$Q_2Q_1Q_0$				13	((SP) − 1)←($rp1_H$)((SP) − 2) ← ($rp1_L$)(SP)←(SP) − 2	
POP		rp1	10100$Q_2Q_1Q_0$				10	($rp1_L$)←((SP)), ($rp1_H$)←((SP) + 1) (SP)←(SP) + 2)	
LXI	*	rp2, word	0$P_2P_1P_0$0100		Low Byte	High Byte	10	(rp2)←(word) String skip when rp2 = H	
8-bit Aritmetic (Register)									
TABLE			01001000	10101000			17	(C)←((PC) + 3 + (A)) B←((PC) + 3 + (A) + 1)	
ADD		A, r	01100000	11000$R_2R_1R_0$			8	(A)←(A) + (r)	
		r, A	↓	01000$R_2R_1R_0$			8	(r)←(r) + (A)	
ADC		A, r	↓	11010$R_2R_1R_0$			8	(A)←(A) + (r) + (CY)	
		r, A	↓	01010$R_2R_1R_0$			8	(r)←(r) + (A) + (CY)	

4

Instruction Groups (cont)

8-Bit Arithmetic (Register) (cont)

Mnemonic	Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
ADDNC	A, r	01100000	10100$R_2R_1R_0$			8	(A)←(A) + (r)	No Carry
	r, A		00100$R_2R_1R_0$			8	(r)←(r) + (A)	No Carry
SUB	A, r		11100$R_2R_1R_0$			8	(A)←(A) − (r)	
	r, A		01100$R_2R_1R_0$			8	(r)←(r) − (A)	
SBB	A, r		11110$R_2R_1R_0$			8	(A)←(A) − (r) − (CY)	
	r, A		01110$R_2R_1R_0$			8	(r)←(r) − (A) − (CY)	
SUBNB	A, r		10110$R_2R_1R_0$			8	(A)←(A) − (r)	No Borrow
	r, A		00110$R_2R_1R_0$			8	(r)←(r) − (A)	No Borrow
ANA	A, r		10001$R_2R_1R_0$			8	(A)←(A) ∧ (r)	
	r, A		00001$R_2R_1R_0$			8	(r)←(r) ∧ (A)	
ORA	A, r	01100000	10011$R_2R_1R_0$			8	(A)←(A)∨(r)	
	r, A		00011$R_2R_1R_0$			8	(r)←(r)∨(A)	
XRA	A, r		10010$R_2R_1R_0$			8	(A)←(A)∨̄(r)	
	r, A		00010$R_2R_1R_0$			8	(r)←(r)∨̄(A)	
GTA	A, r		10101$R_2R_1R_0$			8	(A) − (r) − 1	No Borrow
	r, A		00101$R_2R_1R_0$			8	(r) − (A) − 1	No Borrow
LTA	A, r		10111$R_2R_1R_0$			8	(A) − (r)	Borrow
	r, A		00111$R_2R_1R_0$			8	(r) − (A)	Borrow
NEA	A, r		11101$R_2R_1R_0$			8	(A) − (r)	No Zero
	r, A		01101$R_2R_1R_0$			8	(r) − (A)	No Zero
EQA	A, r		11111$R_2R_1R_0$			8	(A) − (r)	Zero
	r, A		01111$R_2R_1R_0$			8	(r) − (A)	Zero
ONA	A, r		11001$R_2R_1R_0$			8	(A)∧(r)	No Zero
OFFA	A, r	01100000	11011$R_2R_1R_0$			8	(A)∧(r)	Zero

8-bit Arithmetic (Memory)

ADDX	rpa	01110000	11000$A_2A_1A_0$			11	(A)←(A) + ((rpa))	
ADCX	rpa		11010$A_2A_1A_0$			11	(A)←(A) + ((rpa)) + (CY)	
ADDNCX	rpa		10100$A_2A_1A_0$			11	(A)←(A) + ((rpa))	No Carry
SUBX	rpa		11100$A_2A_1A_0$			11	(A)←(A) − ((rpa))	
SBBX	rpa		11110$A_2A_1A_0$			11	(A)←(A) − ((rpa)) − (CY)	
SUBNBX	rpa		10110$A_2A_1A_0$			11	(A)←(A) − ((rpa))	No Borrow
ANAX	rpa		10001$A_2A_1A_0$			11	(A)←(A) − ((rpa))	
ORAX	rpa		10011$A_2A_1A_0$			11	(A)←(A)∨((rpa))	
XRAX	rpa		10010$A_2A_1A_0$			11	(A)←(A)∨̄((rpa))	
GTAX	rpa		10101$A_2A_1A_0$			11	(A) − ((rpa)) − 1	No Borrow
LTAX	rpa		10111$A_2A_1A_0$			11	(A) − ((rpa))	Borrow
NEAX	rpa		11101$A_2A_1A_0$			11	(A) − ((rpa))	No Zero
EQAX	rpa		11111$A_2A_1A_0$			11	(A) − ((rpa))	Zero
ONAX	rpa		11001$A_2A_1A_0$			11	(A)←((rpa))	No Zero
OFFAX	rpa		11011$A_2A_1A_0$			11	(A)←((rpa))	Zero

Instruction Groups (cont)

