# OKI semiconductor
# MSM66207/66P207

## OKI ORIGINAL HIGH PERFORMANCE
## CMOS SINGLE CHIP 8/16-BIT MICROCONTROLLER

## GENERAL DESCRIPTION

The OKI MSM66207 is a new generation, high performance single chip microcontroller implemented in silicon gate complementary metal oxide semiconductor technology (CMOS). Integrated within this chip are 16-bit ALU, 32K bytes of mask program ROM, 1K bytes of data RAM, 48 I/O lines, built-in 16-bit timers, 10-bit A/D converter, serial I/O port, pulse width modulator (PWM), and oscillator. Also available is the MSM66P207, which replaces the on-chip program memory with one-time PROM.

## FEATURES

- 8-Bit External Data Bus Interface

- 16-Bit Internal Architecture

- 64K address space for program memory (including 32K bytes on-chip ROM)

- 64K address space for data memory (including 1K bytes on-chip RAM)

- High speed execution
  Minimum Cycle for Instruction:
  400ns (10MHz)

- The Abundance of Powerful Instructions
  8/16 data transfer operation
  8/16 bit arithmetic operation
    16(8) bit x 16(8) bit→32(16) bit
    32(16) bit x 16(8) bit→32(16) bit
    16(8) bit ± 16(8) bit→16(8) bit
  8/16 logic operation
  Bit operation
  ROM table access operation

- The same instruction allows both byte and word width operation according to Data Descriptor.
  That is to say, the same algorithm and the same source program lines are applicable to byte and word width data manipulation with only changing Data Descriptor.

- Many Addressing Modes

- 8 Input lines 40 Input/Output lines

- Built-in 16 bit timer x 4
  Each timer has the following 4 modes.
    Auto reload timer mode
    Clock output mode
    Capture register mode
    Real time output mode

- Serial Port x 1 ch.
  (variable bit length, baud rate generators for transmitter & receiver)
    Asynchronous normal mode
    Asynchronous multi processor communication mode
    Synchronous normal mode
    Synchronous multi processor communication mode

- 16 bit Pulse Width Modulator x 2

- Transition Detector x 4

- 10 bit A/D converter (8 channel)

- 1 non-maskable interrupt, 16 maskable interrupts

- Stand-by Function
  Software Clock stop
  Software CPU stop
  Hardware CPU stop

- 64 pin Shrink DIP/64 pin QFP/68 pin PLCC

# PIN CONFIGURATION

| | Pin | | Pin | |
|---|---|---|---|---|
| AD0/P0₀ | 1 | | 64 | VDD |
| AD1/P0₁ | 2 | | 63 | VREF |
| AD2/P0₂ | 3 | | 62 | AGND |
| AD3/P0₃ | 4 | | 61 | PS₇/AI7 |
| AD4/P0₄ | 5 | | 60 | PS₆/AI6 |
| AD5/P0₅ | 6 | | 59 | PS₅/AI5 |
| AD6/P0₆ | 7 | | 58 | PS₄/AI4 |
| AD7/P0₇ | 8 | | 57 | PS₃/AI3 |
| A8/P1₀ | 9 | | 56 | PS₂/AI2 |
| A9/P1₁ | 10 | | 55 | PS₁/AI1 |
| A10/P1₂ | 11 | | 54 | PS₀/AI0 |
| A11/P1₃ | 12 | | 53 | P4₇/TRANS3 |
| A12/P1₄ | 13 | | 52 | P4₆/TRANS2 |
| A13/P1₅ | 14 | | 51 | P4₅/TRANS1 |
| A14/P1₆ | 15 | | 50 | P4₄/TRANS0 |
| A15/P1₇ | 16 | | 49 | P4₃/PWM1 |
| P2₀ | 17 | | 48 | P4₂/PWM0 |
| P2₁ | 18 | | 47 | P4₁/TM1CK |
| P2₂ | 19 | | 46 | P4₀/TM0CK |
| CLKOUT/P2₃ | 20 | | 45 | P3₇/TM3IO |
| RESOUT | 21 | | 44 | P3₆/TM2IO |
| ALE | 22 | | 43 | P3₅/TM1IO |
| PSEN | 23 | | 42 | P3₄/TM0IO |
| RD | 24 | | 41 | P3₃/INT1 |
| WR | 25 | | 40 | P3₂/INT0 |
| READY | 26 | | 39 | P3₁/RXD |
| EA | 27 | | 38 | P3₀/TXD |
| FLT | 28 | | 37 | P2₇/RXC |
| RES | 29 | | 36 | P2₆/TXC |
| OSC0 | 30 | | 35 | P2₅/HLDA |
| OSC1 | 31 | | 34 | P2₄/HOLD |
| GND | 32 | | 33 | NMI |

# FUNCTIONAL BLOCK DIAGRAM
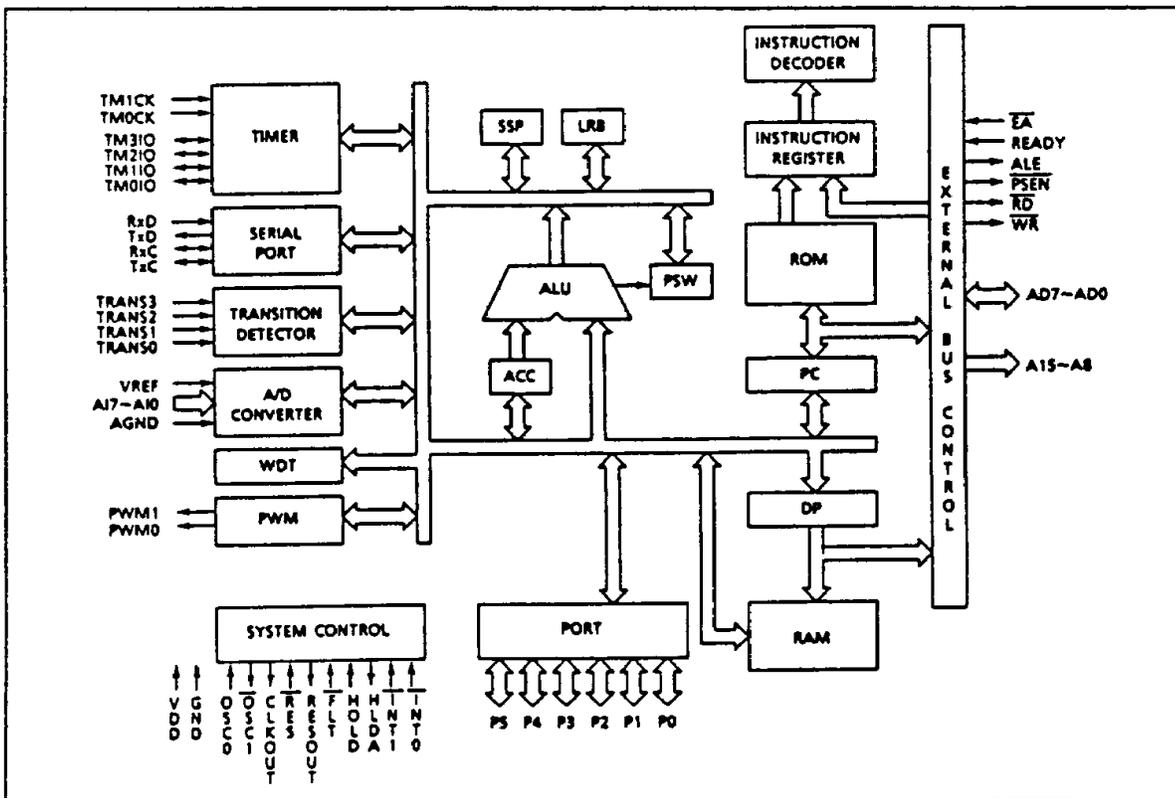
# PIN DESCRIPTION

| Designation | Input/Output | Function |
|---|---|---|
| P0₀ – P0₇/<br><br>AD0 – AD7 | I/O | P0: 8-bit I/O port. Each bit can be assigned to input or output.<br><br>AD: Outputs the lower 8 bits of program counter during external program memory fetch, and receives the addressed instruction under the control of $\overline{PSEN}$. Also outputs the address, Outputs or inputs data during an external data memory access instruction, under the control of ALE, $\overline{RD}$, and $\overline{WR}$. |
| P1₀ – P1₇/<br><br>A8 – A15 | I/O | P1: 8-bit I/O port. Each bit can be assigned to input or output.<br><br>A: Outputs the upper 8 bits of program counter ($PC_{8-15}$) during external program memory fetch. Also outputs the upper 8 bits of address during an external data memory access instructions. |
| P2₀ – P2₂<br><br>P2₃/CLKOUT<br><br>P2₄/HOLD<br><br>P2₅/HLDA<br><br>P2₆/TxC<br>P2₇/RxC | I/O | P2: 8-bit I/O port. Each bit can be assigned to input or output.<br><br>CLKOUT: Clock output pin. Output frequency range is equal to or twice the system clock.<br><br>HOLD: Input pin to request the CPU to enter the hardware power-down state.<br><br>HLDA: HOLD ACKNOWLEDGE: the HLDA signal appears in response to the HOLD signal and indicates that the CPU has entered the power-down state.<br><br>TxC: Transmitter clock input/output pin.<br><br>RxC: Receiver clock input/output pin. |
| P3₀/TxD<br><br>P3₁/RxD<br>P3₂/INT0<br>P3₃/INT1<br><br>P3₄/TM0IO<br><br>P3₅/TM1IO<br><br>P3₆/TM2IO<br><br>P3₇/TM3IO | I/O | P3: 8-bit I/O port. Each bit can be assigned to input or output.<br><br>TxD: Transmitter data output pin.<br><br>RxD: Receiver data input pin.<br><br>$\overline{INT}$: Interrupt Request Input pin.<br>Falling edge trigger or level trigger is selectable.<br><br>TM0IO~TM3IO: One of the following signals is output or input.<br>● clock twice the frequency range of the 16 bit timer overflow<br>● load trigger signal to the capture register input<br>● setting value output<br><br>Whether the signal is input or output depends on the mode. |

