T-49-19-57

CATALYST SEMICONDUCTOR, INC.
2231 CALLE DE LUNA, SANTA CLARA, CA 95054 Telephone: (408) 748-7700

CAT62C720 Smart Card Microcomputer

Preliminary

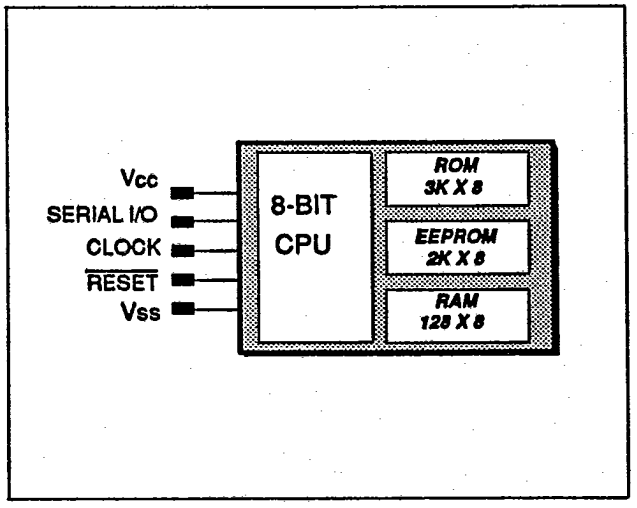
DESCRIPTION:

The **CAT62C720** is an advanced single-chip 8-bit microcomputer with 16K-bits of EEPROM, 3K-bytes of ROM, and 128 bytes of RAM. Designed specifically for IC card applications, it incorporates extensive hardware and software security to protect the program and data memories and it features on-board hardware error correction. The **CAT62C720** is ideal for **"Portable Database"** applications where a significant amount of data storage is required, such as financial transactions and personal health record IC cards.

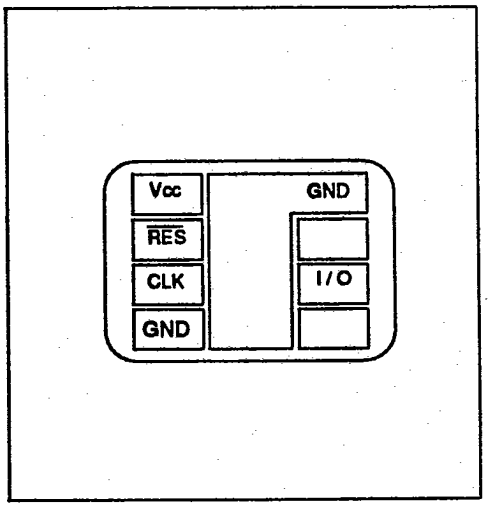
FEATURES:

- 8-Bit CPU, RAM, EEPROM, and ROM in a single chip
- Low power CMOS technology
- Hardware and software security
- Speed: 800 ns instruction cycle at 5 MHz
- Clock frequency: D.C. to 5 MHz
- Single pin, high speed serial I/O interface
- 22 internal registers
- 100 instructions
- 9 addressing modes
- Upward compatible with the CAT62C780
- ECC
- 10,000 EEPROM erase/write cycles per byte
- 10 year EEPROM data retention

