PRELIMINARY DATA SHEET

NEC

MOS INTEGRATED CIRCUIT

μPD16434

1/8, 1/16 DUTY LCD CONTROLLER/DRIVER

DESCRIPTION

μPD16434 is LCD controler/driver containing the interfaing feathres for a dot-matrix mode 8-, 16-times division LCD and a microprocessor. uPD16434 contains a 5 by 7 dot matrix character generator corresponding to ASCII/JIS. Therefore, user origenal patterns can be easily displayed.

FEATURES

- Dot matrix LCD controller/driver
- 8 or 16-times division drive possible wiht a single chip
  8-time division - 400(50 by 8) dots
  16-time division - 672(42 by 16) dots
- 8 or 16-times division drive possible wiht n0 chip
  8-time division - n×400(50 by 8) dots
  16-time division - n×800(50 by 16) dots
- Display data storage RAM - 2×50×8 bits
- Programmer specified dot(graphic) display
- Capable of alphanumeric and symbolic displays through built-in ROM (5 by 7 dots)
  160 characters
- Parallel data input/output (Switchable between 4 and 8 bits)
- Cursor manipulation command
- Upgraded version of μPD7228, μPD7228A, μPD7229, μPD7229A

ORDERING INFORMATION

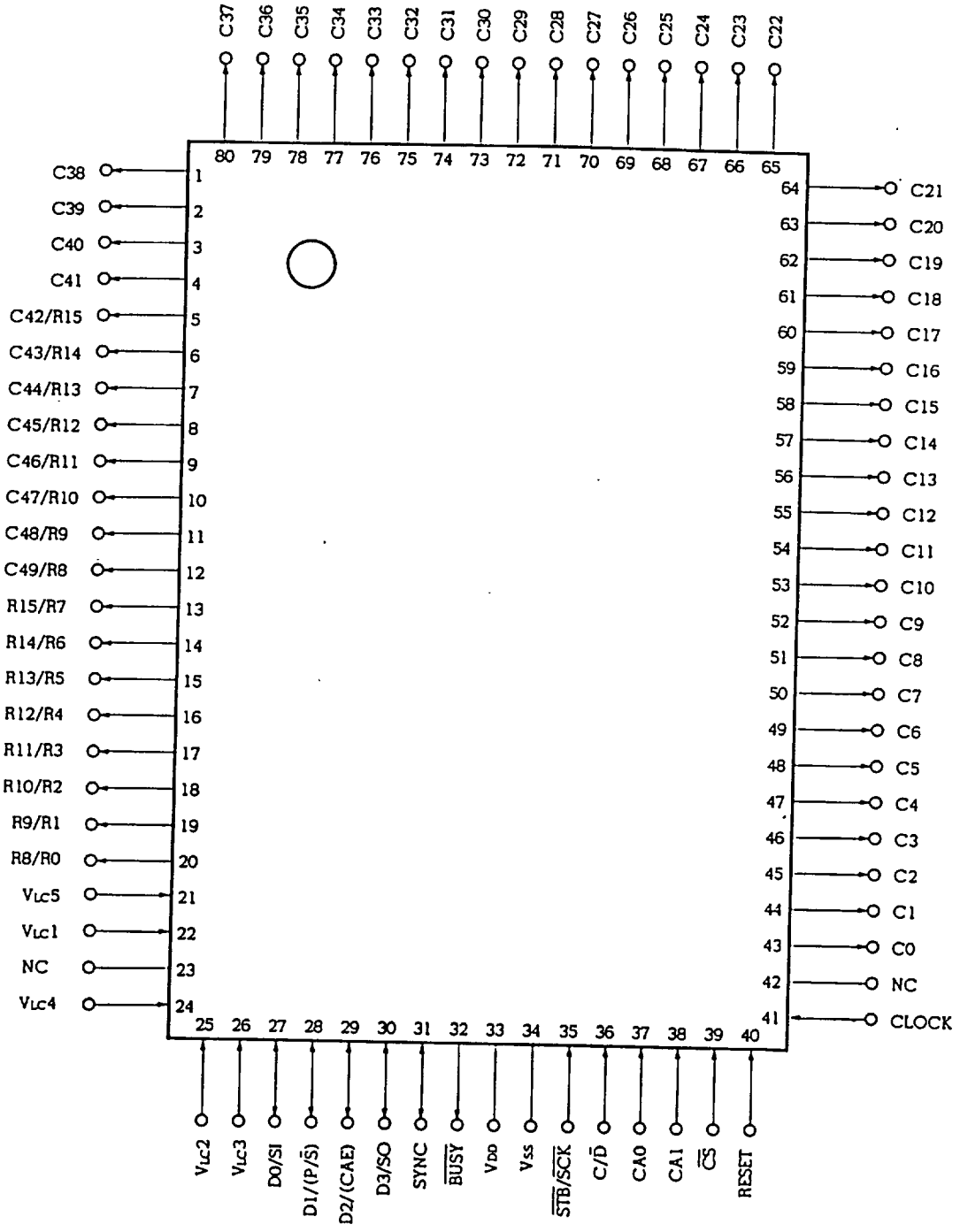
Part Number	Package
μPD16434-001-G12	Chip/wafer, Standard ROM code

The information contained in this document is being issued in advance of the production cycle for the device. The parameters for the device may change before final production or NEC Corporation, at its own discretion, may withdraw the device prior to its production.

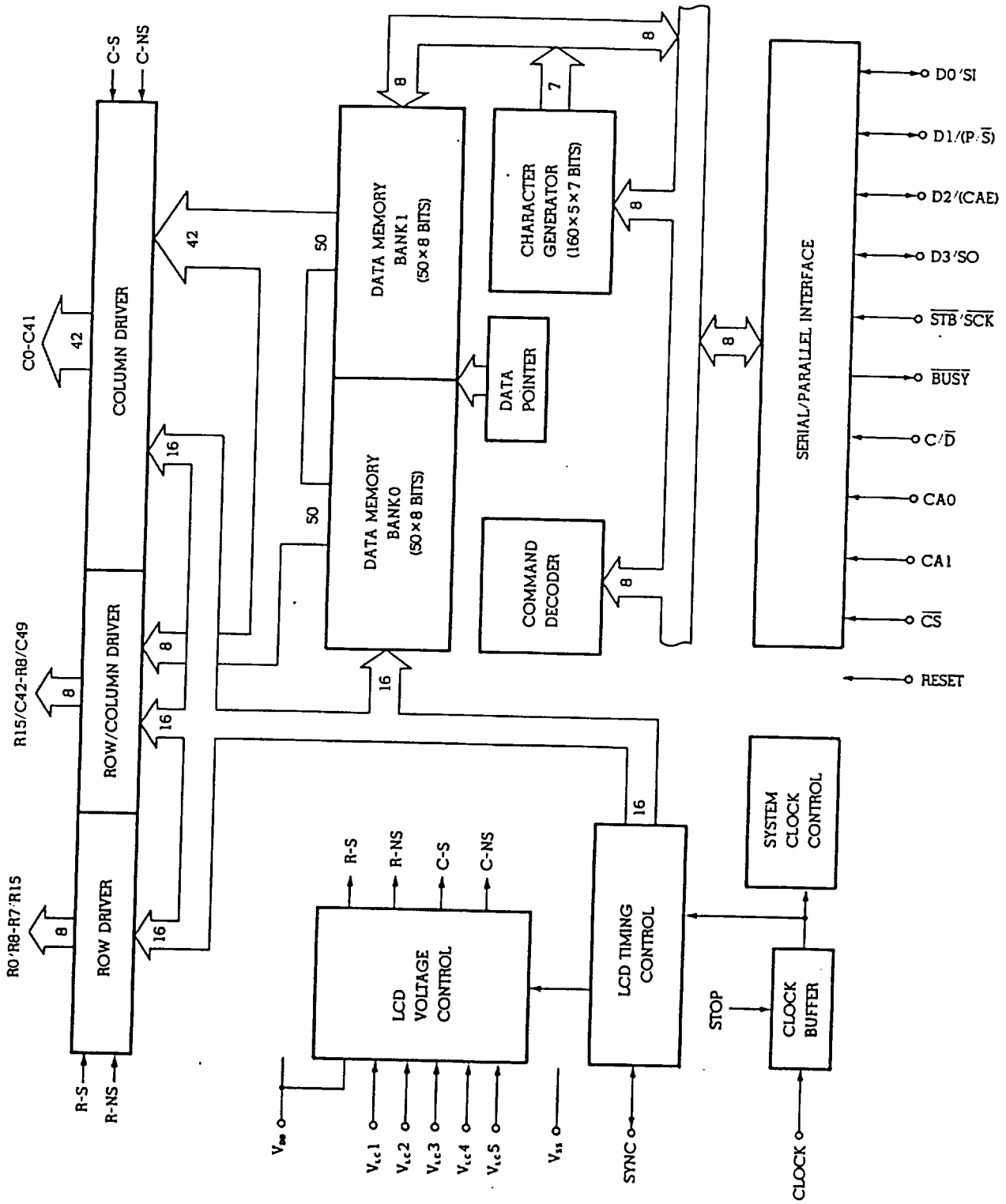
Document No. S10299EJ1V0DS00 (1st edition)
(O.D. No. IP-3593)
Date Published March 1996 P
Printed in Japan

Pin Configuration (Top View)

80-pin plastic QFP (14 x 20mm)



Block Diagram



CONTENTS

1. PIN FUNCTIONS

1.1 D0-D3 (Data Bus) ... 3-state input/output

In the parallel interface mode, these pins serve as 4-bit parallel data input/output pins.
Data on the D0-D3 lines is read at the $\overline{STB}$ signal rising edge. The 4-bit data, read at the first rising edge of the $\overline{STB}$, is loaded into the upper 4 bits of the serial/parallel register, and the data read at the second rising edge is loaded into the lower 4 bits of the register.
The serial/parallel register contents are output to the D0-D3 pins in synchronization with the $\overline{STB}$ signal falling edge. In the same manner as read operation, the upper 4 bits of the serial/parallel register are output in the first $\overline{STB}$ signal falling edge, and the lower 4 bits are output in the second $\overline{STB}$ falling edge.
In the serial interface mode, the D0 serves as the serial data input pin (SI), and the D3 pin serves as the serial data output pin (SO).
The D1 pin serves as the parallel/serial interface mode selection pin (P/$\overline{S}$), and the D2 pin serves as the chip address enable pin (CAE).

1.2 SI (Serial Data In) ... Also serves as D0 input

This pin serves as the serial data input pin in the serial interface mode. Data on the SI line is loaded into the serial/parallel register at the $\overline{SCK}$ rising edge. The first data becomes the MSB. This is a Schmitt trigger input with hysteresis, in order to prevent erroneous operation caused by noise.

1.3 SO (Serial Data Out) ... Also serves as D3 output

This pin serves as the serial data output pin in the serial interface mode. The serial/parallel register contents are output to the SO pin with the MSB first in synchronization with the $\overline{SCK}$ pin falling edge.

1.4 P/$\overline{S}$ (Parallel/Serial Select) ... Also serves as D1 input

This pin is sampled at the RESET signal falling edge (when the reset is released). If this pin is high, the parallel interface mode is set. If it is low, the serial interface mode is set. This is a Schmitt trigger input with hysteresis in order to prevent erroneous operation caused by noise.

1.5 CAE (Chip Address Enable) ... Also serves as D2 input

The CAE input has a meaning, if P/$\overline{S}$ input is low (when the serial interface mode is specified) at the RESET signal falling edge (when reset is released). If the CAE signal is high at this timing, the chip address function is enabled. If the CAE signal is low, the chip address function is disabled. This is a Schmitt trigger input with hysteresis in order to prevent erroneous operation caused by noise.

1.6 CA0, CA1 (Chip Address) ... Input

This is the input pin used to allocate the inherent address to select each μPD16434 chip, when interfacing with the CPU in a multi-chip configuration. In the parallel interface mode, CA0 and CA1 inputs are compared with the chip address information sent from the CPU, regardless of the CAE input. In the serial interface mode, these inputs are compared with the chip address information sent from the CPU, when the chip address selection function is enabled by the CAE input.

Table 1-1 Processing CA1, CA0 Pins

Mode	CA1, CA0
With chip address function - Always in parallel mode - When CAE=1 in serial mode	Set to 00, 01, 10, or 11 (always 00 in single chip configuration)
Without chip address function - When CAE=0 in serial mode	Always set to 00

Notes: In a multi-chip configuration in the serial interface mode, chip selection is also possible by providing decoded $\overline{CS}$ signals for the number of chips used, without using the chip address function. In this case, CAE for each chip must be set to 0, and CA1 and CA0 pins must be set to 00.

These are Schmitt trigger inputs with hysteresis in order to prevent erroneous operation caused by noise.

1.7 $\overline{CS}$ (Chip Select) ... Input

This is the chip select input, which is low active.
When the chip address function is not used, if a low is input to the $\overline{CS}$ input, the $\overline{STB}/\overline{SCK}$ and $C/\overline{D}$ inputs become effective, so that commands and data can be input/output.
When the chip address function is used, in order for the $\overline{STB}/\overline{SCK}$ and $C/\overline{D}$ inputs to become effective, the chip address information and CA0 and CA1 inputs must coincide, and moreover, the $\overline{CS}$ input should become low.
When the $\overline{CS}$ input is set to high, D3-D0 and $\overline{BUSY}$ pins unconditionally become high impedance.
This is a Schmitt trigger input with hysteresis in order to prevent erroneous operation caused by noise.

1.8 $\overline{STB}/\overline{SCK}$ (Strobe/Serial Clock) ... Input

In the parallel interface mode, this pin serves as the strobe signal input pin ($\overline{STB}$) for 4-bit parallel data input/output operation. In the serial interface mode, this pin serves as the serial clock input pin ($\overline{SCK}$) for serial data input/output operation.

1.9 C/$\overline{D}$ (Command/Data) ... Input

This pin is used to identify whether serial or parallel data input is a command or data. When inputting a command, set the C/$\overline{D}$ pin to high. When inputting data, set to low.
When inputting a command or data in the parallel interface mode, the command or data is latched at the second $\overline{STB}$ rising edge. In the serial interface mode, the command or data is latched at the rising edge of the 8th $\overline{SCK}$. However, in parallel input, switching C/$\overline{D}$ must be performed, before the falling edge of the 1st $\overline{STB}$.
When outputting data, C/$\overline{D}$ input must always be set to low, regardless of whether the mode is parallel or serial.
This is a Schmitt trigger input with hysteresis in order to prevent erroneous operation caused by noise.

1.10 $\overline{BUSY}$ (Busy) ... 3-state output

This pin outputs a $\overline{BUSY}$ signal which indicates to the CPU that the μPD16434 is busy because of internal processing.
If this signal is low, the μPD16434 is busy, and the CPU cannot execute read/write to the μPD16434.
The $\overline{BUSY}$ signal becomes low at the second rising edge of the $\overline{STB}$ signal in the parallel interface mode. In the serial interface mode, the $\overline{BUSY}$ signal becomes low at the rising edge of the 8th $\overline{SCK}$.
The μPD16434 sets the $\overline{BUSY}$ signal to high, when the μPD16434 completes the internal processing.
The $\overline{BUSY}$ output becomes high impedance, when the chip is not selected ($\overline{CS}$=high or the chip address does not coincide).

1.11 SYNC (Synchronous) ... 3-state input/output

In a multi-chip configuration, in which the row drive signal is commonly used, this pin inputs/outputs the synchronous signal in order to synchronize the phases of all LCD drive alternate cycle signals (row/column signals) with the frame period.
One chip in the multi-chip configuration is selected as the master, and the SYNC pin of the master is set to the output mode. The remaining chips all serve as slave chips, and these SYNC pins are set to the input mode.
The SMM command is used to specify whether the pin functions as an input or output pin.
The master chip, set in the output mode, outputs the SYNC pulse in the last cycle in each frame. A slave chip reads the SYNC pulse output from the master chip for synchronization with the master chip.
Figures. 1-1 and 1-2 show SYNC pulse output timing waveforms in 8-time-division and 16-time-division modes, respectively.
In single chip configuration, the SYNC pin can be set in either the input or output mode. However, when it is set in the input mode, the SYNC pin must be fixed to V_{SS}. If it is set in the output mode, the SYNC pin must be left open.

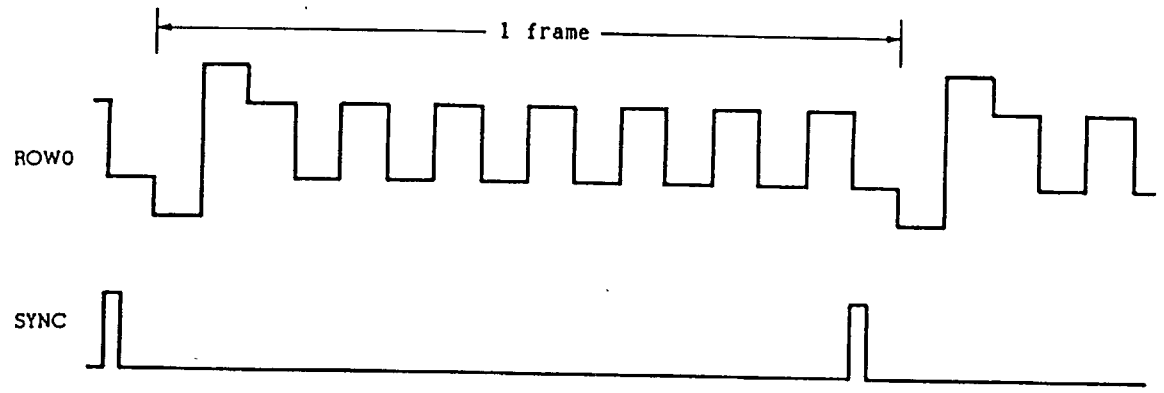


Fig. 1-1 SYNC Signal in 8-Time-Division Mode

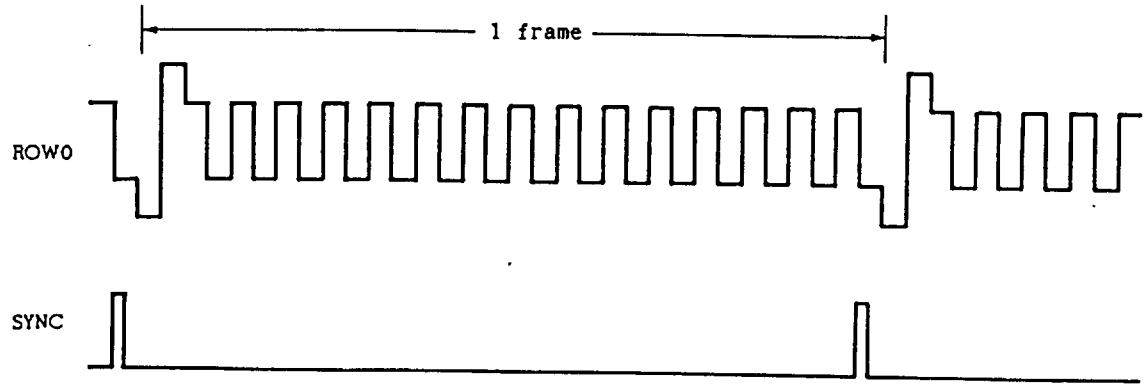


Fig. 1-2 SYNC Signal in 16-Time-Division Mode

1.12 C0-C41 (Column) ... Output

These pins serve as LCD column drive signal output pins.

1.13 R8/C49-R15/C42 (Row/Column) ... Output

These pins serve as LCD row drive signals R8 to R15 or column drive signals C49 to C42 output pins. Whether or not these pins are used as row or column pins is specified by the SMM command.

