

ABOV SEMICONDUCTOR Co., Ltd.
4-BIT SINGLE-CHIP MICROCONTROLLERS

MC40P5004B

MC40P5104B

MC40P5204B

MC40P5404B

User's Manual



REVISION HISTORY

Ver.1.7.(Feb, 27, 2015)

Fixed operation range at 1.2~1.8V(Data Sheet)

Ver.1.6.C (May, 4, 2012) This book

Added DC characteristic data at 1.5V.

Ver.1.6 (Mar, 19, 2012)

Added MC40P5404BR to ordering information table.

Added Circuit guide note for IR LED resistor at circuit diagrams.

Ver.1.5 (Aug 3, 2011)

Removed 0.1uF bypass capacitor at the application circuit diagram.

Ver.1.4.1 (Dec 28, 2010)

Added 0.1uF bypass capacitor at the application circuit diagram.

Ver.1.3 (Sep 3, 2010)

Modified Supply Voltage of Absolute Maximum Ratings.

Ver.1.2 (Aug 12, 2010)

Modified circuit diagram of MC40P5004B.

Added circuit diagram of MC40P5404B.

Added diagram of IR LED current by power.

Ver.1.1 (Jul 16, 2010)

Added 16SOPN Package diagram.

Added explanation of PGND pin as internally N.C. pin.

Ver.1.0 (Mar 10, 2010)

First released version.

Version 1.7

Published by ABU

©2010 ABOV Semiconductor Co., Ltd. All rights reserved.

Additional information of this manual may be served by ABOV Semiconductor offices in Korea or Distributors. ABOV Semiconductor reserves the right to make changes to any information here in at any time without notice. The information, diagrams and other data in this manual are correct and reliable; however, ABOV Semiconductor is in no way responsible for any violations of patents or other rights of the third party generated by the use of this manual.

Table of Contents

1. OVERVIEW	4
1.1 Description.....	4
1.2 Features.....	4
1.3 Ordering Information	5
2. BLOCK DIAGRAM	6
2.1 MC40P5004B (20 pin package).....	6
2.2 MC40P5104B (16 pin package).....	7
2.3 MC40P5204B (24 pin package).....	8
2.4 MC40P5404B (20 pin package).....	9
3. PIN ASSIGNMENT	10
4. PACKAGE DIAGRAM	11
5. PIN FUNCTION	14
6. PORT STRUCTURES	15
7. ELECTRICAL CHARACTERISTICS	17
7.1 Absolute Maximum Ratings ($T_a = 25^\circ\text{C}$)	17
7.2 Recommended Operating Conditions	17
7.4 Electrical characteristics ($T_a=25^\circ\text{C}$, $V_{DD}= 1.5\text{V}$).....	18
8. Architecture	21
8.1 Program Memory (EPROM).....	21
8.2 EPROM Address Register	22
8.3 Data memory (RAM).....	23
8.4 X-register (X).....	24
8.5 Y-register (Y).....	24
8.6 Accumulator (Acc)	24
8.7 Arithmetic and Logic Unit (ALU)	24
8.8 State Counter (SC).....	24
8.9 Clock Generator	25
8.10 Pulse Generator.....	26
8.11 Reset Operation.....	27
8.12 STOP Operation.....	29
9. Instruction	30
9.1 Instruction Table	31
9.2 DETAILS OF INSTRUCTION SYSTEM.....	33
9.3 Assembler Macro.....	46
10. SPGM (Serial Program)	47
10.1 Summary of Protocol	47
OPTION PROGRAM / READ DATA Format	48
4K MTP (Multi Time Programming).....	49
11. APPLICATION	51

MC40P5004B MC40P5104B MC40P5204B MC40P5404B

CMOS SINGLE-CHIP 4-BIT MICROCONTROLLER

1. OVERVIEW

1.1 Description

The MC40P5x04B series is 4-bit remote control MCU which uses CMOS technology and the 4K bytes EPROM version. This enables transmission code outputs of different configurations, multiple custom code output, and double push key output for easy fabrication. The MC40P5x04B series is suitable for remote control of TV, VCR, FANS, Air-conditioners, Audio Equipments, Toys, Games etc.

1.2 Features

- Program memory : 4,096 bytes
- MTP : 1K * 4, 2K * 2, 4K * 1 times programmable
- Data memory : 32 × 4 bits
- 43 types of instruction set
- 3 levels of subroutine nesting
- Operating frequency : 2.4MHz ~ 4MHz
- Instruction cycle : $f_{osc}/48$ or $f_{osc}/12$
- CMOS process
- Stop mode (Through internal instruction)
- Released stop mode by key input
- Built in Power-on Reset circuit
- Built in Transistor for I.R LED Drive
 - $I_{OL} = 220$ mA at $V_{DD} = 1.5$ V and $V_O = 0.3$ V
 - $I_{OL} = 290$ mA at $V_{DD} = 1.5$ V and $V_O = 0.52$ V
- Built in Low Voltage reset circuit
- Built in a watch dog timer (WDT)
- Low operating voltage : 1.2 ~ 1.8V (1 Battery supported only)
- 16 SOPN, 20 SOP/TSSOP, 24 SOP package

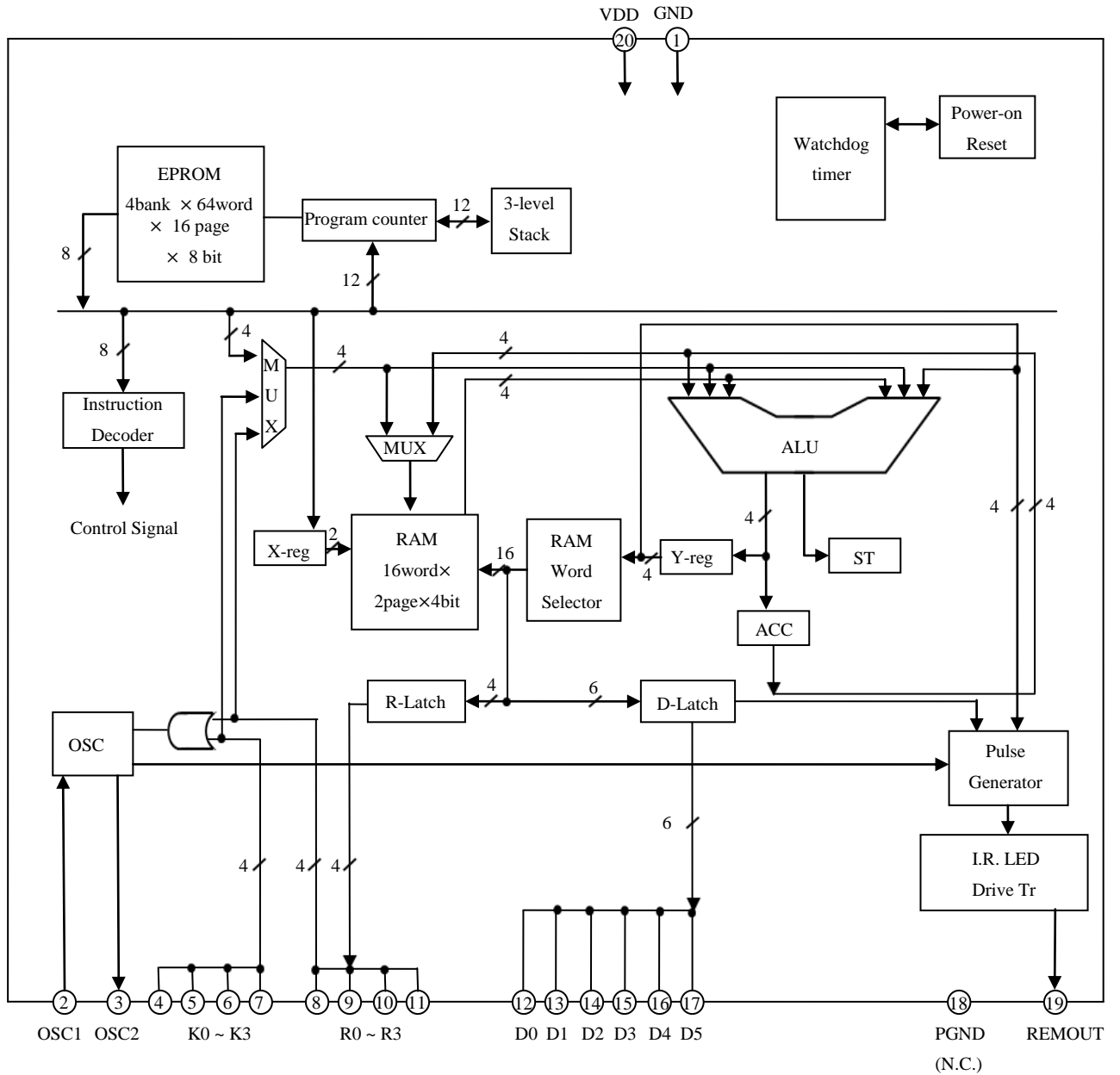
1.3 Ordering Information

Series	MC40P5004BD	MC40P5104BM	MC40P5204BD	MC40P5404BD
Program memory	4,096	4,096	4,096	4,096
Data memory	32 x 4	32 x 4	32 x 4	32 x 4
I/O Ports	2	2	2	2
Input Ports	6	4	6	6
Output Ports	6 (D0~D5)	5 (D2~D6)	10 (D0~D9)	7 (D0~D6)
Built-in Drive Tr.	0	0	0	0
Package	20SOP	16SOPN	24SOP	20SOP

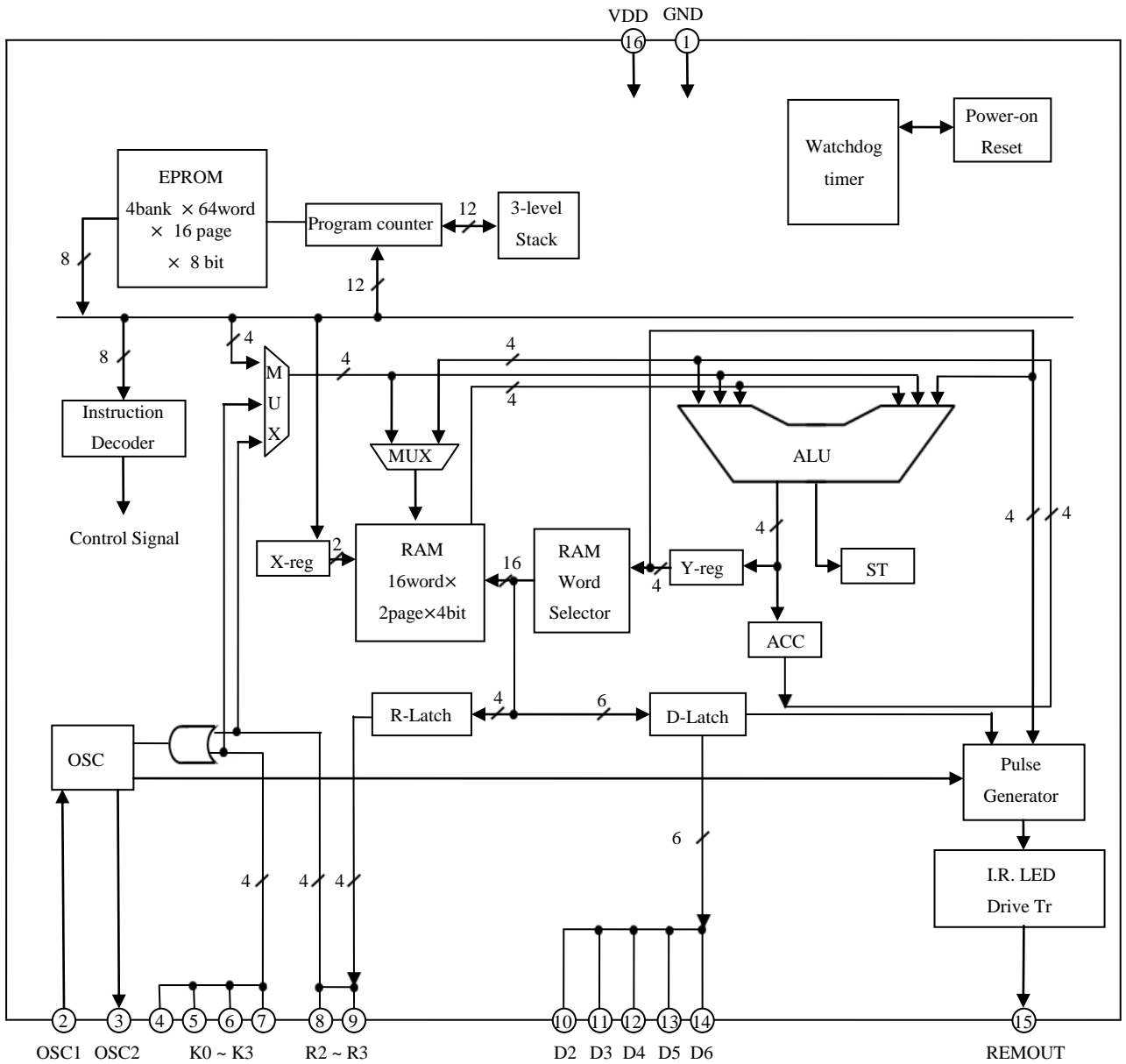
Series	MC40P5004BR	MC40P5404BD
Program memory	4,096	4,096
Data memory	32 x 4	32 x 4
I/O Ports	2	2
Input Ports	6	6
Output Ports	6 (D0~D5)	7 (D0~D6)
Built-in Drive Tr.	0	0
Package	20TSSOP	20TSSOP