3

# PIN DESCRIPTION (Continued)

| Designation | Input/Output | Function |
|---|---|---|
| $P4_0$/TM0CK<br><br>$P4_1$/TM1CK<br><br>$P_{42}$/PWM0<br><br><br>$P4_3$/PWM1<br>$P4_4 - P4_7$/<br>TRANS0 - TRANS3 | I/O | P4:     8-bit I/O port. Each bit can be assigned to input or output<br><br>TM0CK, TM1CK:    Clock input pins of timer 0, timer 1.<br><br>TRANS:    Transition Detector.<br>      The input pins which sense the falling edge and set the flag.<br><br>PWM:    Pulse Wide Modulator output pin. |
| $P5_0 - P5_7$/<br><br>AI0 - AI7 | INPUT | P5:   8-bit input port.<br><br>AI:   Analog signal input pin for A/D converter. |
| RESOUT | OUTPUT | Output 'H' level when the CPU is in RESET cycle.<br>Reset to 'L' level by program. |
| ALE | OUTPUT | Address Latch Enable:<br>    The timing pulse to latch the lower 8 bit of the address output from port 0 when the CPU accesses the external memory. |
| $\overline{PSEN}$ | OUTPUT | Program Store Enable:<br>    The strobe pulse to fetch to external program memory. |
| $\overline{RD}$ | OUTPUT | Output strobe activated during a bus read cycle.<br>Used to enable data on to the bus from the external data memory. |
| $\overline{WR}$ | OUTPUT | Output strobe during a bus write cycle.<br>Used as write strobe to external data memory. |
| READY | INPUT | Used when the CPU accesses low speed peripherals. |
| $\overline{EA}$ | INPUT | Normaly set to 'H' level.<br>If set to 'L' level, the CPU fetches the code from external program memory. |
| $\overline{FLT}$ | INPUT | If $\overline{FLT}$ is 'H' level, ALE, $\overline{WR}$, $\overline{RD}$, $\overline{PSEN}$ are set 'H' level when reset.<br>If $\overline{FLT}$ is set to 'L', ALE, $\overline{WR}$, $\overline{RD}$, $\overline{PSEN}$ are set to floating level when reset. |
| $\overline{RES}$ | INPUT | RESET input pin. |
| $OSC_0$, $\overline{OSC_1}$ | | Oscillation circuit input and output. |
| $\overline{NMI}$ | INPUT | Non maskable interrupt input pin (falling edge) |
| VREF | INPUT | Reference voltage input pin for A/D converter |
| AGND | INPUT | Ground for A/D converter |
| VDD | | System power supply |
| GND | | Ground |

4

# MSM66207/66P207 ELECTRICAL CHARACTERISTICS

## ■ Absolute Maximum Rating

| Item | Symbol | Conditions | Rating | Unit |
|---|---|---|---|---|
| Power supply voltage | $V_{DD}$ | | $-0.3 \sim 7.0$ | V |
| Input voltage | $V_I$ | | $-0.3 \sim V_{DD} + 0.3$ | V |
| Output voltage | $V_O$ | $Ta = 25°C$ | $-0.3 \sim V_{DD} + 0.3$ | V |
| Analog reference voltage | $V_R$ | | $-0.3 \sim V_{DD} + 0.3$ | V |
| Analog input voltage | $V_{AI}$ | | $-0.3 \sim V_R$ | V |
| Power Dissipation | $P_D$ | $Ta = 25°C$, per package | 400 MAX | mW |
| | | $Ta = 25°C$, per output | 50 MAX | mW |
| Storage temperature | $T_{STG}$ | ——— | $-55 \sim +150$ | °C |

## ■ Operation Range

| Item | Symbol | Conditions | | Rating | Unit |
|---|---|---|---|---|---|
| Power supply voltage | $V_{DD}$ | $f_{(OSC)} \leqq 10MHz$ | | $4.5 \sim 5.5$ | V |
| Memory sustaining voltage | $V_{DDH}$ | $f_{(OSC)} = 0$ Hz | | $2 \sim 5.5$ | V |
| Operating frequency | $f_{(OSC)}$ | $V_{DD} = 5V \pm 10\%$ | | $0 \sim 10$ | MHz |
| Operating temperature | Top | ——— | | $-40 \sim +85$ | °C |
| Fan out | N | MOS load | | 20 | |
| | | TTL load | P0 | 2 | |
| | | | P1, P2, P3, P4 | 1 | |

5

## ■ DC Characteristics

$(V_{DD} = 5V \pm 10\%, Ta = -40 \sim +85°C)$

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| "H" input voltage   *1 *3 *6 | $V_{IH}$ | —— | 2.4 | | $V_{DD} + 0.3$ | V |
| "H" input voltage   *5 *7 | $V_{IH}$ | —— | 4.0 | | $V_{DD} + 0.3$ | V |
| "H" input voltage   *8 | $V_{IH}$ | —— | 4.2 | | $V_{DD} + 0.3$ | V |
| "H" input voltage   *2 | $V_{IH}$ | —— | 3.6 | | $V_{DD} + 0.3$ | V |
| "L" input voltage *1 *2 *3 *6 | $V_{IL}$ | —— | -0.3 | | 0.8 | V |
| "L" input voltage   *5 *7 | $V_{IL}$ | —— | -0.3 | | 0.8 | V |
| "L" input voltage   *8 | $V_{IL}$ | —— | -0.3 | | 0.4 | V |
| "H" output voltage *1 *4 | $V_{OH}$ | $I_O = -400\mu A$ | 4.2 | | | V |
| "H" output voltage *2 | $V_{OH}$ | $I_O = -200\mu A$ | 4.2 | | | V |
| "L" output voltage *1 *4 | $V_{OL}$ | $I_O = 3.2mA$ | | | 0.4 | V |
| "L" output voltage *2 | $V_{OL}$ | $I_O = 1.6mA$ | | | 0.4 | V |
| Input leakage current   *3 *6 *7 | $I_{IH}/I_{IL}$ | $V_I = V_{DD}/0V$ | | | 1/-1 | $\mu A$ |
| Input current   *5 | $I_{IH}/I_{IL}$ | $V_I = V_{DD}/0V$ | | | 1/-20 | $\mu A$ |
| Input current   *8 | $I_{IH}/I_{IL}$ | $V_I = V_{DD}/0V$ | | | 10/-10 | $\mu A$ |
| "H" output current *1 | $I_{OH}$ | $V_O = 2.4V$ | -2 | | | mA |
| "H" output current *2 | $I_{OH}$ | $V_O = 2.4V$ | -1 | | | mA |
| "L" output current *1 | $I_{OL}$ | $V_O = 2.4V$ | 10 | | | mA |
| "L" output current *2 | $I_{OL}$ | $V_O = 2.4V$ | 5 | | | mA |
| Output leakage current   *1 *2 *4 | $I_{LO}$ | $V_O = V_{DD}/0V$ | | | $\pm 2$ | $\mu A$ |
| Input capacity | $C_I$ | $f = 1MHz, Ta = 25°C$ | | 5 | | pF |
| Output capacity | $C_O$ | $f = 1MHz, Ta = 25°C$ | | 7 | | pF |
| Current consumption (during STOP) | $I_{DDS}$ | $V_{DD} = 2V, Ta = 25°C$ ** | | 0.2 | 10 | $\mu A$ |
| | | ** | | 1 | 100 | $\mu A$ |
| Current consumption (during HALT) | $I_{DDH}$ | $f_{(OSC)} = 10MHz$, No Load | | 6 | 10 | mA |
| Current consumption | $I_{DD}$ | $f_{(OSC)} = 10MHz$, No Load | | 20 | 35 | mA |

*1 : Applied to P0
*2 : Applied to P1, P2, P3, and P4
*3 : Applied to P5
*4 : Applied to $\overline{ALE}$, $\overline{PSEN}$, $\overline{RD}$, $\overline{WR}$, and RESOUT
*5 : Applied to $\overline{RES}$ and NMI
*6 : Applied to $\overline{READY}$ and $\overline{EA}$
*7 : Applied to FLT
*8 : Applied to OSCo
** : VDD or 0V for ports serving as the input pin.no-load for any other

6

# ■ AC Characteristics

- ● External program memory control

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| Clock (OSC) Pulse | $t_\phi W$ | — | 50 | | | nS |
| ALE pulse width | $t_{AW}$ | | $3t_{\phi W} - 20$ | | | nS |
| $\overline{PSEN}$ pulse width | $t_{PW}$ | | $4t_{\phi W} - 20$ | | | nS |
| $\overline{PSEN}$ pulse delay time | $t_{PAD}$ | | $t_{\phi W} - 20$ | | $t_{\phi W} + 20$ | nS |
| Low address set time | $t_{AAS}$ | | $2t_{\phi W} - 35$ | | $2t_{\phi W} + 20$ | nS |
| Low address hold time | $t_{AAH}$ | CL = 50pF | $t_{\phi W} - 20$ | | $t_{\phi W} + 40$ | nS |
| High address delay time | $t_{AAD}$ | | $t_{\phi W} - 20$ | | $t_{\phi W} + 40$ | nS |
| High address hold time | $t_{APH}$ | | $t_{\phi W} - 20$ | | $t_{\phi W} + 40$ | nS |
| Instruction set time | $t_{IS}$ | | 100 | | | nS |
| Instruction hold time | $t_{IH}$ | | 0 | | $t_{\phi W} - 20$ | nS |

● External data memory control

$(V_{DD} = 5V \pm 10\%, Ta = -40 \sim +85°C)$

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| Clock (OSC) pulse | $t_\phi W$ | — | 50 | | | nS |
| ALE pulse width | $t_{AW}$ | | $3t_\phi W - 20$ | | | nS |
| $\overline{RD}$ pulse width | $t_{RW}$ | | $4t_\phi W - 20$ | | | nS |
| $\overline{WR}$ pulse width | $t_{WW}$ | | $4t_\phi W - 20$ | | | nS |
| $\overline{RD}$ pulse delay time | $t_{RAD}$ | | $t_\phi W - 20$ | | $t_\phi W + 20$ | nS |
| $\overline{WR}$ pulse delay time | $t_{WAD}$ | | $t_\phi W - 20$ | | $t_\phi W + 20$ | nS |
| Low address set time | $t_{AAS}$ | | $2t_\phi W - 35$ | | $2t_\phi W + 20$ | nS |
| Low address hold time | $t_{AAH}$ | $CL = 50pF$ | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |
| High address delay time | $t_{AAD}$ | | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |
| High address hold time | $t_{ARH}$ | | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |
| High address hold time | $t_{AWH}$ | | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |
| Memory data set time | $t_{MS}$ | | 100 | | | nS |
| Memory data hold time | $t_{MH}$ | | 0 | | $t_\phi W - 20$ | nS |
| Data delay time | $t_{DD}$ | | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |
| Data hold time | $t_{DH}$ | | $t_\phi W - 20$ | | $t_\phi W + 40$ | nS |