1.14 R0/R8-R7/R15 (Row) ... Output

These pins serve as LCD row drive signals R0 to R7 or R8 to R15. Whether or not these pins are used as R0 to R7 or R8 to R15 is specified by the SMM command.

1.15 V_{LC1}-V_{LC5} (LCD Drive Voltage Supply) ... Input

These pins input the reference voltage for determining the LCD row/column drive signal voltage level.

1.16 CLOCK (Clock) ... Input

This pin inputs the external clock.

1.17 RESET (Reset) ... Input

This is the high active reset signal input pin. The reset operation has priority over all other operations.
This input is also used for clearing the standby mode or operation to retain data in the data memory at a low power supply voltage.

1.18 V_{DD}

Positive voltage power supply pin.

1.19 V_{SS}

GND

2. INTERNAL BLOCK FUNCTIONS

2.1 Serial/Parallel Interface

The μPD16434 contains both serial and parallel interface functions. Whether the serial interface or the parallel interface is used is determined by whether the P/$\overline{S}$ input is high (specifying the parallel interface) or low (specifying the serial interface) at the RESET signal falling edge.

The interface circuit is used to write commands and data from the CPU or output data to the CPU.

The operation of the serial/parallel interface differs, depending on the data processing mode setting. When a RESET is input, the data processing mode is initialized to the write mode, so that the first command input can be accepted. Afterwards, the mode can be set to write related or read related data processing mode by the data processing set command.

If the data processing mode is set to the write, AND, OR, or the character write related mode, the serial/parallel interface is set to the data input mode, and the μPD16434 clocks in the data from the SI pin (serial data) or from the D3-D0 pins (4-bit parallel data) in synchronization with the rising edge of the $\overline{SCK}$ or the $\overline{STB}$, respectively.

If the data processing mode is set to the read mode, the serial/parallel interface becomes the data output mode and outputs data from the SO pin (serial data) or from the D3-D0 pins (4-bit parallel data) in synchronization with the falling edge of the $\overline{SCK}$ or the $\overline{STB}$.

The serial/parallel register serves as the buffer, between 8-bit serial data or two 4-bit parallel data transferred through the serial input/output (SI, SO) or parallel input/output (D3-D0) and 8-bit parallel data of the data memory.

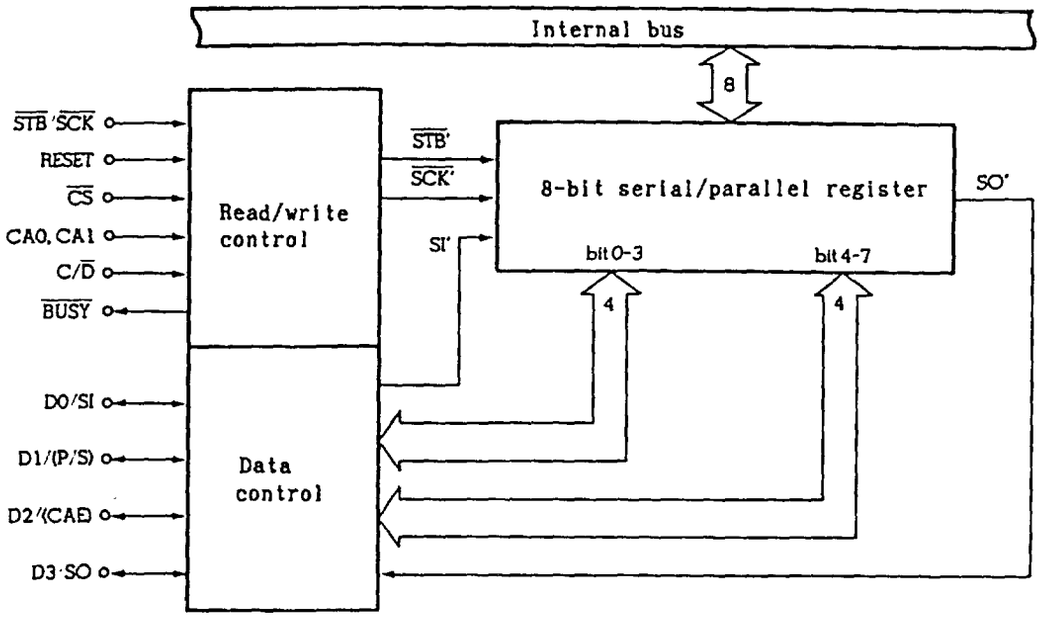


Fig. 2-1 Serial/Parallel Interface

If the C/$\overline{D}$ input indicates command specifications, the data input from the CPU to the serial/parallel interface is sent from the serial/parallel register to the command decoder for decoding.

In the write mode, if the C/D input indicates data specifications, the data loaded to the serial/parallel register is directly transferred to the data memory. In the AND or OR mode, the data loaded into the serial/parallel register is ANDed or ORed with the data memory contents, and the result is transferred to the data memory. In the character mode, the data loaded into the serial/parallel register is regarded as ASCII or JIS code and is sent to the character generator. It is decoded to 5 x 7-bit character display pattern, and is stored into 5 successive data memory addresses.

Only when set to the read mode, can the serial/parallel interface output data to the CPU. When set to the read mode, the serial/parallel interface always reads 8-bit data from the data memory and sets it in the serial/parallel register for the next read operation.

In the serial interface mode, the data in the serial/parallel register is output from the SO pin with the MSB first at each $\overline{SCK}$ falling edge.

In the parallel interface mode, the upper 4 bits of the data in the serial/parallel register are output from the D3 to D0 pins at the first falling edge of the $\overline{STB}$, and the lower 4 bits of the data in the serial/parallel register are output from those pins at the second falling edge of the $\overline{STB}$.
In either the serial/parallel interface mode, each time 8 bits of data are output, the next 8 bits of data are automatically read out from the data memory and set in the serial/parallel register.

2.2 Command Decoder

If the 8-bit data, input through the serial/parallel interface, is specified as a command (C/$\overline{D}$=1), the data is clocked in as a command, and is decoded to generate an internal control signal.

2.3 Character Generator

The character generator becomes effective, when a character mode setting command (SCML, SCMR) is executed. In this case, 8-bit data written through the serial/parallel interface is interpreted as a character code, and the 5 x 7-dot matrix pattern, corresponding to the code, will be generated. It is transferred to the 5 successive addresses in the data memory (7 bits x 5 times).
The character generator contains the following 160 different pattern data:

ASCII		JIS	
Upper-case alphabets	26	KATAKANA	55
Lower-case alphabets	26	Symbols	9
Numerical characters	10		
Symbols	34		

Figure 2-2 shows correspondence of character codes (ASCII/JIS) and 5 dots x 7 dots display patterns. 96 codes of 20H to 7FH correspond to ASCII characters, and A0H to DFH correspond to JIS characters.

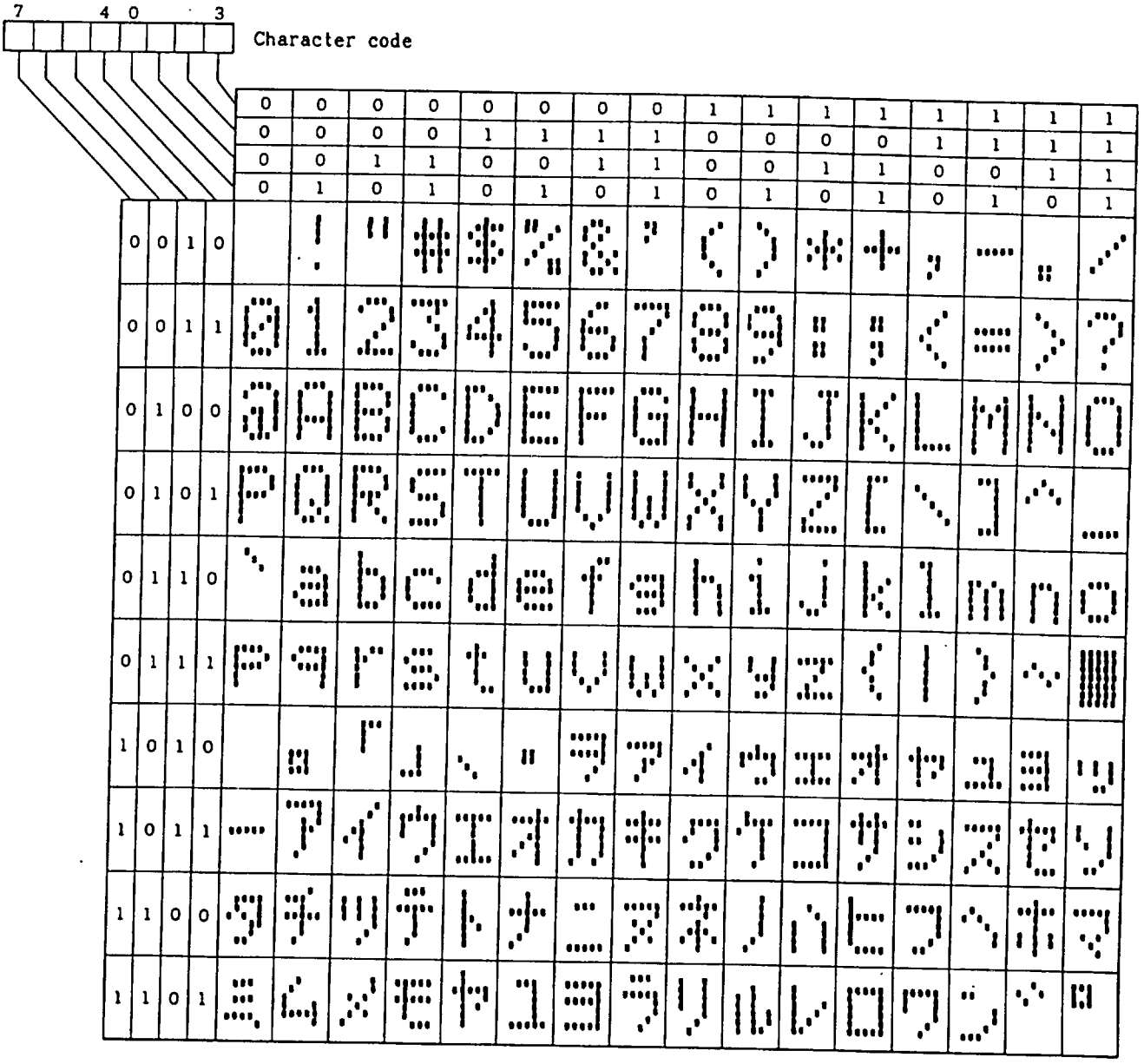
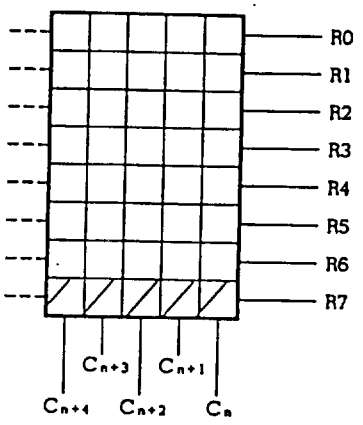


Fig. 2-2 Character codes and Display Pattern (Standard ROM code : 001)

Figure 2-3 shows the LCD configuration for the character generator. A character is configured in 5 x 7-dot configuration. The most significant bit (bit 7) of the data memory is not used by the character generator. Therefore, LCD dots, corresponding to the most significant bit (R7 in 8-time-division, or R7 and R15 in 16-time-division), can be used for the cursor or indicator display pattern, independently from the character generator. The most significant bits are manipulated by a cursor manipulation command (WRCURS, CLCURS), etc.

(a) 8-time-division



(b) 16-time-division

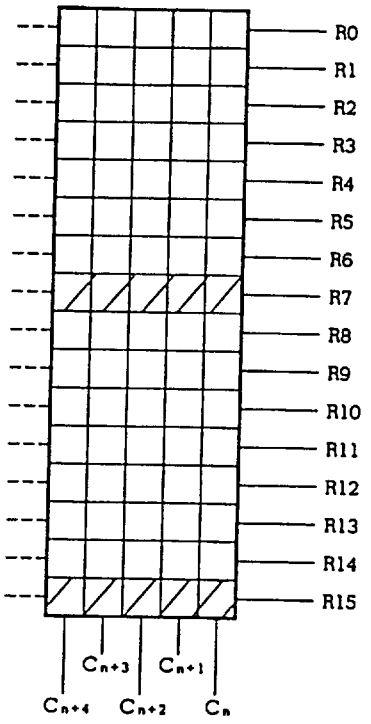


Fig. 2-3 LCD Configuration When Using Character Generator

2.4 Data Pointer

The data pointer consists of a 6-bit binary counter (DP5-DP0) and 1-bit bank flag (BNKF). It specifies the data memory address.

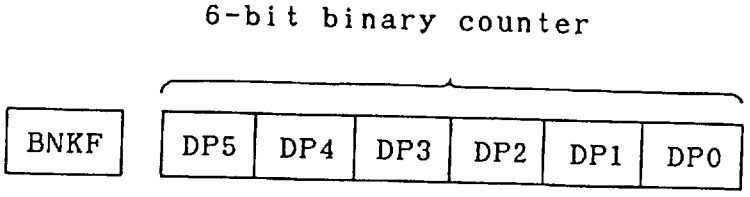


Fig. 2-4 Data Pointer Organization

The contents of the bank flag and the 6-bit binary counter are set simultaneously by the immediate data from the LDPI command. The bank flag contents specify the data memory bank (BNKF=0: Bank 0, BNKF=1: Bank 1), and the 6-bit binary counter contents specify the address (00H to 31H) in the bank specified by the bank flag. The 6-bit binary counter is an up/down counter. Its contents are modified (+1, -1, or held) each time a read, write, or AND operation is performed, or an 8-bit data is input/output in the OR mode, according to the specifications for each mode setting command.

When the BSET or BRESET command is executed, the 6-bit binary counter contents are modified (+1, -1, or held) each time according to the command specification.

In addition, in the character mode, the 6-bit binary counter contents are modified (+5 or -5) each time an 8-bit data is input or a cursor processing command is executed.

Notes: The 6-bit binary counter value can exceed the limit of the data memory address space. For example, if the 6-bit binary counter is decremented (-1) from 00H, the value will be 3FH, or it it is incremented (+1) from 31H, the value will be 32H. However, the data memory will perform nothing for the command specifying an address from 32H to 3FH.

2.5 Data Memory

The data memory is a static RAM configured by two 50-word x 8-bit banks. It is used for storing display data.

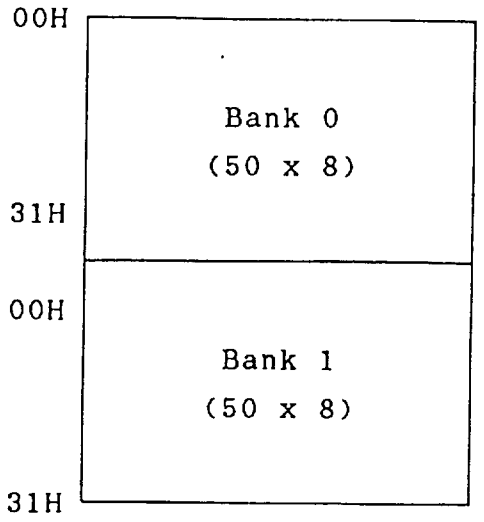


Fig. 2-5 Data Memory Configuration

The data memory bank is specified by the bank flag in the data pointer, and the address in the bank is specified by the 6-bit binary counter in the data pointer.

The 8-bit data, written to the serial/parallel interface by the CPU, is used for operation or decoded according to the specified data processing mode, and is written into the data memory.

The data memory contents can be directly manipulated by a bit manipulation instruction.

When the μPD16434 is set in the read mode, the data memory contents are output to the CPU through the serial/parallel interface.

The data memory contents are read out in bit units in synchronization with the row drive signal and are sent to the column driver for driving the LCD. This operation is performed independently from command/data write/read operation with the CPU, which is performed through the serial/parallel interface. Display data read out operation differs, depending on the number of time-divisions.

(1) 8-time-division (single/multi-chip configuration)

The contents of the display data in bank 0 or bank 1, whichever is specified by the SMM command, are read out to the column driver.

Figure 2-9 shows bits correspondence for the row driver and column driver for the data memory. If the data located at the Rn and Cm intersection is 1, the corresponding LCD dot is ON. If the data is 0, the dot is OFF.

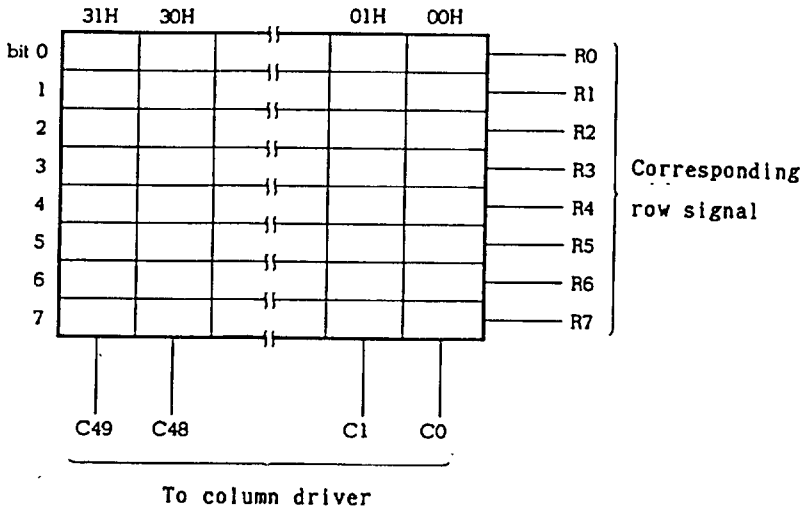


Fig. 2-6 Data Memory (8-Time-Division, Single/Multi-Chip)

(2) 16-time-division (single-chip configuration)

Bank 0 and bank 1 are used in a pair, and the contents are read out to the column driver as 42 x 16-bit display data. Figure 2-10 shows correspondence of bits for the row driver and column driver for the data memory. If the data located at the Rn and Cm interesection is 1, the corresponding LCD dot is ON. If the data is 0, the dot is OFF.

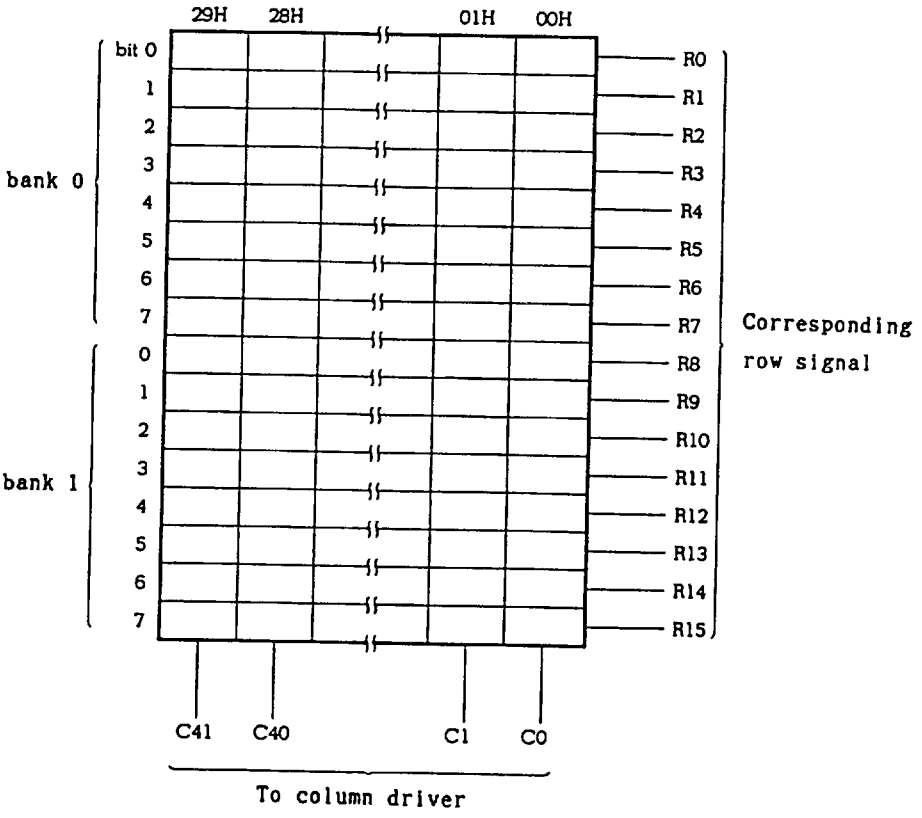


Fig. 2-7 Data Memory (16-Time-Division, Single-Chip)

(3) 16-time-division (multi-chip configuration)

Bank 0 and bank 1 are used in a pair, and the contents are read out to the column driver as 50 x 16-bit display data. The row drive signals, output from each μPD16434, are R0-R7 or R8-R15.