2. BLOCK DIAGRAM

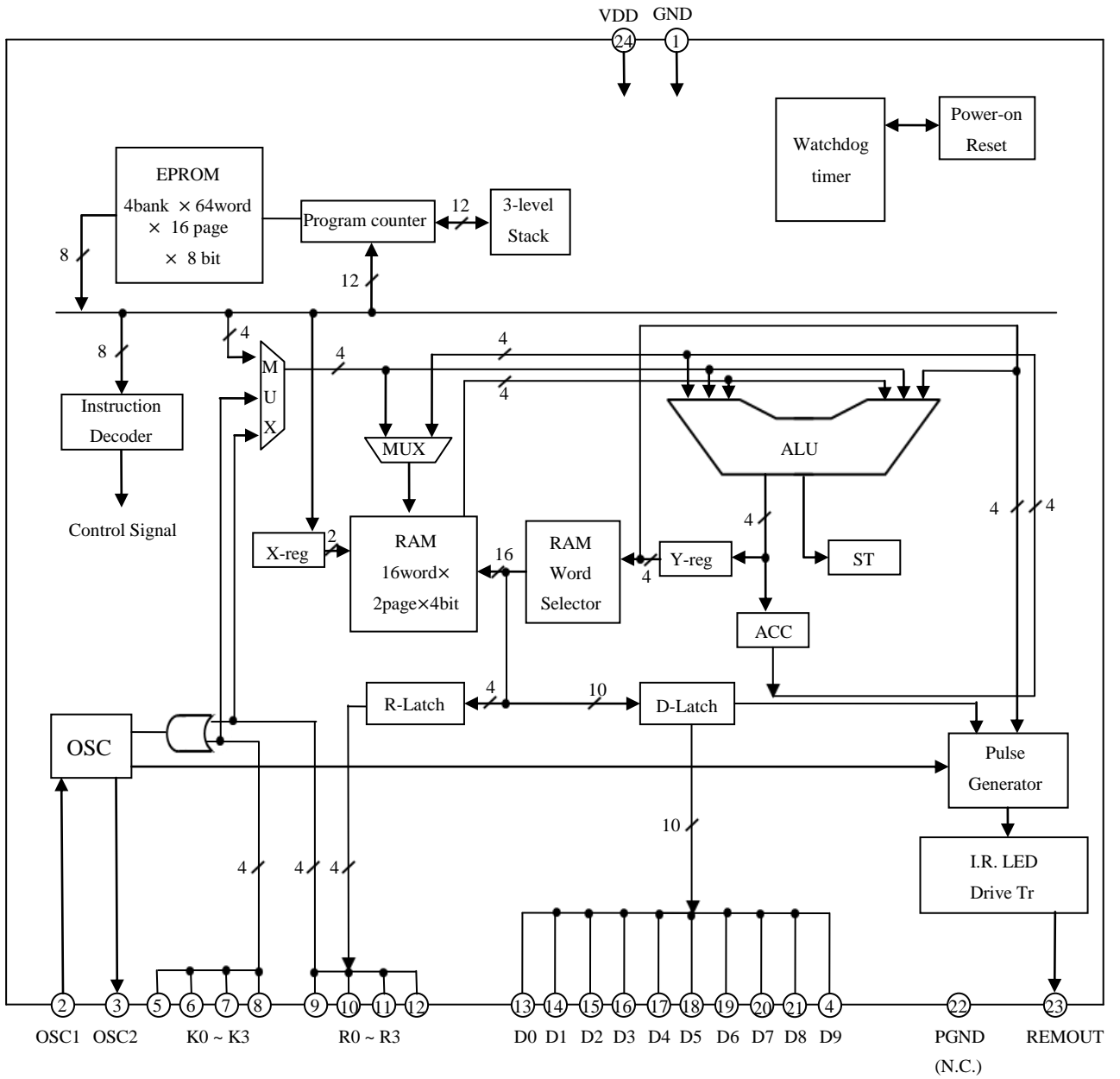
2.1 MC40P5004B (20 pin package)



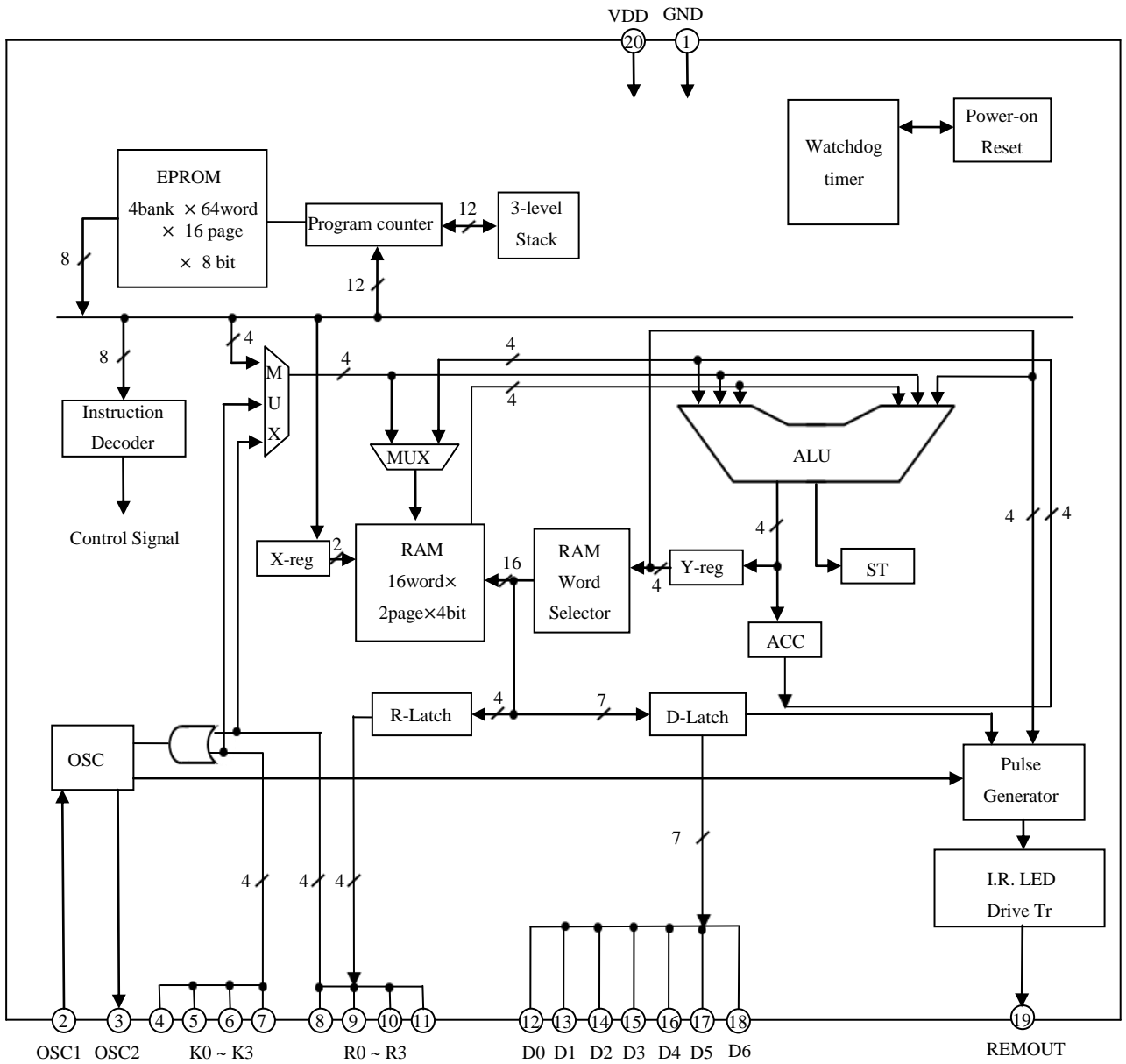
2.2 MC40P5104B (16 pin package)



2.3 MC40P5204B (24 pin package)



2.4 MC40P5404B (20 pin package)



3. PIN ASSIGNMENT

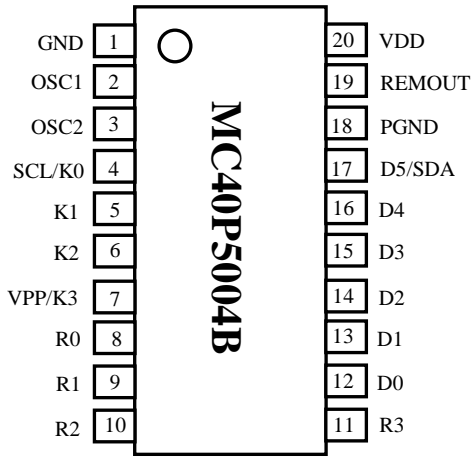


Fig 3-1 MC40P5004B (20 PIN)
 REMOUT : open drain output
 VPP : K3 (PIN No.7)

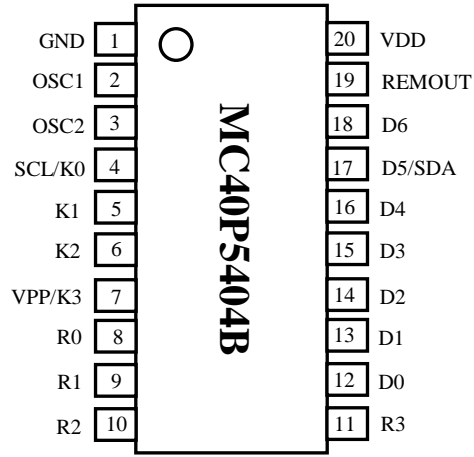


Fig 3-2 MC40P5404B (20 PIN)
 REMOUT : open drain output
 VPP : K3 (PIN No.7)

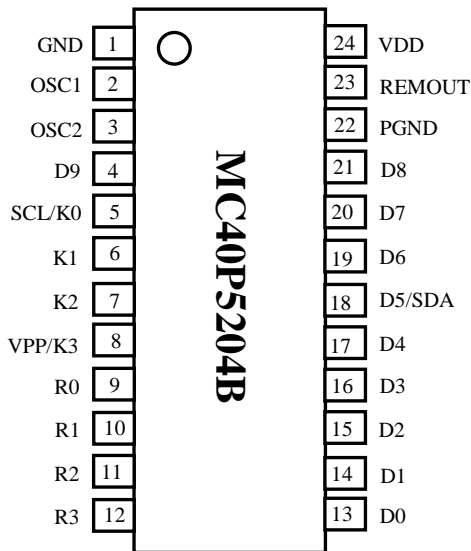


Fig 3-3 MC40P5204B (24 PIN)
 REMOUT : open drain output
 VPP : K3 (PIN No.8)

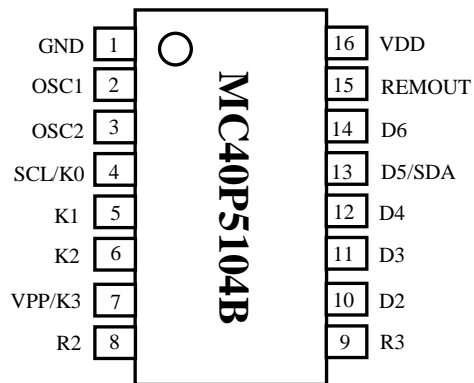


Fig 3-4 MC40P5104B (16 PIN)
 REMOUT : open drain output
 VPP : K3 (PIN No.7)

4. PACKAGE DIAGRAM

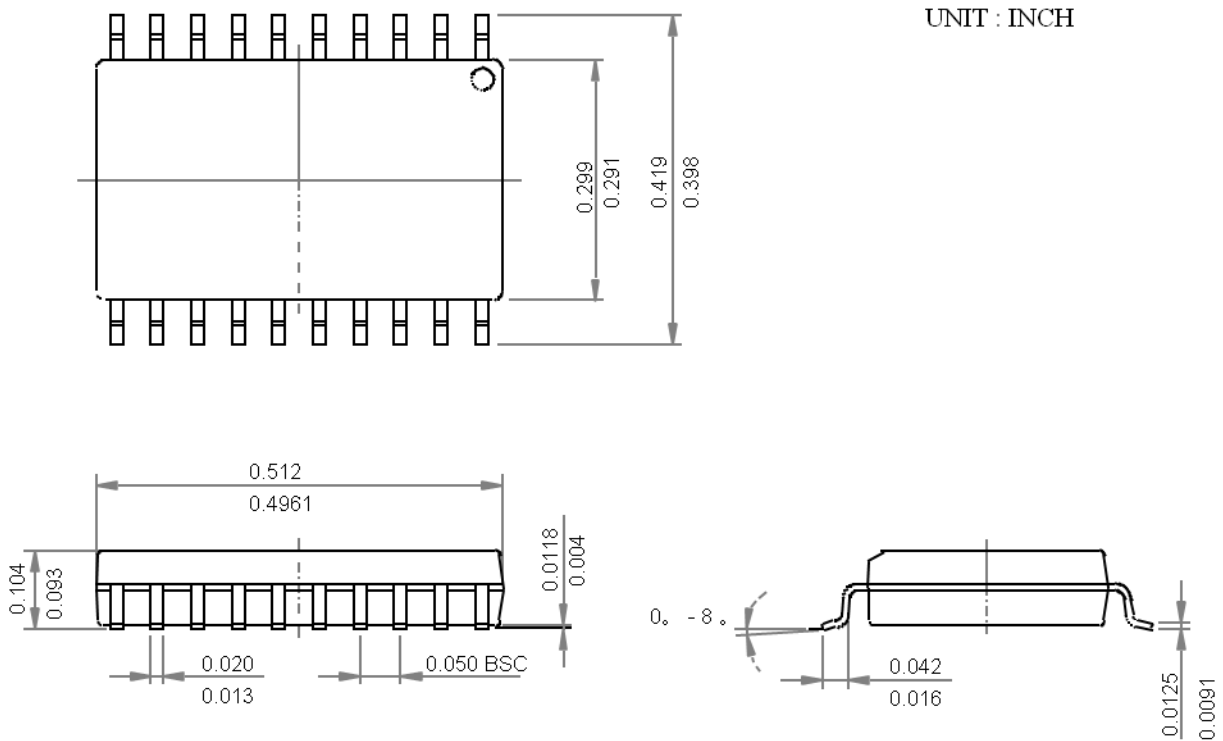


Fig 4-1 20SOP (300MIL)