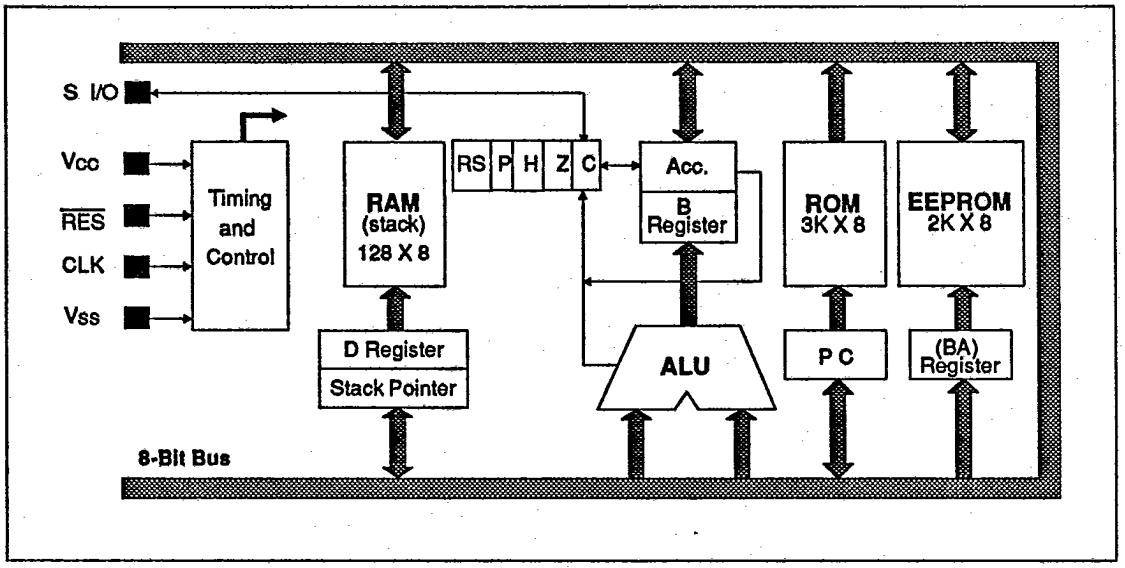
BLOCK DIAGRAM



COB MICRO MODULE



FUNCTIONAL BLOCK DIAGRAM



OPERATING CONDITIONS

Parameter	Symbol	Limits	Unit
Power supply voltage	Vcc	4.5 to 5.5	Volts
Operating temp. range	Top	0 to +70	°C

PIN DESCRIPTION

Pin	Function	Input/Output
Vcc	Power supply pin, +5 Volts ± 10%	
Vss	Power supply pin, 0 Volts	
CLOCK	CPU Clock input pin. Pulled down internally by approximately 1.5 Meg. resistor	INPUT
$\overline{\text{RESET}}$	Resets the CPU. Pin is an active low input and is pulled down internally by approximately 1.5 Meg. resistor	INPUT
SERIAL I/O	Serial data input/output pin or pseudo bidirectional pin. The pin is pulled up by approx. 10K resistor, and is set high at CPU reset.	INPUT/OUTPUT

ABSOLUTE MAXIMUM RATINGS

Parameter	Symbol	Conditions	Limits	Unit
Power supply voltage	V_{CC}	$T_A = 25^oC$	-0.5 to 7	Volts
Input Voltage	V_{IN}	$T_A = 25^oC$	-0.5 to V_{CC} +0.5	Volts
Output Voltage	V_{OUT}	$T_A = 25^oC$	-0.5 to V_{CC} +0.5	Volts
Storage Temperature	T_{stg}		-55 to 150	°C

D.C. CHARACTERISTICS

(V_{CC} = 5V ± 10%, T_A = 0° C to +70°C)

Parameter		Symbol	Conditions	Min.	Typ.	Max.	Unit
Supply Current		I_{CC}	f = 5 MHz	-	8.0	16	mA
Low Input Voltage	CLOCK	V_{IL}	-	-0.3	-	0.5	Volts
	$\overline{\text{RESET}}$			-0.3	-	0.6	
	SERIAL I/O			-0.3	-	0.8	
High Input Voltage	CLOCK	V_{IH}	-	2.4	-	V_{CC} +0.3	Volts
	$\overline{\text{RESET}}$			4.0	-	V_{CC} +0.3	
	SERIAL I/O			2.0	-	V_{CC} +0.3	
Low Output Voltage		V_{OL}	I_{OL} MAX=1.6mA	0	-	0.4	Volts
High Output Voltage		V_{OH}	I_{OH} MAX ≥ -100µA	2.4	-	V_{CC}	Volts
Input Current (CLOCK, $\overline{\text{RESET}}$)		I_{IL1}	V_{CC} = 5.5V, V_{IN} = 0.0V	-	-	-10	µA
		I_{IH1}	V_{CC} = 5.5V, V_{IN} = 5.5V	-	-	10	µA
Input Current (SIO) SERIAL I/O		I_{IL2}	V_{CC} = 5.5V, V_{IN} = 0.0V	-	-	-1	mA
		I_{IH2}	V_{CC} = 5.5V, V_{IN} = 5.5V	-	-	10	µA
Input Capacitance		C_I	f = 1 MHz T_A = 25°C	-	10	20	pF
Output Capacitance		C_O		-	10	20	pF

NOTE: CLOCK and $\overline{\text{RESET}}$ pins are pulled down internaly, and SERIAL I/O pin is pulled up internally.