Figure 2-11 shows bits correspondence for the row driver and column driver for the data memory for each chip.

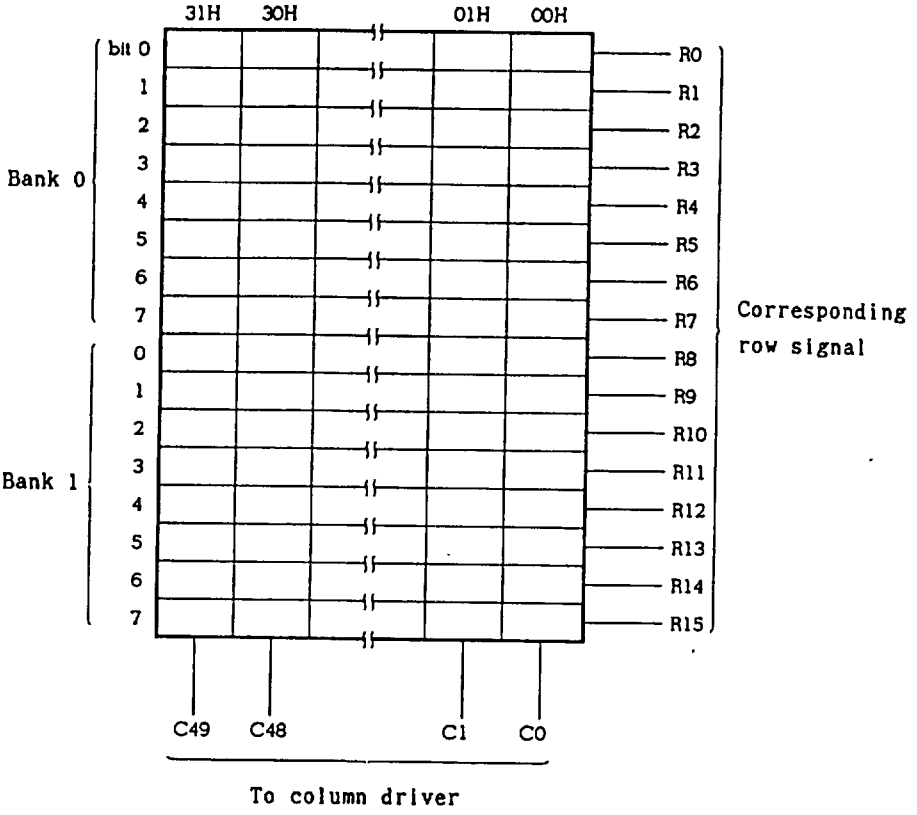


Fig. 2-8 Data Memory (16-Time-Division, Multi-Chip)

2.6 LCD Voltage Control Circuit

This circuit multiplexes the DC voltage supplied from the LCD drive reference voltage inputs ($V_{LC}1$ to $V_{LC}5$) with the AC signal synchronized with the CLOCK and SYNC signal, and supplies the signals to determine the select and non-select level of the row and column signals and the phase for the row and column drivers. Table 2-1 indicates these signal levels and the phase.

Table 2-1 LCD Drive Voltage Signal Levels and Phase

		8-time-divisions		16-time-divisions	
		-	+	-	+
Row	Select	$V_{LC}0$	$V_{LC}5$	$V_{LC}0$	$V_{LC}5$
	Non-select	$V_{LC}4$	$V_{LC}1$	$V_{LC}4$	$V_{LC}1$
Column	Select	$V_{LC}5$	$V_{LC}0$	$V_{LC}5$	$V_{LC}0$
	Non-select	$V_{LC}2$*	$V_{LC}2$	$V_{LC}3$	$V_{LC}2$

*: $V_{LC}2=V_{LC}3$

2.7 LCD Timing Control Circuit

This circuit generates the timing signals from the clock signal, according to the frame frequency specified by the SFF command, and the number of time divisions specified by the SMM command. The timing signals are necessary for automatically reading the display data and driving the LCD, and are supplied to the data memory row/column driver, and LCD voltage control circuit.
If the SYNC signal is set to the output mode by the SMM command, the SYNC signal is output for each frame. If the SYNC mode is specified to the input mode, the SYNC signal supplied from some other chip is input to generate the timing signals in synchronization with each frame interval.
The SYNC signal input/output function is used to synchronize the LCD drive timing between chips in multi-chip configuration.

2.8 Row/Column Driver

The row/column driver consists of the column driver for C0-C41 signals, row/column driver for R15-R8 and C42-C49, and a dual mode row driver for R0-R7 or R8-R15.
The dual mode row driver function is determined by the SMM command.

Table 2-2 LCD Dual Mode Row Driver Function Selection

<table>
<tr><th>$M_2M_1M_0$*1</th><th>Number of time divisions</th><th>R0/R8-R7/R15</th><th>R15/C42-R8/C49</th></tr>
<tr><td>0 0 0</td><td rowspan="4">8</td><td rowspan="4">R0-R7</td><td rowspan="6">C42-C49</td></tr>
<tr><td>0 0 1</td></tr>
<tr><td>0 1 0</td></tr>
<tr><td>0 1 1</td></tr>
<tr><td>1 0 0</td><td rowspan="4">16</td><td rowspan="2">R8-R15*2</td></tr>
<tr><td>1 0 1</td></tr>
<tr><td>1 1 0</td><td rowspan="2">R0-R7*3</td></tr>
<tr><td>1 1 1</td><td>R15-R8</td></tr>
</table>

Note 1: $M_2M_1M_0$ is code specified by SMM command.
2: Some other chip handles R0-R7 outputs.
3: Some other chip handles R8-R15 outputs.
$M_2M_1M_0$=111 is for single chip configuration. In this case, R15/C42-R8/C49 are used as row signal outputs, and all 16 row signals are output from this chip.

These drivers perform switching of the analog level in correspondence to the contents of the desplay data read out from the data memory and the timing signals supplied from the LCD timing control circuit, according to the select level, non-select level, and phase supplied from the LCD voltage control circuit, and generates and outputs row and column drive signals in order to directly drive the LCD.

3. DATA INPUT/OUTPUT OPERATION

In the μPD16434, a command/data consists of 1 byte (8 bits), and processing is performed each time a byte of data is transferred in either the serial or parallel mode.

The end of a byte data transfer is confirmed by the byte counter (octal/binary counter) which counts eight $\overline{SCK}$ counts or two $\overline{STB}$ counts.

This counter is unconditionally cleared, when $\overline{CS}$=high or RESET=high, and becomes ready to count a new byte or data. Therefore, if $\overline{CS}$ is set to high or RESET is input in the middle of a byte transfer, the byte transfer is not guaranteed.

In the serial interface mode, data is treated as 8-bit serial data. It is regarded that 1 byte of data has been input or output, when eight serial clock pulses ($\overline{SCK}$) are counted in the chip selected condition, then internal processing is started. At the 8th rising edge of the $\overline{SCK}$, the μPD16434 sets the $\overline{BUSY}$ signal to low to inform the CPU that the μPD16434 is in a busy state. When the internal processing completes, the μPD16434 sets the $\overline{BUSY}$ signal to high to inform the CPU that the μPD16434 is ready for the next byte transfer.

The serial data is input/output with the MSB first (refer to Figs. 3-1 and 3-2).

If the chip address selection function is specified in the serial interface mode, the 8-bit serial data (only the lower 2 bits have a meaning) for chip address information must be written first after the $\overline{CS}$ falling edge. Only the chip, whose address coincides with this information, can enter command input or data input/output operation (refer to Figs. 3-3 and 3-4).

In the parallel interface mode, since the data bus (D3-D0) is a 4-bit bus, data is treated as 4-bit x 2 parallel data. When the parallel data strobe signal ($\overline{STB}$) is counted twice in the chip selected state, it is regarded that a byte of data has been input/output, then the μPD16434 enters the internal processing.

At the 2nd rising edge of the $\overline{STB}$, the μPD16434 sets the $\overline{BUSY}$ signal to low, to inform the CPU that the μPD16434 is in a busy state. When the internal processing completes, the μPD16434 sets the $\overline{BUSY}$ signal to high, to inform the CPU that the μPD16434 is ready for the next byte transfer.

In both input and output operation, the upper 4 bits of paralel data correspond to the first $\overline{STB}$, and the lower 4 bits of parallel data correspond to the second $\overline{STB}$.

The parallel interface of the μPD16434 is compatible with the uPD82C43 I/O expander, so that the parallel data can be input to the μPD16434 in the same manner as sending 4-bit data twice to the uPD82C43. In addition, 8-bit data can be read out from the serial/parallel register of the μPD16434 in the same way as reading 4-bit data twice from the uPD82C43.

The chip address selection function is always specified in the parallel interface mode. After the $\overline{CS}$ falling edge, the data on the D1 and D0 lines, read at the first falling edge of the $\overline{STB}$, becomes the chip address information. The lower 2 bits of the command code, output from the CPU as the data for selecting port 4 to port 7 of the uPD82C43, are used as the chip address information. After the $\overline{CS}$ falling edge, the command code, output from the CPU at the second and successive falling edges of the $\overline{STB}$, has no meaning for the μPD16434 (refer to Figs. 3-5 and 3-6).

Refer to Section 4 for details on chip address function selection.

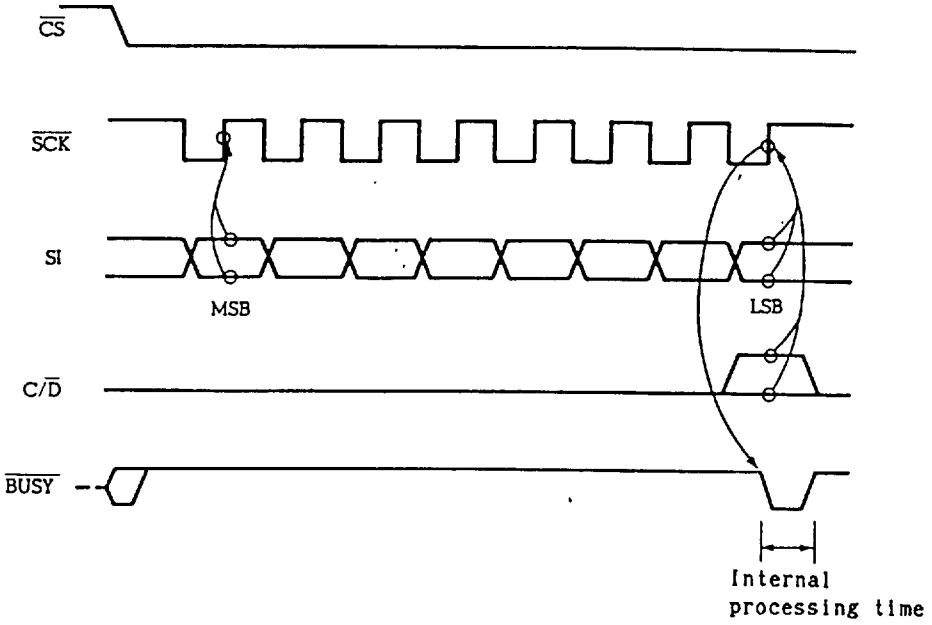


Fig. 3-1 Serial Input Timing Waveforms (Without Chip Address Selection Function)

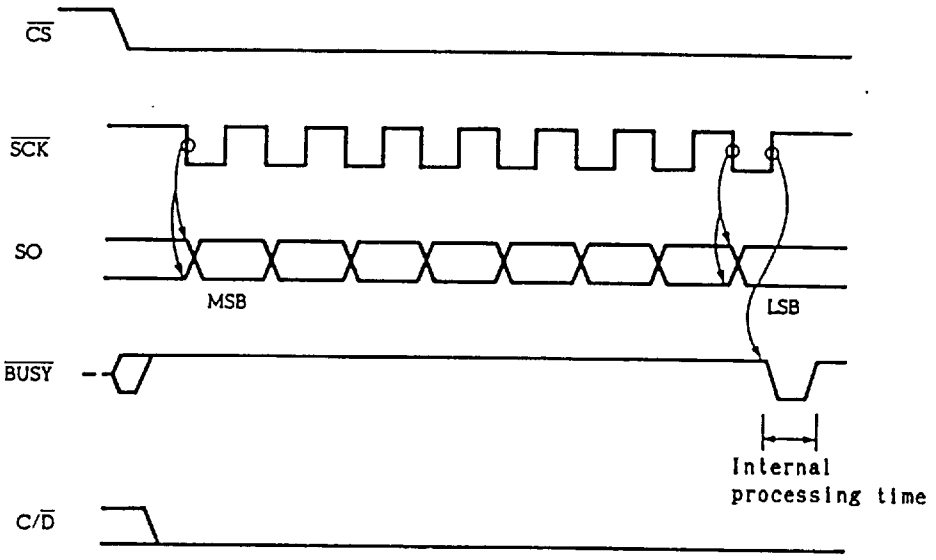


Fig. 3-2 Serial Output Timing Waveforms

Fig. 3-3 Serial Input Timing Waveforms (With Chip Address Selection Function)

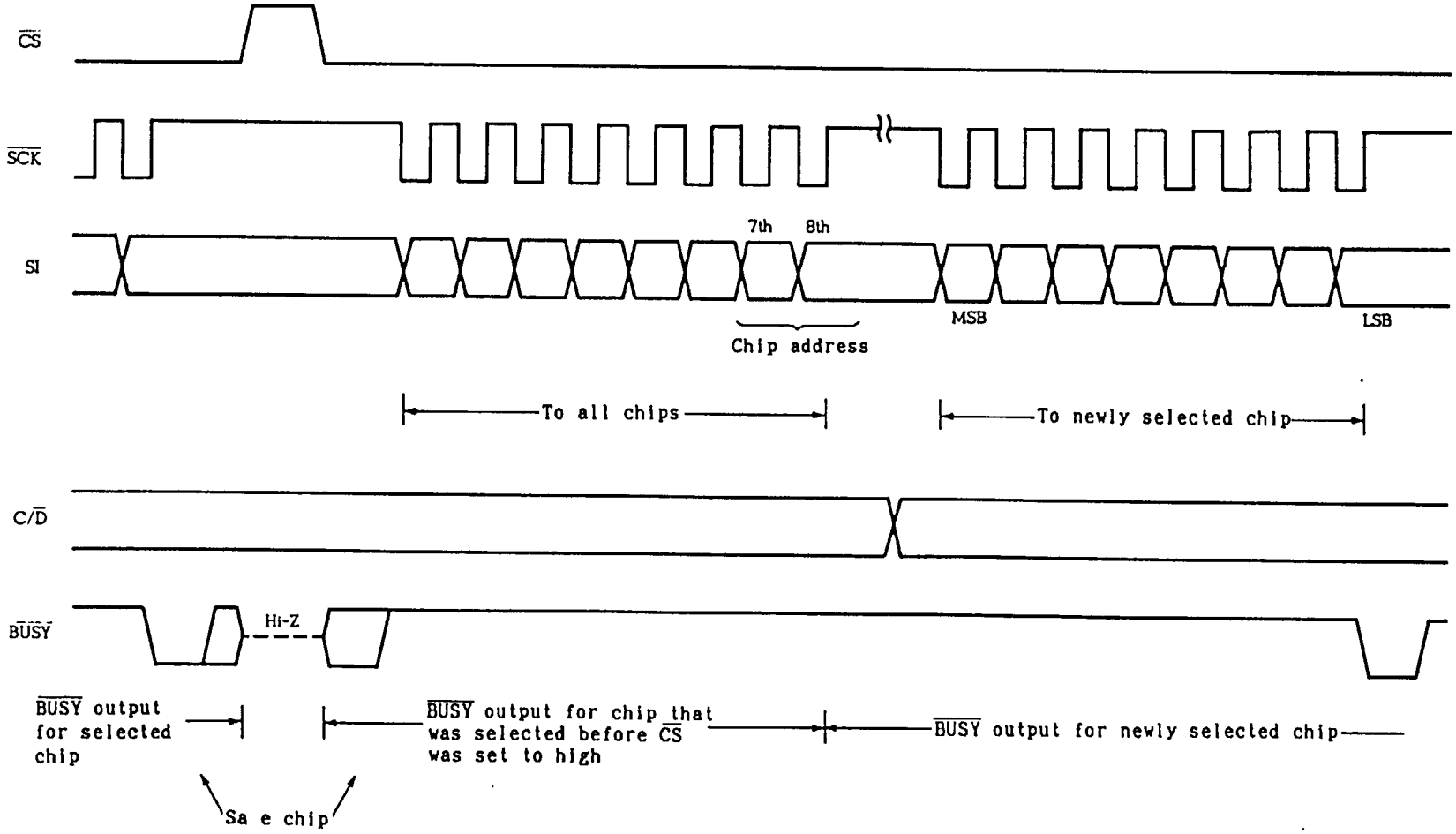
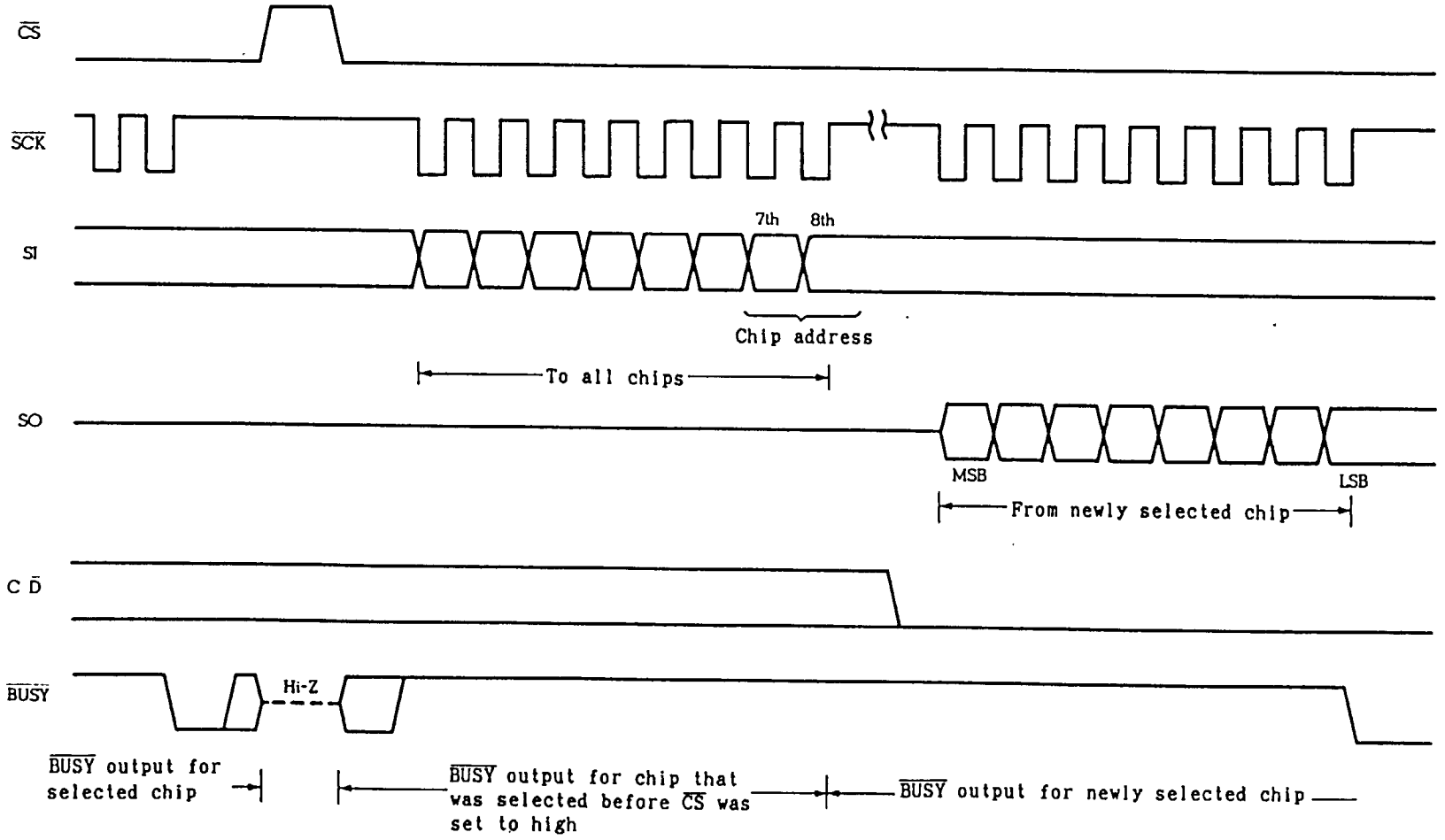