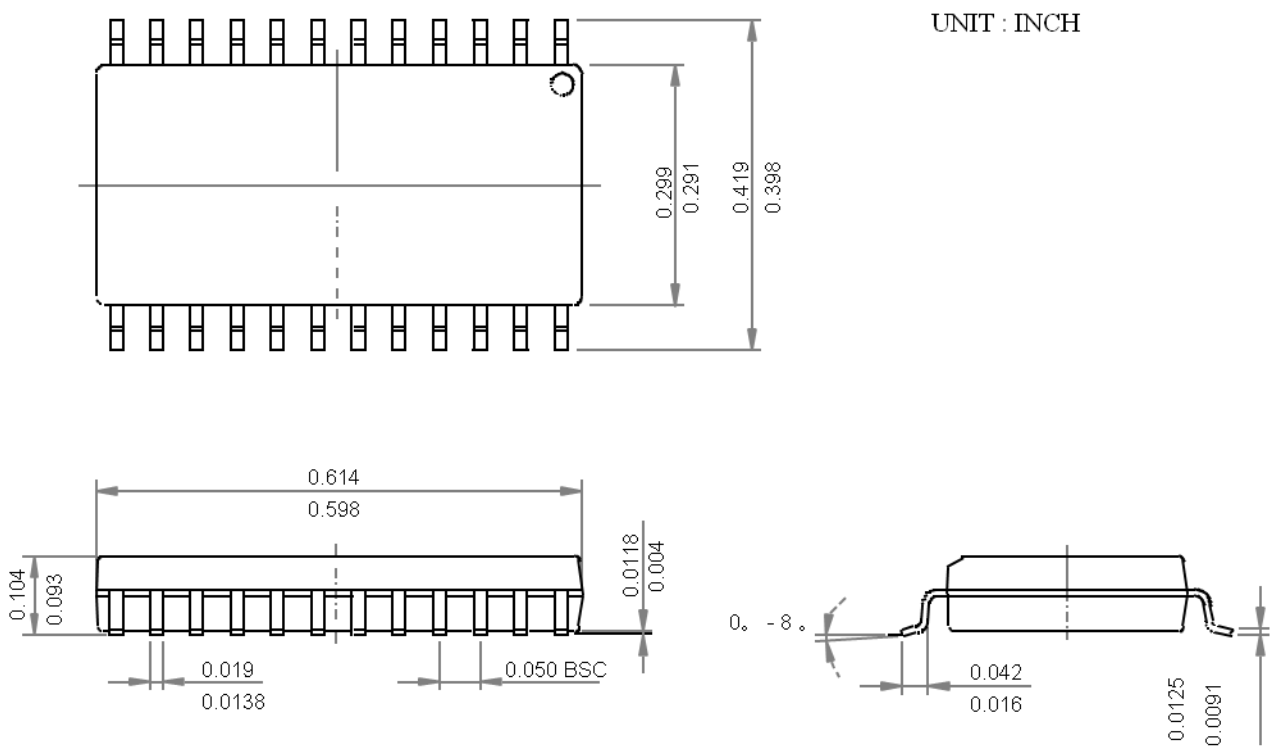


Fig 4-2 24SOP (300MIL)

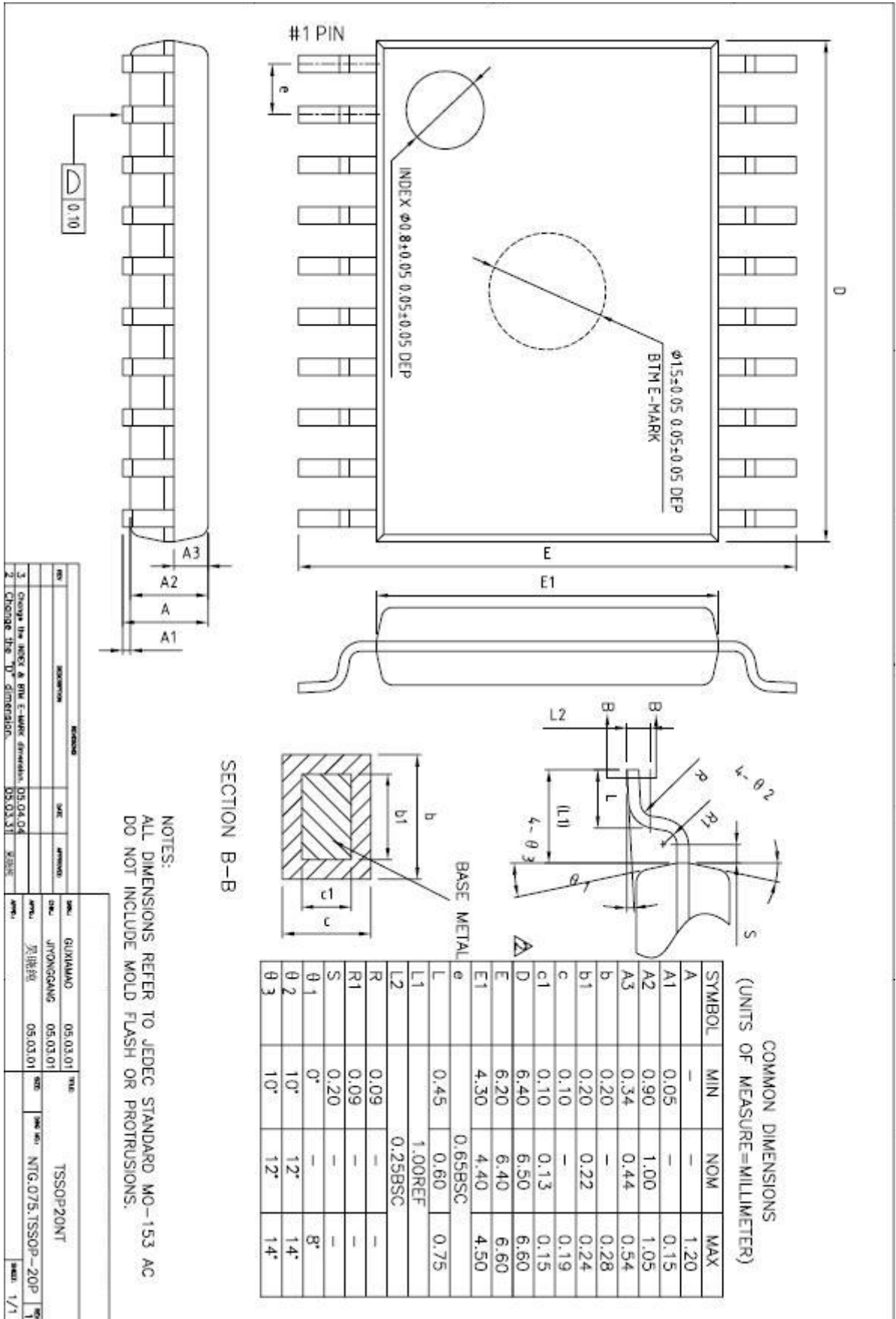


Fig 4-4 20TSSOP

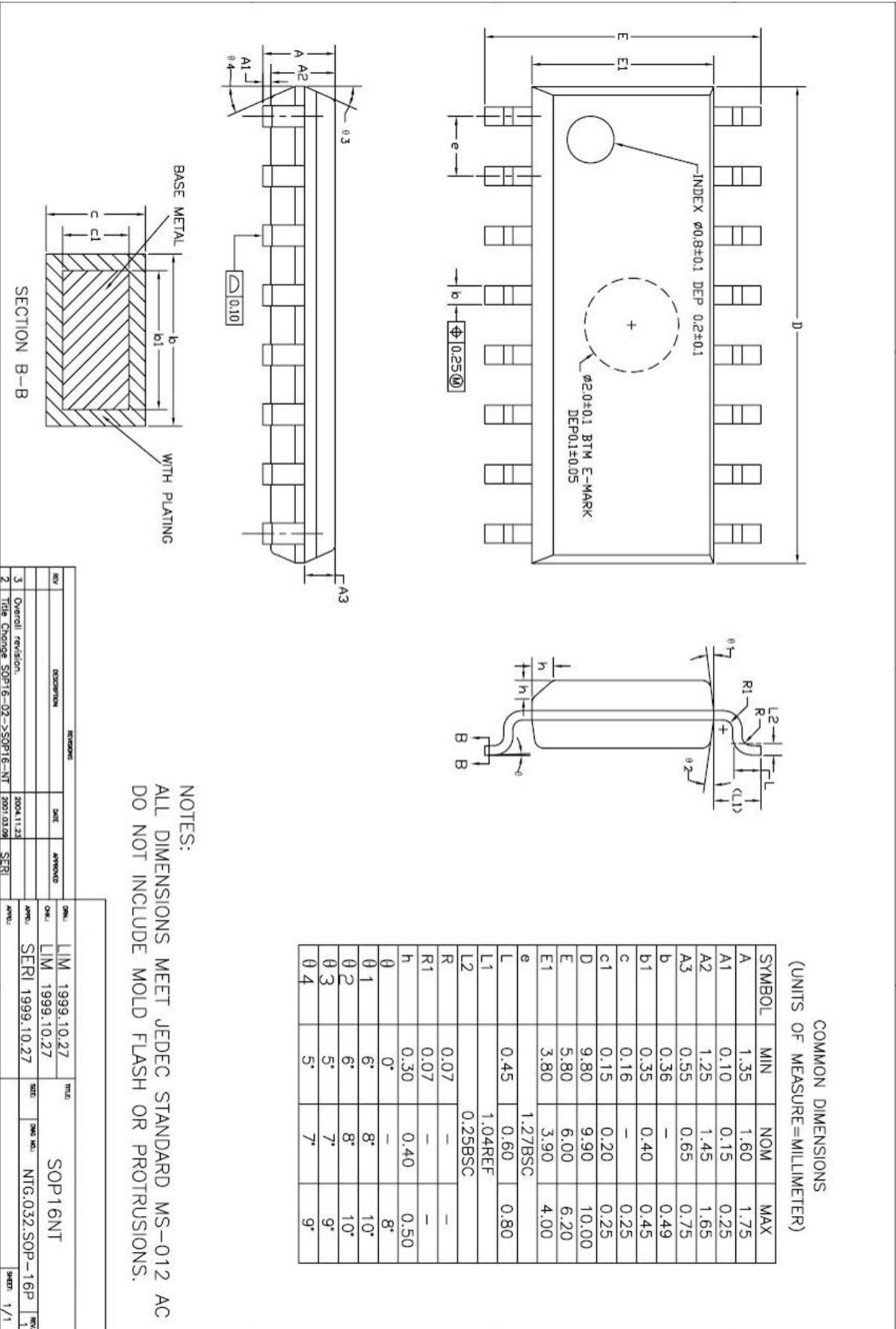


Fig 4-5 16SOPN

5. PIN FUNCTION

Pin	I/O	Function
VDD	-	Connected to 1.2~ 1.8V power supply
GND	-	Connected to 0V power supply.
K0 ~ K3	Input	4-bit input port with built in pull-up resistor. STOP mode is released by "L" input of each pin.
D0 ~ D9	Output	Each can be set and reset independently. The output is the structure of N-channel-open-drain. D0~D3 are "L" output at STOP MODE. D4 ~D9 keep the state before stop mode at STOP MODE.
R0 ~ R1	Input	2-bit input port with built in pull-up resistor. STOP mode is released by "L" input of each pin.
R2 ~ R3	I/O	2-bit I/O port with built in pull-up resistor. STOP mode is released by "L" input of each pin. Input mode is set only when each of them output "H". The output is in the form of C-MOS. In outputting, each can be set and reset independently(or at once.)
OSC1	Input	Oscillator input. Input to the oscillator circuit and connection point for ceramic resonator. A feedback resistor is connected between this pin and OSC2.
OSC2	Output	Connect a resonator between this pin and OSC1.
PGND	internally N.C.(Not Connected)	PGND pin is just nominal for old compatibility, so this can be open or connected to any other point.
REMOUT	Output	High current output port for driving I.R.LED. The output is in the form N-channel open drain.

Note: D port pin mapping is

MC40P5004B (D0~D5), MC40P5204B(D0~D9),
MC40P5404B(D0~D6), MC40P5104B(D2~D6)

6. PORT STRUCTURES

Pin	I/O	I/O circuit	Note
R0 ~ R1	I		- Built in MOS Tr for pull-up, about 140 kΩ.
R2 ~ R3	I/O		- CMOS output. - "H" output at reset. - Built in MOS Tr for pull-up, about 140 kΩ.
K0 ~ K3	I		- Built in MOS Tr for pull-up, about 140 kΩ.
D0 ~ D9	O		- Open drain output. - "L" output at reset.
REMOUT	O		- Open drain output - Output Tr. Disable at Reset or stop.
PGND	-	internally N.C.(Not Connected)	PGND pin is just nominal for old compatibility, so this can be open or connected to any other point.

<p>OSC2</p>	<p>O</p>		<p>- Built in feedback-resistor about 1 MΩ</p>
<p>OSC1</p>	<p>I</p>		

Note: The gate voltage of REMOUT port (NMOS Transistor) is always higher than VDD voltage when it drives REMOUT port as Logic "LOW"..