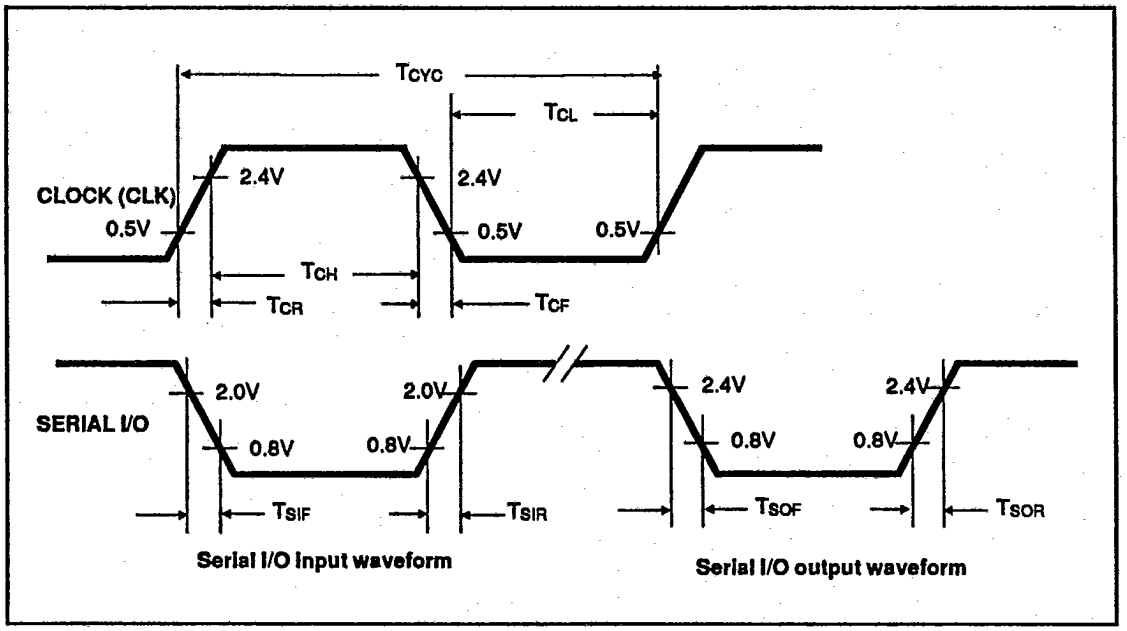
A.C. CHARACTERISTICS

(V_{CC} = 5 Volts ±10%, Ta = 0° C to +70°C)