CLK

$t_{\Phi W}$  $t_{\Phi W}$

ALE

$t_{AW}$

$\overline{PSEN}$

$t_{PAD}$  $t_{PW}$

AD$_{0-7}$   PC$_{0-7}$   INST$_{0-7}$

$t_{AAS}$  $t_{AAH}$   $t_{IS}$  $t_{IH}$

A$_{8-15}$   PC$_{8-15}$

$t_{AAD}$   $t_{APH}$

$\overline{RD}$

$t_{RAD}$  $t_{RW}$

AD$_{0-7}$   RAP$_{0-7}$   DIN$_{0-7}$

$t_{AAS}$  $t_{AAH}$   $t_{MS}$  $t_{MH}$

A$_{8-15}$   RAP$_{8-15}$

$t_{AAD}$   $t_{ARH}$

$\overline{WR}$

$t_{WAD}$  $t_{WW}$

AD$_{0-7}$   RAP$_{0-7}$   DOUT$_{0-7}$

$t_{AAS}$  $t_{AAH}$   $t_{DH}$

$t_{DD}$

A$_{8-15}$   RAP$_{8-15}$

$t_{AAD}$   $t_{AWH}$

9
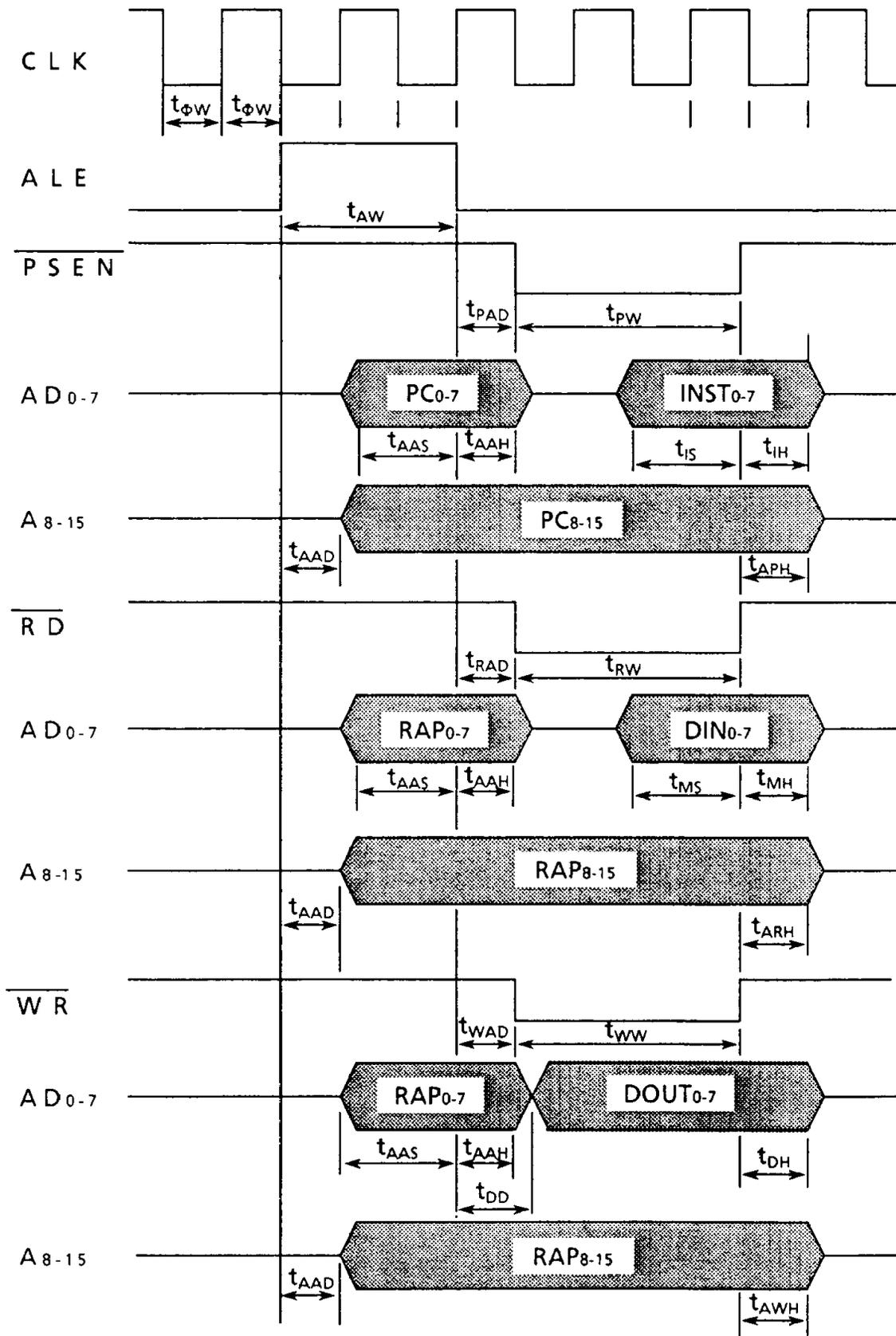
● Serial Port Control

Master mode                                    $(V_{DD} = 5V \pm 10\%, Ta = -40 \sim +85°C)$

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| Clock (OSC) pulse width | $t_\phi W$ | —— | 50 | | | nS |
| Serial clock pulse width | $t_{SCKW}$ | —— | $8t_\phi W$ | | | nS |
| Data set time | $t_{STMXS}$ | | $8t_\phi W + 40$ | | | nS |
| Data hold time | $t_{STMXH}$ | $C_L = 50pF$ | $6t_\phi W - 20$ | | | nS |
| Input data set time | $t_{SRMXS}$ | | $2t_\phi W + 10$ | | | nS |
| Input data hold time | $t_{SRNXH}$ | | 50 | | | nS |

● Serial Port Control

Slave mode                                     $(V_{DD} = 5V \pm 10\%, Ta = -40 \sim +85°C)$

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| Clock (OSC) pulse width | $t_\phi W$ | —— | 50 | | | nS |
| Serial clock pulse width | $t_{SCKW}$ | —— | $8t_\phi W$ | | | nS |
| Data set time | $t_{STSXS}$ | | $6t_\phi W + 40$ | | | nS |
| Data hold time | $t_{STSXH}$ | $C_L = 50pF$ | $6t_\phi W - 20$ | | | nS |
| Input data set time | $t_{SRSXS}$ | | 100 | | | nS |
| Input data hold time | $t_{SRSXH}$ | | 100 | | | nS |

OSC

$t_{\phi W}$    $t_{\phi W}$

SCK

$t_{SCKW}$    $t_{SCKW}$

SDOUT

(TXD)

$t_{STMXH}$    $t_{STMXS}$

SDIN

(RXD)

Valid    Valid

$t_{SRMXH}$    $t_{SRMXS}$

SCK

$t_{SCKW}$    $t_{SCKW}$

SDOUT

(TXD)

$t_{STSXH}$    $t_{STSXS}$

SDIN

(RXD)

Valid    Valid

$t_{SRSXH}$    $t_{SRSXS}$

11
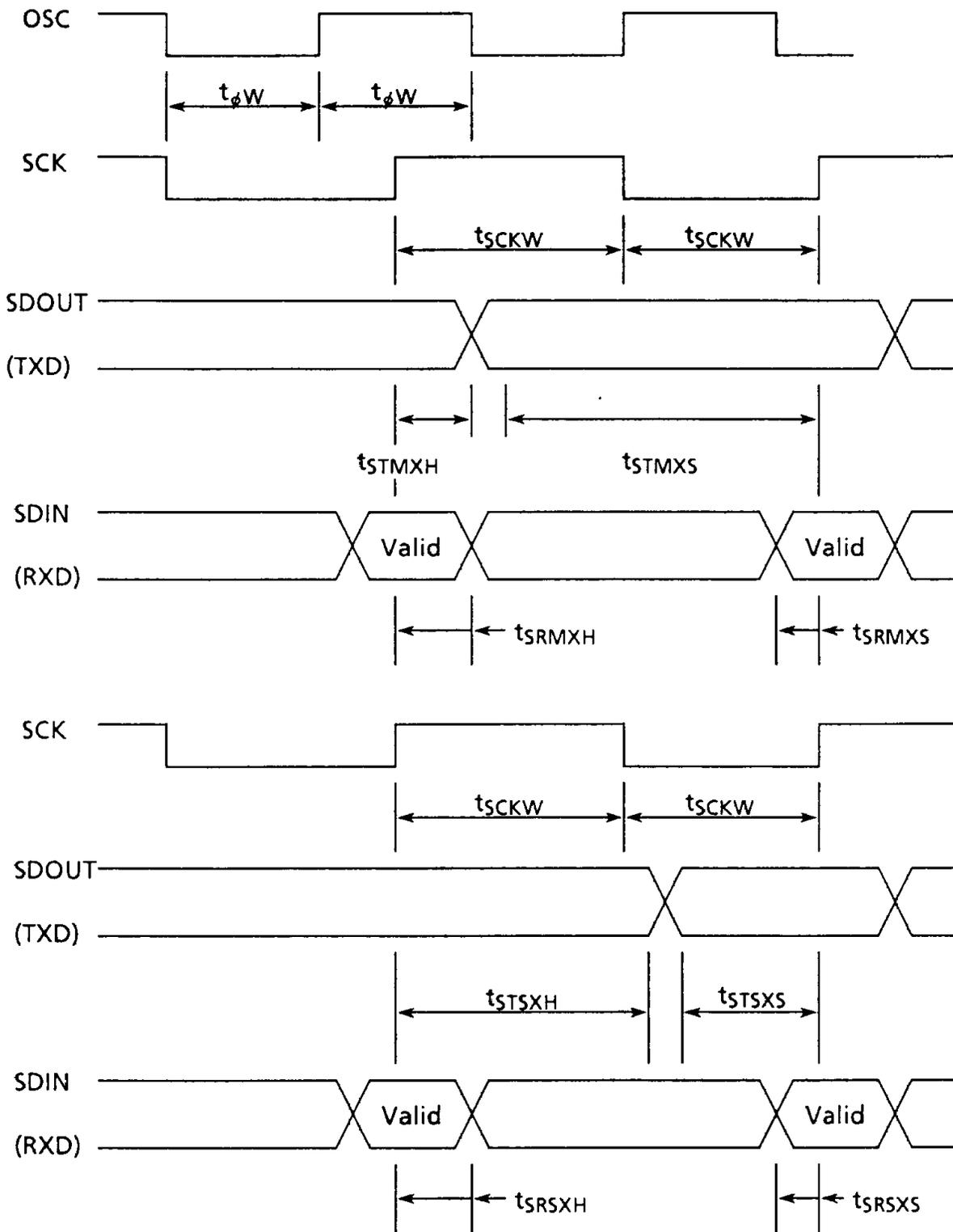
## ■ A/D Converter Characteristics

### ● Operation range

| Item | Symbol | Conditions | MIN | TYP | MAX | Unit |
|---|---|---|---|---|---|---|
| Power Supply voltage | $V_{DD}$ | $f(OSC) \leq 10MHz$ | 4.5 | | 5.5 | V |
| Analog reference voltage | $V_R$ | | 4.5 | | $V_{DD}$ | V |
| Analog input voltage | $V_{AI}$ | $V_{AG} = GND = 0V$ | $V_{AG}$ | | $V_R$ | V |
| Analog reference power voltage resistance | $R_R$ | | | 16 | | K$\Omega$ |
| Operating temperature | $T_{OP}$ | $V_{DD} = 5V \pm 10\%$ | -40 | | +85 | °C |

● A/D converter accuracy

Normal operation mode    $(V_{DD} = 5V \pm 10\%, f(OSC) = 10MHz, Ta = -40\sim +85°C)$

| Item | Symbol | Conditions | MIN | MIN * | TYP | TYP * | MAX | MAX * | Unit |
|---|---|---|---|---|---|---|---|---|---|
| Resolution | n | See the recommended circuit $V_R = V_{DD}$ $V_{AG} = GND = 0V$ analog input source impedance $\leq 5k\Omega$ one channel conversion time $t_C = 64\mu S$ | | | | | 10 | 10 | Bit |
| Absolute error | $E_A$ | | | | | | +3.0 −3.5 | +2.0 −3.5 | LSB |
| Relative error | $E_R$ | | | | | | ±1.5 | ±1.0 | LSB |
| Zero point error | $E_Z$ | | 0 | 0 | | | +3.0 | +2.0 | LSB |
| Full scale error | $E_F$ | | −0.5 | −1.0 | | | −3.5 | −3.5 | LSB |
| Differential linearity error | $E_D$ | | | | | | +3.0 | +2.0 | LSB |
| Crosstalk | $E_C$ | | | | ±0.5 | ±0.5 | | | LSB |