7. ELECTRICAL CHARACTERISTICS

7.1 Absolute Maximum Ratings (Ta = 25°C)

Parameter	Symbol	Max. rating	Unit
Supply Voltage	V _{DD}	-0.3 ~ 5	V
Power dissipation	P _D	700 *	mW
Storage temperature range	T _{STQ}	-55 ~ 125	°C
Input voltage	V _{IN}	-0.3 ~ V _{DD} +0.3	V
Output voltage	V _{OUT}	-0.3 ~ V _{DD} +0.3	V

*Thermal derating above 25°C: 6mW per degree °C rise in temperature

7.2 Recommended Operating Conditions

Parameter	Symbol	Condition	Rating	Unit
Supply Voltage	V _{DD}	2.4MHz ~ 4MHz	1.2 ~ 1.8	V
Operating temperature	T _{OPR}	-	-20 ~ +70	°C

7.3 Electrical characteristics (Ta=25°C, V_{DD}= 1.5V)

Parameter	Symbol	Limits			Unit	Condition	
		Min.	Typ.	Max.			
Input H current	I _{IH}	-	-	1	uA	VI=V _{DD}	
K Pull-up Resistance	R _{PU1}	70	140	300	kΩ	VI=GND	
R Pull-up Resistance	R _{PU2}	70	140	300	kΩ	VI=GND, Output off	
Feedback Resistance	R _{FD}	0.241	0.805	2.415	MΩ	V _{OSC1} =GND, V _{OSC2} =VDD	
K, R input H voltage	V _{IH1}	1.05	-	-	V	-	
K, R input L voltage	V _{IL1}	-	-	0.45	V	-	
D, R output L voltage	V _{OL2} *1	-	0.315	0.84	V	I _{OL} =3mA	
OSC2 output L voltage	V _{OL3}	-	0.41	0.92	V	I _{OL} =150uA	
OSC2 output H voltage	V _{OH3}	0.97	1.15	-	V	I _{OH} =-150uA	
REMOUT output L current	I _{OL4} *2	-	218 289	-	mA	V _{OL4} =0.3V V _{OL4} =0.52V	
D, R output leakage current	I _{OLK2}	-	-	1	uA	V _{OUT} =V _{DD} , Output off	
Low Voltage Reset voltage	V _{LVR}	-	1.15	-	V		
Current on STOP mode	I _{STP}	-	-	1	uA	At STOP mode	
Operating Supply current	I _{DD2} *3	-	-	0.598	mA	f _{OSC} =4MHz	
System clock frequency	f _{OSC} /48	f _{OSC}	2.4	-	4	MHz	MHz version

*1 Refer to Fig.7-1 < I_{OL1} vs. V_{OL1} Graph>

*2 Refer to Fig.7-2 < I_{OL4} vs. V_{OL4} Graph>

*3 I_{DD1}, I_{DD2}, is measured at RESET mode.

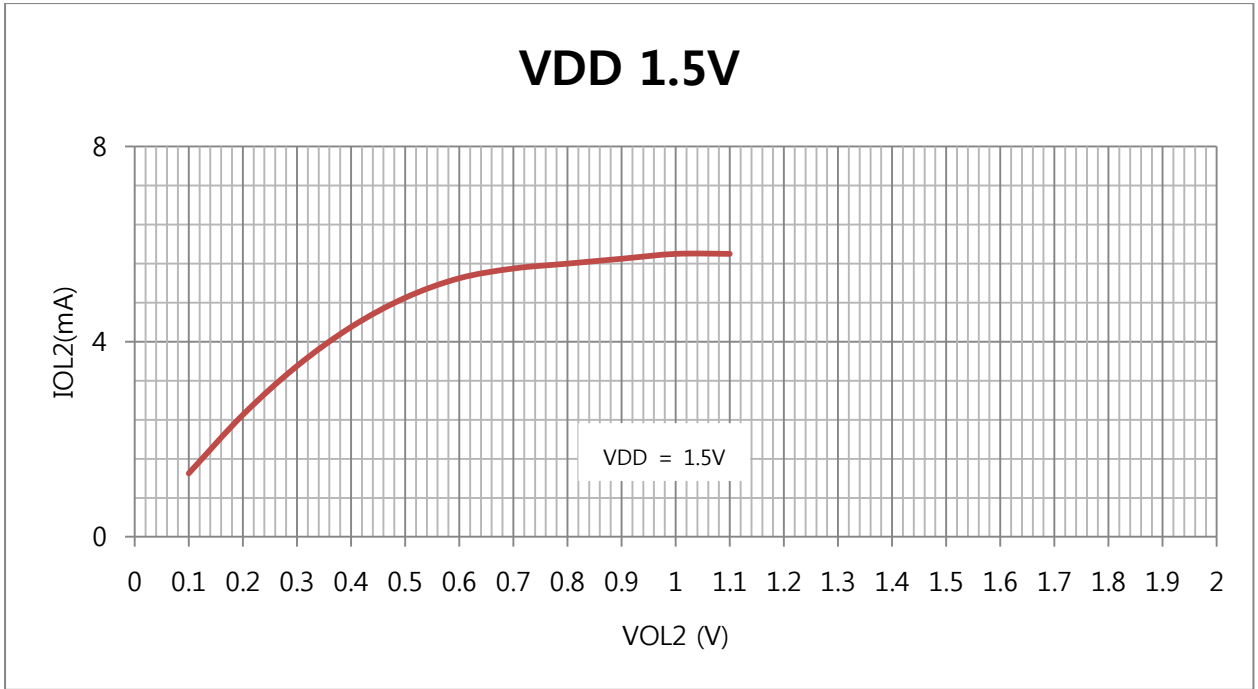


Fig 7-1. I_{OL2} vs. V_{OL2} Graph. (D, R Port)

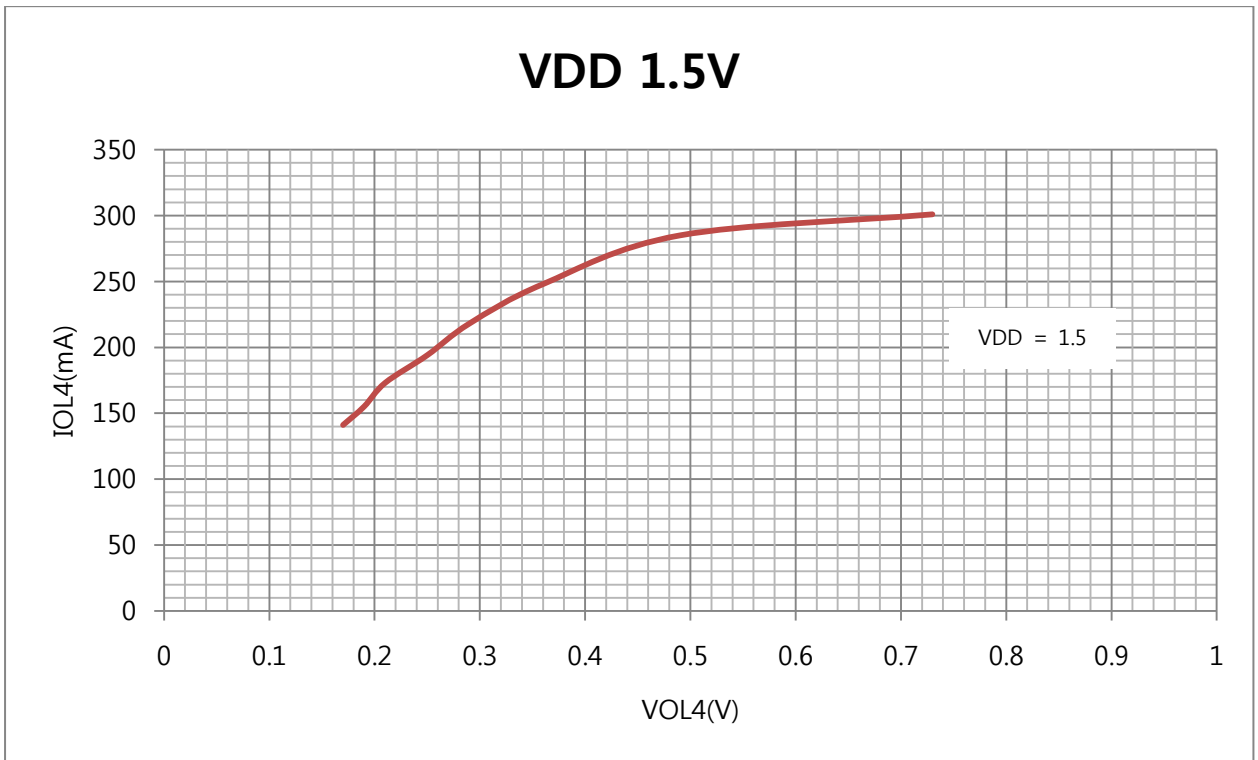


Fig 7-2. I_{OL4} vs. V_{OL4} Graph (REMOUT Port with built-in Transistor)

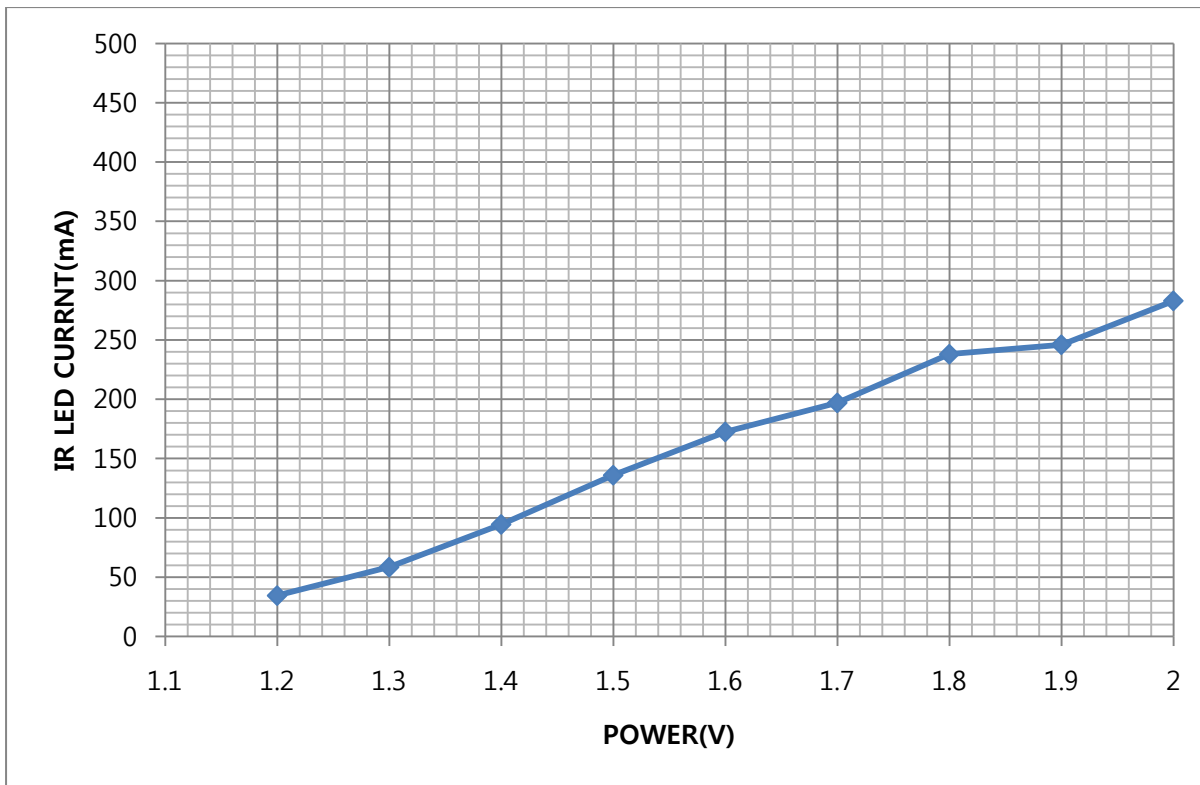


Fig 7-3. Approximate IR LED CURRENT (Reference data of sample)
 (IR LED : BIR-BM13E4G-2-KR)

8. Architecture

8.1 Program Memory (EPROM)

The MC40P5x04B series can incorporate maximum 4,096 words (4 bank x 64 words x 16 page x 8bits) for program memory. Program counter PC (A0~A5), Page address register (A6~A9) and Bank address register (A10, A11) are used to address the whole area of program memory having an instruction (8bits) to be next executed.

The program memory consists of 64 words on each page, and thus each page can hold up to 64 steps of instructions.

The program memory is composed as shown below

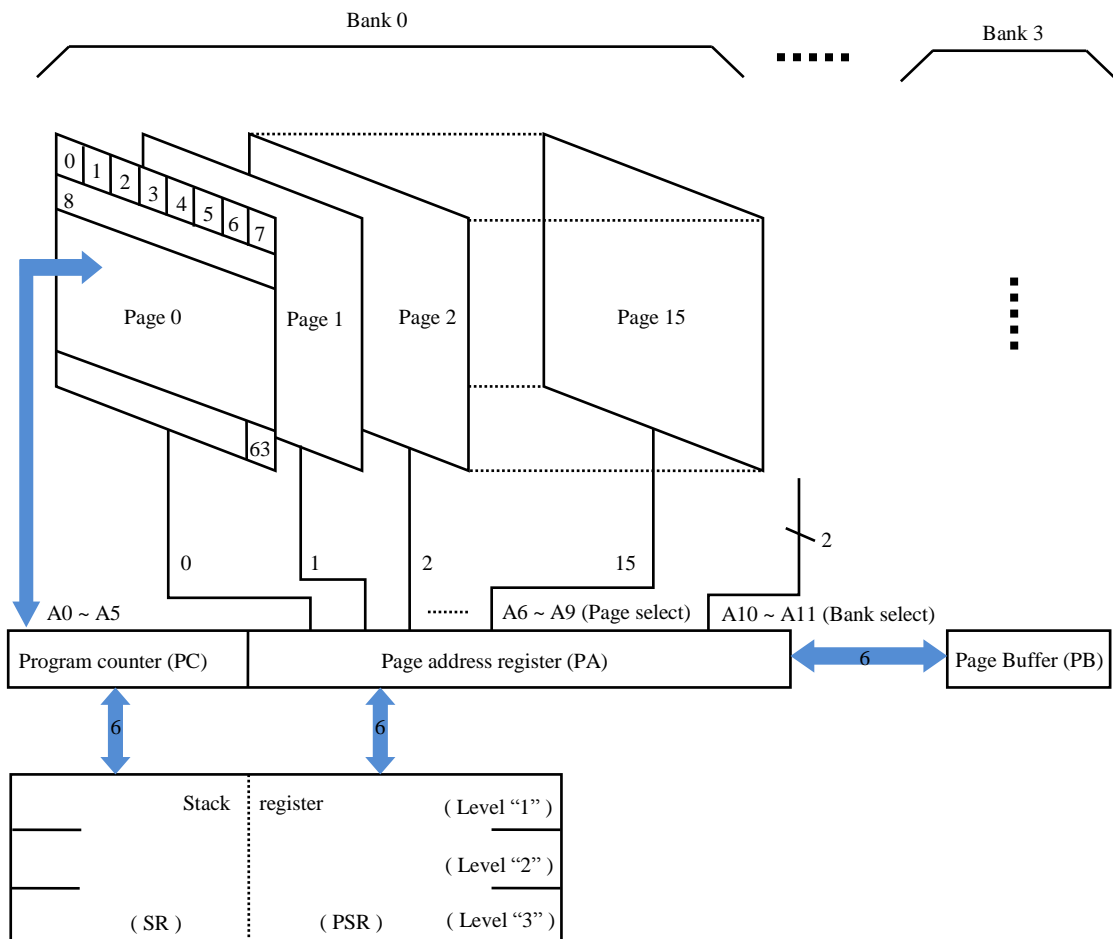


Fig 8-1 Configuration of Program Memory

8.2 EPROM Address Register

The following registers are used to address the EPROM.