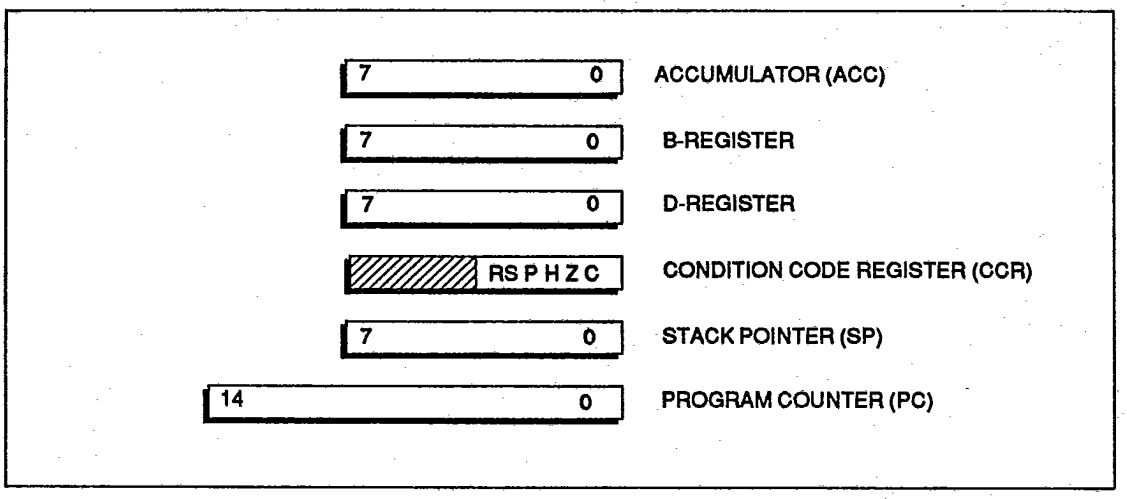
Parameter	Symbol	Min	Typ	Max	Unit
Clock cycle time	T_{CYC}	200	-	-	ns
Clock low level time	T_{CL}	0.4*T_{CYC}	-	0.6*T_{CYC}	ns
Clock high level time	T_{CH}	0.4*T_{CYC}	-	0.6*T_{CYC}	ns
Clock rise time	T_{CR}	-	-	5.0	µs
Clock fall time	T_{CF}	-	-	5.0	µs
$\overline{\text{RESET}}$ pulse width	T_{RW}	8*T_{CY}	-	-	µs
Serial I/O rise time (input)	T_{SIR}	-	-	5.0	µs
Serial I/O fall time (input)	T_{SIF}	-	-	5.0	µs
Serial I/O rise time (output)	T_{SOR}	-	-	1.0	µs
Serial I/O fall time (output)	T_{SOF}	-	-	1.0	µs

NOTE: Output load capacitance = 30pF.