Mnemonic		Operand	Immediate Data Opcode B1	B2	B3	B4	State [1]	Operation	Skip Condition
ADI	*	A, byte	01000110	←Data→			7	(A)←(A) + byte	
		r, byte	01110100	$01000R_2R_1R_0$	Data		11	(r)←(r) + byte	
		sr2, byte	0110 ↓	$S_31000S_2S_1S_0$	↓		20	(sr2)←(sr2) + byte	
ACI	*	A, byte	01010110	←Data→			7	(A)←(A) + byte + (CY)	
		r, byte	01110100	$01010R_2R_1R_0$	Data		11	(r)←(r) + byte + (CY)	
		sr2, byte	0110 ↓	$S_31010S_2S_1S_0$	↓		20	(sr2)←(sr2) + byte + (CY)	
ADINC	*	A, byte	00100110	←Data→			7	(A)←(A) + byte	No Carry
		r, byte	01110100	$00100R_2R_1R_0$	Data		11	(r)←(r) + byte	No Carry
		sr2, byte	0110 ↓	$S_30100S_2S_1S_0$	↓		20	(sr2)←(sr2) + byte	No Carry
SUI	*	A, byte	01100110	←Data→			7	(A)←(A) - byte	
		r, byte	01110100	$01100R_2R_1R_0$	Data		11	(r)←(r) - byte	
		sr2, byte	0110 ↓	$S_31100S_2S_1S_0$	↓		20	(sr2)←(sr2) - byte	
SBI	*	A, byte	01110110	←Data→			7	(A)←(A) - byte - (CY)	
		r, byte	01110100	$01110R_2R_1R_0$	Data		11	(r)←(r) - byte - (CY)	
		sr2, byte	0110 ↓	$S_31110S_2S_1S_0$	↓		20	(sr2)←(sr2) - byte - (CY)	
SUINB	*	A, byte	00110110	←Data→			7	(A)←(A) - byte	No Borrow
		r, byte	01110100	$00110R_2R_1R_0$	Data		11	(r)←(r) - byte	No Borrow
		sr2, byte	0110 ↓	$S_30110S_2S_1S_0$	↓		20	(sr2)←(sr2) - byte	No Borrow
ANI	*	A, byte	00000111	←Data→			7	(A)←(A)∧byte	
		r, byte	01110100	$00001R_2R_1R_0$	Data		11	(r)←(r)∧byte	
		sr2, byte	01100100	$S_30001S_2S_1S_0$	↓		20	(sr2)←(sr2)∧byte	
ORI	*	A, byte	00010111	←Data→			7	(A)←(A)∨byte	
		r, byte	01110100	$00011R_2R_1R_0$	Data		11	(r)←(r)∨byte	
		sr2, byte	0110 ↓	$S_30011S_2S_1S_0$	↓		20	(sr2)←(sr2)∨byte	
XRI	*	A, byte	00010110	←Data→			7	(A)←(A)¥byte	
		r, byte	01110100	$00010R_2R_1R_0$	Data		11	(r)←(r)¥byte	
		sr2, byte	0110 ↓	$S_30010S_2S_1S_0$	↓		20	(sr2)←(sr2)¥byte	
GTI	*	A, byte	00100111	←Data→			7	(A) - byte - 1	No Borrow
		r, byte	01110100	$00101R_2R_1R_0$	Data		11	(r) - byte - 1	No Borrow
		sr2, byte	0110 ↓	$S_30101S_2S_1S_0$	↓		14	(sr5) - byte - 1	No Borrow
LTI	*	A, byte	00110111	←Data→			7	(A) - byte	Borrow
		r, byte	01110100	$00111R_2R_1R_0$	Data		11	(r) - byte	Borrow
		sr2, byte	0110 ↓	$S_30111S_2S_1S_0$	↓		14	(sr5) - byte	Borrow
NEI	*	A, byte	01100111	←Data→			7	(A) - byte	No Zero
		r, byte	01110100	$01101R_2R_1R_0$	Data		11	(r) - byte	No Zero
		sr2, byte	0110 ↓	$S_31101S_2S_1S_0$	↓		14	(sr5) - byte	No Zero
EQI	*	A, byte	01110111	←Data→			7	(A) - byte	Zero
		r, byte	01110100	$01111R_2R_1R_0$	Data		11	(r) - byte	Zero
		sr2, byte	0110 ↓	$S_31111S_2S_1S_0$	↓		14	(sr5) - byte	Zero

4

Instruction Groups (cont)

Immediate Data (cont)

Mnemonic		Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
ONI	*	A, byte	01000111	←Data→			7	(A).∧byte	No Zero
		r, byte	01110100	$01001R_2R_1R_0$	Data		11	(r).∧byte	No Zero
		sr2, byte	0110 ↓	$S_31001S_2S_1S_0$	↓		14	(sr5).∧byte	No Zero
OFFI	*	A, byte	01010111	←Data→			7	(A).∧byte	Zero
		r, byte	01110100	$01011R_2R_1R_0$	Data		11	(r).∧byte	Zero
		sr2, byte	0110 ↓	$S_31011S_2S_1S_0$	↓		14	(sr5).∧byte	Zero