- Page address register (PA)
Holds EPROM's page number (0~Fh) and bank address (0 ~ 3h)to be addressed.
- Page buffer register (PB)

Value of PB is loaded by an LPBI command when newly addressing a page. Then it is shifted into the PA when rightly executing a branch instruction (BR) and a subroutine call (CAL).

When addressing more than 1K, LPBI command must be used continuously. First LPBI value will be written at LSB 4bits, second is written at MSB 2bits.

	PA/PB	PC
MC40P5x01	4 bit	6 bit
MC40P5x04B	6 bit	6 bit

Fig 8-2 Compare MC40P5x01 with MC40P5x04B

※ Example of command flow

Command	PA	PB	PC
⋮	XX_XXXX	XX_XXXX	pc
LPBI #7 (high 2bits 00, low 4bits #7)	XX_XXXX	00_0111	Next pc
LPBI #3 (high 2bits #3, low 4bits keep)	XX_XXXX	11_0111	Next pc
BR #2A	11_0111	11_0111	#2A

- Program counter (PC)
Available for addressing word on each page.
- Stack register (SR)
Stores returned-word address in the subroutine call mode.

(1) Page address register and page buffer register

Address one of pages #0 to #15 in the EPROM by the 4-bit binary counter. Unlike the program counter, the page address register is usually unchanged so that the program will repeat on the same page unless a page changing command is issued. To change the page address, take two steps such as (1) writing in the page buffer what page to jump (execution of LPBI) and (2) execution of BR or CAL, because instruction code is of eight bits so that page and word can not be specified at the same time.

In case a return instruction (RTN) is executed within the subroutine that has been called In the other page, the page address will be changed at the same time.

(2) Program counter

This 6-bit binary counter increments for each fetch to address a word in the currently addressed page having an instruction to be next executed. For easier programming, at turning on the power, the program counter is reset to the zero location. The PA is also set to "0". Then the program counter specifies the next EPROM address in random sequence. When BR, CAL or RTN instructions are decoded, the switches on each step are turned off not to update the address. Then, for BR or CAL, address data are taken in from the instruction operands (a0 to a5), or for RTN, and address is fetched from stack register No. 1.

(3) Stack register

This stack register provides three stages each for the program counter (6bits) and the page address register (4bits) so that subroutine nesting can be made on three levels.

8.3 Data memory (RAM)

Up to 32 nibbles (16 words x 2pages x 4bits) is incorporated for storing data. The whole data memory area is indirectly specified by a data pointer (X, Y). Page number is specified by zero bit of X register, and words in the page by 4 bits in Y-register. Data memory is composed in 16 nibbles/page. Fig. 8-3 shows the configuration.

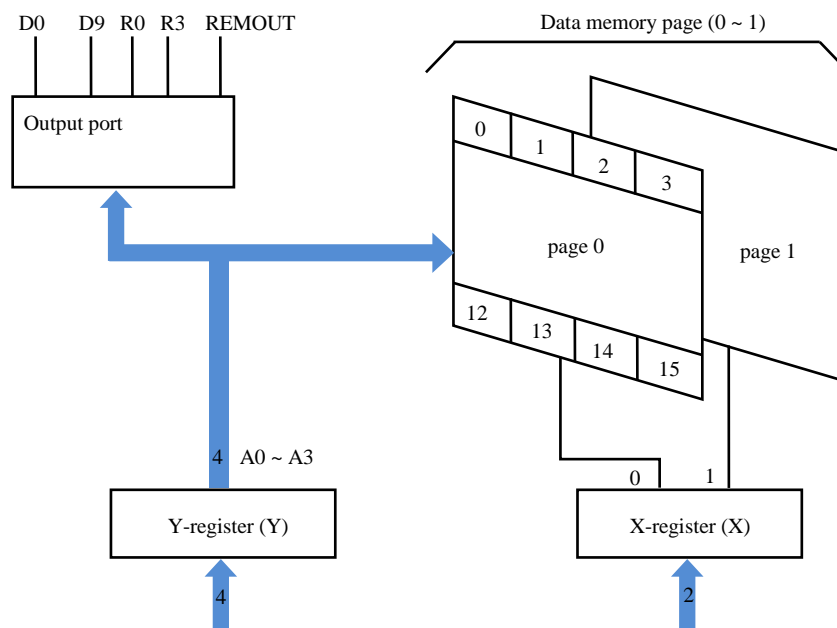


Fig 8-3 Configuration of Data Memory

8.4 X-register (X)

X-register is consist of 2bit, X0 is a data pointer of page in the RAM, X1 is only used for selecting of D8~D9 with value of Y-register

	X1 = 0 (X=0 or 1)	X1 = 1 (X=2 or 3)
Y = 0	D0	D8
Y = 1	D1	D9

Table 8-1 Mapping table between X and Y register for port access

8.5 Y-register (Y)

Y-register has 4 bits. It operates as a data pointer or a general-purpose register.

Y-register specifies and address (a0~a3) in a page of data memory, as well as it is used to specify an output port. Further it is used to specify a mode of carrier signal outputted from the REMOUT port. It can also be treated as a general-purpose register on a program.

8.6 Accumulator (Acc)

The 4-bit register for holding data and calculation results.

8.7 Arithmetic and Logic Unit (ALU)

In this unit, 4bits of adder/comparator are connected in parallel as it's main components and they are combined with status latch and status logic (flag.)

(1) Operation circuit (ALU)

The adder/comparator serves fundamentally for full addition and data comparison. It executes subtraction by making a complement by processing an inversed output of Acc (Acc +1)

(2) Status logic

This is to bring an ST, or flag to control the flow of a program. It occurs when a specified instruction is executed in three cases such as overflow or underflow in operation and two inputs unequal.

8.8 State Counter (SC)

A fundamental machine cycle timing chart is shown below. Every instruction is one byte length. Its execution time is the same. Execution of one instruction takes 48 clocks for fetch cycle and 48clocks for execute cycle (96 clocks in total).

Virtually these two cycles proceed simultaneously, and thus it is apparently completed in 48clocks (one machine cycle). Exceptionally BR, CAL and RTN instructions is normal execution time since they change an addressing sequentially. Therefore, the next instruction is prefetched so that its execution is completed within the fetch cycle.

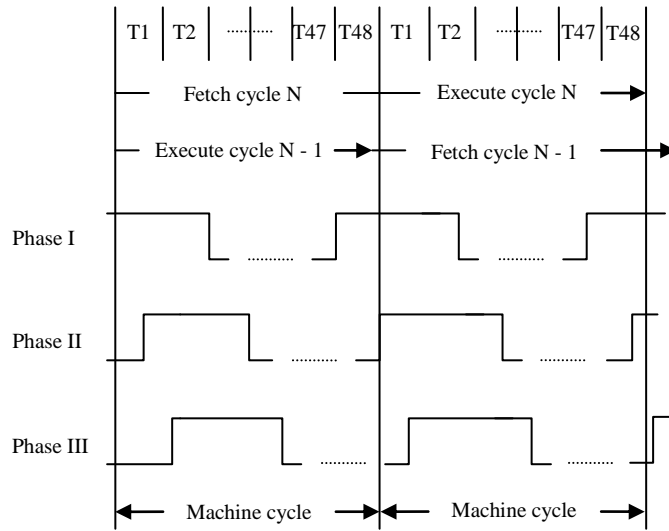


Fig 8-4 Fundamental timing chart

8.9 Clock Generator

The oscillator circuit is designed to operate with an external ceramic resonator. Oscillator circuit is able to organize by connecting ceramic resonator to outside.

* It is necessary to connect capacitor to outside in order to change ceramic resonator, you must refer to a manufacturer`s resonator matching guide.

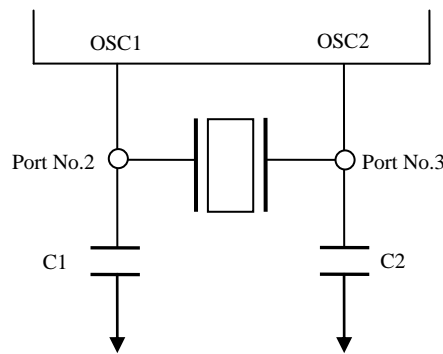
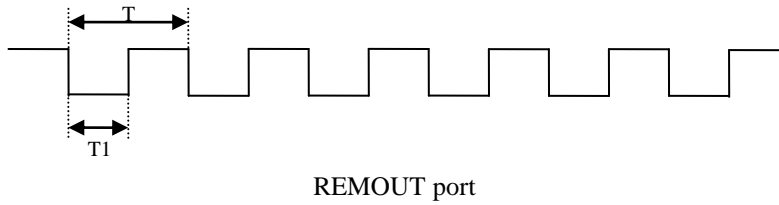


Figure 8-5 Oscillator circuit with external capacitor

8.10 Pulse Generator

The following frequency and duty ratio are selected for carrier signal outputted from the REMOUT port depending on a PMR (Pulse Mode Register) value set in a program.



PMR	REMOUT signal	Carrier frequency ($f_{OSC} = 3.64\text{MHz}$)
0	$T=1/f_{PUL} = 96/ f_{OSC}, T1/T = 1/2$	37.92KHz
1	$T=1/f_{PUL} = 96/ f_{OSC}, T1/T = 1/3$	37.92KHz
2	$T=1/ f_{PUL} = 64/ f_{OSC}, T1/T = 1/2$	56.88KHz
3	$T=1/ f_{PUL} = 65/ f_{OSC}, T1/T = 22/65$	56.00KHz
4	$T=1/ f_{PUL} = 87/ f_{OSC}, T1/T = 1/3$	41.84KHz
5	No Carrier (same to inversion of D0~D9)	-
6	$T=1/ f_{PUL} = 91/ f_{OSC}, T1/T = 31/91$	40.00KHz
7	$T= 1/ f_{PUL} = 101/ f_{OSC}, T1/T = 34/101$	36.04KHz

*Default value is “0”

* f_{PUL} = Carrier Pulse frequency, f_{OSC} = Oscillation frequency

Table 8-3 PMR selection table

8.11 Reset Operation

MC40P5x04B series have three reset sources. One is a built-in Power-on reset circuit, Another is a built-in Low VDD Detection circuit, the other is the overflow of Watch Dog Timer (WDT). All reset operations are internal in the MC40P5x04B.

8.11.1 Built-in Power On Reset Circuit

MC40P5x04B series has a built-in Power-on reset circuit consisting of an about 1 MΩ Resistor and a 3pF Capacitor. When the Power-on reset pulse occurs, system reset signal is latched and WDT is cleared. After the overflow time of WDT(2¹³ x System clock time), system reset signal is released.

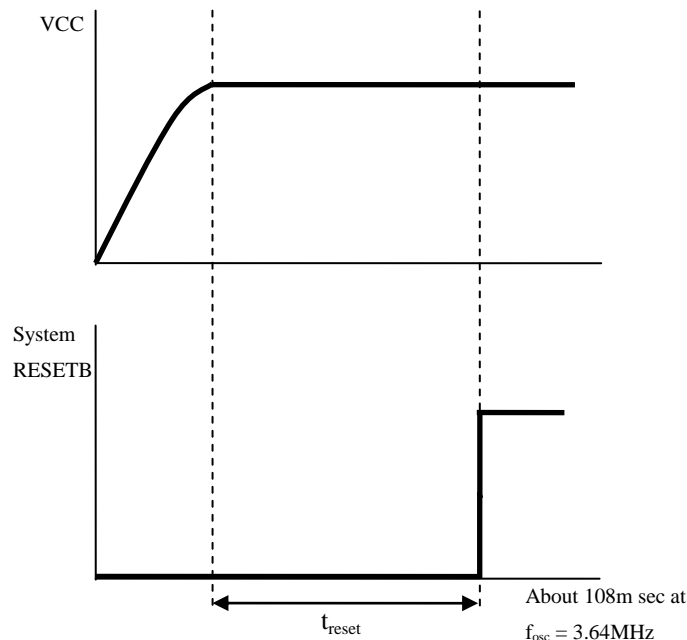
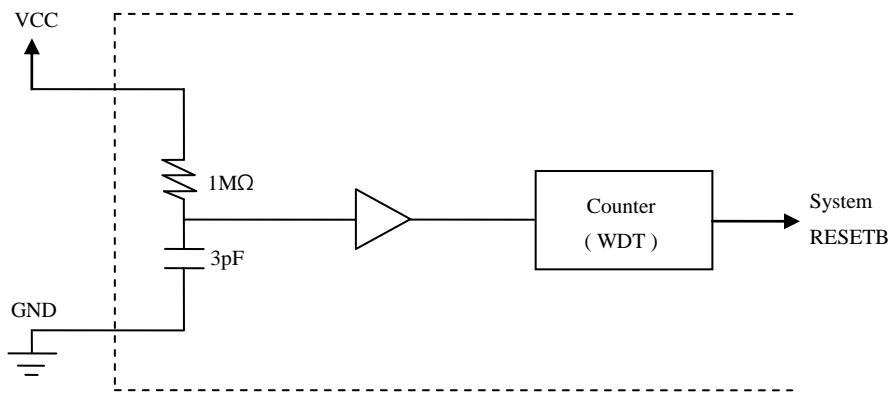


Fig 8-6 Power –On Reset Circuit and Timing Chart

8.11.2 Built-in Low VDD Reset Circuit

MC40P5x04B series have a Low VDD detection circuit.

If VDD become Reset Voltage of Low VDD Detection circuit at a active status, system reset occur and WDT is cleared.

After VDD is increased upper Reset Voltage again, WDT is re-counted and if WDT is overflowed, system reset is released.