Fig. 3-4 Serial Output Timing Waveforms (With Chip Address Selection Function)



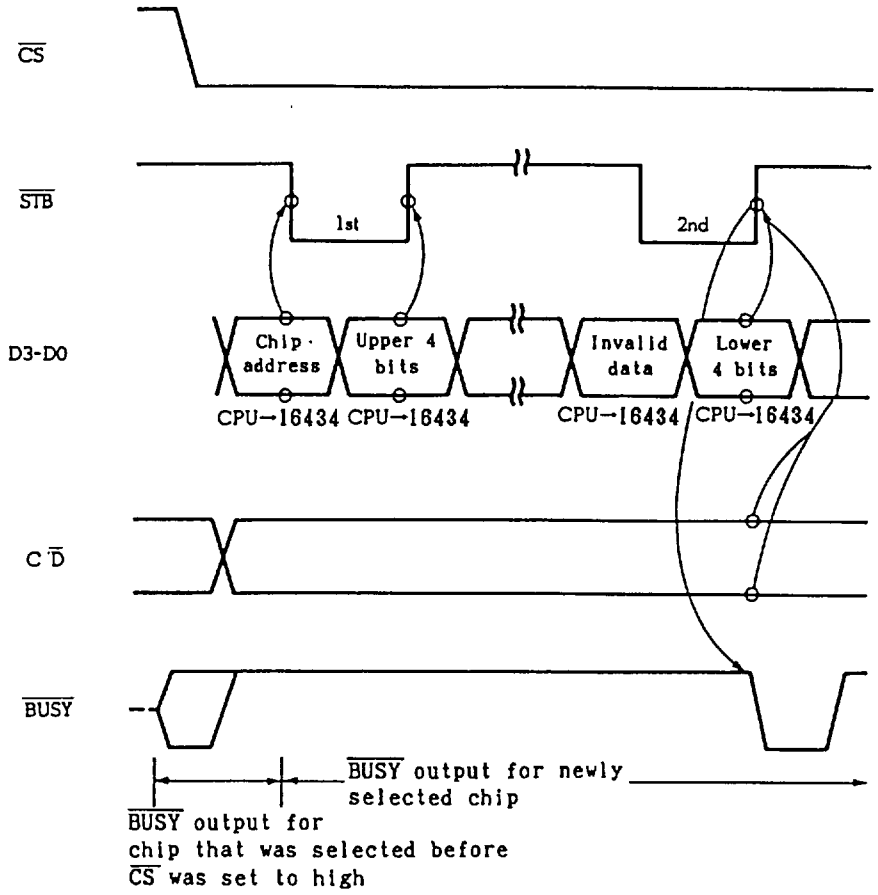


Fig. 3-5 Parallel Input Timing

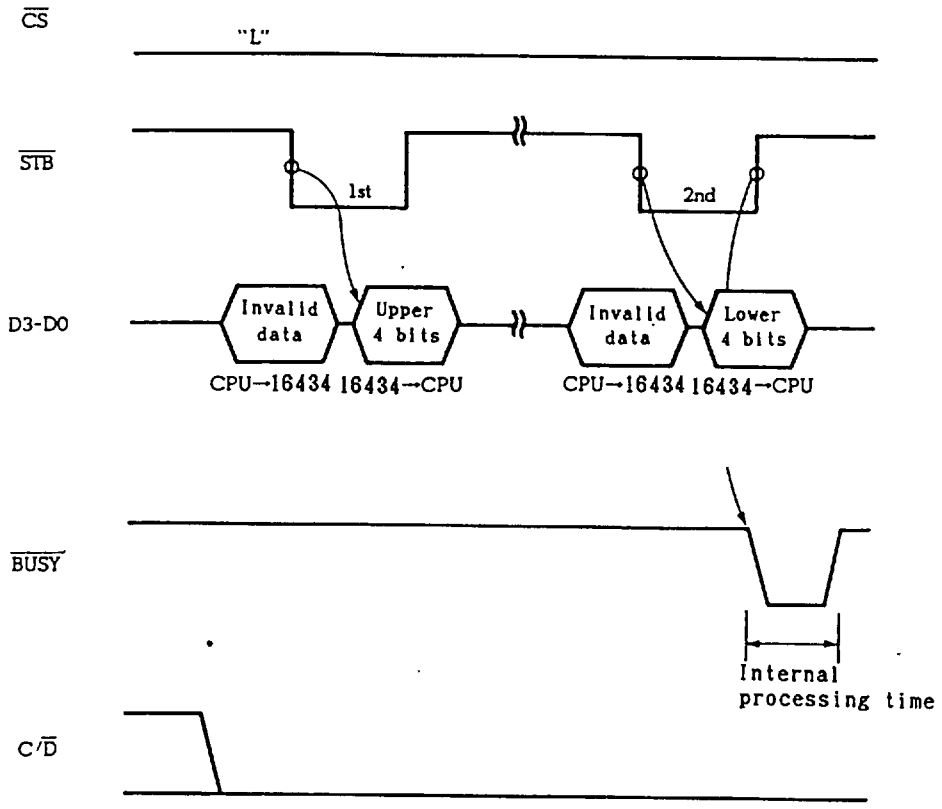


Fig. 3-6 parallel Output Timing

4. SELECTING μPD16434 INTERFACE FUNCTION WITH CPU

The command/data for the μPD16434 is 8 bits long. However, serial interfacing with the CPU is made in 8-bit transfer or parallel interfacing is made in two 4-bit transfers. In addition, the μPD16434 is provided with a chip address selection function for multi-chip system configuration.
Whether the serial or parallel interface is used and whether or not the chip address select function is used are specified by the data on the D2 (CAE) line and D1 (P/$\overline{S}$) line at the RESET signal release timing (falling edge).

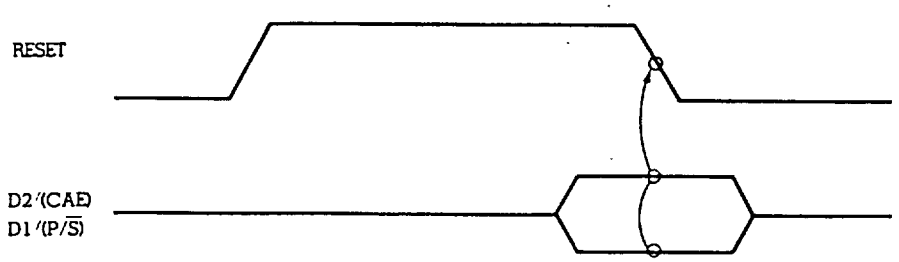


Fig. 4-1 Interface Specification Timing Waveforms

Table 4-1 Interface Specification Code

D2/(CAE)	D1/(P/$\overline{S}$)	Serial/parallel specification	Chip address selection function
0	0	Serial	Unprovided
1	0		Provided
0/1	1	Parallel	

4.1 Functions of Shared Pins

The functions of the $\overline{STB}/\overline{SCK}$, D3/SO, and the D0/SI pins used for clock input and data input/output differ, depending on whether the serial interface or the parallel interface is specified, as indicated in Table 4-2.

Table 4-2 Functions of Shared Pins

Pin name	Serial (P/$\overline{S}$=0)	Parallel (P/$\overline{S}$=1)
$\overline{STB}/\overline{SCK}$	$\overline{SCK}$ input	$\overline{STB}$ input
D3/SO	SO output	D3-D0 input/ output (4-bit parallel data bus)
D2/(CAE)	-	
D1/(P/$\overline{S}$)	-	
D0/SI	SI input	

4.2 Chip Address Selection Function

In a multi-chip system configuration, the chip address selection function compares the chip address assigned to each μPD16434 (by CA0, CA1 inputs) in advance and the chip address information (2 bits) sent from the CPU in the serial or parallel data format. Only the chip whose address coincides with the chip address information is selected (enables command/data input/output). Thus, the CPU need not send two or more chip select signals ($\overline{CS}$). This function is unconditionally provided in the parallel interface mode. However, in the serial interface mode, this function is provided, when D2/(CAE)=1 (at reset release), is specified.

(1) Parallel interface mode (refer to Figs. 3-5 and 3-6)

After the falling edge of the $\overline{CS}$, the data read into D1 (corresponds to CA1), and D0 (corresponds to CA0) at the first falling edge of the $\overline{STB}$ becomes the 2-bit chip address information.
The parallel interface is equivalent to that for the uPD82C43 I/O expander. Therefore, the chip address information (0-3) for the μPD16434 can be obtained on the D1 and D0 lines at the falling edge of the $\overline{STB}$ by executing an output or input instruction for port 4 to port 7 of the uPD82C43, when the uPD50H is connected to the μPD16434 using the uPD82C43 interface function.

(2) Serial interface mode (refer to Figs. 3-3 and 3-4)

After the falling edge of the $\overline{CS}$, the data read in to SI at the rising edge of the 7th $\overline{SCK}$ (corresponds to CA1) and 8th $\overline{SCK}$ (corresponds to CA0), that is the lower 2 bits of the first 8-bit serial data, becomes the chip address information.

Notes 1: When a RESET is input, the chip address comparison data (data compared with CA1 and CA0) in the μPD16434 is cleared to "00". Therefore, in multi-chip configuration, if the $\overline{CS}$ is set to low immediately after the RESET input is released, a chip whose CA1 and CA0 are set to "00" sets the $\overline{BUSY}$ to high, informing the CPU that the chip can be accessed.
If no chip address is sent, a chip whose CA1 and CA0 are "00" will be accessed.

2: The following points must be noted for a multi-chip configuration system using the parallel interface; when transferring the process from chip A, which has already been in the read mode to chip B, and again selecting chip A after that, the data pointer must be set by the data pointer load command before reading data.

5. LCD DRIVE REFERENCE VOLTAGE SUPPLY

The value of the LCD drive reference voltage to the μPD16434 differs, depending on whether the number of time divisions is 8 or 16, so that the LCD drive reference voltage should be set as shown in Figures 5-1 and 5-2.

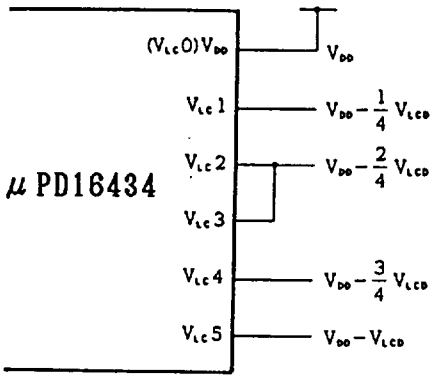


Fig. 5-1 8-Time-Divisions

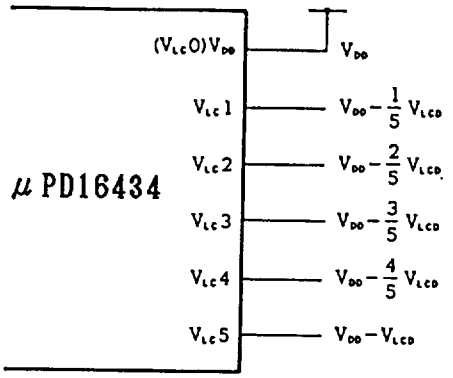


Fig. 5-2 16-Time-Divisions

Remarks: For both 8 and 16-time-divisions, the LCD drive voltage (L_{LCD}) must not exceed V_{DD}.

5.1 Supplying LCD Drive Reference Voltage by Resistor Network

Figures 5-3 and 5-4 show circuit examples, which supply the LCD drive reference voltage indicated in Figures 5-1 and 5-2 using register networks which divide the voltage level between V_{DD}-V_{SS}.

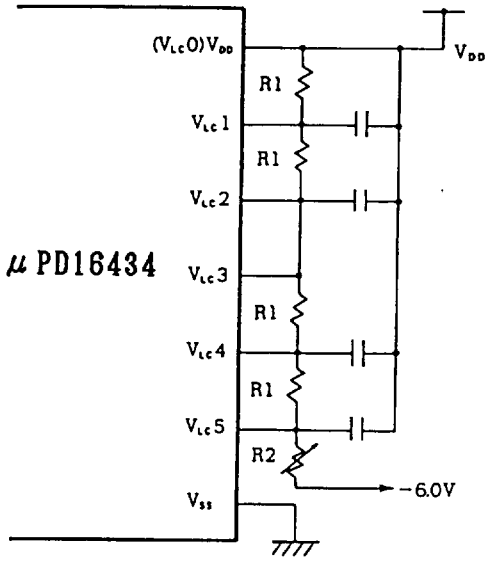


Fig. 5-3 8-Time-Divisions Circuit Example

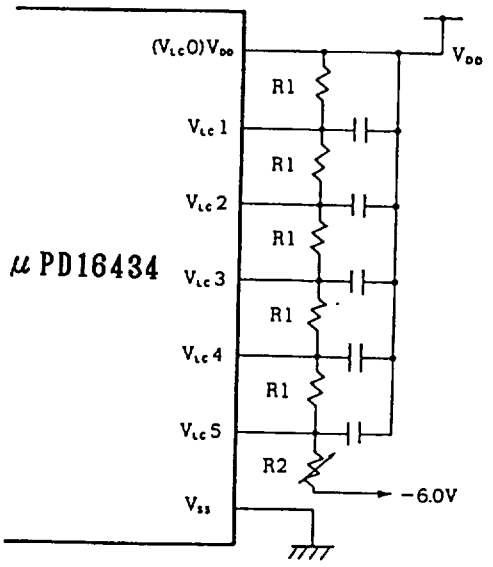


Fig. 5-4 16-Time-Divisions Circuit Example

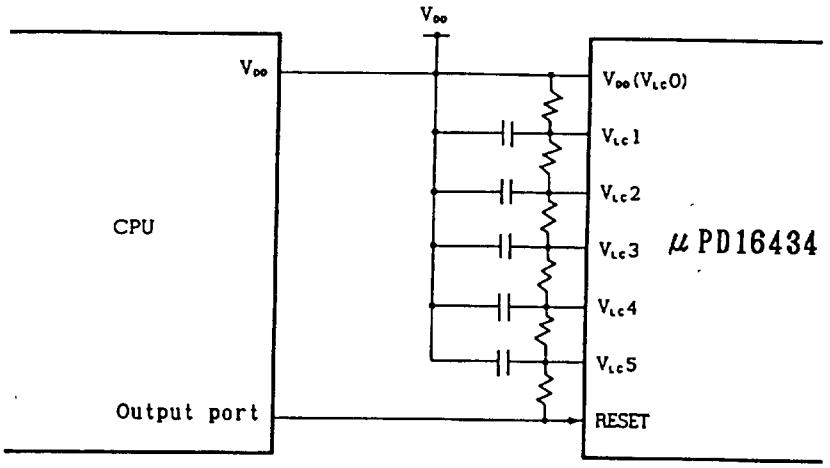
The values of R1 and R2, which divide the voltage for 8-time-divisions and 16-time-divisions, are determined by the following expressions:

$$R1 = \frac{V_{LCD}}{4(V_{DD} - V_{LCD})} \times R2 \quad \text{(8-time-divisions)}$$

$$R1 = \frac{V_{LCD}}{5(V_{DD} - V_{LCD})} \times R2 \quad \text{(16-time-divisions)}$$

5.2 Reduction in Current Consumption by RESET Signal

If a resistor network is used to supply the LCD drive reference voltage, some current is drained by the resistor network connected across V_{DD} and V_{SS} when no display operation is performed, such as when the μPD16434 is in the STOP mode or when it is being reset. Therefore, for a system to which reducing the current draw is extremely important, the current path through the resistor network must be cut off by an external circuit, when no displaying is performed, to eliminate unnecessary current flow. Figure 5-5 shows a circuit which cuts off the current to the resistor network during reset state (RESET=high) using the RESET signal level instead of the V_{SS} level.