Working Register

ADDW		wa	01110100	11000000	Offset		14	(A)←(A) + ((V)•(wa))	
ADCW		wa		1101			14	(A)←(A) + ((V)•(wa)) + (CY)	
ADDNCW		wa		1010			14	(A)←(A) + ((V)•(wa))	No Carry
SUBW		wa		1110			14	(A)←(A) − ((V)•(wa))	
SBBW		wa		1111			14	(A)←(A) − ((V)•(wa)) − (CY)	
SUBNBW		wa		1011			14	(A)←(A) − ((V)•(wa))	No Borrow
ANAW		wa		10001000			14	(A)←(A).∧((V)•(wa))	
ORAW		wa	↓	1001 ↓	↓		14	(A)←(A)∨((V)•(wa))	
XRAW		wa	01110100	10010000	Offset		14	(A)←(A)∀((V)•(wa))	
GTAW		wa		10101000			14	(A) − ((V)•(wa)) − 1	No Borrow
LTAW		wa		1011			14	(A) − ((V)•(wa))	Borrow
NEAW		wa		1110			14	(A) − ((V)•(wa))	No Zero
EQAW		wa		1111			14	(A) − ((V)•(wa))	Zero
ONAW		wa		1100			14	(A).∧((V)•(wa))	No Zero
OFFAW		wa	↓	1101	↓		14	(A).∧((V)•(wa))	Zero
ANIW	*	wa, byte	00000101	←Offset→	Data		19	((V)•(wa))←((V)•(wa)).∧byte	
ORIW	*	wa, byte	0001				19	((V)•(wa))←((V)•(wa))∨byte	
GTIW	*	wa, byte	0010				13	((V)•(wa)) − byte − 1	No Borrow
LTIW	*	wa, byte	0011				13	((V)•(wa)) − byte	Borrow
NEIW	*	wa, byte	0110				13	((V)•(wa)) − byte	No Zero
EQIW	*	wa, byte	0111				13	((V)•(wa)) − byte	Zero
ONIW	*	wa, byte	0100				13	((V)•(wa)).∧byte	No Zero
OFFIW	*	wa, byte	0101 ↓	↓	↓		13	((V)•(wa)).∧byte	Zero

16-bit Arithmetic

EADD		EA, r2	01110000	$010000R_1R_0$			11	(EA)←(EA) + (r2)	
DADD		EA, rp3	0100	$110001P_1P_0$			11	(EA)←(EA) + (rp3)	
DADC		EA, rp3		1101			11	(EA)←(EA) + (rp3) + (CY)	
DADDNC		EA, rp3	↓	1010	↓		11	(EA)←(EA) + (rp3)	No Carry
ESUB		EA, r2	0000	$011000R_1R_0$			11	(EA)←(EA) − (r2)	
DSUB		EA, rp3	01110100	$111001P_1P_0$			11	(EA)←(EA) − (rp3)	
DSBB		EA, rp3		1111			11	(EA)←(EA) − (rp3) − (CY)	
DSUBNB		EA, rp3		1011			11	(EA)←(EA) − (rp3)	No Borrow
DAN		EA, rp3	↓	$100011P_1P_0$			11	(EA)←(EA) − (rp3)	

Instruction Groups (cont)

Mnemonic		Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State 1	Operation	Skip Condition
16-Bit Arithmetic (cont)									
DOR		EA, rp3	01110100	$100111P_1P_0$			11	(EA)←(EA)∨(rp3)	
DXR		EA, rp3		$100101P_1P_0$			11	(EA)←(EA)∀(rp3)	
DGT		EA, rp3		$101011P_1P_0$			11	(EA) - (rp3) - 1	No Borrow
DLT		EA, rp3		1011			11	(EA) - (rp3)	Borrow
DNE		EA, rp3		1110			11	(EA) - (rp3)	No Zero
DEQ		EA, rp3		1111			11	(EA) - (rp3)	Zero
DON		EA, rp3		1100			11	(EA).∧(rp3)	No Zero
DOFF		EA, rp3		1101			11	(EA).∧(rp3)	Zero
Multiply/Divide									
MUL		r2	01001000	$001011R_1R_0$			32	(EA)←(A) × (r2)	
DIV		r2	↓	0011 ↓			59	(EA)←(EA) + (r2), (r2)←Remainder	
Increment/Decrement									
INR		r2	$010000R_1R_0$				4	(r2) - (r2) + 1	Carry
INRW	*	wa	00100000	←Offset→			16	((V), (wa))←((V), (wa)) + 1	Carry
INX		rp	$00P_1P_00010$				7	(rp)←(rp) + 1	
		EA	10101000				7	(EA)←(EA) + 1	
DCR		r2	$010100R_1R_0$				4	(r2)←(r2) - 1	Borrow
DCRW	*	wa	00110000	←Offset→			16	((V), (wa))←((V), (wa)) - 1	Borrow
DCX		rp	$00P_1P_00011$				7	(rp)←(rp) - 1	
		EA	10101001				7	(EA)←(EA) - 1	
Others									
DAA			01100001				4	Decimal Adjust Accumulator	
STC			01001000	00101011			8	(CY)←1	
CLC			↓	00101010			8	(CY)←0	
NEGA			↓	00111010			8	(A)←(A) + 1	
Rotate and Shift									
RLD			01001000	00111000			17	Rotate Left Digit	
RRD				↓ 1001			17	Rotate Right Digit	
RLL		r2	01001000	$001101R_1R_0$			8	$(r2_{M+1})$←$(r2_M)$, $(r2_0)$←(CY), (CY)←$(r2_7)$	
RLR		r2		↓ $00R_1R_0$			8	$(r2_{M+1})$←$(r2_M)$, $(r2_7)$←(CY), (CY)←$(r2_0)$	
SLL		r2		$001001R_1R_0$			8	$(r2_{M+1})$←$(r2_M)$, $(r2_0)$←0, (CY)←$(r2_7)$	
SLR		r2		↓ $00R_1R_0$			8	$(r2_{M-1})$←$(r2_M)$, $(r2_7)$←0, (CY)←$(r2_0)$	
SLLC		r2		$000001R_1R_0$			8	$(r2_{M+1})$←$(r2_M)$, $(r2_0)$←0, (CY)←$(r2_7)$	Carry
SLRC		r2		↓ $00R_1R_0$			8	$(r2_{M-1})$←$(r2_M)$, $(r2_7)$←0, (CY)←$(r2_0)$	Carry