HALT/HOLD operation mode $(V_{DD} = 5V \pm 10\%, f(OSC) = 10MHz, Ta = -40\sim +85°C)$

| Item | Symbol | Conditions | MIN | MIN * | TYP | TYP * | MAX | MAX * | Unit |
|---|---|---|---|---|---|---|---|---|---|
| Resolution | n | See the recommended circuit $V_R = V_{DD}$ $V_{AG} = GND = 0V$ analog input source impedance $\leq 5k\Omega$ one channel conversion time $t_C = 64\mu S$ | | | | | 10 | 10 | Bit |
| Absolute error | $E_A$ | | | | | | +2.0 −3.5 | +1.0 −2.0 | LSB |
| Relative error | $E_R$ | | | | | | ±1.0 | ±0.5 | LSB |
| Zero point error | $E_Z$ | | +0.5 | +0.5 | | | +2.0 | +1.0 | LSB |
| Full scale error | $E_F$ | | −1.0 | −1.5 | | | −3.5 | −2.0 | LSB |
| Differential linearity error | $E_D$ | | | | | | +2.0 | +1.0 | LSB |
| Crosstalk | $E_C$ | | | | ±0.5 | ±0.5 | | | LSB |

\* $V_{DD} = 5V, Ta = 25°C$
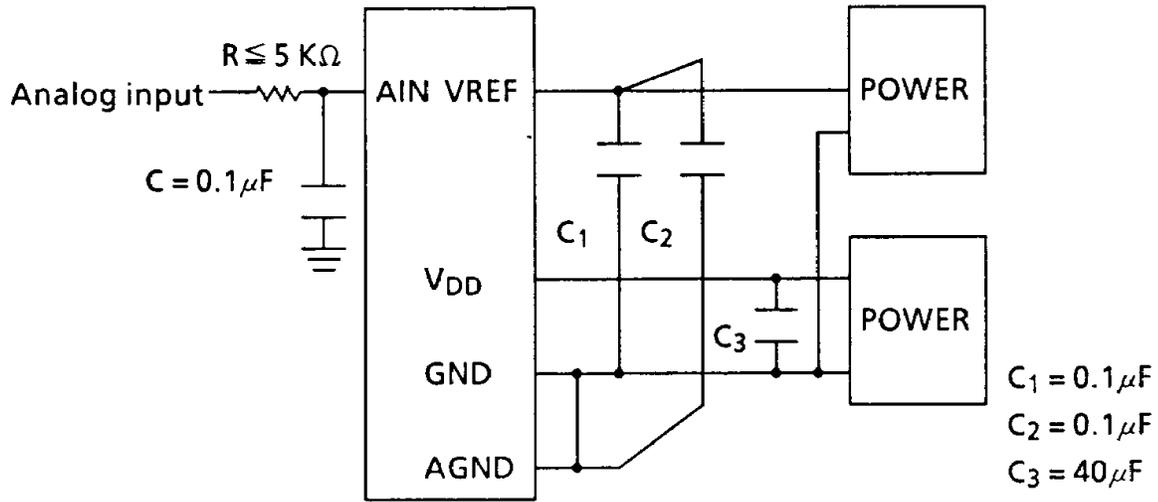
13

○ Recommended circuit



$R \leqq 5 K\Omega$

Analog input

$C = 0.1\mu F$

AIN VREF

$V_{DD}$

GND

AGND

$C_1$  $C_2$

$C_3$

POWER

POWER

$C_1 = 0.1\mu F$
$C_2 = 0.1\mu F$
$C_3 = 40\mu F$

○ A/D Converter Conversion Characteristics ①



[HEX]

3FF

Conversion
code

000

$E_Z$
MIX

$E_Z$
MAX

$E_F$
MAX

$E_F$
MIX

Ideal conversion
(center line)

------ Actual
conversion width

Actual Conversion
(center line)

VREF    [V]

Analog input

Conversion Characteristics Diagram ①

14
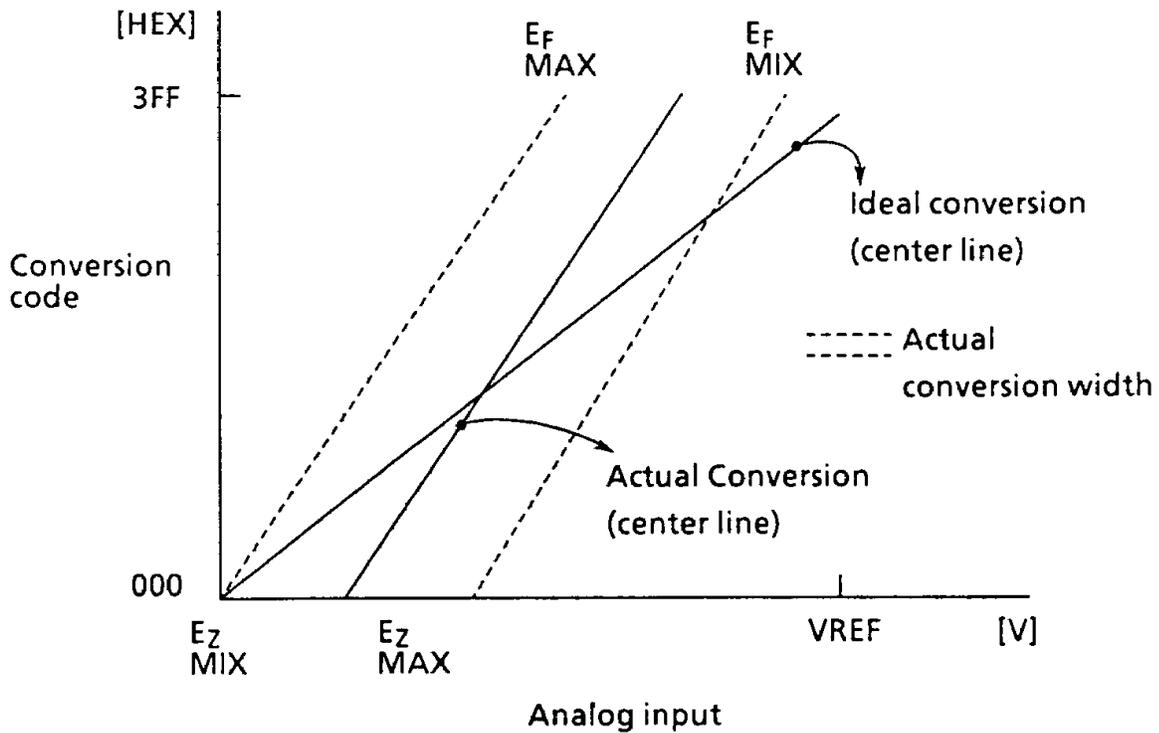
Absolute error (EA)

The absolute error indicates a difference between actual conversion and ideal conversion, excluding a quantizing error.
The absolute error of the A/D converter gets larger as it approaches the zero point or full scale. (Refer to Conversion Characteristics Diagram ①.)

Relative error (ER)

The relative error indicates a deviation from a line which connects the center point of the zero point conversion width with that of the full scale conversion width, excluding a quantizing error.

The relative error of this A/D converter is almost due to a differential linearity error.

Zero point error (Ez) and full scale error (EF)

The zero point error and full scale error indicate a difference between actual conversion and ideal conversion at the zero point and full scale, respectively. (Refer to Conversion Characteristics Diagram ①.)

○    A/D converter conversion characteristics ② (temperature characteristics)
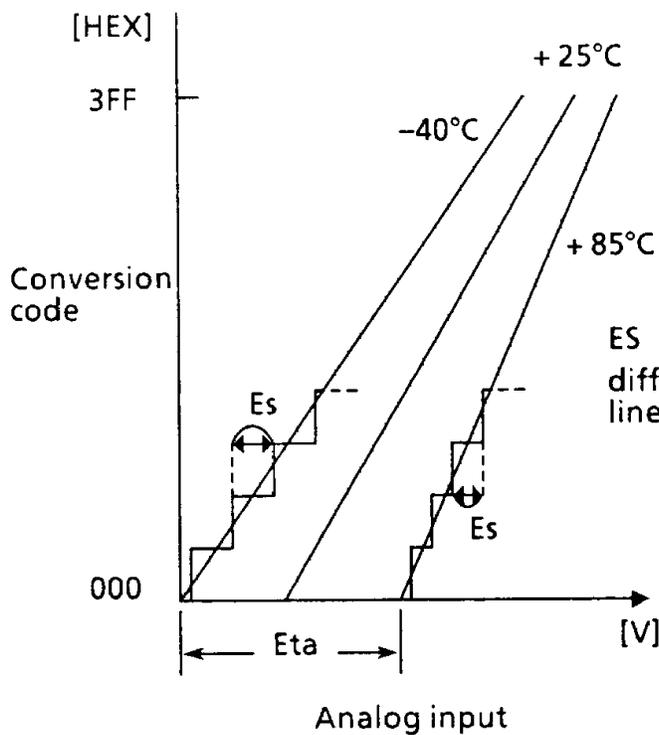


Conversion Characteristics
Diagram ②-1

Conversion Characteristics
Diagram ②-2

15
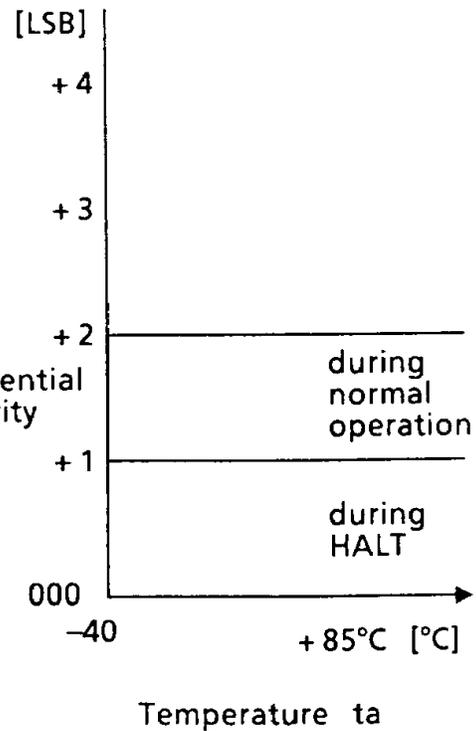
## Differential linearity error

The differential linearity error indicates a difference between the actual conversion width (actual step width) and ideal value (1LSB).

With this A/D converter, a voltage for actual conversion is shifted and the inclination of a voltage is changed, with changes of temperature (See Conversion Characteristics Diagram ②-1.) Specifications described in the foregoing tables are established from Eta shown in Conversion Characteristics Diagram ②-1. Conversion Characteristics Diagram ②-2 shows temperature characteristics of differential linearity of ES in Conversion Characteristics Diagram ②-1.