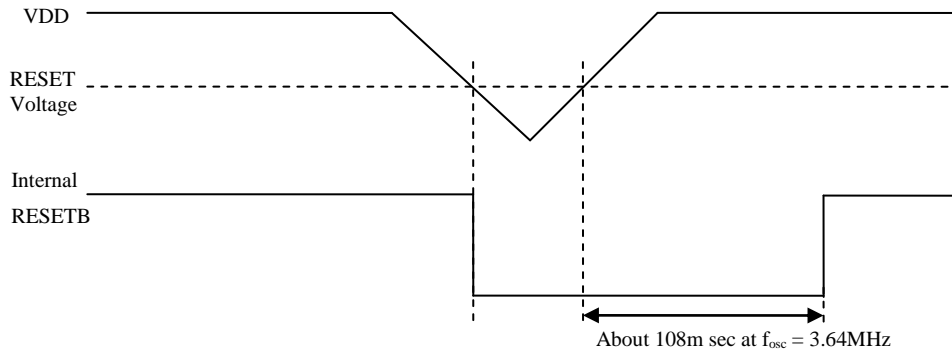


Fig 8-7 Low Voltage Detection Timing Chart

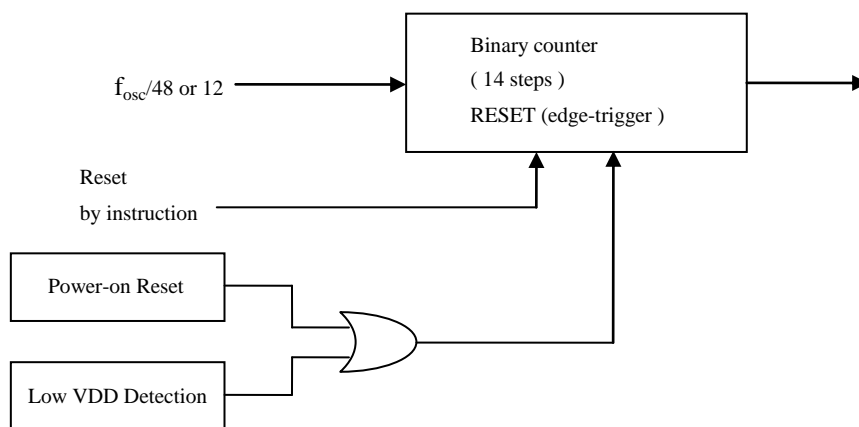
8.11.3 Watch Dog Timer (WDT)

Watch dog timer is organized binary of 14 steps. The signal of $f_{osc} / 48$ or $f_{osc} / 12$ cycle comes in the first step of WDT after WDT reset. If this counter was overflowed, reset signal automatically come out so that internal circuit is initialized.

The overflow time is $8 \times 6 \times 2^{13} / f_{osc}$ (108.026ms at $f_{osc} = 3.64\text{MHz}$)

Normally, the binary counter must be reset before the overflow by using reset instruction (WDTR), Power-on reset pulse or Low VDD detection pulse.

* It is constantly reset in STOP mode. When STOP is released, counting is restarted. (Refer to STOP operation)



8.12 STOP Operation

Stop mode can be achieved by STOP instructions.

In stop mode

1. Oscillator is stopped, the operating current is low.
2. Watch dog timer is reset, D0~D3 output are "L", and REMOUT output are High-Z.
3. Part of output pin other than WDT, D0~D3 and REMOUT output have a value before come into stop mode.

Stop mode is released when one of K or R input is going to "L".

1. State of D0~D7 output and REMOUT output is return to state of before stop mode is achieved.
2. After 1,024 x 8 enable clocks for stable oscillating, First instruction start to operate.
3. In return to normal operation, WDT is counted from zero again.

But, at executing stop instruction, if one of K or R input is chosen to "L", stop instruction is same to NOP (No Operation) instruction.

Value of X-reg	Value of Y-reg	Operation
0 or 1	0 ~ 7	S0 : D(Y) ← 1, R0 : D(Y) ← 0
0 or 1	8	REMOUT port repeats "H" and "L" in pulse frequency. (When PMR = 5, it is fixed at "H" or "L") S0 : REMOUT ← 0 R0 : REMOUT(PMR) ← 1 (High-Z)
0 or 1	9	S0 : D0 ~ D9 ← 1 (High-Z) R0 : D0 ~ D9 ← 0
0 or 1	A ~ D	S0 : R(Y-Ah) ← 1 R0 : R(Y-Ah) ← 0
0 or 1	E	S0 : R0 ~ R3 ← 1 R0 : R0 ~ R3 ← 0
0 or 1	F	S0 : D0 ~ D9 ← 1, R0 ~ R3 ← 1 R0 : D0 ~ D9 ← 0, R0 ~ R3 ← 0
2 or 3	0	S0 : D(8) ← 1 R0 : D(8) ← 0
2 or 3	1	S0 : D(9) ← 1 R0 : D(9) ← 0

9. Instruction

INSTRUCTION FORMAT

All of the 43 instruction in MC40P5x04 series is format in two fields of OP code and operand which consist of eight bits. The following formats are available with different types of operands.

*Format I

All eight bits are for OP code without operand.

*Format II

Two bits are for operand and six bits for OP code. Two bits of operand are used for specifying bits of RAM and X-register (bit 1 and bit 7 are fixed at "0")

*Format III

Four bits are for operand and the others are OP code. Four bits of operand are used for specifying a constant loaded in RAM or Y-register, a comparison value of compare command, or page addressing in ROM.

*Format IV

Six bits are for operand and the others are OP code. Six bits of operand are used for word addressing in the ROM.

9.1 Instruction Table

The MC40P5x04B series provides the following 43 basic instructions.

	Category	Mnemonic	Function	ST*1
1	Register to Register	LAY	$A \leftarrow Y$	S
2		LYA	$Y \leftarrow A$	S
3		LAZ	$A \leftarrow 0$	S
4	RAM to Register	LMA	$M(X, Y) \leftarrow A$	S
5		LMAIY	$M(X, Y) \leftarrow A, Y \leftarrow Y + 1$	S
6		LYM	$Y \leftarrow M(X, Y)$	S
7		LAM	$A \leftarrow M(X, Y)$	S
8		XMA	$A \leftrightarrow M(X, Y)$	S
9	Immediate	LYI i	$Y \leftarrow i$	S
10		LMIIY i	$M(X, Y) \leftarrow i, Y \leftarrow Y + 1,$	S
11		LXI n	$X \leftarrow n$	S
12	RAM Bit Manipulation	SEM n	$M(n) \leftarrow 1$	S
13		REM n	$M(n) \leftarrow 0$	S
14		TM n	TEST $M(n) \leftarrow 1$	E
15	ROM Address	BR a	if ST = 1 then Branch	S
16		CAL a	if ST = 1 then Subroutine call	S
17		RTN	Return from Subroutine	S
18		LPBI i	$PB \leftarrow i$ ((PB3~0 or PB5~4 loading) ^{*3}	S
19	Arithmetic	AM	$A \leftarrow A + M(X, Y)$	C
20		SM	$A \leftarrow M(X, Y) - A$	B
21		IM	$A \leftarrow M(X, Y) + 1$	C
22		DM	$A \leftarrow M(X, Y) - 1$	B
23		IA	$A \leftarrow A + 1$	S
24		IY	$Y \leftarrow Y + 1$	C

25	Arithmetic	DA	$A \leftarrow A - 1$	B
26		DY	$Y \leftarrow Y - 1$	B
27		EORM	$A \leftarrow A \oplus M(X, Y)$	S
28		NEGA	$A \leftarrow \bar{A} + 1$	Z
29	Comparison	ALEM	TEST $A \leq M(X, Y)$	E
30		ALEI i	TEST $A \leq i$	E
31		MNEZ	TEST $M(X, Y) \neq 0$	N
32		YNEA	TEST $Y \neq A$	N
33		YNEI i	TEST $Y \neq i$	N
34		KNEZ	TEST $K \neq 0$	N
35		RNEZ	TEST $R \neq 0$	N
36	Input/ Output	LAK	$A \leftarrow K$	S
37		LAR	$A \leftarrow R$	S
38		SO	Output(Y) $\leftarrow 1^{*2}$	S
39		RO	Output(Y) $\leftarrow 0^{*2}$	S
40	Control	WDTR	Watch Dog Timer Reset	S
41		STOP	Stop operation	S
42		LPY	$PMR \leftarrow Y$	S
43		NOP	No operation	S

Note) i = 0~f, n = 0~3, a = 6bit PC Address

*1 Column ST indicates conditions for changing status. Symbols have the following meanings

- S: On executing an instruction, status is unconditionally set.
- C: Status is only set when carry or borrow has occurred in operation.
- B: Status is only set when borrow has not occurred in operation.
- E: Status is only set when equality is found in comparison.
- N: Status is only set when equality is not found in comparison.
- Z: Status is only set when the result is zero.

*2 Operation is settled by a value of Y-register..

*3 LPBI instruction loads data to PB3~0 or PB5~4 according to position.

Refer to ABL and ACALL macro.

9.2 DETAILS OF INSTRUCTION SYSTEM

All 43 basic instructions of the MC40P5X04B Series are one by one described in detail below.

Description Form.

Each instruction is headlined with its mnemonic symbol according to the instructions table given earlier. Then, for quick reference, it is described with basic items as shown below. After that, detailed comment follows.

*Items :

- Naming : Full spelling of mnemonic symbol
- Status : Check of status function
- Format :Categorized into I to IV
- Operand : Omitted for Format I
- Function

(1) LAY

Naming : Load Accumulator from Y-Register

Status : Set

Format : I

Function : $A \leftarrow Y$

<Comment> Data of four bits in the Y-register is unconditionally transferred to the accumulator. Data in the Y-register is left unchanged.

(2) LYA

Naming : Load Y-register from Accumulator

Status : Set

Format : I

Function : $Y \leftarrow A$

<Comment> Load Y-register from Accumulator

(3) LAZ

Naming : Clear Accumulator

Status : Set

Format : I

Function : $A \leftarrow 0$

<Comment> Data in the accumulator is unconditionally reset to zero.

(4) LMA

Naming : Load Memory from Accumulator

Status : Set

Format : I

Function : $M(X,Y) \leftarrow A$

<Comment> Data of four bits from the accumulator is stored in the RAM location addressed by the X-register and Y-register. Such data is left unchanged.

(5) LMAIY

Naming : Load Memory from Accumulator and Increment Y-Register

Status : Set

Format : I

Function : $M(X,Y) \leftarrow A, Y \leftarrow Y+1$

<Comment> Data of four bits from the accumulator is stored in the RAM location addressed by the X register and Y-register. Such data is left unchanged.

(6) LYM

Naming : Load Y-Register form Memory

Status : Set

Format : I

Function : $Y \leftarrow M(X,Y)$

<Comment> Data from the RAM location addressed by the X-register and Y-register is loaded into the Y-register. Data in the memory is left unchanged.

(7) LAM

Naming : Load Accumulator from Memory

Status : Set

Format : I

Function : $A \leftarrow M(X,Y)$

<Comment> Data from the RAM location addressed by the X-register and Y-register is loaded into the Y-register. Data in the memory is left unchanged.

(8) XMA

Naming : Exchanged Memory and Accumulator

Status : Set

Format : I

Function : $M(X,Y) \leftrightarrow A$

<Comment> Data from the memory addressed by X-register and Y-register is exchanged with data from the accumulator. For example, this instruction is useful to fetch a memory word into the accumulator for operation and store current data from the accumulator into the RAM. The accumulator can be restored by another XMA instruction.

(9) LYI i

Naming : Load Y-Register from Immediate

Status : Set

Format : III

Operand : Constant $0 \leq i \leq 15$

Function : $Y \leftarrow i$

<Purpose> To load a constant in Y-register. It is typically used to specify Y-register in a particular RAM word address, to specify the address of a selected output line, to set Y-register for specifying a carrier signal outputted from OUT port, and to initialize Y-register for loop control. The accumulator can be restored by another XMA instruction.

<Comment> Data of four bits from operand of instruction is transferred to the Y-register.

(10) LMIY i

Naming : Load Memory from Immediate and Increment Y-Register
 Status : Set
 Format : III
 Operand : Constant $0 \leq i \leq 15$
 Function : $M(X,Y) \leftarrow i, \quad Y \leftarrow Y + 1$
 <Comment> Data of four bits from operand of instruction is stored into the RAM location addressed by the X-register and Y-register. Then data in the Y-register is incremented by one.

(11) LXI n

Naming : Load X-Register from Immediate
 Status : Set
 Format : II
 Operand : X file address $0 \leq n \leq 3$
 Function : $X \leftarrow n$
 <Comment> A constant is loaded in X-register. It is used to set X-register in an index of desired RAM page. Operand of 1 bit of command is loaded in X-register.

(12) SEM n

Naming : Set Memory Bit
 Status : Set
 Format : II
 Operand : Bit address $0 \leq n \leq 3$
 Function : $M(X,Y,n) \leftarrow 1$
 <Comment> Depending on the selection in operand of operand, one of four bits is set as logic 1 in the RAM memory addressed in accordance with the data of the X-register and Y-register.

(13) REM n

Naming : Reset Memory Bit
 Status : Set
 Format : II
 Operand : Bit address $0 \leq n \leq 3$
 Function : $M(X,Y,n) \leftarrow 0$
 <Comment> Depending on the selection in operand of operand, one of four bits is set as logic 0 in the RAM memory addressed in accordance with the data of the X-register and Y-register.

(14) TM n

Naming : Test Memory Bit

Status : Comparison results to status

Format : II

Operand : Bit address $0 \leq n \leq 3$

Function : $M(X,Y,n) = 1 ?$

$ST \leftarrow 1$ when $M(X,Y,n)=1$, $ST \leftarrow 0$ when $M(X,Y,n)=0$

<Purpose> A test is made to find if the selected memory bit is logic. 1 Status is set depending on the result.