4

Instruction Groups (cont)

Rotate and Shift (cont)

Mnemonic	Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
DRLL	EA	01001000	10110100			8	$(EA_{N+1})\leftarrow(EA_N)$, $(EA_0)\leftarrow(CY)$, $(CY)\leftarrow(EA_{15})$	
DRLR	EA		↓ 0000			8	$(EA_{N-1})\leftarrow(EA_N)$, $(EA_{15})\leftarrow(CY)$, $(CY)\leftarrow(EA_0)$	
DSLL	EA		10100100			8	$(EA_{N+1})\leftarrow(EA_N)$, $(EA_0)\leftarrow 0$, $(CY)\leftarrow(EA_{15})$	
DSLR	EA		↓ 0000			8	$(EA_{N-1})\leftarrow(EA_N)$, $(EA_{15})\leftarrow 0$	

Jump

Mnemonic		Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
JMP	*	word	01010100	←Low Addr→	High Addr		10	(PC)←word	
JB			00100001				4	$(PC_H)\leftarrow(B)$, $(PC_L)\leftarrow(C)$	
JR		word	11←jdisp 1→				10	(PC)←(PC) + 1 + jdisp 1	
JRE	*	word	0100111←	jdisp→			10	(PC)←(PC) + 2 + jdisp	
JEA			01001000	00101000			8	(PC)←EA	

Call

Mnemonic		Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
CALL	*	word	01000000	←Low Addr→	High Addr		16	$((SP)-1)\leftarrow((PC)+3)_H$, $((SP)-2)\leftarrow((PC)+3)_L$, (PC)←word, (SP)←(SP) − 2	
CALB			01001000	00101001			17	$((SP)-1)\leftarrow((PC)+2)_H$, $((SP)-2)\leftarrow((PC)+2)_L$, $(PC_H)\leftarrow(B)$, (SP)←(SP) − 2	
CALF	*	word	01111←	fa→			13	$((SP)-1)\leftarrow((PC)+2)_H$, $((SP)-2)\leftarrow((PC)+2)_L$, $(PC_{15-11})\leftarrow 00001$, $(PC_{10-0})\leftarrow$fa, (SP)←(SP) − 2	
CALT		word	100←ta→				16	$((SP)-1)\leftarrow((PC)+1)_H$, $((SP)-2)\leftarrow((PC)+1)_L$ $(PC_L)\leftarrow(128+2ta)$, $(PC_H)\leftarrow(129+2ta)$, (SP)←(SP) − 2	
SOFTI			01110010				16	$((SP)-1)\leftarrow$(PSW), $((SP)-2)\leftarrow((PC)+1)_H$, $((SP)-3)\leftarrow((PC)+1)_L$, (PC)←0060H, (SP)←(SP) − 3	

Return

Mnemonic	Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
RET		10111000				10	$(PC_L)\leftarrow((SP))$, $(PC_H)\leftarrow((SP)+1)$ (SP)←(SP) + 2	
RETS		↓ 1001				10	$(PC_L)\leftarrow((SP))$, $(PC_H)\leftarrow((SP)+1)$ (SP)←(SP) + 2, (PC)←(PC) + n	
RETI		01100010				13	$(PC_L)\leftarrow((SP))$, $(PC_H)\leftarrow((SP)+1)$ (PSW)←((SP) + 2), (SP)←(SP) + 3	Unconditional Skip

Skip

Mnemonic	Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
BIT	bit, wa	$01011B_2B_1B_0$	←Offset→			10	Bit Test	((V), (wa)) bit = 1

CPU Control

Mnemonic	Operand	Opcode B1	Opcode B2	Opcode B3	Opcode B4	State [1]	Operation	Skip Condition
SK	f	01001000	$0001F_2F_1F_0$			8	Skip if f = 1	f = 1
SKN	f	↓	0001 ↓			8	Skip if f = 0	f = 0
SKIT	irf		$0101_4I_3I_2I_1I_0$			8	Skip if irf = 1, then reset Irf	irf = 1

Instruction Groups (cont)

CPU Control (cont)								
		Opcode						
Mnemonic	Operand	B1	B2	B3	B4	State [1]	Operation	Skip Condition
SKNIT	irf	01001000	$0111_4 I_3 I_2 I_1 I_0$			8	Skip if irf = 0 Reset irf, if irf = 1	irf = 1
NOP		00000000				4	No Operation	
EI		10101010				4	Enable Interrupt	
DI		10111010				4	Disable Interrupt	
HLT		01001000	00111011			11	Halt	

Notes: * 1 :In the case of skip condition, the idle states are as follows:
1-byte instruction: 4 states 2-byte instruction (with *): 7 states
2-byte instruction: 8 states 3-byte instruction (with *): 10 states
3-byte instruction: 11 states 4-byte instruction (with *): 14 states
* 2 : B2 (Data):rpa2 = D + byte. H + byte.
* 3 : Right side of slash (/) in states indicate case rpa2, rpa3 = D + byte, H + A, H + B, H + EA, H + byte.
* 4 : B3 (Data): rpa3 = D + byte, H + byte.