# REGISTERS

- **ACCUMULATOR**

```
15                    0
┌─────────────────────┐
│         ACC         │
└─────────────────────┘
```
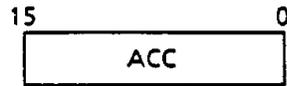
- **CONTROL REGISTERS (CR)**

  PROGRAM STATUS WORD

  ```
  ┌─────────────┐
  │     PSW     │
  └─────────────┘
  ```

  PROGRAM COUNTER

  ```
  ┌─────────────┐
  │     PC      │
  └─────────────┘
  ```

  LOCAL REGISTER BASE

  ```
  ┌─────────────┐
  │     LRB     │
  └─────────────┘
  ```

  SYSTEM STACK POINTER

  ```
  ┌─────────────┐
  │     SSP     │
  └─────────────┘
  ```
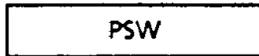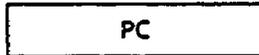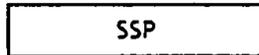
- **POINTING REGISTERS (PR)**

  INDEX REGISTER1

  INDEX REGISTER2

  DATA POINTER

  USER STACK POINTER

  ```
  ┌─────────────┐
  │     X1      │
  ├─────────────┤
  │     X2      │
  ├─────────────┤
  │     DP      │
  ├─────────────┤
  │     USP     │
  └─────────────┘
  ```

- **LOCAL REGISTERS**

  | | | |
  |-----|-----|-----|
  | er0 | r1  | r0  |
  | er1 | r3  | r2  |
  | er2 | r5  | r4  |
  | er3 | r7  | r6  |
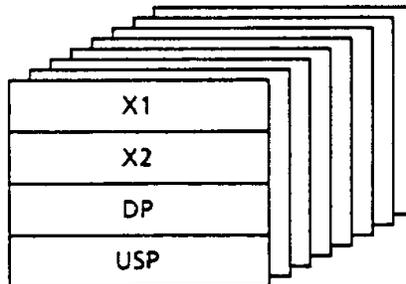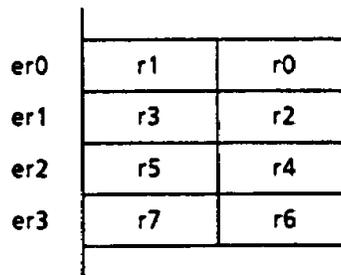
- **SPECIAL FUNCTION REGISTERS (SFR)**

  All of the I/O functions are controlled by SFRs. Also, some of the internal functions (Timer, WDT, etc, ...) are controlled by SFRs. SFRs are located in the top of RAM space (000H~007FH).

# MSM66207 SPECIAL FUNCTION REGISTERS

| Address (HEX) | SFR Name | Abbreviated Name | R/W | 8/16-bit operation | Reset Value (HEX) |
|---|---|---|---|---|---|
| 0000 | System Stack Printer | SSP (ASSP) | | | FFH |
| 0001 | | | | | FFH |
| 0002 | Local Register Base | LRB (ALRB) | | | undefined |
| 0003 | | | | 8/16 | |
| 0004 | Program Status Word | PSWL (APSW) | R/W | | C8H |
| 0005 | | PSWH | | | 0CH |
| 0006 | Accumulator | ACC | | | 00H |
| 0007 | | | | | 00H |
| 0010 | Standby control register | SBYCON | | | F8H or F0H |
| 0011 | Watchdog timer | WDT | W | 8 | 00H/WDT is stopped |
| 0012 | Peripheral control register | PRPHF | R/W | | FDH |
| 0013 | Stop code acceptor | STPACP | W | 8 | "0" |
| 0018 | Interrupt request flag | IRQ | | | 00H |
| 0019 | | | | 8/16 | 00H |
| 001A | Interrupt Enable flag | IE | | | 00H |
| 001B | | | | | 00H |
| 001C | External interrupt control register | EXICON | | | FCH |
| 0020 | Port 0 data register | P0 | | | undefined |
| 0021 | Port 0 mode register | P0IO | | | 00H |
| 0022 | Port 1 data register | P1 | | | undefined |
| 0023 | Port 1 mode register | P1IO | R/W | | 00H |
| 0024 | Port 2 data register | P2 | | | undefined |
| 0025 | Port 2 mode register | P2IO | | | 00H |
| 0026 | Port 2 secondary function control register | P2SF | | 8 | 07H |
| 0028 | Port 3 data register | P3 | | | undefined |
| 0029 | Port 3 mode register | P3IO | | | 00H |
| 002A | Port 3 secondary function control register | P3SF | | | 00H |
| 002C | Port 4 data register | P4 | | | undefined |
| 002D | Port 4 mode register | P4IO | | | 00H |
| 002E | Port 4 secondary function control register | P4SF | | | 00H |

18

| Address (HEX) | SFR Name | Abbreviated Name | R/W | 8/16-bit operation | Reset Value (HEX) |
|---|---|---|---|---|---|
| 002F | Port 5 | P5 | R | 8 | – |
| 0030 | Timer 0 counter | TM0 | | | 00H |
| 0031 | | | | | 00H |
| 0032 | Timer 0 register | TMR0 | | | 00H |
| 0033 | | | | | 00H |
| 0034 | Timer 1 counter | TM1 | | | 00H |
| 0035 | | | | | 00H |
| 0036 | Timer 1 register | TMR1 | R/W | 16 | 00H |
| 0037 | | | | | 00H |
| 0038 | Timer 2 counter | TM2 | | | 00H |
| 0039 | | | | | 00H |
| 003A | Timer 2 register | TMR2 | | | 00H |
| 003B | | | | | 00H |
| 003C | Timer 3 counter | TM3 | | | 00H |
| 003D | | | | | 00H |
| 003E | Timer 3 register | TMR3 | | | 00H |
| 003F | | | | | 00H |
| 0040 | Timer 0 Control register | TCON0 | | | 00H |
| 0041 | Timer 1 Control register | TCON1 | | | 00H |
| 0042 | Timer 2 Control register | TCON2 | | | 00H |
| 0043 | Timer 3 Control register | TCON3 | | | 00H |
| 0046 | Transition detector register | TRNSIT | | | undefined |
| 0048 | Serial port transmission baud rate generator counter | STTM | | | 00H |
| 0049 | Serial port transmission baud rate generator register | STTMR | R/W | | 00H |
| 004A | Serial port transmission control register | STTMC | | | 0CH |
| 004C | Serial port receiving baud rate generator counter | SRTM | | | 00H |
| 004D | Serial port receiving baud rate generator register | SRTMR | | 8 | 00H |
| 004E | Serial port receiving control register | SRTMC | | | 0EH |
| 0050 | Serial port transmission mode control register | STCON | | | 80H |
| 0051 | Serial port transmission data buffer register | STBUF | W | | undefined |
| 0054 | Serial port receiving mode control register | SRCON | R/W | | 00H |
| 0055 | Serial port receiving data buffer register | SRBUF | R | | undefined |
| 0056 | Serial port receiving error register | SRSTAT | | | F0H |
| 0058 | A/D scan mode register | ADSCAN | R/W | | 80H |
| 0059 | A/D select mode register | ADSEL | | | A0H |

| Address (HEX) | SFR Name | Abbreviated Name | R/W | 8/16-bit operation | Reset Value (HEX) |
|---|---|---|---|---|---|
| 0060 | A/D conversion result register 0 | ADCR0 | | | |
| 0061 | | | | | |
| 0062 | A/D conversion result register 1 | ADCR1 | | | |
| 0063 | | | | | |
| 0064 | A/D conversion result register 2 | ADCR2 | R | | undefined |
| 0065 | | | | | |
| 0066 | A/D conversion result register 3 | ADCR3 | | | |
| 0067 | | | | | |
| 0068 | A/D conversion result register 4 | ADCR4 | | 8/16 | |
| 0069 | | | | | |
| 006A | A/D conversion result register 5 | ADCR5 | | | |
| 006B | | | | | |
| 006C | A/D conversion result register 6 | ADCR6 | | | |
| 006D | | | | | |
| 006E | A/D conversion result register 7 | ADCR7 | | | |
| 006F | | | | | |
| 0070 | PWM 0 counter | PWMC0 | | | 00H |
| 0071 | | | | | 00H |
| 0072 | PWM 0 register | PWMR0 | | | 00H |
| 0073 | | | | | 00H |
| 0074 | PWM 1 counter | PWMC1 | R/W | | 00H |
| 0075 | | | | | 00H |
| 0076 | PWM 1 register | PWMR1 | | | 00H |
| 0077 | | | | | 00H |
| 0078 | PWM 0 control register | PWCON0 | | 8 | 00H |
| 007A | PWM 1 control register | PWCON1 | | | 00H |

# MEMORY MAP

- Program Memory Space

| | | | |
|---|---|---|---|
| 0000H | Internal ROM | 0000H | Vector Table Area (40 byte) |
| | | 0027H | |
| | | 0028H | VCAL Table Area (16 byte) |
| 7FFFH | | 0037H | |
| 8000H | External ROM | 0038H | |
| 0FFFFH | | 7FFFH | |

- Data Memory Space

| | | | |
|---|---|---|---|
| 0000H | SFRs | 0000H | Special Function Registers |
| 007FH | | | PORT.A/DC TIMER. PWM etc..... |
| 0080H | PRs | | |
| 00BFH | | | |
| 00C0H | | 007FH | |
| 00FFH | | 0080H | PR0 |
| 0100H | | | PR1 |
| | | | PR2 |
| 047FH | | | PR3 |
| 0480H | | | PR4 |
| | | | PR5 |
| | | | PR6 |
| 00BFH | | 00BFH | PR7 |
| 00C0H | | 00C0H | |
| 0FFFFH | | 047FH | |

Zero Page: 0080H – 00FFH

Internal RAM: 0000H – 047FH

X 1
X 2
D P
USP

# ADDRESSING MODE

The MSM66207 supports 64KB of data space and 64KB of program space with verious types of addressing modes. These modes divide into the following types.