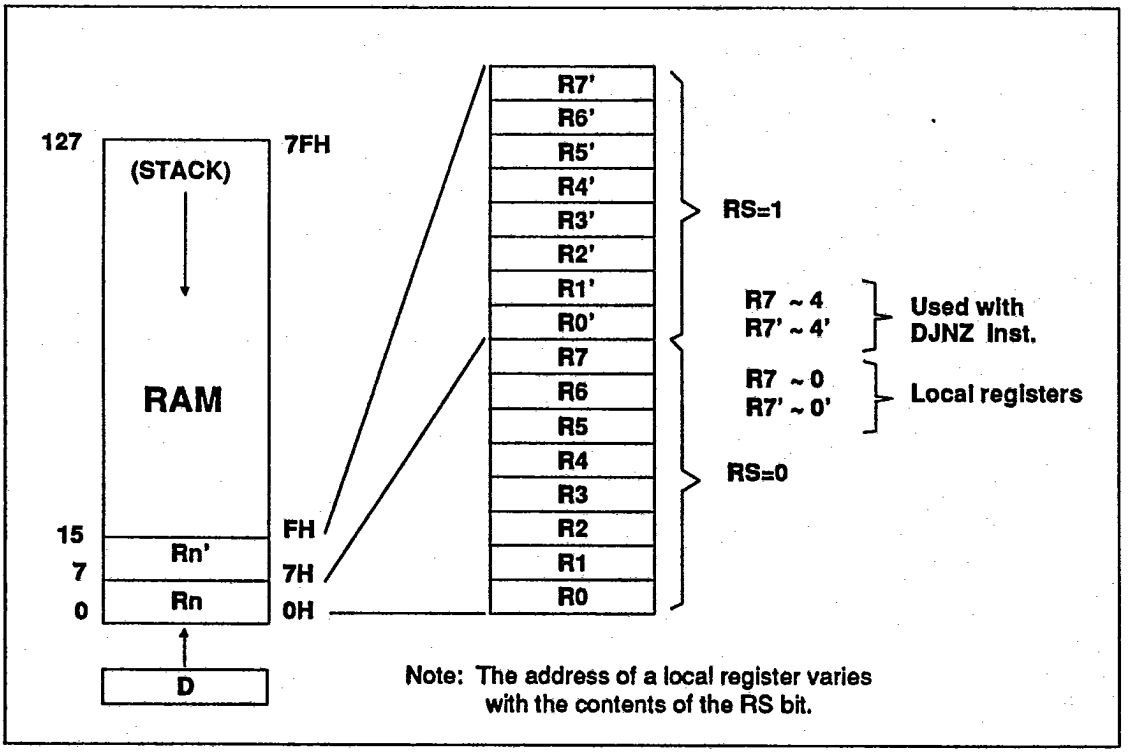
TIMING DIAGRAM



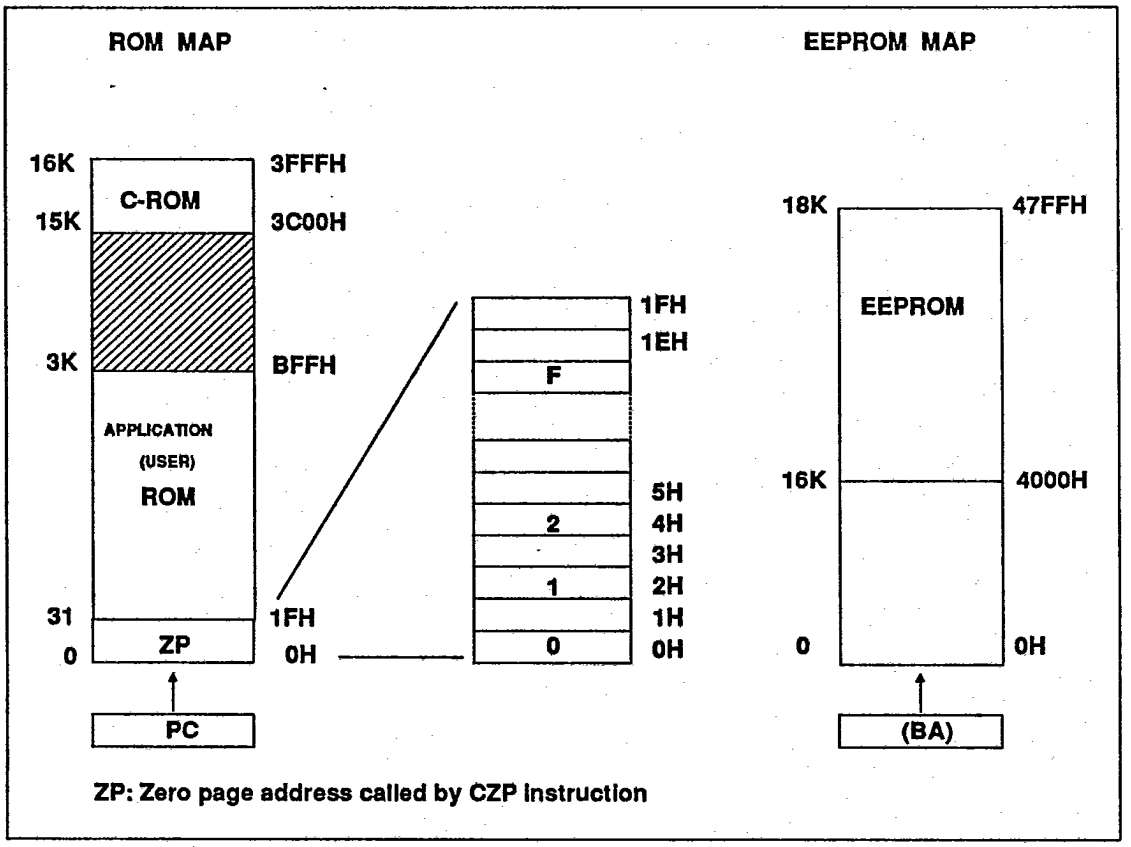
REGISTER SET



MEMORY MAP - RAM



MEMORY MAP - ROM & EEPROM



ZP: Zero page address called by CZP instruction

EEPROM WRITING

The EEPROM write operations are controlled by EEPROM write subroutines installed in the built-in control ROM area. The EEPROM Byte write subroutine is called by the EWR instruction. The EEPROM data read is executed by the MOV@D, @BA instruction in the same manner as the CAT62C780.

BYTE-WRITE MODE

The byte-write mode writes a single byte of data to the address specified. The following instructions specify how to use this byte-write mode.