Emulating the μPD7811

To emulate the μPD7811: tie MODE0 to ground and pull up MODE1 through a 10-kΩ resistor; insert a 2732A or 2764 into the upper terminals of the μPD78PG11. If a 2732 is used it should be inserted so that pin 1 of the 2732A goes into terminal 3 (see pin configuration). If a 2764 is used, address line A_{12} will be held low so that only memory locations 0-OFFFH (the lower 4K bytes) of the 2764 will be accessed. This simulates accessing 4K bytes of masked-ROM in the μPD7811. In other respects μPD78PG11 is functionally equivalent to μPD7811.

Input/Output

8 Analog Input Lines
44 Digital I/O Lines: five 8-bit ports (Port A, Port B, Port C, Port D, Port F) and 4 input lines (AN_4-AN_7)

1. Analog Input Lines
   AN_0-AN_7 are configured as analog input lines for on-chip A/D converter.
2. Port Operation
   —Port A, Port B, Port C, Port F
   Each line of these ports can be individually programmed as an input or as an output. When used as I/O ports, all have latched outputs, high-impedance inputs.
   —Port D
   Port D can be programmed as a byte input or a byte output.
   —AN_4-AN_7
   The high-order analog input lines, AN_4-AN_7 can be used as digital input lines for falling edge detection.
3. Control Lines
   Under software control, each line of Port C can be configured individually to provide control lines for serial interface, timer, and timer/event counter.
4. Memory Expansion
   In addition to the single-chip operation mode, μPD78PG11 has 4 memory expansion modes. Under software control, Port D can provide multiplexed low-order address and data bus and Port F can provide high-order address bus. The relation between memory expansion modes and the pin configurations of Port D and Port F is shown in the table that follows.

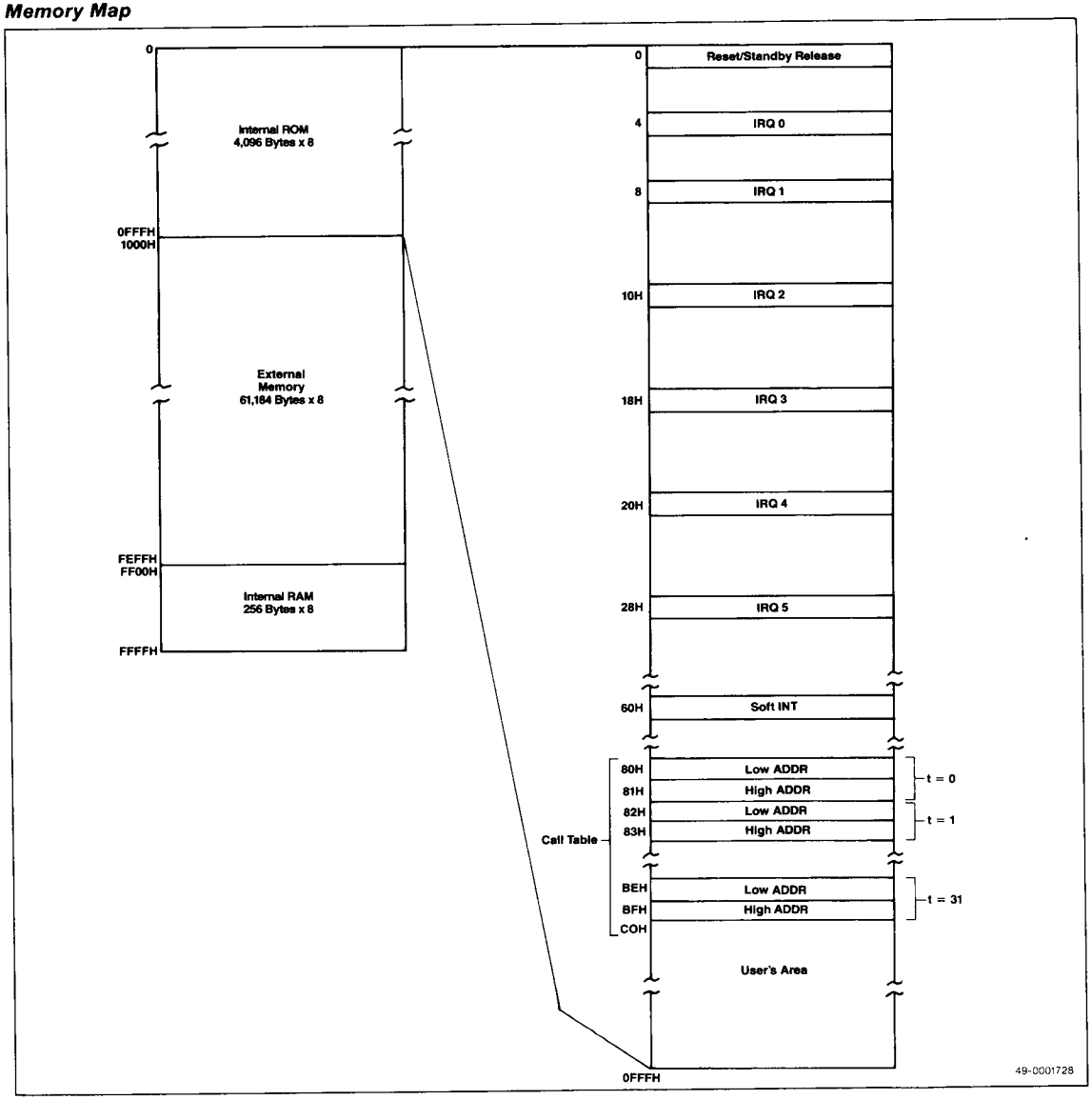
Memory Expansion	Port Configuration	
None	Port D	I/O Port
	Port F	I/O Port
256 Bytes	Port D	Multiplexed Address/Data Bus
	Port F	I/O Port
4K Bytes	Port D	Multiplexed Address/Data Bus
	Port F_0 — F_3	Address Bus
	Port F_4 — F_7	I/O Port
16K Bytes	Port D	Multiplexed Address/Data Bus
	Port F_0 — F_5	Address Bus
	Port F_6 — F_7	I/O Port
60K Bytes	Port D	Multiplexed Address/Data Bus
	Port F	Address Bus