## 1. RAM ADDRESSING (FOR DATA SPACE)

### 1.1 Register Direct Addressing

```
ex.
      ROR      DP
               ==
               :·············           ////// DP //////
```
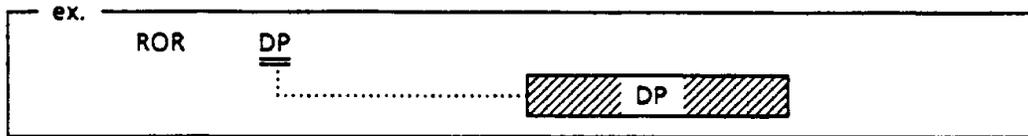
### 1.2 Displacement Addressing
a) Zero Page

```
ex.
      L      A, 18H                                    000H
             ==                          SFR
             :
             :·········          /////////////////     0018H
```

b) Direct Page

```
ex.
      ST     A, off   XX10H                             xx00H
                      =====                RAM
                        :
                        :········          /////////////////   xx10H
```
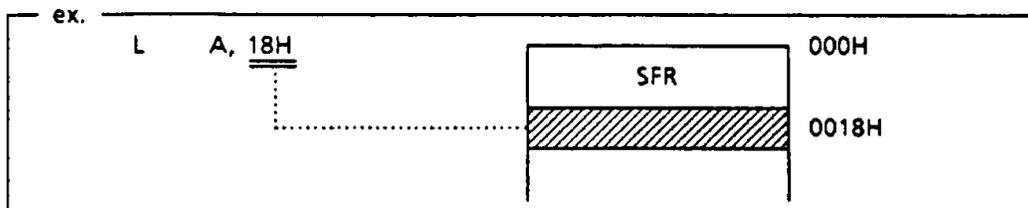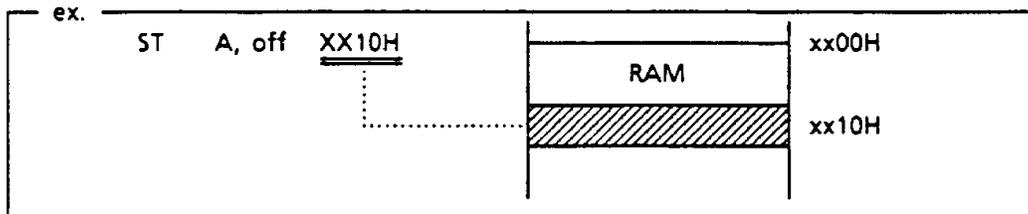
### 1.3 Pointing Register Indirect Addressing
a) Data Pointer (DP) Indirect

```
ex.
      SLL        [DP]
                 ====
                  :                                  RAM
                  :
          ┌──────────────┐
          │     DP       │────────────►  /////////////////
          └──────────────┘
```

b) User Stack Pointer (USP) Indirect

```
ex.
      SRL     10H  [USP]
              ==   =====
               │     │
               │   ┌──────────────┐
               │   │     USP      │              RAM
               │   └──────────────┘
               │          │
               │          ▼
               └────────►(+)────────►  /////////////////
          - 128~ + 127
```

c) Index register (X1, X2) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────────┐
│                                                                 │
│          INC      300H [X1]                                     │
│                   ═══════╤════                                  │
│                   ┌──────┴────────────┐      ┌───────────────┐  │
│                   │        X1         │      │     RAM       │  │
│                   └───────────────────┘      │               │  │
│                 ┌───────────┐    │           ├───────────────┤  │
│                 │ 0-65535   │    ↓           │///////////////│  │
│                 └─────────→ ⊕ ─────────────→ ├───────────────┤  │
│                                              │               │  │
│                                              └───────────────┘  │
└─────────────────────────────────────────────────────────────────┘
```
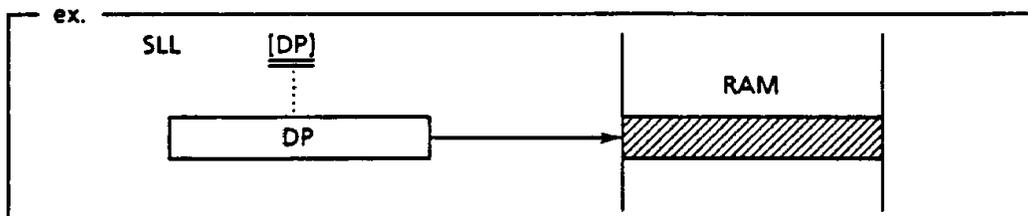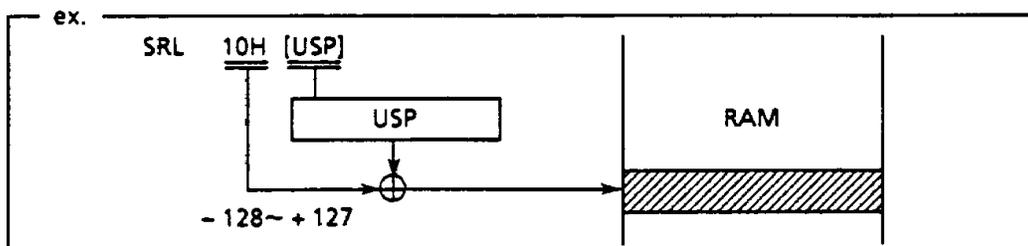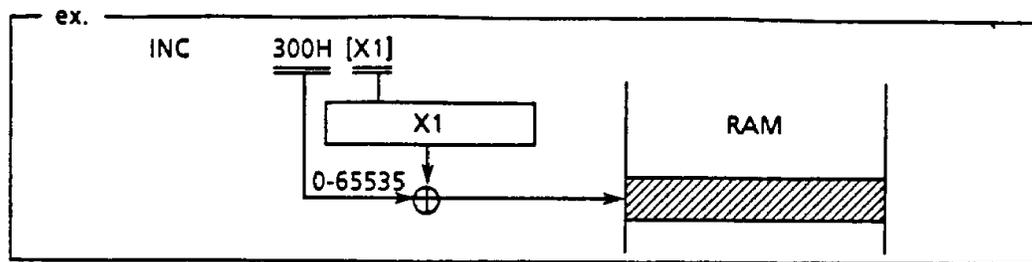
### 1.4    Immediate Addressing

```
┌─ ex. ──────────────────────────────────────────────────────────┐
│                                                                 │
│          MOV      SSP, #    ┌──────────────┐                    │
│                             │     27FH     │                    │
│                             └──────────────┘                    │
│                               ════════════                      │
└─────────────────────────────────────────────────────────────────┘
```

## 2.   ROM ADDRESSING (FOR PROGRAM SPACE)

### 2.1    Direct Addressing

```
┌─ ex. ──────────────────────────────────────────────────────────┐
│                                                                 │
│          LC       A, 2000H                    ┌───────────────┐ │
│                      ╌╌╌╌╌                     │     ROM       │ │
│                          ┊                    ├───────────────┤ │
│                          └╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌╌→ │///////////////│ │
│                                               ├───────────────┤ │
│                                               │               │ │
│                                               └───────────────┘ │
└─────────────────────────────────────────────────────────────────┘
```
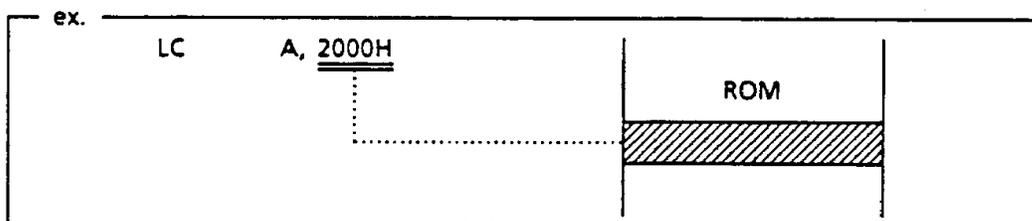
### 2.2    Simple Indirect Addressing
a) Local Register Indirect
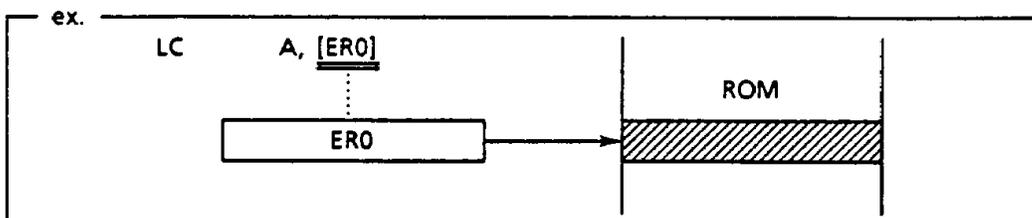
```
┌─ ex. ──────────────────────────────────────────────────────────┐
│                                                                 │
│          LC       A, [ER0]                    ┌───────────────┐ │
│                      ══════                    │     ROM       │ │
│                        ┊                      ├───────────────┤ │
│               ┌────────┴──────────┐           │///////////////│ │
│               │       ER0         │─────────→ ├───────────────┤ │
│               └───────────────────┘           │               │ │
│                                               └───────────────┘ │
└─────────────────────────────────────────────────────────────────┘
```

23

b) Pointing Register Indirect

①  Data Pointer (DP) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────┐
│        LC          A, [DP]                                   │
│                       ┆                          ROM         │
│             ┌──────────────────┐       ┌─────────────────┐   │
│             │       DP         │──────▶│/////////////////│   │
│             └──────────────────┘       └─────────────────┘   │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```

②  User Stack Pointer (USP) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────┐
│        LC          A, [USP]                                  │
│                       ┆                          ROM         │
│             ┌──────────────────┐       ┌─────────────────┐   │
│             │       USP        │──────▶│/////////////////│   │
│             └──────────────────┘       └─────────────────┘   │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```

③  Index Register (X1, X2) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────┐
│        LC          A, [X1]                                   │
│                       ┆                          ROM         │
│             ┌──────────────────┐       ┌─────────────────┐   │
│             │       X1         │──────▶│/////////////////│   │
│             └──────────────────┘       └─────────────────┘   │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```
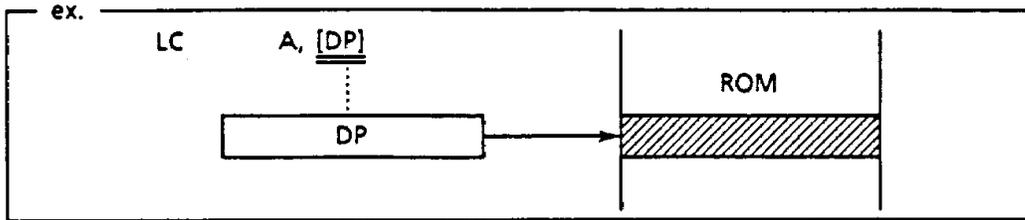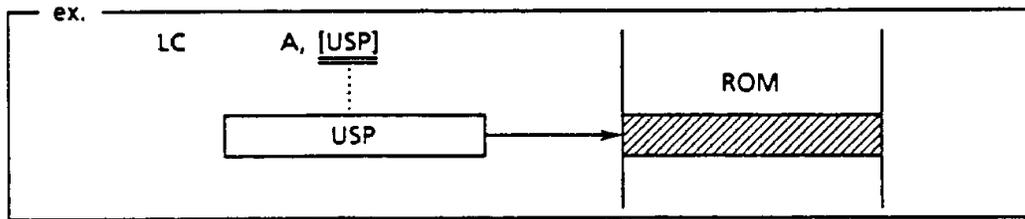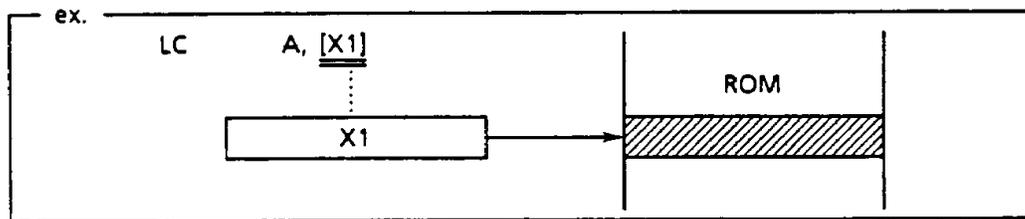
c) System Stack Pointer (SSP) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────┐
│        LC          A, [SSP]                                  │
│                       ┆                          ROM         │
│             ┌──────────────────┐       ┌─────────────────┐   │
│             │       SSP        │──────▶│/////////////////│   │
│             └──────────────────┘       └─────────────────┘   │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```
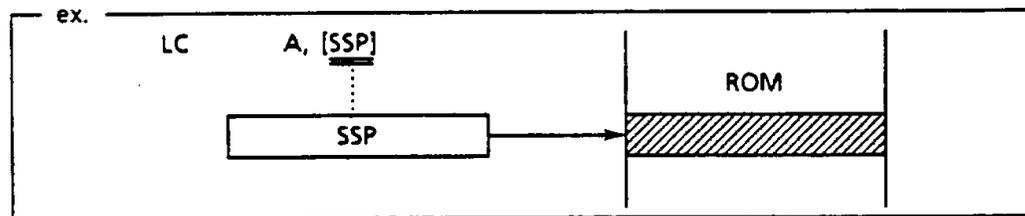
d) Local Register Base (LRB) Indirect