(15) BR a

Naming : Branch on status 1

Status : Conditional depending on the status

Format : IV

Operand : Branch address a (Addr)

Function : When $ST = 1$, $PA \leftarrow PB$, $PC \leftarrow a(\text{Addr})$

When $ST = 0$, $PC \leftarrow PC + 1$, $ST \leftarrow 1$

Note : PC indicates the next address in a fixed sequence that is actually pseudo-random count.

<Purpose> For some programs, normal sequential program execution can be change. A branch is conditionally implemented depending on the status of results obtained by executing the previous instruction.

<Comment>

- Branch instruction is always conditional depending on the status.
 - a. If the status is reset (logic 0), a branch instruction is not rightly executed but the next instruction of the sequence is executed.
 - b. If the status is set (logic 1), a branch instruction is executed as follows.
- Branch is available in two types - short and long. The former is for addressing in the current page and the latter for addressing in the other page. Which type of branch to execute is decided according to the PB register. To execute a long branch, data of the PB register should in advance be modified to a desired page address through the LPBI instruction.

(16) CAL a

Naming :	Subroutine Call on status 1		
Status :	Conditional depending on the status		
Format :	IV		
Operand :	Subroutine code address a(Addr)		
Function :	When ST = 1 ,	PC ← a(Addr)	PA ← PB
		SR1 ← PC + 1,	PSR1 ← PA
		SR2 ← SR1	PSR2 ← PSR1
		SR3 ← SR2	PSR3 ← PSR2
	When ST = 0,	PC ← PC + 1	PB ← PS ST ← 1

Note : PC actually has pseudo-random count against the next instruction.

- <Comment>
- In a program, control is allowed to be transferred to a mutual subroutine. Since a call instruction preserves the return address, it is possible to call the subroutine from different locations in a program, and the subroutine can return control accurately to the address that is preserved by the use of the call return instruction (RTN). Such calling is always conditional depending on the status.
 - a. If the status is reset, call is not executed.
 - b. If the status is set, call is rightly executed.
 - The subroutine stack (SR) of three levels enables a subroutine to be manipulated on three levels. Besides, a long call (to call another page) can be executed on any level.
 - For a long call, an LPBI instruction should be executed before the CAL. When LPBI is omitted (and when PA=PB), a short call (calling in the same page) is executed.

(17) RTN

Naming :	Return from Subroutine		
Status :	Set		
Format :	I		
Function :	PC ← SR1	PA, PB ← PSR1	
	SR1 ← SR2	PSR1 ← PSR2	
	SR2 ← SR3	PSR2 ← PSR3	
		ST	← 1

<Purpose> Control is returned from the called subroutine to the calling program.

<Comment> Control is returned to its home routine by transferring to the PC the data of the return address that has been saved in the stack register (SR1). At the same time, data of the page stack register (PSR1) is transferred to the PA and PB.

(18) LPBI i

Naming : Load Page Buffer Register from Immediate
 Status : Set
 Format : III
 Operand : ROM page address $0 \leq i \leq 15$
 Function : $PB \leftarrow i$
 <Purpose> A new ROM page address is loaded into the page buffer register (PB). This loading is necessary for a long branch or call instruction. Refer to ABL and ACALL macro.
 <Comment> The PB register is loaded together with three bits from 4 bit operand.

(19) AM

Naming : Add Accumulator to Memory and Status 1 on Carry
 Status : Carry to status
 Format : I
 Function : $A \leftarrow M(X,Y)+A, \quad ST \leftarrow 1(\text{when total} > 15),$
 $ST \leftarrow 0(\text{when total} \leq 15)$
 <Comment> Data in the memory location addressed by the X and Y-register is added to data of the accumulator. Results are stored in the accumulator. Carry data as results is transferred to status. When the total is more than 15, a carry is caused to put "1" in the status. Data in the memory is not changed.

(20) SM

Naming : Subtract Accumulator to Memory and Status 1 Not Borrow
 Status : Carry to status
 Format : I
 Function : $A \leftarrow M(X,Y) - A \quad ST \leftarrow 1(\text{when } A \leq M(X,Y))$
 $ST \leftarrow 0(\text{when } A > M(X,Y))$
 <Comment> Data of the accumulator is, through a 2's complemental addition, subtracted from the memory word addressed by the Y-register. Results are stored in the accumulator. If data of the accumulator is less than or equal to the memory word, the status is set to indicate that a borrow is not caused. If more than the memory word, a borrow occurs to reset the status to "0".

(21) IM

Naming : Increment Memory and Status 1 on Carry
 Status : Carry to status
 Format : I
 Function : $A \leftarrow M(X,Y) + 1 \quad ST \leftarrow 1(\text{when } M(X,Y) \geq 15)$
 $ST \leftarrow 0(\text{when } M(X,Y) < 15)$
 <Comment> Data of the memory addressed by the X and Y-register fetched. Adding 1 to this word, results are stored in the accumulator. Carry data as results is transferred to the status. When the total is more than 15, the status is set. The memory is left unchanged.

(22) DM

Naming : Decrement Memory and Status 1 on Not Borrow

Status : Carry to status

Format : I

Function : $A \leftarrow M(X,Y) - 1$ $ST \leftarrow 1(\text{when } M(X,Y) \geq 1)$
 $ST \leftarrow 0(\text{when } M(X,Y) = 0)$

<Comment> Data of the memory addressed by the X and Y-register is fetched, and one is subtracted from this word (addition of Fh)> Results are stored in the accumulator. Carry data as results is transferred to the status. If the data is more than or equal to one, the status is set to indicate that no borrow is caused. The memory is left unchanged.

(23) IA

Naming : Increment Accumulator

Status : Set

Format : I

Function : $A \leftarrow A+1$

<Comment> Data of the accumulator is incremented by one. Results are returned to the accumulator. A carry is not allowed to have effect upon the status.

(24) IY

Naming : Increment Y-Register and Status 1 on Carry

Status : Carry to status

Format : I

Function : $Y \leftarrow Y + 1$ $ST \leftarrow 1(\text{when } Y = 15)$
 $ST \leftarrow 0(\text{when } Y < 15)$

<Comment> Data of the Y-register is incremented by one and results are returned to the Y-register. Carry data as results is transferred to the status. When the total is more than 15, the status is set.

(25) DA

Naming : Decrement Accumulator and Status 1 on Borrow

Status : Carry to status

Format : I

Function : $A \leftarrow A - 1$ $ST \leftarrow 1(\text{when } A \geq 1)$
 $ST \leftarrow 0(\text{when } A = 0)$

<Comment> Data of the accumulator is decremented by one. As a result (by addition of Fh), if a borrow is caused, the status is reset to "0" by logic. If the data is more than one, no borrow occurs and thus the status is set to "1".

(34) KNEZ

Naming : K Not Equal Zero
Status : The status is set only when not equal
Format : I
Function : When $K \neq 0$, $ST \leftarrow 1$
<Purpose> A test is made to check if K is not zero.
<Comment> Data on K are compared with zero. Results are transferred to the status. For input data not equal to zero, the status is set.

(35) RNEZ

Naming : R Not Equal Zero
Status : The status is set only when not equal
Format : I
Function : When $R \neq 0$, $ST \leftarrow 1$
<Purpose> A test is made to check if R is not zero.
<Comment> Data on R are compared with zero. Results are transferred to the status. For input data not equal to zero, the status is set.

(36) LAK

Naming : Load Accumulator from K
Status : Set
Format : I
Function : $A \leftarrow K$
<Comment> Data on K are transferred to the accumulator

(37) LAR

Naming : Load Accumulator from R
Status : Set
Format : I
Function : $A \leftarrow R$
<Comment> Data on R are transferred to the accumulator

(38) SO

Naming : Set Output Register Latch

Status : Set

Format : I

Function : $D(Y) \leftarrow 1 \quad 0 \leq Y \leq 7$
 $REMOUT \leftarrow 0 \quad Y = 8$
 $D0\sim D9 \leftarrow 1 \text{ (High-Z)} \quad Y = 9$
 $R(Y) \leftarrow 1 \quad Ah \leq Y \leq Dh$
 $R \leftarrow 1 \quad Y = Eh$
 $D0\sim D9, R \leftarrow 1 \quad Y = Fh$

<Purpose> A single D output line is set to logic 1, if data of Y-register is between 0 to 7. Carrier frequency comes out from REMOUT port, if data of Y-register is 8.
 All D output line is set to logic 1, if data of Y-register is 9. It is no operation, if data of Y-register between 10 to 15.
 When Y is between Ah and Dh, one of R output lines is set at logic 1.
 When Y is Eh, the output of R is set at logic 1.
 When Y is Fh, the output D0~D9 and R are set at logic 1.

<Comment> Data of Y-register is between 0 to 7, selects appropriate D output.
 Data of Y-register is 8, selects REMOUT port.
 Data of Y-register is 9, selects all D port.
 Data in Y-register, when between Ah and Dh, selects an appropriate R output (R0~R3).
 Data in Y-register, when it is Eh, selects all of R0~R3.
 Data in Y-register, when it is Fh, selects all of D0~D9 and R0~R3.

(39) RO

Naming : Reset Output Register Latch

Status : Set

Format : I

Function : $D(Y) \leftarrow 0 \quad 0 \leq Y \leq 7$
 $REMOUT \leftarrow 1(\text{High-Z}) \quad Y = 8$
 $D0\sim D9 \leftarrow 0 \quad Y = 9$
 $R(Y) \leftarrow 0 \quad Ah \leq Y \leq Dh$
 $R \leftarrow 0 \quad Y = Eh$
 $D0\sim D9, R \leftarrow 0 \quad Y = Fh$

<Purpose> A single D output line is set to logic 0, if data of Y-register is between 0 to 9.
 REMOUT port is set to logic 1(High-Z), if data of Y-register is 9.
 All D output line is set to logic 0, if data of Y-register is 9.
 When Y is between Ah and Dh, one of R output lines is set at logic 0.
 When Y is Eh, the output of R is set at logic 0
 When Y is Fh, the output D0~D9 and R are set at logic 1.

<Comment> Data of Y-register is between 0 to 7, selects appropriate D output.
 Data of Y-register is 8, selects REMOUT port.
 Data of Y-register is 9, selects D port.
 Data in Y-register, when between Ah and Dh, selects an appropriate R output (R0~R3).
 Data in Y-register, when it is Eh, selects all of R0~R3.
 Data in Y-register, when it is Fh, selects all of D0~D9 and R0~R3.

(40) WDTR

Naming : Watch Dog Timer Reset

Status : Set

Format : I

Function : Reset Watch Dog Timer (WDT)

<Purpose> Normally, you should reset this counter before overflowed counter for dc watch dog timer.
 this instruction controls this reset signal.

(41) STOP

Naming : STOP

Status : Set

Format : I

Function : Operate the stop function

<Purpose> Stopped oscillator, and little current.
 (See STOP function.)

(42) LPY

Naming : Pulse Mode Set

Status : Set

Format : I

Function : PMR ← Y

<Comment> Selects a pulse signal outputted from REMOUT port.

(43) NOP

Naming : No Operation

Status : Set

Format : I

Function : No operation

9.3 Assembler Macro

(44) CALL a (2 or 3byte) : Long_call Macro

Page call within same Bank (2bytes instruction) :

LPBI i ; i = low_page address(4bits), PB3~0(low_page address) <-- i
 CAL a ; see you "CAL" instruction.

Page call out of Bank (3bytes instruction) :

LPBI i ; i = low_page address(4bits), PB3~0(low_page address) <-- i
 LPBI i ; i = high_page address(2bits), PB5~4(high_page address) <-- i
 CAL a ; see you "CAL" instruction.

(45) BL a (2 or 3byte) : Long_branch Macro

Page branch within same Bank (2bytes instruction) :

LPBI i ; i = low_page address(4bits), PB0~3(low_page address) <-- i
 BR a ; see you "BR" instruction.

Page branch out of Bank (3bytes instruction) :

LPBI i ; i = low_page address(4bits), PB3~0(low_page address) <-- i
 LPBI i ; i = high_page address(2bits), PB5~4(high_page address) <-- i
 BR a ; see you "BR" instruction.

(46) ACALL a (3byte) : Absolute call Macro

Full-range ROM address call (3bytes instruction) :

LPBI i ; i = low_page address(4bits), PB3~0(low_page address) <-- i
 LPBI i ; i = high_page address(2bits), PB5~4(high_page address) <-- i
 CAL a ; see you "CAL" instruction.

(47) ABL a (3byte) : Absolute branch Macro

Full-range ROM address branch (3bytes instruction) :

LPBI i ; i = low_page address(4bits), PB3~0(low_page address) <-- i
 LPBI i ; i = high_page address(2bits), PB5~4(high_page address) <-- i
 BR a ; see you "BR" instruction.

* a(address) : 6bits, i (low_page address) : 4bits, i (high_page address) : 2bits
 Bank(16pages unit)

10. SPGM (Serial Program)

10.1 Summary of Protocol

※ The I²C Bus Protocol

The I²C bus protocol is a method of communication. It physically consists of 2 active wires. The active wires, called SCL and SDA, are both bi-directional. SCL is the Serial Clock line. It is used to synchronize all data transfers over the I²C bus. and SDA is the Serial Data line. The SCL & SDA lines are connected to all devices on the I²C bus.

• **necessary pins (5pins)**

- Serial Data (SDA) : D5
- Serial Clock (SCL) : K0
- Programming Power(VPP) : K3
- VDD
- VSS

OPTION PROGRAM / READ DATA Format

-	-	-	CLKSEL	Offset3	Offset2	Offset1	Offset0
---	---	---	--------	---------	---------	---------	---------

CLKSEL : change the Main Clock fosc/48, fosc/12 (default fosc/48) . If you select fosc/12, instruction cycle is 4 times faster than fosc/48, but Carrier Frequency isn't affected.

“0” : fosc/12

“1” : fosc/48

OFFSET : at normal mode, change the ROM address

-	-	-	-	LOCK 3	LOCK 2	LOCK 1	LOCK 0
---	---	---	---	--------	--------	--------	--------

- LOCK : when program is read by PGM-PLUS, Data Protection

ID 7	ID 6	ID 5	ID 4	ID 3	ID 2	ID 1	ID 0
------	------	------	------	------	------	------	------

- ID7 – ID0 : it can be treated as User ID.