4

Memory Map

The μPD78PG11 can directly address up to 64K bytes of memory. Except for the EPROM (0-4,095) and RAM (65,280-65,535), any memory location can be used as ROM or RAM. The following memory map defines the 0-64K-byte memory space for the μPD78PG11.

Memory Map



Timers

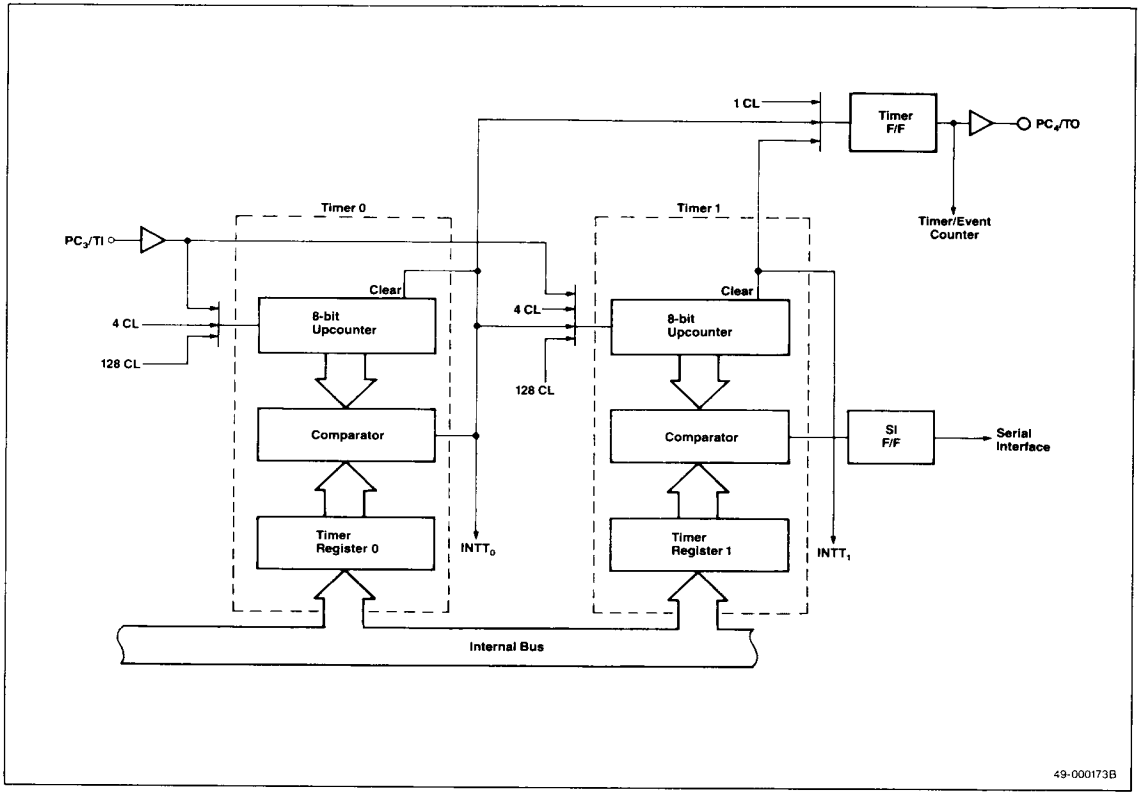
The timer/event counter consists of two 8-bit timers. The timers may be programmed independently or may be cascaded and used as a 16-bit timer. The timer can be set in software to increment at intervals of 4 machine cycles (1μs at 12MHz operation) or 128 machine cycles (32μs at 12MHz), or to increment on receipt of a pulse at T1.