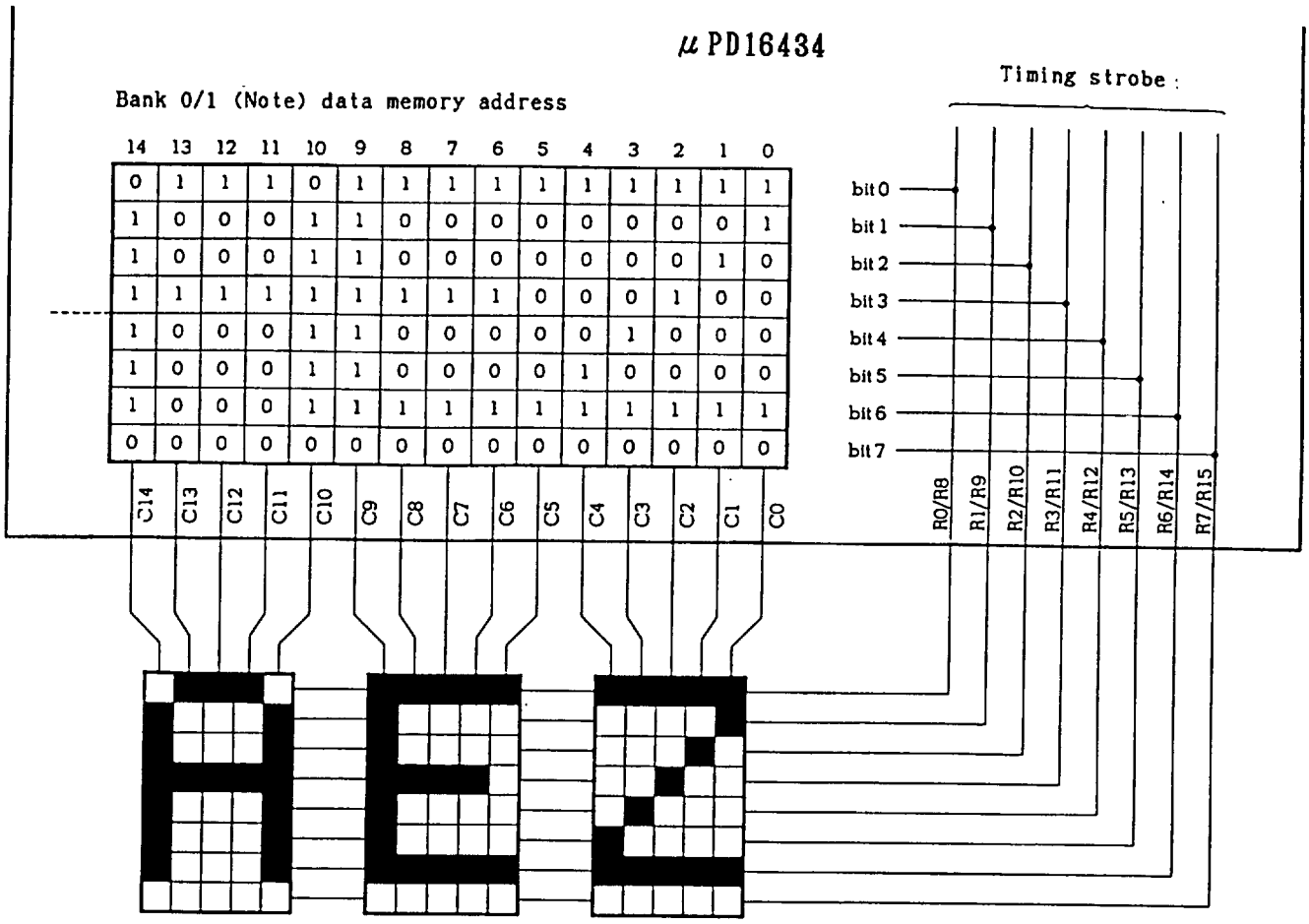
Remarks: The power to the CPU and the μPD16434 must be from the same source.

Fig. 5-5 Example of Controlling Current Path Using RESET Signal

6. DISPLAY EXAMPLES

Figure 6-1 shows how the data memory contents and LCD display pattern are corresponded, when displaying characters "AEZ" in 8-time-division mode.
This example is to display 3 digits of 5 x 7 (5 x 8) dot characters, and uses data memory addresses 00H to 0EH (0 to 14) and column signals C0 to C14.
Figure 6-2 shows the timing waveforms for displaying character "A" in columns C14 to C10 for the display example, shown in Figure 6-1.

Figure 6-3 shows how the data memory contents and display pattern are corresponded, when displaying characters $"^{ANZ}_{8,5}"$ in 16-time-division mode.
This example is to display 5 x 7 (5 x 8) dot characters in two rows, 3 digits in each row, and uses data memory addresses 00H to 0EH (0 to 14) in data memory banks 0 and 1 and column signals C0 to C14.
Figure 6-4 shows the timing waveforms for displaying character $"^{A}_{8}"$ in columns C14 to C10 for display example, shown in Fig. 6-3.
In Figures 6-2 and 6-4, when the differential waveform levels between the row signal and column signal are V_{LCD} and $-V_{LCD}$, the LCD dot corresponding to these signals will be lit.



Note: Display data is read out from either bank 0 or bank 1.

Fig. 6-1 8-Time-divisions

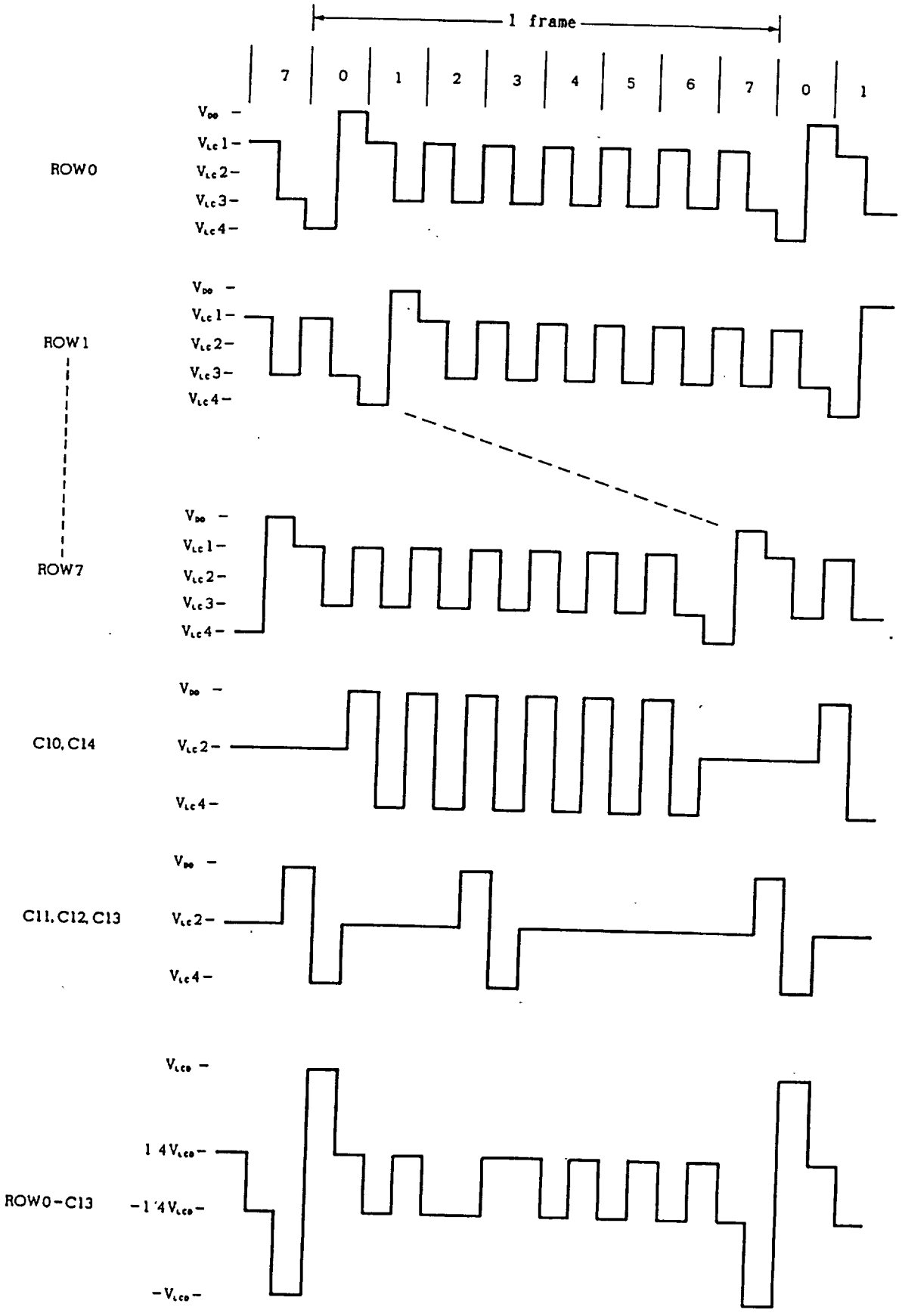


Fig. 6-2 8-Time-divisions (When Displaying Character A)

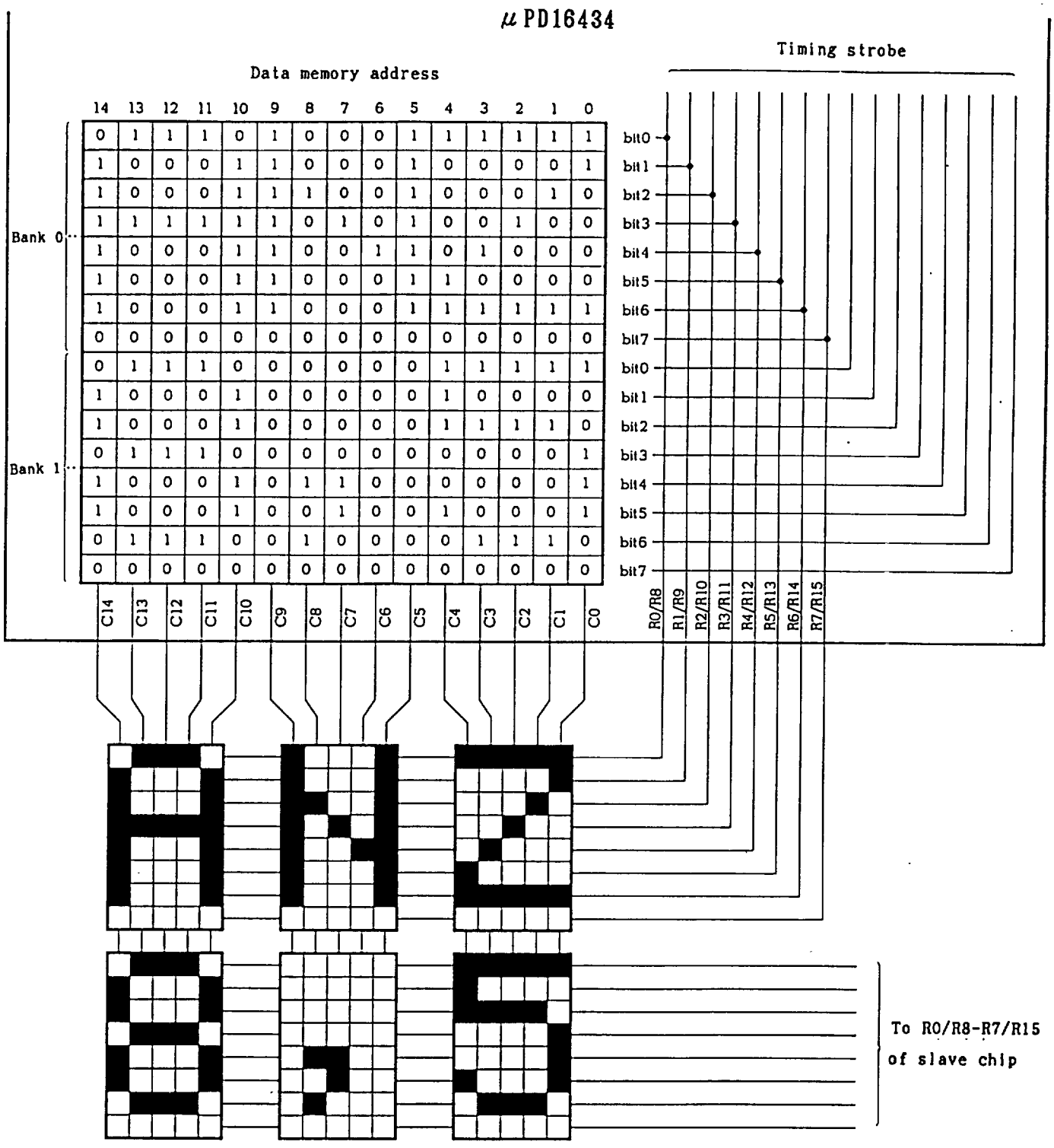


Fig. 6-3 16-Time-divisions

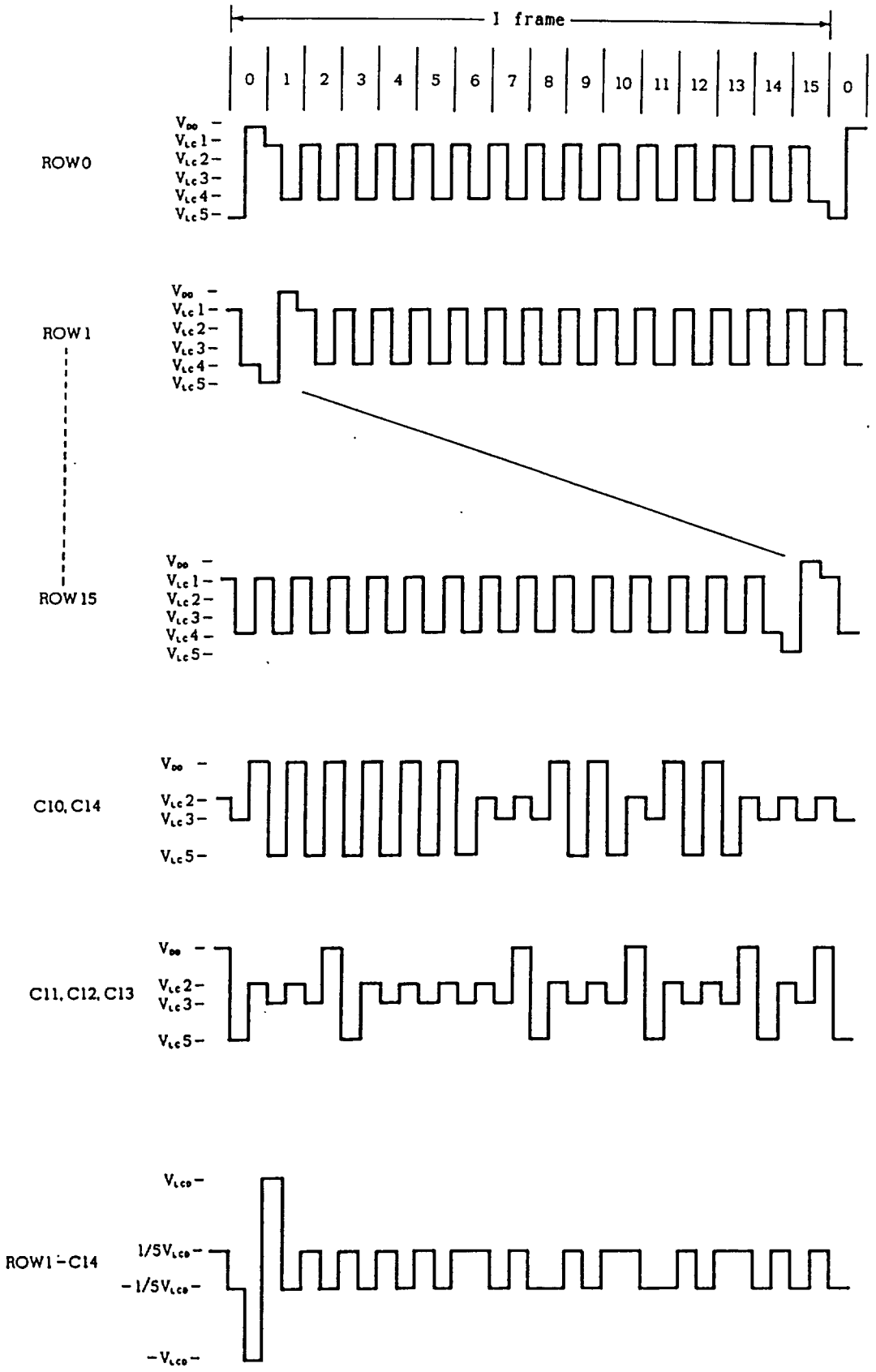


Fig. 6-4 16-Time-divisions (When Displaying Characters A and 8)

7. STANDBY MODE

The μPD16434 offers the standby mode in order to reduce the power consumption, when displaying operation is not necessary.
The standby mode is set, by executing the STOP command. The standby mode is actually set, when the $\overline{BUSY}$ signal is set to high after the STOP command is executed. In the standby mode, the μPD16434 stops supplying the CLOCK signal to the LCD timing control circuit and the clock control circuit by internally masking the CLOCK signal. In addition, the μPD16434 initializes the data processing mode to the auto-increment (I_1I_0=00) write mode. However, no other modes are affected by RESET operation, so that the interface mode and the display mode will be retained. The standby mode is cleared when a byte of data (command or data) is input, or when the RESET signal falls. However, the processing necessary during the standby mode and operation after clearing the standby mode differ, depending on which method is used.
In addition, the CLOCK signal to the μPD16434 can be stopped during the standby mode. In this case, the power consumption can be further reduced, compared to when the CLOCK is only internally masked.

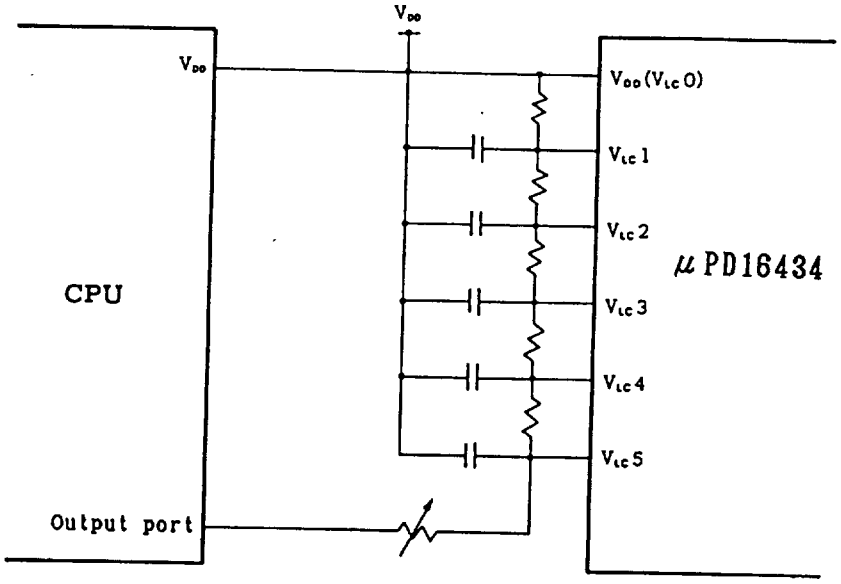
7.1 Clearing Standby Mode

(1) Clearing standby mode by writing a byte of data

If the previous operation modes (except the data processing mode), used before entering the standby mode, needs to be maintained, the standby mode can be cleared by writing a byte of data (command or data).
In the serial interface mode, the standby mode is cleared, when writing 8 bits of serial data is completed (at the rising edge of the 8th $\overline{SCK}$). In the parallel interface, the standby mode is cleared, when writing the second 4-bit data is completed (at the rising edge of the 2nd $\overline{STB}$).
When the chip address selection function is used in the

serial interface mode, if $\overline{CS}$ is set to high in the standby mode, the first 8-bit data, after $\overline{CS}$ is set to low, is used as the chip address information, when the standby mode is cleared. Therefore, the standby mode is cleared, when writing the next 8-bit data is completed.

Notes: During the standby mode, the clock necessary for driving the LCD by alternating current is stopped. Therefore, the LCD drive signal level, before entering the standby mode, is maintained in the standby mode. This means that a DC voltage remains applied to the LCD in the standby mode. To avoid this, control the $V_{LC}5$ pin voltage using the CPU output port, as shown in Fig.7-1, and output a high level from the output port, before executing the STOP command, to eliminate voltage differential between the V_{DD} and $V_{LC}5$.



Remarks: The power fed to the CPU and the μPD16434 must be from the same source.

Fig. 7-1 Controlling LCD Drive Voltage

(2) Clearing standby mode by RESET signal

For a system for which only the contents of the data memory need be held when the standby mode is cleared, the RESET signal can be used to clear the standby mode.
When using the RESET signal for clearing the standby mode, set the RESET signal to high after the standby mode is initiated (this can be checked by determining whether the $\overline{\text{BUSY}}$ is set to high), then set the RESET signal to low when clearing the standby mode. While the RESET signal is high, the LCD drive voltage becomes the same as when it is in the reset operation, and no voltage is applied to the LCD. However, unlike normal reset operation, the contents of the data memory will not become undefined by the RESET signal. The data which existed before entering the standby mode is retained, and it can be used after clearing the standby mode.
In a system which uses this method, the current path control by the RESET signal shown in Fig. 5-5 can be used.