MC40P5004B ID: 1000 1111b

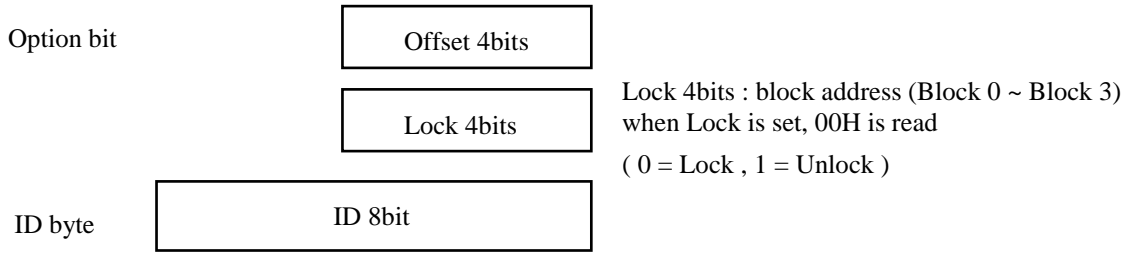
MC40P5204B ID: 1010 1111b

MC40P5404B ID: 1100 1111b

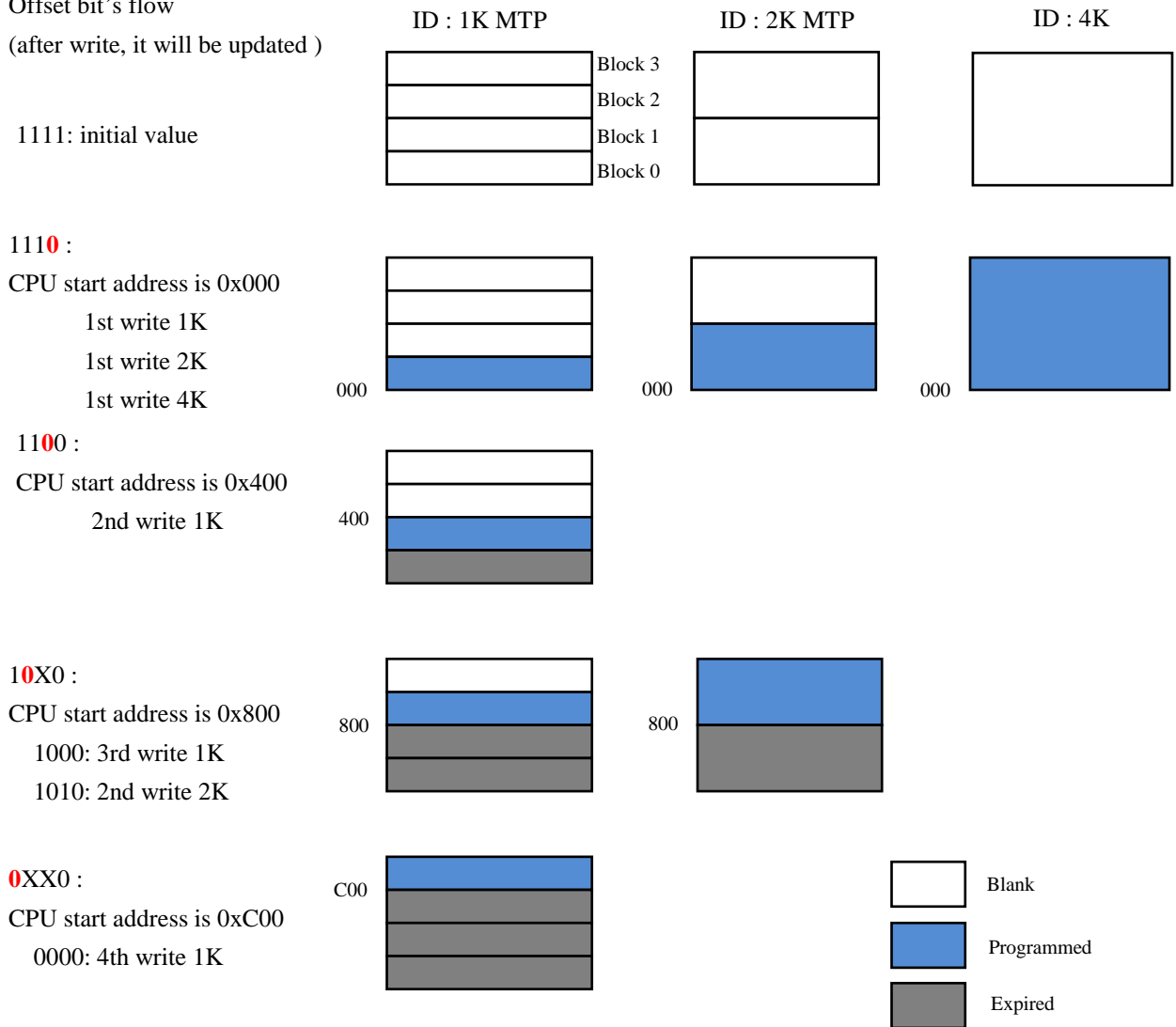
MC40P5104B ID: 1001 1111b

※ For protection the written program code, in other words it can not be read, you have to clear the Option bit to “0”, and for this, you have to write the Option Register to 1111_1110b. In this time, ID7 – ID0 keep the existing value without any effect

4K MTP (Multi Time Programming)



Offset bit's flow (after write, it will be updated)



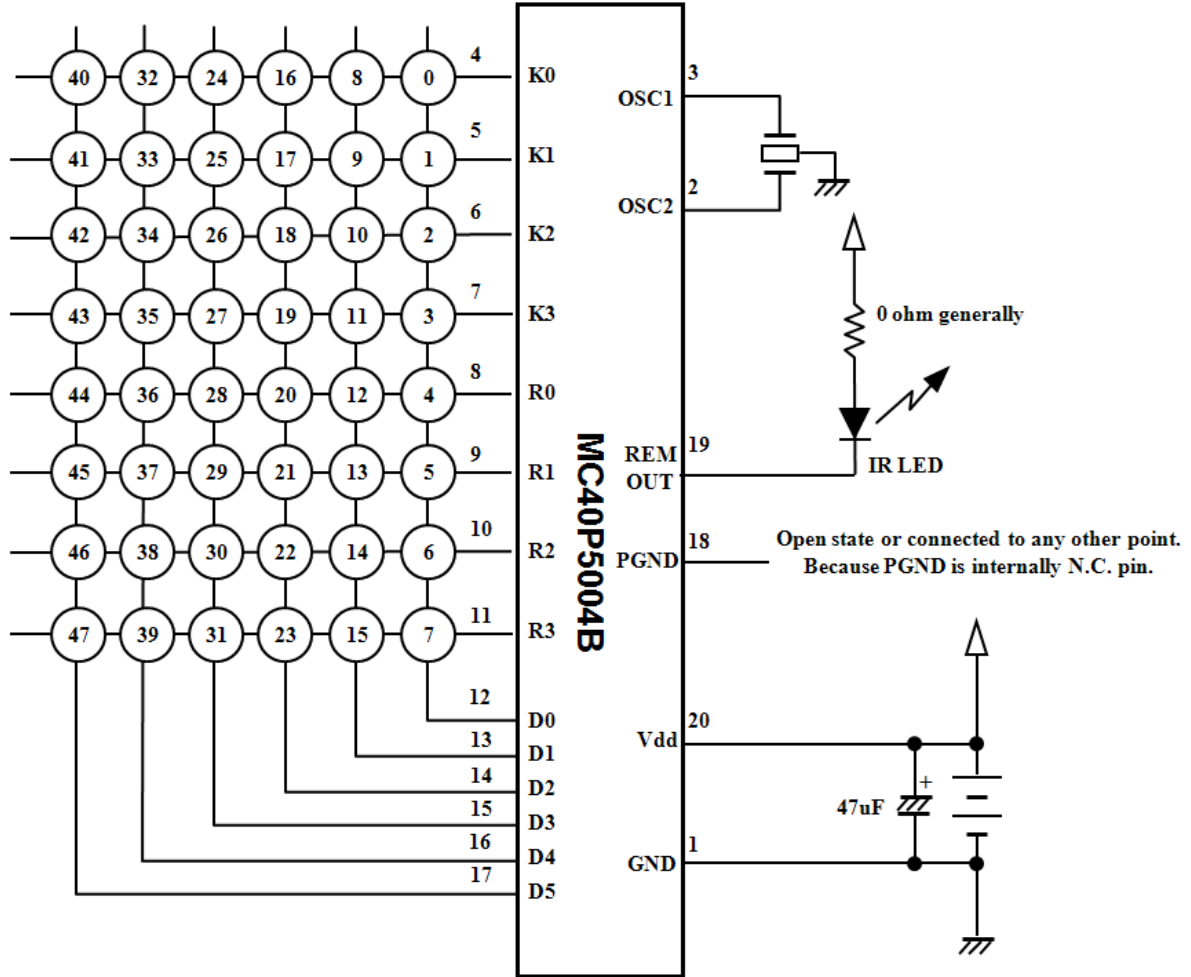
• Case of Available MTP

Case	1K	1K	1K	1K	Available
1K*4					O
2K*2					O
4K*1					O
1K, 1K, 2K					X
1K, 2K					O
1K, 2K, 1K					X
2K, 1K, 1K					O

	1 st PGM
	2 nd PGM
	3 rd PGM
	4 th PGM
	Expired

11. APPLICATION

Circuit Diagram of MC40P5004B



*NOTE

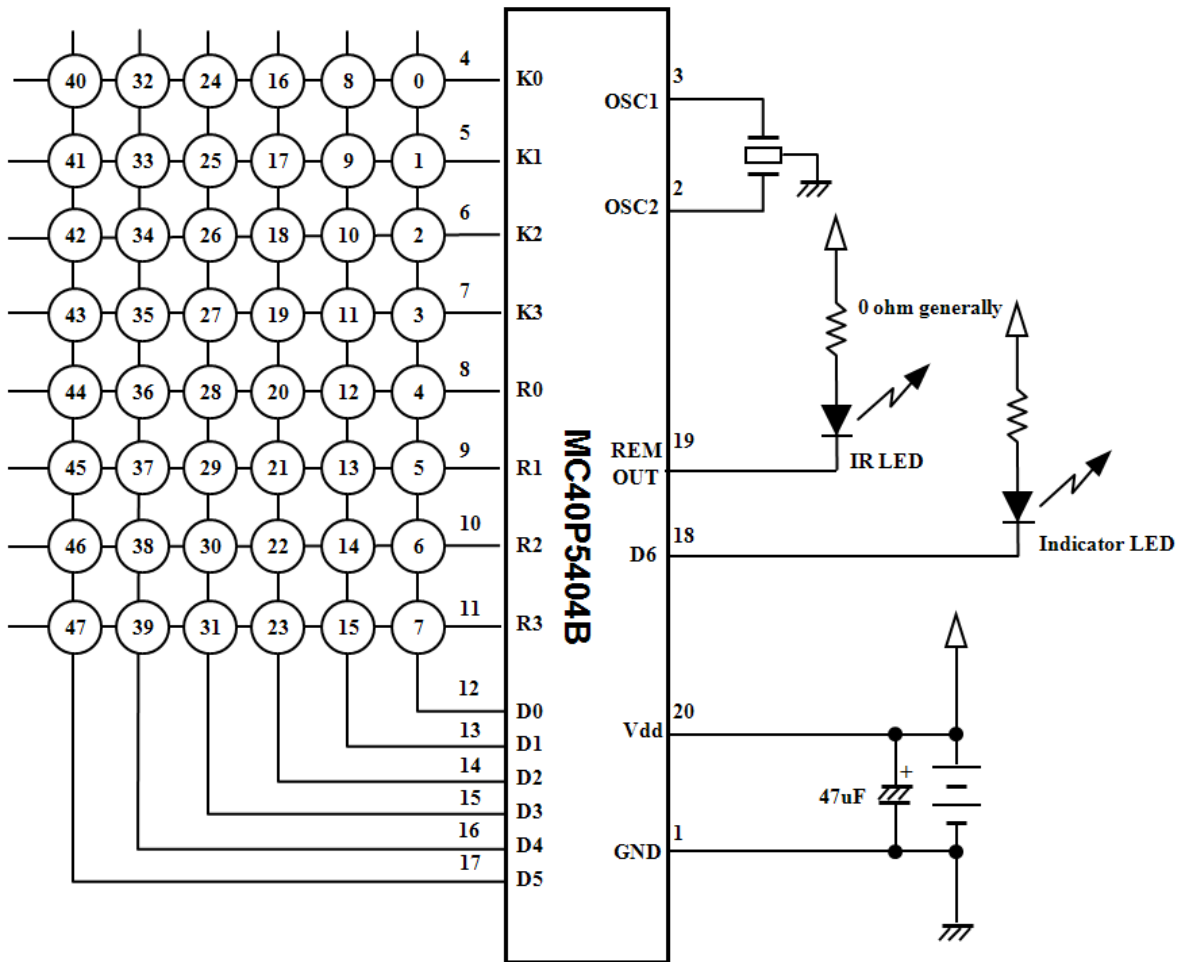
1. Power stabilization capacitor :

is recommended to be placed as close to VDD/GND pins as possible for better MCU operation.

2. IR LED serial resistance :

Coin cell batteries generally causes large VDD level drop during IR LED emission due to low current capacity and high internal resistance. Therefore, users have to optimize the circuit conditions by adjusting IR LED serial resistance to reduce IR LED current and to avoid early occurrence of LVD reset and to meet aiming IR reaching range and longer battery lifetime.

Circuit Diagram of MC40P5404B



***NOTE**

1. Power stabilization capacitor :

is recommended to be placed as close to VDD/GND pins as possible for better MCU operation.

2. IR LED serial resistance :

Coin cell batteries generally causes large VDD level drop during IR LED emission due to low current capacity and high internal resistance. Therefore, users have to optimize the circuit conditions by adjusting IR LED serial resistance to reduce IR LED current and to avoid early occurrence of LVD reset and to meet aiming IR reaching range and longer battery lifetime.