Timer/Event Counter

The 16-bit multifunctional timer/event counter can be used for the following operations:

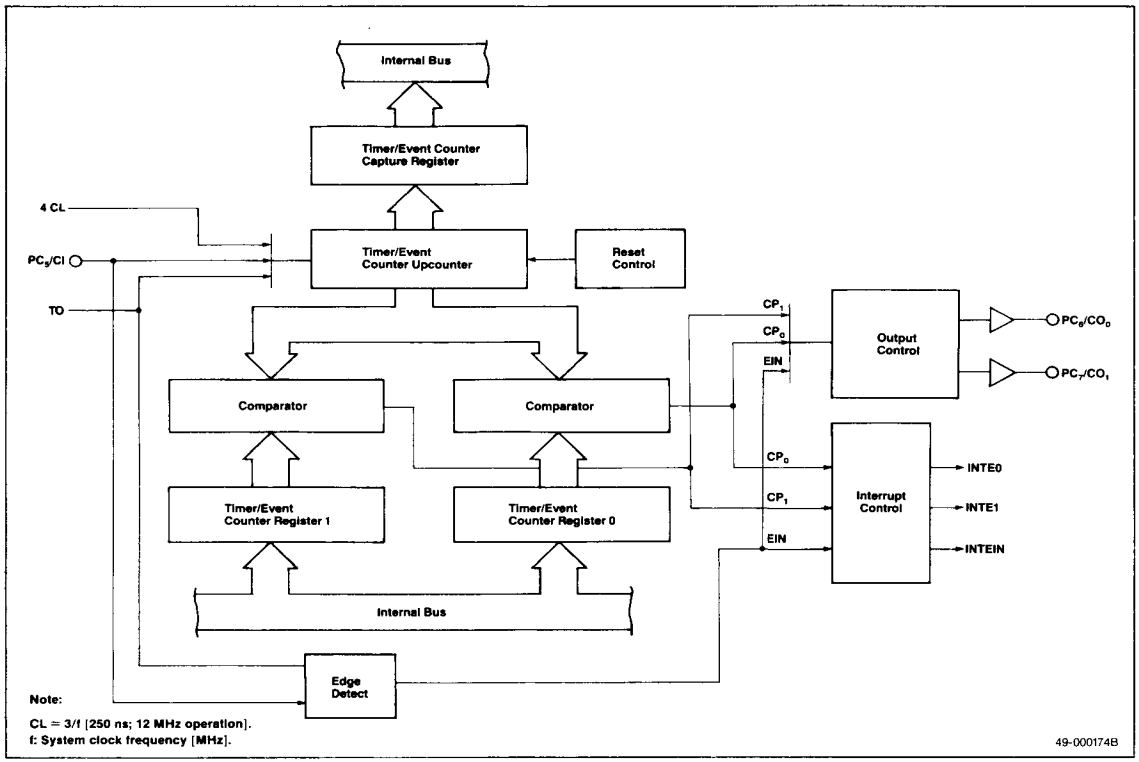
- ☐ Interval timer
- ☐ External event timer
- ☐ Frequency measurement
- ☐ Pulse width measurement
- ☐ Programmable square-wave output

Timer Block Diagram



4

Block Diagram for Timer/Event Counter



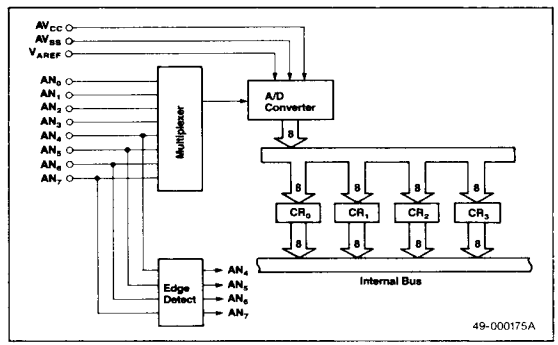
8-Bit A/D Converter

- ☐ 8 input channels
- ☐ 4 conversion result registers
- ☐ 2 powerful operation modes
  - —Autoscan mode
  - —Channel select mode
- ☐ Successive approximation technique
- ☐ Absolute accuracy: ±1.5 LSB (±0.6%)
- ☐ Conversion range: 0V to 5V
- ☐ Conversion time: 50μs
- ☐ Interrupt generation

Analog/Digital Converter

The μPD78PG11 features an 8-bit, high-speed, high-accuracy A/D converter. The A/D converter consists of a 256-resistor ladder and a successive approximation register (SAR). There are four conversion result registers (CR_0-CR_3). The 8-channel analog input may be operated in either of two modes. In the select mode, the conversion value of one analog input is sequentially stored in CR_0-CR_3. In the scan mode, the upper four channels or the lower four channels may be specified. Then those four channels will be consecutively selected and the conversion results stored sequentially in the four conversion result registers.

A/D Converter Block Diagram

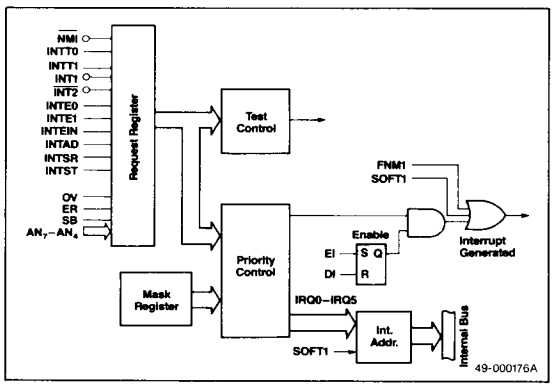


Interrupt Structure

There are 11 interrupt sources. Three are external interrupts and 8 are internal. These 11 interrupt sources are divided into 6 priority levels as shown in the table below.