7.2 Stopping Clock Supply and Retaining Data at Low Voltage in Standby Mode

In the standby mode, only the data memory contents can be retained at a reduced voltage level. In this case, the power consumption can be further reduced by stopping the CLOCK supply to the μPD16434. When stopping the CLOCK supply to the μPD16434 in the standby mode, check that the μPD16434 is in the standby mode (check that the $\overline{BUSY}$ signal is high, after executing the STOP command), set the RESET signal to high, then stop the CLOCK supply to the μPD16434 after the specified time period.

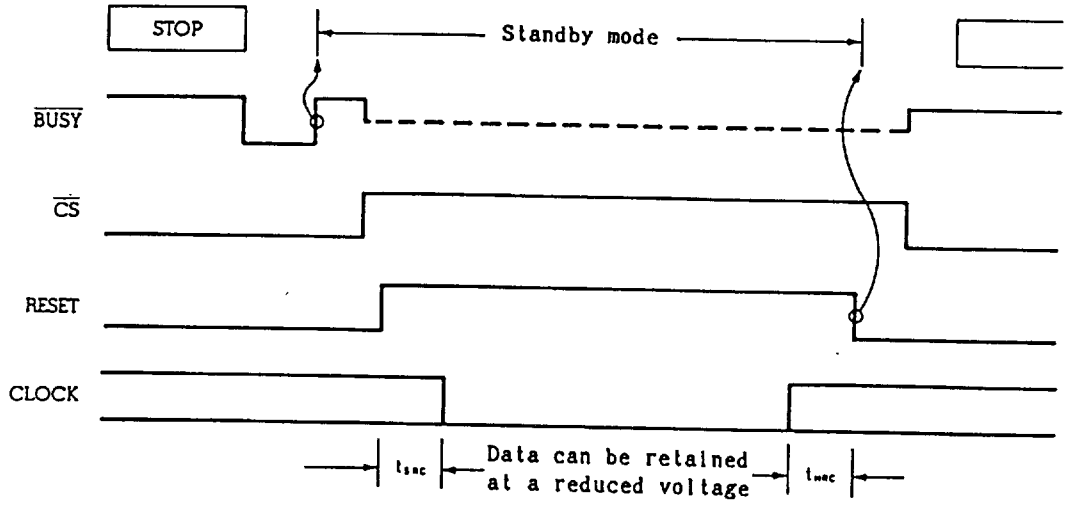


Fig. 7-2 CLOCK Supply Stop Timing Waveforms in Standby Mode

8. RESET OPERATION

The μPD16434 is initialized as follows, when a high level is input to the RESET pin:

- The chip address compare data (compared with CA1, CA0 inputs) is initialized to 00.
  In a multi-chip configuration, $\overline{BUSY}$ output operation will differ, depending on whether CA1 and CA0 of the chip are 00 (coinciding address) or not (non-coinciding address) (refer to Fig. 8-1).
  When CA1, CA0 = 00: Sets $\overline{BUSY}$ output to low, if $\overline{CS}$=0. If $\overline{CS}$=1, sets $\overline{BUSY}$ output to high impedance.
  Other than 00 : Sets $\overline{BUSY}$ output to high impedance, regardless of $\overline{CS}$ input.
  In a single chip configuration, BUSY output operation is the same as that when CA1 and CA0 are 00.

- All processing operations (command/data processing, reading timing signal and display data to the row and column driver) are stopped.

- $V_{LC}3$ level DC current is output from each LCD drive signal output pin (C0-C41, R15/C42-R8/C49, R0/R8-R7/R15).

- The internal functions are set as follows (to the same conditions as when these commands are executed):
  SWM (I_1I_0=00) : Auto-increment mode
  LDPI (D_6-D_0=0000000): Data pointer is cleared to 0
  SMM (M_2-M_0=000) : 8-time-divisions, R0/R8-R7/R15 pins serve as R0-R7 pins, SYNC pin is set in the input mode, the data memory is set to bank 0.
  SFF (F_2-F_0=000) : Frame frequency is set to $f_{CL}/2^{14}$.

- The byte transfer end counter is cleared.

- If the μPD16434 is in the standby mode, the standby mode is maintained.

- The data memory contents become undefined.

When the high level input to the RESET pin is returned to low, the operation becomes possible, according to the initialized contents. In addition, the next processing will be performed at the falling edge of the RESET signal. The display output will be the same as when the DISP OFF is executed.

- The interface specification code (serial/ parallel specification, chip address selection function provided/unprovided) is read from the D2/(CAE) and D1/(P/$\overline{S}$) pins.

- A chip, whose CA1 and CA0 values are 0, becomes selected state.

- If a RESET is executed during the standby mode, the standby mode is cleared. In this case, the data memory contents are retained.

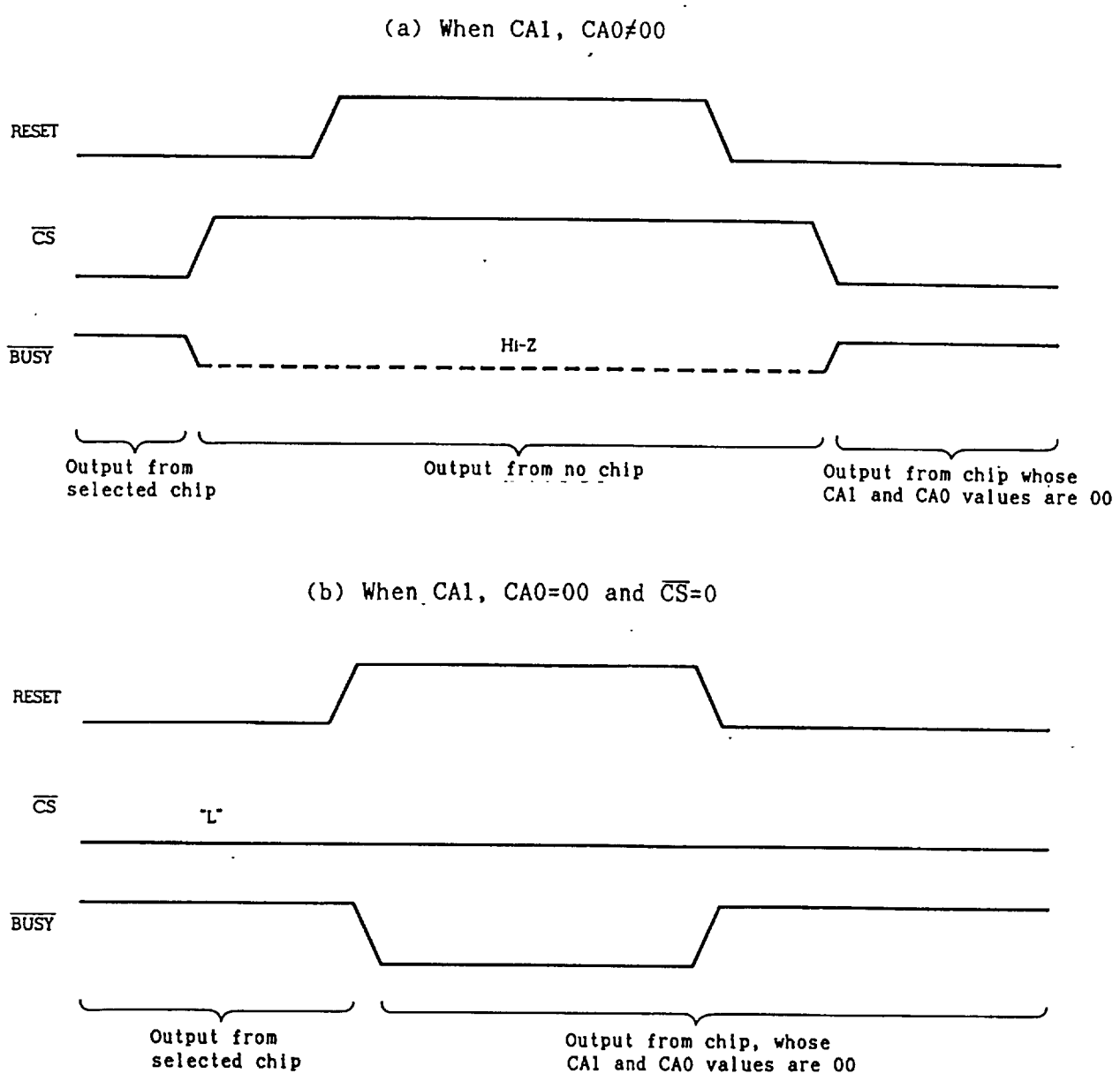


Fig. 8-1 Example of $\overline{\text{BUSY}}$ Output Timing Waveforms by RESET Input

9. COMMANDS

The μPD16434 offers the following 16 different commands, each consisting of 1 byte (8 bits):

Table 9-1 List of Commands

Mnemonic	Operation	Hexadecimal code
SFF	Set Frame Frequency	10-14
SMM	Set Multiplexing Mode	18-1F
DISP OFF	Display Off	08
DISP ON	Display On	09
LDPI	Load Data Pointer with Immediate	80-B1, C0-F1
SRM	Set Read Mode	60-63
SWM	Set Write Mode	64-67
SORM	Set OR Mode	68-6B
SANDM	Set AND Mode	6C-6F
SCML	Set Character Mode with Left entry	71
SCMR	Set Character Mode with Right entry	72
BRESET	Bit Reset	20-3F
BSET	Bit Set	40-5F
CLCURS	Clear Cursor	7C
WRCURS	Write Cursor	7D
STOP	Set Stop Mode	01

9.1 LCD Display Mode Setting Commands

The following commands are provided for LCD display mode setting:

SFF (Set Frame Frequency)
SMM (Set Multiplexing Mode)
DISP OFF (Display Off)
DISP ON (Display On)

(1) SFF (Set Frame Frequency)

0	0	0	1	0	F_2	F_1	F_0

This command sets the frame frequency. The frame frequency will be the clock frequency input from the CLOCK pin divided by the frequency dividing ratio specified by F_2-F_0.

F_2	F_1	F_0	Frame frequency
0	0	0	$f_{CL}/2^{14}$
0	0	1	$f_{CL}/2^{13}$
0	1	0	$f_{CL}/2^{12}$
0	1	1	$f_{CL}/2^{11}$
1	0	0	$f_{CL}/2^{10}$
1	0 to 1	1	These settings are not allowed

f_{CL}: Clock frequency

(2) SMM (Set Multiplexing Mode)

0	0	0	1	1	M_2	M_1	M_0

This command specifies the number of time divisions, and the functions of the row driver, and the row/column driver, input/output for the SYNC pin, and the data memory bank.

M_2	M_1	M_0	Number of time divisions	R0/R8-R7/R15	R15/C42-R8/C49	SYNC pin	Memory bank
0	0	0	8	R0-R7	C42-C49	Input	0
0	0	1					1
0	1	0				Output	0
0	1	1					1
1	0	0	16	R8-R15		Input	0, 1
1	0	1					
1	1	0		R0-R7		Output	
1	1	1			R15-R8		

(3) DISP OFF (Display Off)

0	0	0	0	1	0	0	0

This command sets the relationship of the row signal and column signal to non-select level, regardless of the display data, and deletes display.

(4) DISP ON (Display On)

0	0	0	0	1	0	0	1

When this command is executed, the display operation will be performed according to the display data.

9.2 Data Pointer Load Command

(1) LDPI (Load Data Pointer with Immediate)

1	D_6	D_5	D_4	D_3	D_2	D_1	D_0

D_{6-0}=00H-31H, 40H-71H

This command loads 7-bit immediate data D_6-D_0 to the data pointer.

9.3 Data Processing Mode Setting Commands

The following six different commands are available as the data processing mode setting commands.

SRM (Set Read Mode)
SWM (Set Write Mode)
SORM (Set OR Mode)
SANDM (Set AND Mode)
SCML (Set Character Mode with Left entry)
SCMR (Set Character Mode with Right entry)

Each of these commands sets the μPD16434 to the respective mode. Afterwards, the μPD16434 processes data in the specified mode until a command to set a different processing mode is executed. The lower 2 bits (I_1I_0) of these data processing mode setting commands specify the data pointer modification operation for each byte data processing.
The data pointer is modified as follows:

I_1	I_0	Data pointer contents of modification
0	0	Automatically incremented (+1) each time a byte of data is processed
0	1	Automatically decremented (-1) each time a byte of data is processed
1	0	This setting is not allowed
1	1	No modification (the same address is maintained)

(1) SRM (Set Read Mode)

0	1	1	0	0	0	I_1	I_0

The data processing mode is set to the read mode by this command. Afterwards, data processing will be performed in the read mode.

When this mode is set, the contents of the data memory, addressed by the current contents of the data pointer, are automatically transferred to the serial/parallel register. The data pointer is then modified according to I_1I_0.

When all 8 bits of the serial/parallel register contents are read by the CPU, the contents of the data memory, addressed by the modified data pointer, are automatically transferred into the serial/parallel register for the next read operation. Afterwards, the same operation is repeated by the CPU, each time an 8-bit data is read.

(2) SWM (Set Write Mode)

0	1	1	0	0	1	I_1	I_0

The data processing mode is set to the write mode by this command. Afterwards, data processing will be performed in the write mode.

When this mode is set, the 8-bit data written into the serial/parallel register by the CPU is stored into the data memory addressed by the current contents of the data pointer. The data pointer is then modified, according to

I_1I_0. Afterwards, the same operation is repeated, each time an 8-bit data is written by the CPU.

(3) SORM (Set OR Mode)

0	1	1	0	1	0	I_1	I_0

The data processing mode is set to the OR mode by this command. Afterwards, data processing will be performed in the OR mode.

When this mode is set, the 8-bit data, written into the serial/parallel register by the CPU, is ORed with the data memory addressed by the current contents of the data pointer, and the result will be stored into the same data memory address. The data pointer is then modified, according to I_1I_0. Afterwards, the same operation is repeated, each time an 8-bit data is written by the CPU.

(4) SANDM (Set AND Mode)

0	1	1	0	1	1	I_1	I_0

The data processing mode is set to the AND mode by this command. Afterwards, data processing will be performed in the AND mode.

When this mode is set, the 8-bit data, written into the serial/parallel register by the CPU, is ANDed with the data memory addressed by the current contents of the data pointer, and the result will be stored into the same data memory address. The data pointer is then modified, according to I_1I_0. Afterwards, the same operation is repeated, each time an 8-bit data is written by the CPU.

(5) SCML (Set Character Mode with Left entry)

0	1	1	1	0	0	0	1

The data processing mode is set to the character mode with left entry by this command. Afterwards, data processing will be performed in the character mode with left entry.
When this mode is set, the 8-bit data written into the serial/parallel register by the CPU is treated as ASCII or JIS code and is decoded to 5 x 7-bit character display data by the character generator. It is written into the lower five consecutive data memory addresses from the address indicated by the current contents of the data memory. As a result, the data pointer contents are subtracted by 5 (-5). Afterwards, the same operation is repeated, each time an 8-bit data is written by the CPU.

(6) SCMR (Set Character Mode with Right entry)

0	1	1	1	0	0	1	0

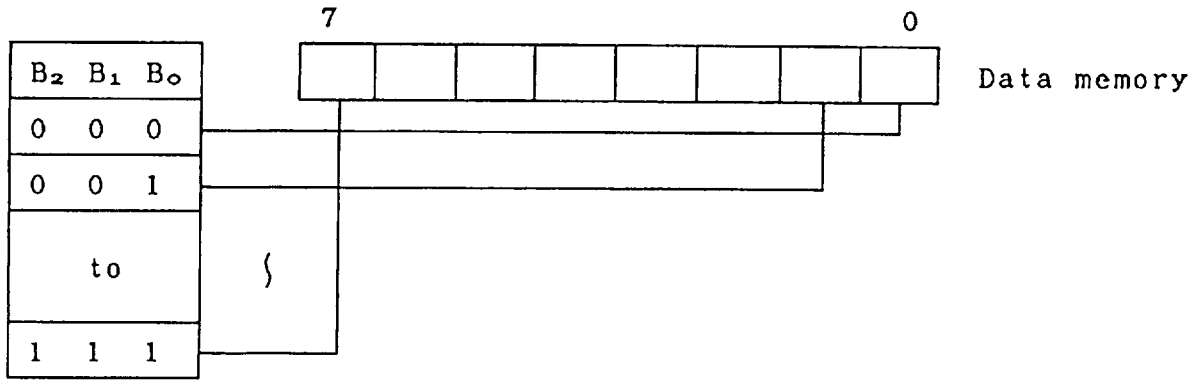
The data processing mode is set to the character mode with right entry by this command. Afterwards, data processing will be performed in the character mode with right entry.
When this mode is set, the 8-bit data written into the serial/parallel register by the CPU is treated as ASCII or JIS code and is decoded to 5 x 7-bit character display data by the character generator, and is written into the subsequent five data memory addresses from the address indicated by the current contents of the data memory. As a result, the data pointer contents are added by 5 (+5). Afterwards, the same operation is repeated, each time an 8-bit data is written by the CPU.

9.4 Memory Bit Manipulation Commands

The following four different memory bit manipulation commands are available:

BRESET (Bit Reset)
BSET (Bit Set)
CLCURS (Clear Cursor)
WRCURS (Write Cursor)

The BRESET and BSET commands can be executed in any data processing mode. After the bit specified by the BRESET/BSET command in the data memory addressed by the data pointer is set/reset, the data pointer is modified according to the lower 2 bits (J_1J_0) of the command byte.
The CLCURS or WRCURS command is used to clear or set the cursor (bit 7 position) in the character mode. After these commands are executed, the data pointer contents are added by 5 or subtracted by 5.
Data pointer manipulation by these memory bit manipulation commands is effective, only when these commands are executed. Afterwards, the data pointer is modified according to the data processing mode that has been set.
However, the data pointer contents are modified by the memory bit manipulation command. Therefore, the modified value will be used as the initial value for the subsequent modification operation.
The figure below shows the BRESET and BSET command bit specifications and modification contents for the data pointer.



J_1	J_0	Data pointer contents for modification
0	0	+1
0	1	-1
1	0	This setting is not allowed
1	1	No modification (the same address is maintained)

(1) BRESET (Bit Reset)

0	0	1	B_2	B_1	B_0	J_1	J_0

This commands resets (to 0) the bit specified by B_2-B_0 of the data memory addressed by the data pointer. Afterwards, the data pointer is modified according to J_1J_0.

(2) BSET (Bit Set)

0	1	0	B_2	B_1	B_0	J_1	J_0

This commands sets (to 1) the bit specified by B_2-B_0 of the data memory addressed by the data pointer. Afterwards, the data pointer is modified according to J_1J_0.

(3) CLCURS (Clear Cursor)

0	1	1	1	1	1	0	0

When this command is executed in the character mode, bit 7 of each data memory of the five subsequent addresses (SCMR mode) or the lower five consecutive addresses (SCML mode) from the address, indicated by the current contents of the data pointer, is reset (to 0).
This command can be used to clear the cursor displayed for 5 x 7-bit configuration character.

(4) WRCURS (Write Cursor)

0	1	1	1	1	1	0	1

When this command is executed in the character mode, bit 7 of each data memory of the five subsequent addresses (SCMR mode) or the lower five consecutive addresses (SCML mode) from the address, indicated by the current contents of the data pointer, is set (to 1).
This command can be used to display the cursor for 5 x 7-bit configuration character.