```
┌─ ex. ──────────────────────────────────────────────────────┐
│        LC          A, [LRB]                                  │
│                       ┆                          ROM         │
│             ┌──────────────────┐       ┌─────────────────┐   │
│             │       LRB        │──────▶│/////////////////│   │
│             └──────────────────┘       └─────────────────┘   │
│                                                              │
└──────────────────────────────────────────────────────────────┘
```

**e) RAM Indirect**

```
 ┌─ ex. ──────────────────────────────────────────────────────┐
 │        LC          A, [0C0H]                                │
 │                      ═══                  ┌──────┐          │
 │                                           │ RAM  │   ┌──────┐
 │                       ┆                   ├──────┤0C0H│//////│
 │                       └┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈┈▶├──────┤───▶│//////│
 │                                           │      │   │ ROM  │
 └────────────────────────────────────────────────────────────┘
```
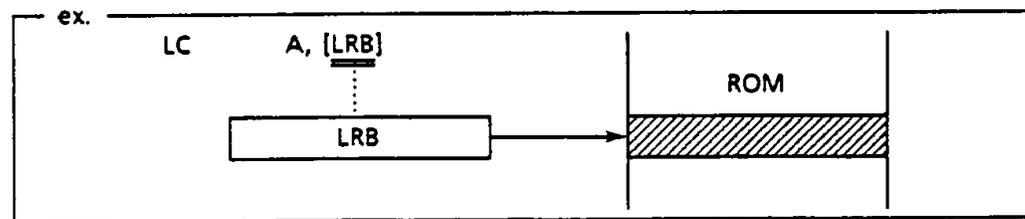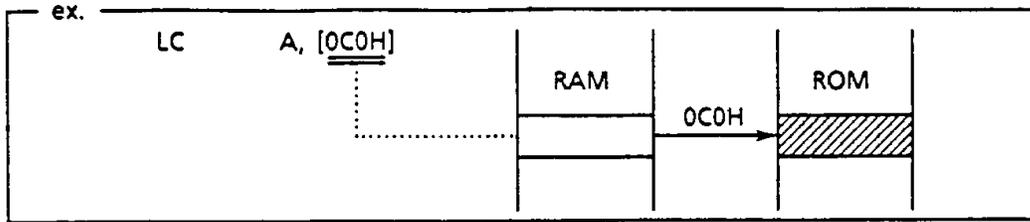
## 2.3   Double Indirect Addressing

**a)   Data Pointer (DP) Double Indirect**

```
 ┌─ ex. ──────────────────────────────────────────────────────┐
 │        LC          A, [[DP]]                                │
 │                        ═══                 ┌──────┐         │
 │                         ┆          RAM     │      │   ROM   │
 │                  ┌──────────┐     ┌──────┐ │      │ ┌──────┐│
 │                  │    DP    │────▶│      │─┼──────┼▶│//////││
 │                  └──────────┘     └──────┘ │      │ └──────┘│
 └────────────────────────────────────────────────────────────┘
```
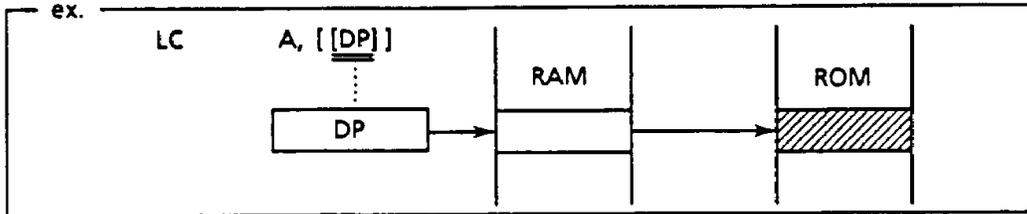
**b)   User Stack Pointer (USP) Double Indirect**

```
 ┌─ ex. ──────────────────────────────────────────────────────┐
 │        LC          A, [−2 [USP]]                            │
 │                       ══  ═════                             │
 │                    ┌──────────┐   RAM          ROM          │
 │                    │   USP    │  ┌──────┐     ┌──────┐      │
 │                    └──────────┘  │      │     │//////│      │
 │                         │        ├──────┤────▶│//////│      │
 │                        ⊕────────▶│      │     │      │      │
 │            −128~ +127            └──────┘                   │
 └────────────────────────────────────────────────────────────┘
```
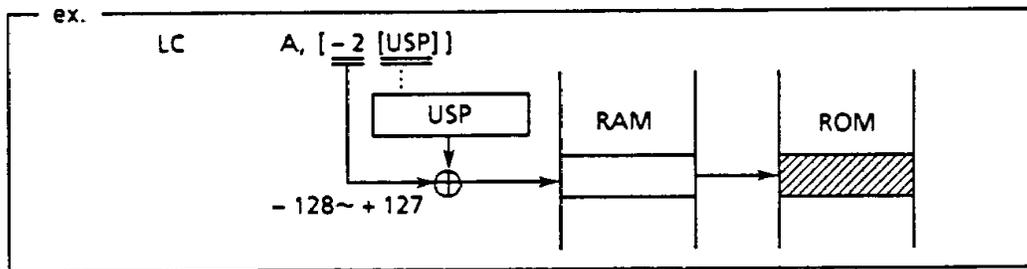
**c)   Index Register (X1, X2) Double Indirect**