Interrupt Request	Interrupt	Type of Interrupt	In/Ext
IRQ0	4	$\overline{NMI}$ (Nonmaskable interrupt)	External
IRQ1	8	INTT0 (Coincidence signal from timer 0) INTT1 (Coincidence signal from timer 1)	Internal
IRQ2	16	INT1 (Maskable interrupt) $\overline{INT2}$ (Maskable interrupt)	External
IRQ3	24	INTE0 (Coincidence signal from timer/event counter) INTE1 (Coincidence signal from timer/event counter)	Internal
IRQ4	32	INTEIN (Falling signal of C1 and T0 counter) INTAD (A/D converter interrupt)	In/External
IRQ5	40	INTSR (Serial receive interrupt) INST (Serial send interrupt)	Internal

Interrupt Structure Block Diagram



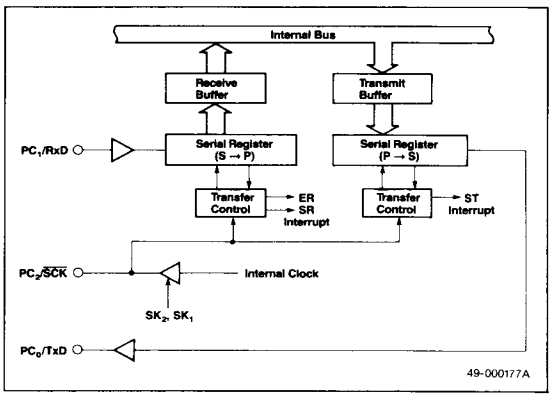
Standby Function

The μPD78PG11 offers a standby function that allows the user to save up to 32 bytes of RAM with backup power (V_{DD}) if the main power (V_{CC}) fails. On powerup the μPD78PG11 checks whether recovery was made from standby mode or from cold start.

Universal Serial Interface

The serial interface can operate in any of three modes: synchronous, asynchronous, and I/O interface. The I/O interface mode transfers data MSB first for ease of communication with certain peripheral devices. Synchronous and asynchronous modes transfer data LSB first. Synchronous operation offers two modes of data reception. In the search mode, data is transferred one bit at a time from serial register to receive buffer. This allows a software search for a sync character. In the nonsearch mode, data transfer from serial register to transmit buffer occurs 8 bits at a time.

Universal Serial Interface Block Diagram



Zero-crossing Detector

The INT1 and $\overline{INT2}$ terminals (used common to TI and PC_3) can be used to detect the zero-crossing point of slow moving AC signals. When driven directly, these pins respond as a normal digital input.

To utilize the zero-cross detection mode, an AC signal of approximately 1—3V AC peak-to-peak magnitude and a maximum frequency of 1kHz is coupled through an external capacitor to these pins.

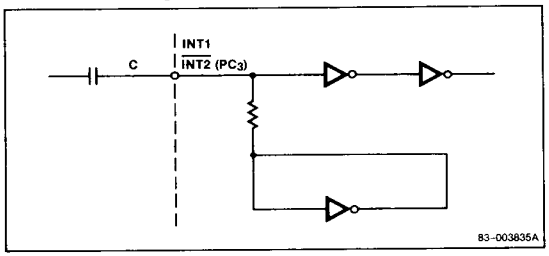
For the INT1 pin, the internal digital state is sensed as a zero until the rising edge crosses the DC average level, when it becomes a one and INT1 interrupt is generated.

For the $\overline{INT2}$ pin, the state is sensed as a one until the falling edge crosses the DC average level, when it becomes a zero and $\overline{INT2}$ interrupt is generated.

The zero-cross detection capability allows the user to make the 50-60Hz power signal the basis for system timing and to control voltage phase sensitive devices.

4

Zero-Crossing Detection Circuit



Operand Format/Description

Format	Description
r	V, A, B, C, D, E, H, L
r1	EAH, EAL, B, C, D, E, H, L
r2	A, B, C
sr	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, SML, EOM, ETMM, TMM, MM, MCC, MA, MB, MC MF, TxB, TM_0, TM_1
sr1	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, EOM, TMM, RxB, CR_0, CR_1, CR_2, CR_3
sr2	PA, PB, PC, PD, PF, MKH, MKL, ANM, SMH, EOM, TMM
sr3	ETM_0, ETM_1
sr4	ECNT, ECPT
rp	SP, B, D, H
rp1	V, B, D, H, EA
rp2	SP, B, D, H, EA
rp3	B, D, H
rpa	B, D, H, D+, H+, D-, H-
rpa1	B, D, H
rpa2	B, D, H, D+, H+, D-, H-, D+byte, H+A , H+B, H+EA, H+byte
rpa3	D, H, D+, H++, D+byte, H+A, H+B, H+EA, H+byte
wa	8-bit immediate data
word	16-bit immediate data
byte	8-bit immediate data
bit	3-bit immediate data
f	CY, HC, Z
irf	FNMI, FTO, FT1, F1, F2, FEO, FE1, FEIN, FAD, FSR, FST, ER, OV, AN_4, AN_5, AN_6, AN_7, SB