1. Load write (transfer) data in RAM
2. Set RAM address containing write data in D register
3. Set EEPROM write address in BA register (4000H~47FFH)
4. EWR XXXXH (byte-write subroutine call)

When the entry address XXXXH is called after proceeding as described above, the write (transfer) process is executed in less than 10ms, and control returns to the main routine.

Note: This subroutine uses 8 stack bytes when executed and all registers as well as RAM are unaffected.

INSTRUCTION SET

MNEMONIC	opr	OPERATION	Bytes	Cycle	FLAGS					
					C	P	H	Z	RS	
MOV A, opr	B	A <-- B	1	1				*		1
	D	A <-- D	1	1				*		2
	@D	A <-- (D)	1	1				*		3
	@D+	A <-- (D), D <-- D+1	1	1				*		4
	@D-	A <-- (D), D <-- D-1	1	2				*		5
	N	A <-- (N)	2	2				*		6
	N+@D	A <-- (N + D)	2	3				*		7
	#N	A <-- #N	2	2				*		8
	Rn	A <-- Rn	1	2				*		9
MOV opr, A	B	B <-- A	1	1						10
	D	D <-- A	1	1						11
	@D	(D) <-- A	1	1						12
	@D+	(D) <-- A, D <-- D+1	1	1						13
	@D-	(D) <-- A, D <-- D-1	1	2						14
	N	(N) <-- A	2	2						15
	N+@D	(N+D) <-- A	2	3						16
	Rn	Rn <-- A	1	2						17
MOV D, opr	#N	D <-- #N	2	2						18
MOV @D, opr	#N	(D) <-- #N	2	2						19
MOV Rn, opr	#N	Rn <-- #N	2	2						20
MOV @BA, opr	@D	(BA) <-- (D)	1	4						21
MOV @D, opr	@BA	(D) <-- (BA)	1	4						22
MOVW BA, opr	#N	A <-- #N1, B <-- #N2	3	3				*		23
XCH A, opr	B	A <--> B	1	1				*		24
	D	A <--> D	1	1				*		25
	@D	A <--> (D)	1	1				*		26
	N	A <--> (N)	2	2				*		27

INSTRUCTION SET

MNEMONIC	opr	OPERATION	Bytes	Cycle	FLAGS					
					C	P	H	Z	RS	
XCH D, opr	B	D <--> B	1	2						28
	SP	D <--> SP	1	2						29
	Rn	D <--> Rn	1	2						30
XCH C, opr	P	C <--> P	1	1	*	*				31
ADD A, opr	@D	A <-- A + (D)	1	1	*		*	*		32
	N	A <-- A + (N)	2	2	*		*	*		33
	#N	A <-- A + #N	2	2	*		*	*		34
ADD A, opr	@D	A <-- A + (D) + C	1	1	*		*	*		35
	N	A <-- A + (N) + C	2	2	*		*	*		36
	#N	A <-- A + #N + C	2	2	*		*	*		37
DAA		Decimal Adjust	1	1	*			*		38
CMP A, opr	@D	A - (D)	1	1	*			*		39
	N	A - (N)	2	2	*			*		40
	#N	A - #N	2	2	*			*		41
CMP @D, opr	@BA	(D) - (BA)	1	4	*			*		42
EOR A, opr	@D	A <-- A ⊻ (D)	1	1				*		43
	N	A <-- A ⊻ (N)	2	2				*		44
	#N	A <-- A ⊻ #N	2	2				*		45
OR A, opr	@D	A <-- A ∨ (D)	1	1				*		46
	N	A <-- A ∨ (N)	2	2				*		47
	#N	A <-- A ∨ #N	2	2				*		48
AND A, opr	@D	A <-- A ∧ (D)	1	1				*		49
	N	A <-- A ∧ (N)	2	2				*		50
	#N	A <-- A ∧ #N	2	2				*		51