```
 ┌─ ex. ──────────────────────────────────────────────────────┐
 │        LC          A, [1000H [X1]]                          │
 │                       ═════  ════                           │
 │                    ┌──────────┐   RAM          ROM          │
 │                    │    X1    │  ┌──────┐     ┌──────┐      │
 │                    └──────────┘  │      │     │//////│      │
 │                         │        ├──────┤────▶│//////│      │
 │                        ⊕────────▶│      │     │      │      │
 │            0~65535               └──────┘                   │
 └────────────────────────────────────────────────────────────┘
```
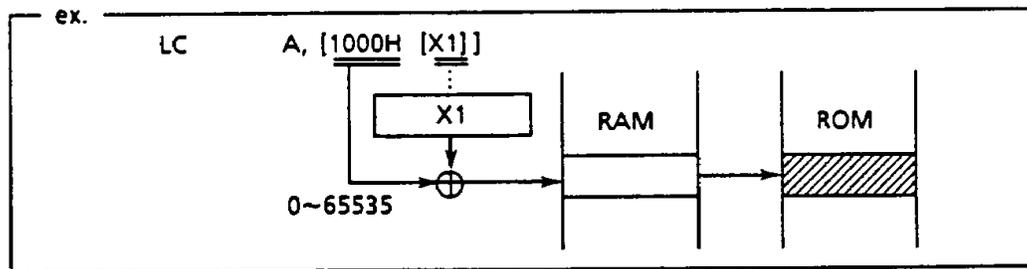
## 2.4 Indirect Addressing with 16 bit offset

### a) Pointing Register Indirect
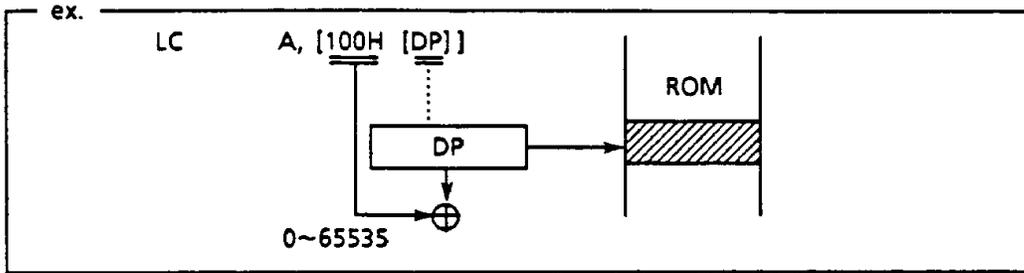
#### ① Data Pointer (DP) Indirect

ex.

LC    A, [100H [DP]]

ROM

DP

0~65535

#### ② User Stack Pointer (USP) Indirect

ex.

LC    A, [100H [USP]]

ROM

USP

0~65535

#### ③ Index Register (X1, X2) Indirect

ex.

LC    A, [100H [X1]]

ROM

X1

0~65535

### b) RAM Indirect

ex.

LC    A, [2000H [80H]]

RAM

80H

ROM

0~65535

26
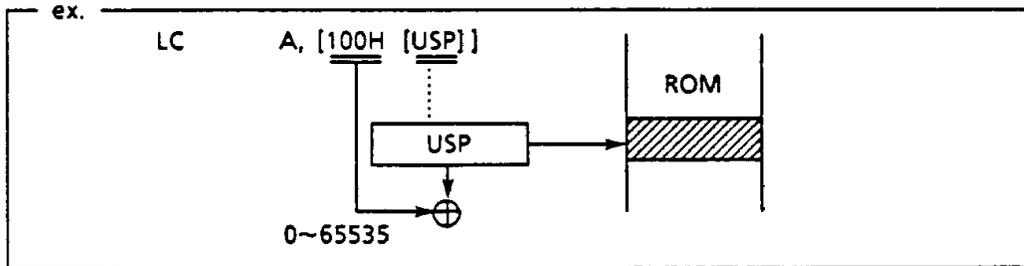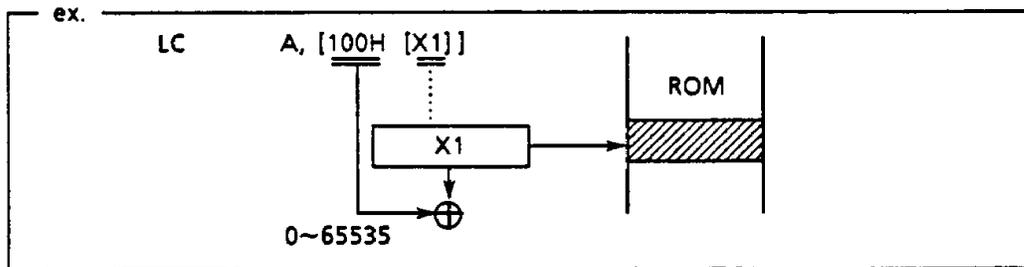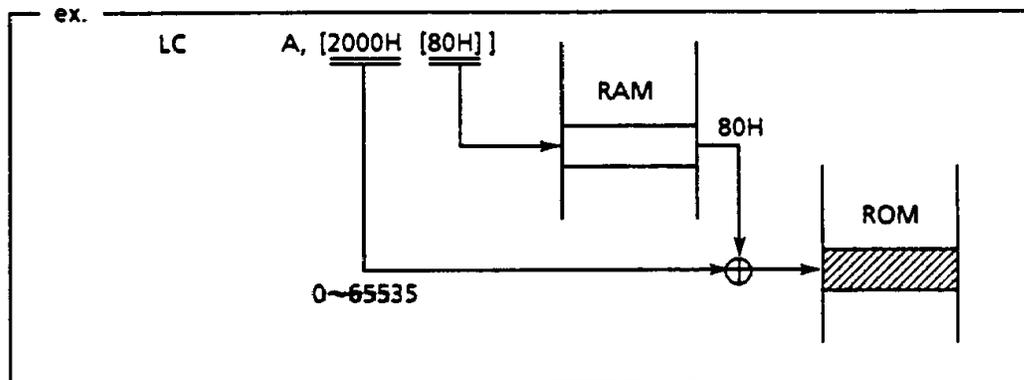
# INSTRUCTIONS

(Note) '*' Addressing mode expression
'#' and 'imme' Immediate value

## Data Transfer

| Instruction | Function |
|---|---|
| L    A, * | A ← * |
| ST    A, * | A → * |
| MOV    *, A | A ← * |
| MOV    *, # | * → Imme. |
| MOV    er.* | er ← * |
| MOV    d, * | (d) ← * |
| MOV    DP, * | DP ← * |
| MOV    X1, * | X1 ← * |
| MOV    X2, * | X2 ← * |
| MOV    USP, * | USP ← * |
| MOV    PSW, * | PSW ← * |
| MOV    SSP, * | SSP ← * |
| MOV    LRB, * | LRB ← * |
| CLR    * | * ← 0 |
| SWAP | $A_{15-8} \leftrightarrow A_{7-0}$ |
| XCHG   A, * | A ↔ * |
| XNBL   A, * | $A_{3-8} \leftrightarrow *_{3-0}$ |

| Instruction | Function |
|---|---|
| LB    A, * | $A_L$ ← * |
| STB    A, * | $A_L$ → * |
| MOVB  A, * | Al ← * |
| MOVB  *, A | * ← AL |
| MOVB  *, # | * ← Imme. |
| MOVB  r, * | r ← * |
| MOVB  d, * | (d) ← * |
| MOVB  PSW0, * | PSW0 ← * |
| MOVB  SCB, * | SCB ← * |
| | |
| | |
| | |
| | |
| CLRB    * | * ← 0 |
| SWAPB | $A_{15-8} \leftrightarrow A_{7-0}$ |
| XCHGB A, * | A ↔ * |

## Push & Pop

| Instruction | Function |
|---|---|
| PUSHS    *, A | * → SYSTEM STACK |
| POPS    *, # | * ← SYSTEM STACK |

## Rotate & Shift

| Instruction | Function |
|---|---|
| ROL    * | Rotate left |
| ROR    * | Rotate right |
| SLL    * | Shift left Logical |
| SRL    * | Shift right Logical |
| SRA    * | Shift right Arithmetic |

| Instruction | Function |
|---|---|
| ROLB    * | Rotate left |
| RORB    * | Rotate right |
| SLLB    * | Shift left Logical |
| SRLB    * | Shift right Logical |
| SRAB    * | Shift right Arithmetic |

## Increment & Decrement

| Instruction | Function |
|---|---|
| INC    * | * ← * + 1 |
| DEC    * | * ← * − 1 |

| Instruction | Function |
|---|---|
| INCB    * | * ← * + 1 |
| DECB    * | * ← * − 1 |

## ROM Table Reference

| Instruction | Function |
|---|---|
| LC    A, * | A ← * (ROM) |
| CMPC    A, * | * − * (ROM) |

| Instruction | Function |
|---|---|
| LCB    A, * | $A_L$ ← * (ROM) |
| CMPCB    A, * | $A_L$ − * (ROM) |

# Arithmetic Operation

| Instruction | | Function |
|---|---|---|
| MUL | | er1: $A \leftarrow A \times er0$ |
| DIV | | er0: $A \leftarrow er0$: A/er2 |
| ADD | A, * | $A \leftarrow A + *$ |
| ADD | *, A | $* \leftarrow * + A$ |
| ADD | *, d | $* \leftarrow * + (d)$ |
| ADD | *, # | $* \leftarrow * + Imme.$ |
| ADC | A, * | $A \leftarrow A + * + C$ |
| ADC | *, A | $* \leftarrow * + A + C$ |
| ADC | *, d | $* \leftarrow * + (d) + C$ |
| ADC | *, # | $* \leftarrow * + Imme. + C$ |
| SUB | A, * | $A \leftarrow A - *$ |
| SUB | *, A | $* \leftarrow * - A$ |
| SUB | *, d | $* \leftarrow * - (d)$ |
| SUB | *, # | $* \leftarrow * - Imme.$ |
| SBC | A, * | $A \leftarrow A - * - C$ |
| SBC | *, A | $* \leftarrow * - A - C$ |
| SBC | *, d | $* \leftarrow * - (d) - C$ |
| SBC | *, # | $* \leftarrow * - Imme. - C$ |

| Instruction | | Function |
|---|---|---|
| MULB | | $A \leftarrow A_L \times r0$ |
| DIVB | | $A \leftarrow A/r0$ |
| ADDB | A, * | $A_L \leftarrow A_L + *$ |
| ADDB | *, A | $* \leftarrow * + A_L$ |
| ADDB | *, d | $* \leftarrow * + (d)$ |
| ADDB | *, # | $* \leftarrow * + Imme.$ |
| ADCB | A, * | $A_L \leftarrow A_L + * + C$ |
| ADCB | *, A | $* \leftarrow * + A_L + C$ |
| ADCB | *, d | $* \leftarrow * + (d) + C$ |
| ADCB | *, # | $* \leftarrow * + Imme. + C$ |
| SUBB | A, * | $A_L \leftarrow A_L - *$ |
| SUBB | *, A | $* \leftarrow * - A_L$ |
| SUBB | *, d | $* \leftarrow * - (d)$ |
| SUBB | *, # | $* \leftarrow * - Imme.$ |
| SBCB | A, * | $A_L \leftarrow A_L - * - C$ |
| SBCB | *, A | $* \leftarrow * - A_L - C$ |
| SBCB | *, d | $* \leftarrow * - (d) - C$ |
| SBCB | *, # | $* \leftarrow * - Imme. - C$ |

## Logical Operation

| Instruction | | Function |
|---|---|---|
| AND | A, * | A ← A and * |
| AND | *, A | * ← * and A |
| AND | *, d | * ← * and (d) |
| AND | *, # | * ← * and Imme. |
| OR | A, * | A ← A or * |
| OR | *, A | * ← * or A |
| OR | *, d | * ← * or (d) |
| OR | *, # | * ← * or Imme. |
| XOR | A, * | A ← A xor * |
| XOR | *, A | * ← xor A |
| XOR | *, d | * ← xor (d) |
| XOR | *, # | * ← * xor Imme. |

| Instruction | | Function |
|---|---|---|
| ANDB | A, * | $A_L$ ← $A_L$ and * |
| ANDB | *, A | * ← * and $A_L$ |
| ANDB | *, d | * ← * and (d) |
| ANDB | *, # | * ← * and Imme. |
| ORB | A, * | $A_L$ ← $A_L$ or * |
| ORB | *, A | * ← * or $A_L$ |
| ORB | *, d | * ← * or (d) |
| ORB | *, # | * ← * or Imme. |
| XORB | A, * | $A_L$ ← $A_L$ xor * |
| XORB | *, A | * ← * xor $A_L$ |
| XORB | *, d | * ← * xor (d) |
| XORB | *, # | * ← * xor Imme. |

## Comparison

| Instruction | | Function |
|---|---|---|
| CMP | A, * | A – * |
| CMP | *, A | * – A |
| CMP | *, d | * – (d) |
| CMP | *, # | * – Imme. |

| Instruction | | Function |
|---|---|---|
| CMPB | A, * | $A_L$ – * |
| CMPB | *, A | * – $A_L$ |
| CMPB | *, d | * – (d) |
| CMPB | *, # | * – Imme. |

## Decimal Adjust

| Instruction | Function |
|---|---|
| DAA | Decimal Adjust for Addition |
| DAS | Decimal Adjust for Substruct |

## Data Type Conversion

| Instruction | Function |
|---|---|
| EXTND | A15 – 8 ← An |

## Bit Operation

| Instruction | | Function |
|---|---|---|
| SBR | * | bit ← 1 |
| RBR | * | bit ← 0 |
| MBR | C, * | C ← bit |
| MBR | *, C | bit ← C |
| TRB | *, C | Z ← bit |

| Instruction | | Function |
|---|---|---|
| SBR | * | bit ← 1 |
| RBR | * | bit ← 0 |
| MBR | C, * | C ← bit |
| MBR | *, C | bit ← C |
| | | |

## Jump & Call

| Instruction | | Function |
|---|---|---|
| SJ | adrs | Short Jump |
| J | * | Jump |
| JC | EQ, adrs | Jump if ' = ' |
| JC | LE, adrs | Jump if ' < = ' |
| JC | GE, adrs | Jump if ' > = ' |
| JBS | bit, adrs | Jump if bit-on |
| JRNZ | DP, adrs | Loop Function |

| Instruction | | Function |
|---|---|---|
| SCAL | adrs | Short Call |
| CAL | * | Call Subroutine |
| JC | NE, adrs | Jump if ' < > ' |
| JC | LT, adrs | Jump if ' < ' |
| JC | GT, adrs | Jump if ' > ' |
| JBR | bit, adrs | Jump if bit-off |
| VCAL | table:adrs | Vector Call |

## Return

| Instruction | Function |
|---|---|
| RT | Return from Subroutine |
| RTI | Return from Interrupt |

## Others

| Instruction | Function |
|---|---|
| SC | $C \leftarrow 1$ |
| SS | STACK FLAG $\leftarrow 1$ |
| NOP | No Operation |

| Instruction | Function |
|---|---|
| RC | $C \leftarrow 0$ |
| BRK | Software Reset |
| | |

# Oki Regional Sales Offices

**Northwest Area**
360 Kiely Blvd.
Suite 115
San Jose, CA 95129
Tel: 408-244-9666
Fax: 408-244-8423

**Southwest Area**
2302 Martin St.
Suite 250
Irvine, CA 92715
Tel: 714-752-1843
Fax: 714-752-2423

**Automotive Electronics**
17177 N. Laurel Park Dr.
Livonia, MI 48152
Tel: 313-464-7200
Fax: 313-464-1724

**Central Area**
2007 N. Collins Blvd.
Suite 303
Richardson, TX 75080
Tel: 214-690-6868
Fax: 214-690-8233

**Southeast Area**
7301 W. Palmetto Park Rd.
Suite 210B
Boca Raton, FL 33433
Tel: 407-394-6099
Fax: 407-395-2716

**Eastern Area**
91 Montvale Ave.
Stoneham, MA 02180
Tel: 617-279-0293
Fax: 617-438-3608

205 South Avenue
Suite 100
Poughkeepsie, NY 12601
Tel: 914-473-8044
Fax: 914-473-7069

**Puerto Rico**
Banco Popular Center
Suite 1819
209 Munoz Rivera Avenue
Hato Rey, San Juan, PR 00918
Tel: 809-766-2422
Fax: 809-763-6135

**OKI**
Semiconductor

785 North Mary Avenue
Sunnyvale, California 94086-2909
Tel: 408-720-1900
Fax: 408-720-1918