9.5 Standby Operation Setting Command

(1) STOP (Set Stop Mode)

0	0	0	0	0	0	0	1

This command sets the STOP mode (standby mode).

The data processing mode is initialized to the auto-increment (I_1I_0=00) write mode.
Other modes are not affected by this command execution.

10. SYSTEM CONFIGURATION EXAMPLE

10.1 Multi-Chip Configuration of μPD16434s

Figure 10-1 shows a circuit example, when four μPD16434s are used in a multi-chip system configuration.

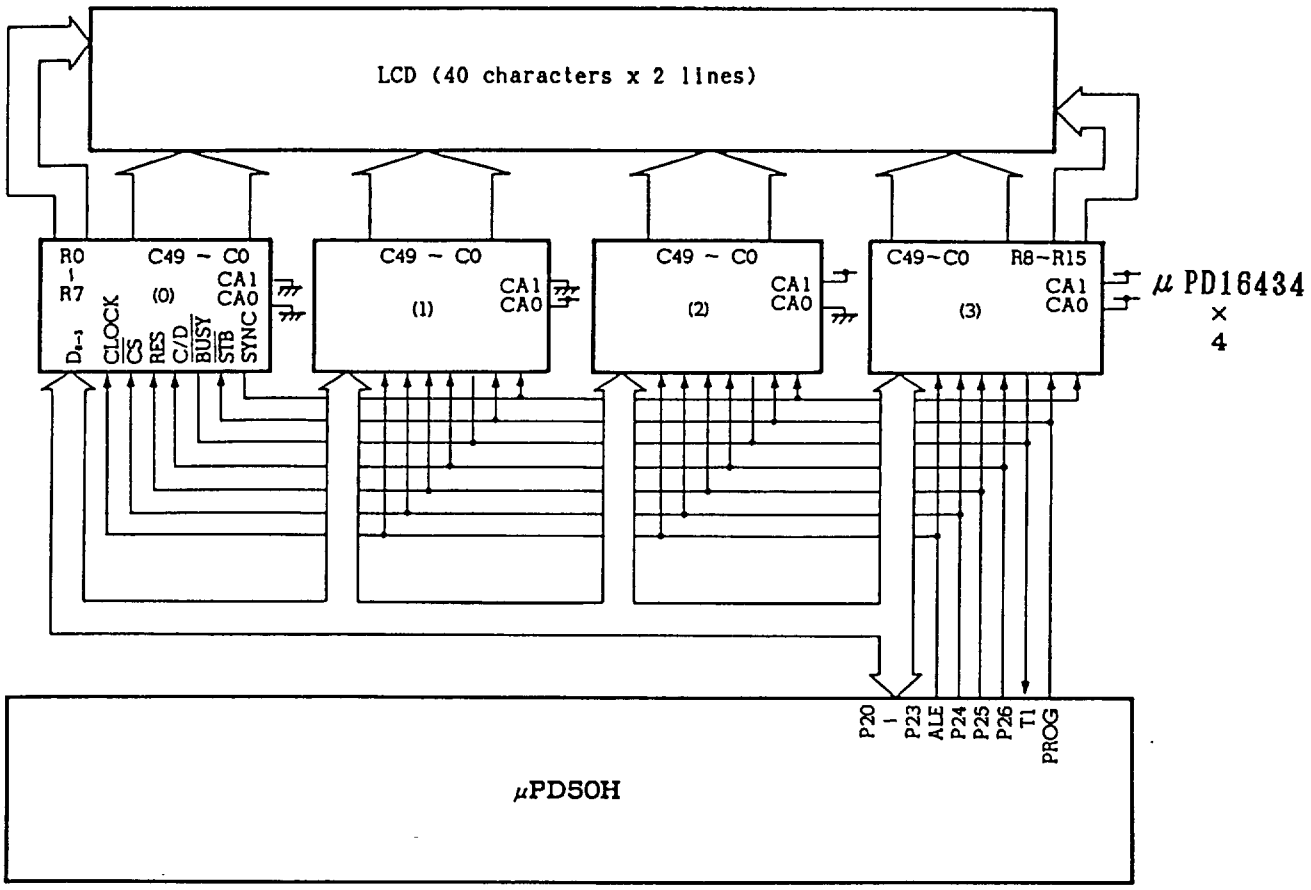


Fig. 10-1 System Configuration Example (Multi-Chip)

10.2 Notes on Interfacing with uPD40H and uPD50H

The interface mode specification data (P/$\overline{S}$) for the μPD16434 is read in synchronization with the falling edge of the RESET signal (when cleared). The specification level (high for parallel specification) must be maintained for at least 4us after the falling edge of the RESET signal.
Therefore, for a system using the external ROM with the uPD40H or uPD50H, this must be noted when parallel interfacing the μPD16434 with the uPD40H or uPD50H.

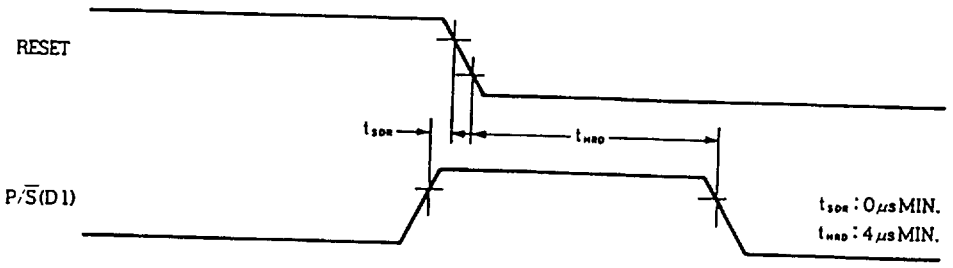


Fig. 10-2 RESET and P/S (D1) Timing Waveforms

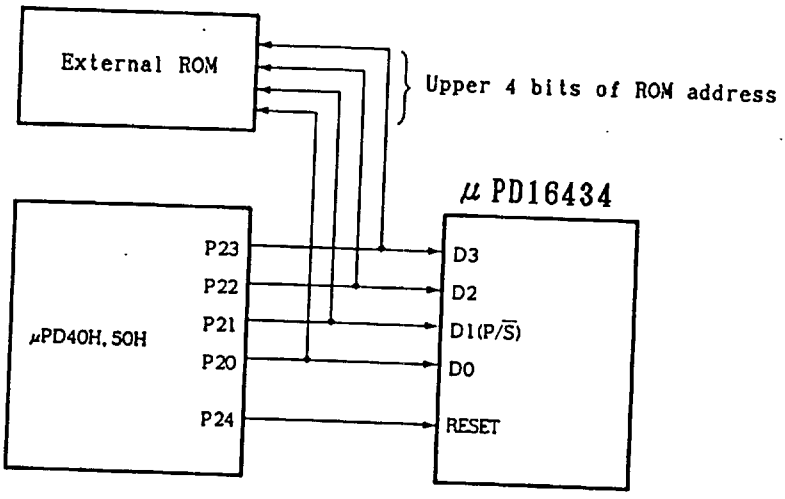


Fig. 10-3 Typical Connection When Using External ROM (Parallel Interface)

In these systems, data transfer between the μPD16434 and the uPD40H or uPD50H is performed through the lower 4 bits (P23-P20) of port 2 for the uPD40H or uPD50H. On the other hand, the contents of the upper 4 bits (PCH: PC11-PC8) of the ROM address are also output to the same pins in time-division. Therefore, if the external ROM access starts immediately after the RESET is cleared by a port 2 output instruction, the data of the lines (P21=D1), corresponding to the P/$\overline{S}$ of the upper 4 bits of the ROM address, must be held high, until t_{HRD}=4us is satisfied.

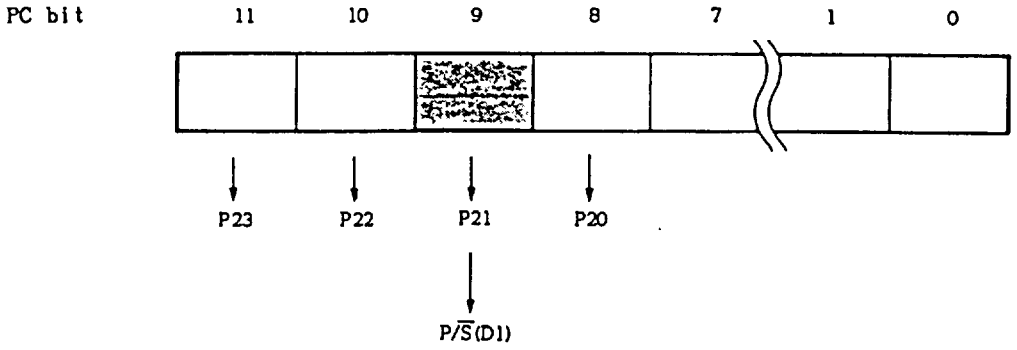


Fig. 10-4 Program Counter (PC) for uPD40H and uPD50H

For example, for the uPD40H (up to 4K bytes of external ROM can be accessed), execute a port 2 output instruction, by which the μPD16434 RESET is cleared within addresses 200H-3FFH, 600H-7FFH, A00H-BFFH, E00H-FFFH and a high level is input to the P/$\overline{S}$. No instruction (jump, call, return) which exceeds these address ranges should be executed until after 4us have elapsed. In addition, no port 2 output instruction, which changes the P/$\overline{S}$, should be executed during this period.

Table 10-1 uPD80C40H Addresses

Address	PC bit								
	11	10	9	8	7	6	.....	1	0
200H-2FFH	0	0	1	0	x	x	.....	x	x
300H-3FFH	0	0	1	1	x	x	.....	x	x
600H-6FFH	0	1	1	0	x	x	.....	x	x
700H-7FFH	0	1	1	1	x	x	.....	x	x
A00H-AFFH	1	0	1	0	x	x	.....	x	x
B00H-BFFH	1	0	1	1	x	x	.....	x	x
E00H-EFFH	1	1	1	0	x	x	.....	x	x
F00H-FFFH	1	1	1	1	x	x	.....	x	x

This is the address for 1

Remarks: x: don't care

11. ELECTRICAL SPECIFICATIONS

Absolute Maximum Rating (Ta=25℃)

Item	Symbol	Conditions	Rating	Unit
Power supply voltage	V_{DD}		-0.3 to +7.0	V
LCD drive voltage1 (Note)	V_{LCD1}	V_{DD}=5.0V	4.5 to 12.5	V
LCD drive voltage2 (Note)	V_{LCD2}	V_{DD}=3.0V	2.5 to 10.5	V
Input voltage	V_I		-0.3 to V_{DD}+0.3	V
Output voltage	V_O		-0.3 to V_{DD}+0.3	V
Operating temperature	T_A		-40 to 85	℃
Storage temperature	T_{stg}		-65 to 150	℃

Note : $V_{LCD} = V_{DD} - V_{LC5}$

DC Characteristics (Ta=-40 to +85°C, V_{DD}=5V±10%, V_{LC}5=-6.0V±10%)

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
Input voltage, high	V_{IH1}	Except $\overline{SCK}$	$0.7V_{DD}$		V_{DD}	V
	V_{IH2}	$\overline{SCK}$	$0.8V_{DD}$		V_{DD}	V
Input voltage, low	V_{IL}		0		$0.3V_{DD}$	V
Input leakage current, high	I_{LIH}	$V_I=V_{DD}$			10	uA
Input leakage current, low	I_{LIL}	V_I=0V			-10	uA
Output voltage, high	V_{OH1}	$\overline{BUSY}$, D0-D3 I_{OH}=-400uA	V_{DD}-0.5			V
	V_{OH2}	SYNC, I_{OH}=-100uA	V_{DD}-0.5			V
Output voltage, low	V_{OL1}	$\overline{BUSY}$, D0-D3 I_{OL}=1.7mA			0.5	V
	V_{OL2}	SYNC, I_{OL}=100uA			0.5	V
Output leakage current, high	I_{LOH}	$V_O=V_{DD}$			10	uA
Output leakage current, low	I_{LOL}	V_O=0V			-10	uA
Row output impedance	R_{ROW}			6	16	kΩ
Row/column output impedance	$R_{ROW/COL}$			7.5	20	kΩ
Column output impedance	R_{COL}			15	30	kΩ
Supply current	I_{DD1}	Operation mode, f_c=400kHz		250	600	uA
	I_{DD2}	STOP mode, CLK=0V			25	uA

Capacitance (Ta=25°C, V_{DD}=0V)

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
Input capacitance	C_{IN}	f=1MHz Unmeasured pins returned to 0V.			10	pF
Output capacitance	C_{OUT}				25	pF
I/O capacitance	C_{IO}				15	pF

AC Characteristics (Ta=-40°C to +85°C, V_{DD}=+5V±10%)

Common operation

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
Clock operation frequency	f_C		100		1100	kHz
Clock high-level pulse width	t_{WHC}		350			ns
Clock low-level pulse width	t_{WLC}		350			ns
RESET high-level width	t_{HRS}		4			us
$\overline{CS}$ ↓ → $\overline{BUSY}$ delay time	t_{DCSB}	C_L=50pF			3	us
$\overline{CS}$ ↑ → $\overline{BUSY}$ float delay time	t_{DCSBF}	C_L=50pF			5	us
$\overline{CS}$ high-level width	t_{WHCS}		4			us
SYNC load capacitance	C_{LSY}				100	pF
Data set time (RESET ↓)	t_{SDR}		0			us
Data hold time (RESET ↓)	t_{HRD}		5			us

Serial input/output operation

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
$\overline{SCK}$ period	t_{CYK}		0.9			us
$\overline{SCK}$ high-level pulse width	t_{WHK}		400			ns
$\overline{SCK}$ low-level pulse width	t_{WLK}		400			ns
High-level $\overline{SCK}$ hold time ($\overline{BUSY}$ ↑)	t_{HBK}		0			ns
SI set time ($\overline{SCK}$ ↑)	t_{SIK}		120			ns
SI hold time ($\overline{SCK}$ ↑)	t_{HKI}		270			ns
$\overline{SCK}$↓→SO delay time	t_{DKO}	C_L=50pF			350	ns
8th $\overline{SCK}$↑→$\overline{BUSY}$ delay time	t_{DKB}	C_L=50pF			4	us
$\overline{BUSY}$ low-level time	t_{WLB}	C_L=50pF	18		64	1/fc
C/$\overline{D}$ set time (first $\overline{SCK}$ ↓)	t_{SDK}		0			us
C/$\overline{D}$ hold time (8th $\overline{SCK}$ ↑)	t_{HKD}		3			us
$\overline{CS}$ hold time (8th $\overline{SCK}$ ↑)	t_{HKCS}		5			us

Parallel input/output operation

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
Command input set time ($\overline{STB}$ ↓)	t_A	C_L=80pF	120			ns
Command input hold time ($\overline{STB}$ ↓)	t_B	C_L=20pF	110			ns
Data input set time ($\overline{STB}$ ↑)	t_C	C_L=80pF	250			ns
Data input hold time ($\overline{STB}$ ↑)	t_D	C_L=20pF	70			ns
Data output delay time	t_{ACC}	C_L=80pF	90		750	ns
Data output hold time	t_H	C_L=20pF	0		150	ns
$\overline{STB}$ low-level pulse width	t_{SL}		700			ns
$\overline{STB}$ high-level time	t_{SH}		1			us
$\overline{STB}$ hold time ($\overline{BUSY}$ ↑)	t_{HBS}		0			us
2nd $\overline{STB}$↑→$\overline{BUSY}$ delay time	t_{DSB}				4	us
$\overline{BUSY}$ low-level time	t_{WLB}	C_L=50pF	18		64	$1/f_c$
C/$\overline{D}$ set time (first $\overline{STB}$ ↓)	t_{SDS}		0			us
C/$\overline{D}$ hold time (2nd $\overline{STB}$ ↑)	t_{HSD}		3			us
$\overline{CS}$ hold time (2nd $\overline{STB}$ ↑)	t_{HSCS}		3			us

DC CHARACSTICS

(UNLESS OTHERWISE SPECIFIED, TA=-40 to +85℃, VDD=3V±10%, VDD-VLC5=9V±10%)

ITEM	SYMBOL	CONDITIONS	MIN.	TYP.	MAX.	UNIT
High Level Input Voltage	V_{IH1}	Except SCK	0.7 V_{DD}		V_{DD}	V
High Level Input Voltage	V_{IH2}	SCK	0.8 V_{DD}		V_{DD}	V
Low Level Input Voltage	V_{IL}		0		0.3 V_{DD}	V
High Level Input Leakage Current	I_{LIH}	$V_I=V_{DD}$			10	μA
Low Level Input Leakage Current	I_{LIL}	$V_I=0V$			-10	μA
High Level Output Voltage	V_{OH1}	BUSY, D0-D3, I_{OH}=-100 μA	V_{DD}-0.5			V
High Level Output Voltage	V_{OH2}	SYNC, I_{OH}=-100 μA	V_{DD}-0.5			V
Low Level Output Voltage	V_{OL1}	BUSY, D0-D3, I_{OL}=500 μA			0.5	V
Low Level Output Voltage	V_{OL2}	SYNC, I_{OL}=100 μA			0.5	V
High Level Output Leakage Current	I_{LOH}	$V_O=V_{DD}$			10	μA
Low Level Output Leakage Current	I_{LOL}	$V_O=0V$			-10	μA
Low Output Impedance	R_{ROW}			8		kΩ
Low/Column Output Impedance	$R_{ROW/COL}$			10		kΩ
Column Output Impedance	R_{COL}			20		kΩ
Supply Current	I_{DD1}	Operation mode, fc=400kHz		150	250	μA
Supply Current	I_{DD2}	STOP mode, CLK=0V			20	μA

CAPACITANCE (T_A=25℃, V_{DD}=0 V)

ITEM	SYMBOL	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Input Capacitance	C_{IN}	f=1MHz			10	pF
Output Capacitance	C_{OUT}	with pins other than that			25	pF
Input/Output Capacitance	C_{IO}	measured at 0V			15	pF

AC CHARACTERISTICS (UNLESS OTHERWISE SPECIFIED, T_A=−40to+85°C、V_{DD} =+3 V±10%)

Common Operations

ITEM	SYMBOL	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Clock Operation Frequency	f_C		20		800	kHz
High Level Clock Pulse Width	t_{WHC}		350			ns
Low Level Clock Pulse Width	t_{WLC}		350			ns
High Level RESET Width	t_{HRS}		4			μs
CS↓→ BUSY Delay Time	t_{DCSB}	C_L=50 pF			4	μs
CS↑→ BUSY Float Delay Time	t_{DCSFB}	C_L=50 pF			6	μs
High Level CS Width	t_{WHCS}		10			μs
SYNC Load Capacitance	C_{LSY}				100	pF
Data Set Time (vs.RESET↓)	t_{SDR}		0			μs
Data Hold Time (vs.RESET↓)	t_{HRD}		10			μs