INSTRUCTION SET

MNEMONIC	opr	OPERATION	Bytes	Cycle	FLAGS					
					C	P	H	Z	RS	
INC opr	A	A <-- A + 1	1	1					*	52
	D	D <-- D + 1	1	1						53
	@D	(D) <-- (D) + 1	1	1					*	54
	N	(N) <-- (N) + 1	2	2					*	55
INCW opr	BA	BA <-- BA + 1	1	2	*			*		56
DEC opr	A	A <-- A- 1	1	1				*		57
	D	D <-- D - 1	1	1						58
	@D	(D) <-- (D) -1	1	1				*		59
	N	(N) <-- (N) -1	2	2				*		60
DECW opr	BA	BA <-- BA - 1	1	2	*			*		61
CAL opr	addr	(SP)<--PC+2, PC<--addr, SP<--SP-2	2	4						62
CZP opr	addr	(SP)<--PC+1, PC<--ZP, SP<--SP-2	1	4						63
RT		PC <-- (SP), SP <-- SP + 2	1	3						64
PUSH	D	(SP) <-- D, SP <-- SP - 1	1	2						65
	PSW	(SP)<--A, (SP-1)<--B, (SP-2)<--CCR, SP<--SP-3	1	4						66
POP opr	D	D<--(SP+1), SP<--SP+1	1	2						67
	PSW	CCR<--(SP+1), B<--(SP+2), A<--(SP+3), SP<--SP+3		4	*	*	*	*	*	68
JMP opr	addr	PC <-- addr	2	2						69
JZ opr	addr	if Z = 1, PC <-- PC + 2 + addr	2	2/3						70
JNZ opr	addr	if Z ≠ 1, PC <-- PC + 2 + addr	2	2/3						71
JC opr	addr	if C = 1, PC <-- PC + 2 + addr	2	2/3						72
JNC opr	addr	if C ≠ 1, PC <-- PC + 2 + addr	2	2/3						73
JB opr	baddr,addr	if (baddr) = 1, PC <-- PC+3+addr	3	3/4						74
JNB opr	baddr,addr	if (baddr) ≠ 1, PC <-- PC+3+addr	3	3/4						75

INSTRUCTION SET

MNEMONIC	opr	OPERATION	Bytes	Cycle	FLAGS					
					C	P	H	Z	RS	
DJNZ opr	Rn, addr	Rn<--Rn-1, if Rn≠0, PC<--PC+2+addr (n=4~7)	2	3/4						76
JMNE opr	#N, addr	if (D) ≠ #N, PC<--PC+3+addr	3	3/4						77
JDNE opr	#N, addr	if D ≠ #N, PC<--PC+3+addr	3	3/4						78
JANE opr	@D, addr	if A ≠ (D), PC <-- PC + 2 + addr	2	3/4	*			*		79
	N, addr	if A ≠ (N), PC <-- PC + 3 + addr	3	3/4	*			*		80
	#N, addr	if A ≠ #N, PC <-- PC + 3 + addr	3	3/4	*			*		81
CLR opr	A	A <-- 0	1	1				*		82
CPL opr	A	A <-- $\overline{A}$	1	1				*		83
	C	C <-- $\overline{C}$	1	1	*					84
RRC opr	A	C -> A 7 ~ 0	1	1	*			*		85
	@D	C -> (D) 7 ~ 0	1	1	*			*		86
	N	C -> (N) 7 ~ 0	2	2	*			*		87
RLC opr	A	C <-- A 7 ~ 0	1	1	*			*		88
	@D	C <-- (D) 7 ~ 0	1	1	*			*		89
	N	C <-- (N) 7 ~ 0	2	2	*			*		90
RC		C <-- 0	1	1	0					91
SC		C <-- 1	1	1	1					92
CHK opr	P	P <-- C, if A = odd parity C <-- 1	1	1	*	*				93
SIN		C <-- SIO	1	1	*					94
SOUT		SIO <-- C	1	1						95
RB	baddr	(baddr) <-- 0	2	2						96
SB	baddr	(baddr) <-- 1	2	2						97
NOP		No operation	1	1						98
DLY opr	#N	Delay N+2 Cycles if N = 0, delay 258 Cycles	2	3~258						99
EWR opr	addr	(SP)<--PC+2,PC<--addr,SP<--SP-2 addr: Entry Address	2	4						100

PACKAGE DIAGRAM <CHIP-ON-BOARD (COB)>