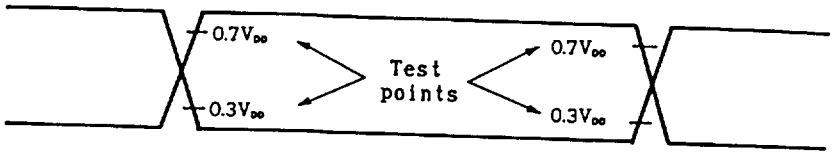
Serial Input/Output Operations

ITEM	SYMBOL	CONDITIONS	MIN.	TYP.	MAX.	UNIT
SCK Cycle	t_{CYK}		1			μs
High Level SCK Pulse Width	t_{WHK}		450			ns
Low Level SCK Pulse Width	t_{WLK}		450			ns
High Level SCK Hold Time (vs.BUSY↑)	t_{HBK}		0			ns
SI Set Time (vs.SCK↑)	t_{SIK}		200			ns
SI Hold Time (vs.SCK↑)	t_{HKI}		500			ns
SCK↓→ SO Delay Time	t_{DKO}	C_L=50 pF			400	ns
8th SCK↑→ BUSY Delay Time	t_{DKB}	C_L=50 pF			5	μs
Low Level BUSY Time	t_{WLB}	C_L=50 pF	18		64	1/f_C
C/D Set Time (vs.1st SCK↓)	t_{SDK}		0			μs
C/D Hold Time (vs.8th SCK↑)	t_{HKD}		4			μs
CS Hold Time (vs.8th SCK↑)	t_{HKCS}		6			μs

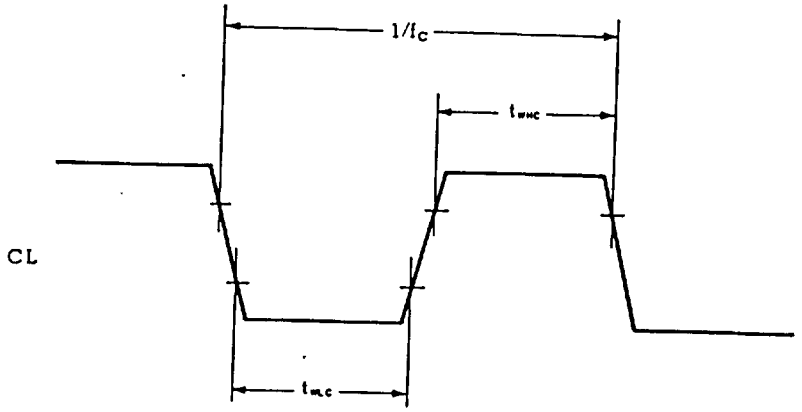
Parallel Input/Output Operations

ITEM	SYMBOL	CONDITIONS	MIN.	TYP.	MAX.	UNIT
Command Input Set Time (vs.STB↓)	t_A	C_L=80 pF	200			ns
Command Input Hold Time (vs.STB↓)	t_B	C_L=20 pF	180			ns
Data Input Set Time (vs.STB↑)	t_C	C_L=80 pF	450			ns
Data Input Hold Time (vs.STB↑)	t_D	C_L=20 pF	100			ns
Data Output Delay Time	t_{ACC}	C_L=80 pF	200		2000	ns
Data Output Hold Time	t_H	C_L=20 pF	0		900	ns
Low Level STB Pulse Width	t_{SL}		2000			ns
High Level STB Time	t_{SH}		3			μs
STB Hold Time (vs.BUSY↑)	t_{HBS}		0			μs
2nd STB↑→BUSY Delay Time	t_{DSB}				5	μs
Low Level BUSY Time	t_{WLB}	C_L=50 pF	18		64	1/fc
C/D Set Time (vs.1st STB↓)	t_{SDS}		0			μs
C/D Hold Time (vs.2nd STB↑)	t_{HSD}		4			μs
CS Hold Time (vs.2nd STB↑)	t_{HSCS}		4			μs

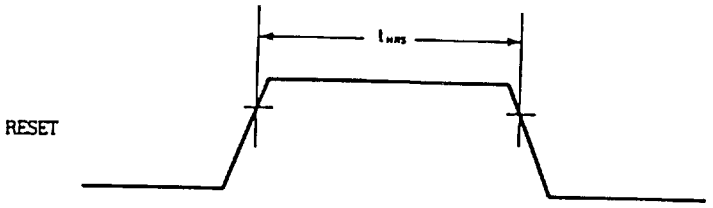
AC timing measurement voltages (except $\overline{STB}/\overline{SCK}$, $\overline{BUSY}$)



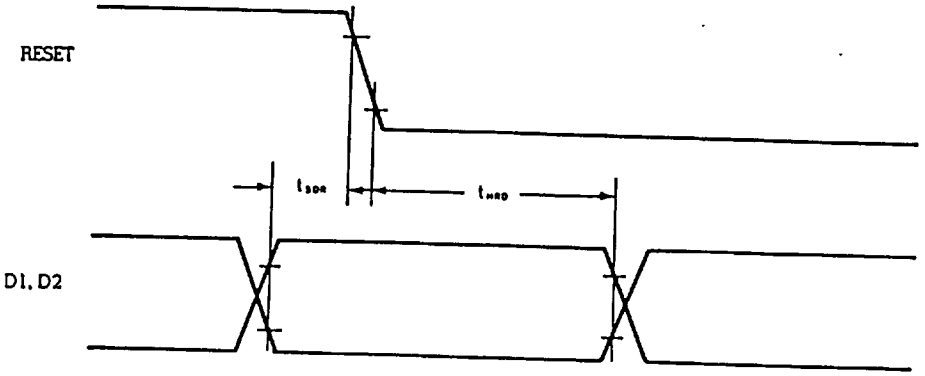
Clock timing waveforms



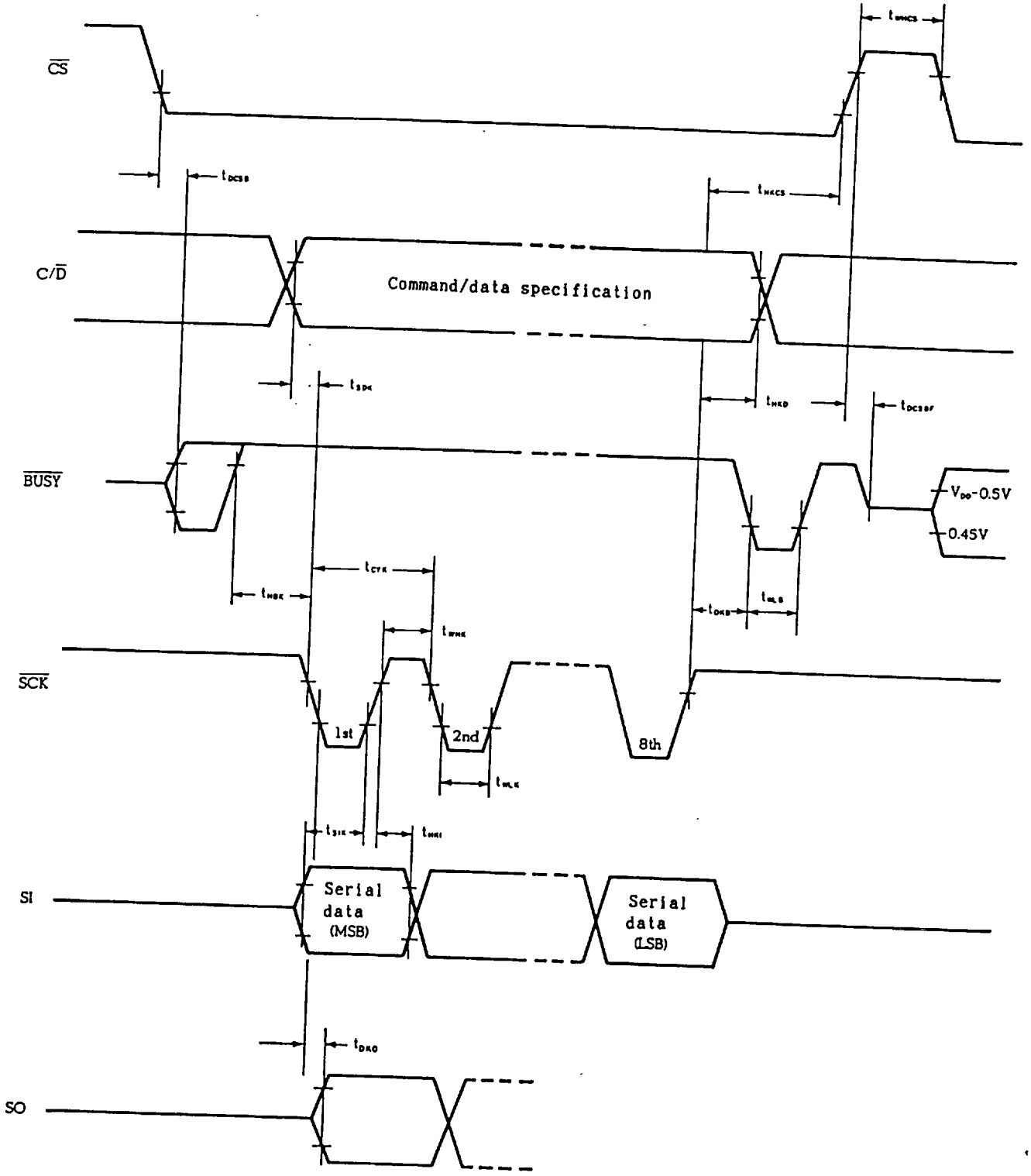
RESET input timing waveforms



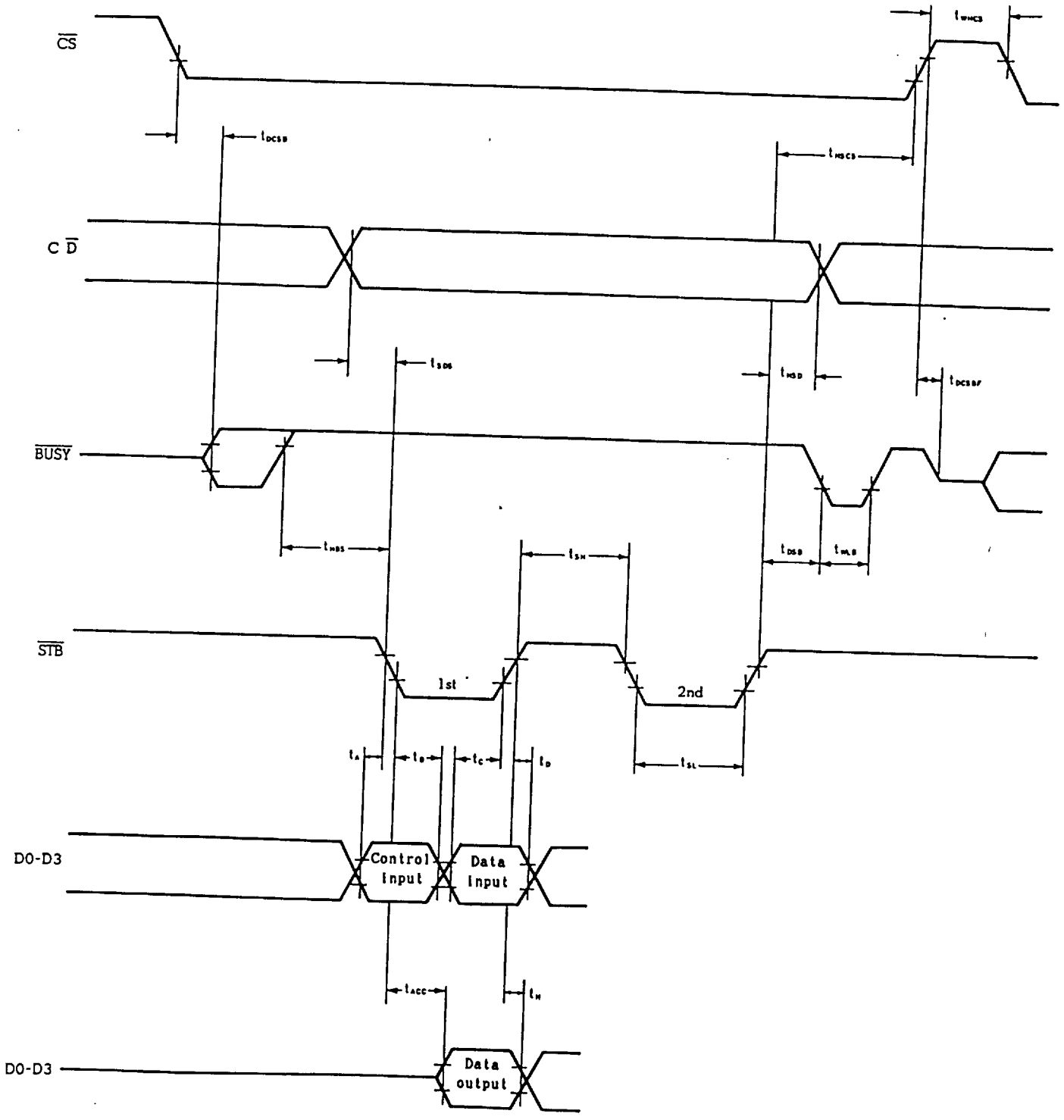
Interface specification timing waveforms



Serial input/output timing waveforms



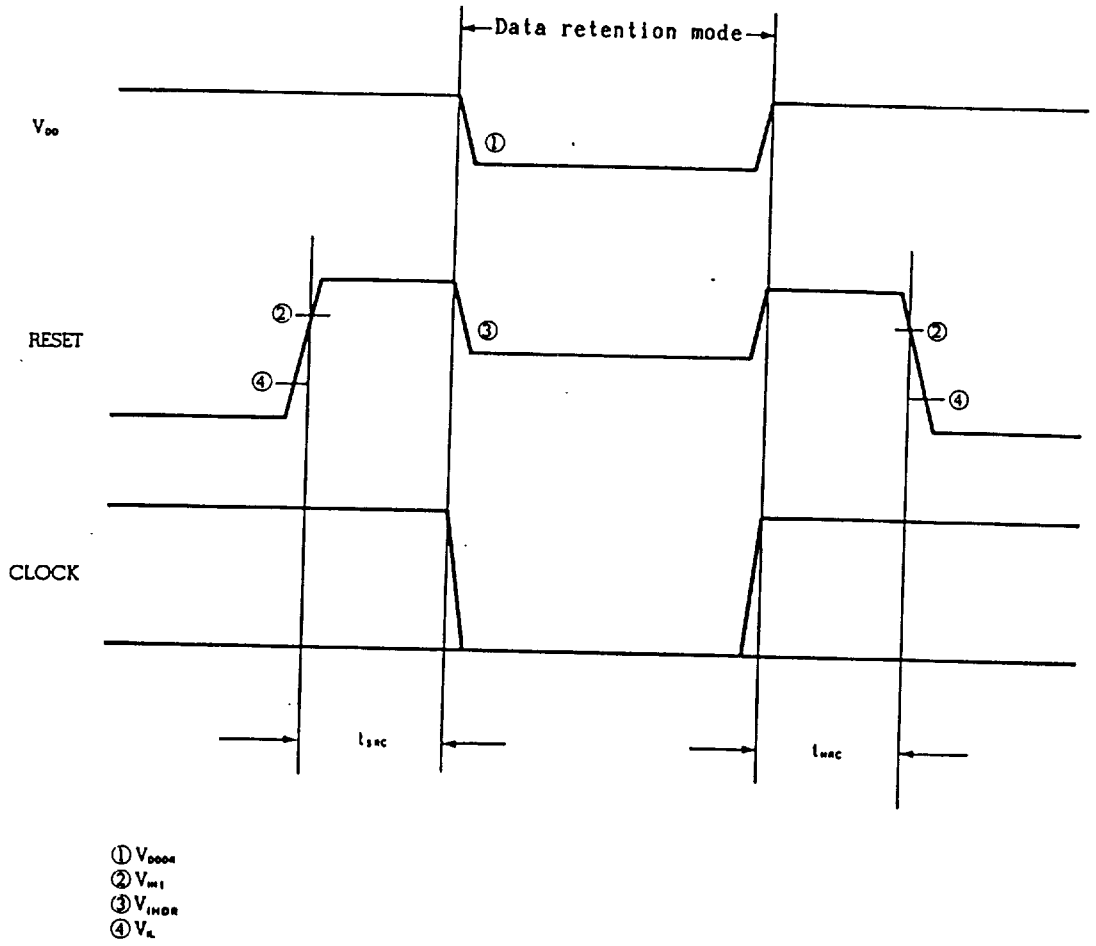
Parallel input/output timing waveforms



Data Memory STOP Mode Low Power Supply Voltage Data Retention Characteristics (Ta=-40 to +85°C)

Item	Symbol	Conditions	MIN	TYP	MAX	Unit
Data retention power supply voltage	V_{DDDR}		2.0			V
Data retention power supply current	I_{DDDR}	V_{DDDR}=2.0V			20	uA
Data retention high level RESET input voltage	V_{IHDR}		0.9V_{DDDR}		V_{DDDR}+0.2	V
RESET, CLOCK setup time	t_{SRC}		10			us
RESET, CLOCK hold time	t_{HRC}		10			us

Data retention timing waveforms

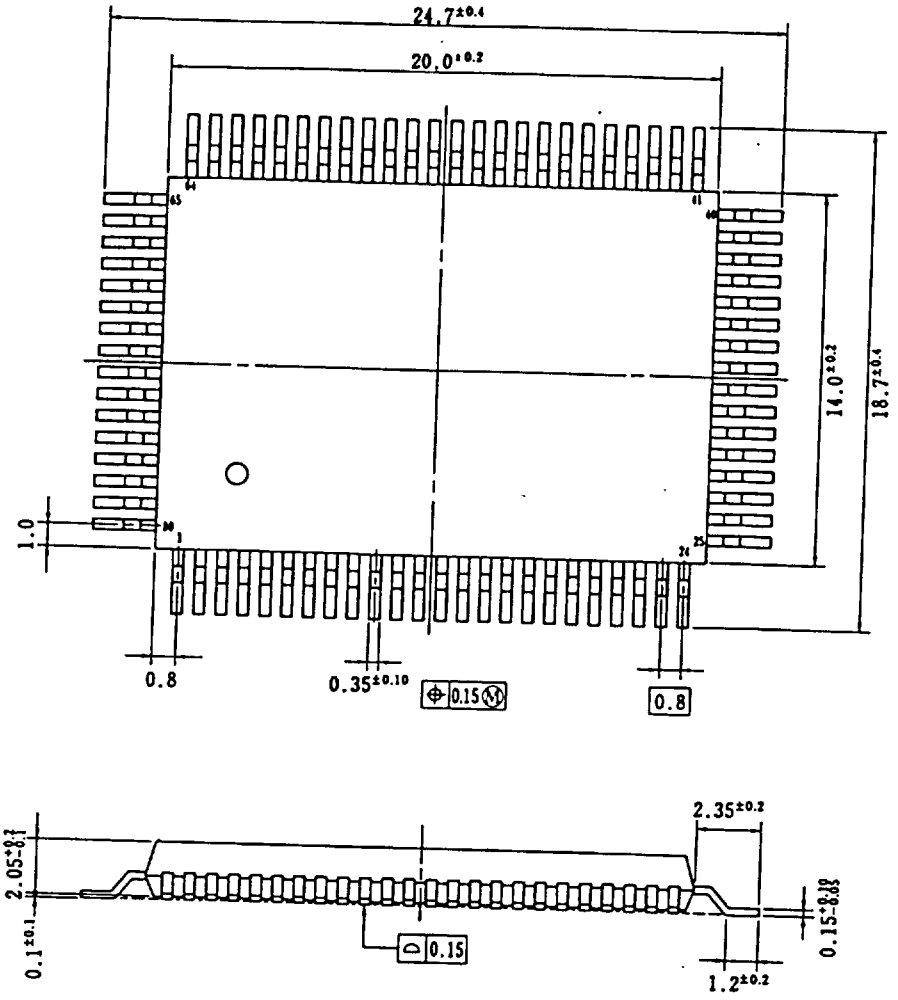


①V_{DDDR}
②V_{IH1}
③V_{IHDR}
④V_{IL}

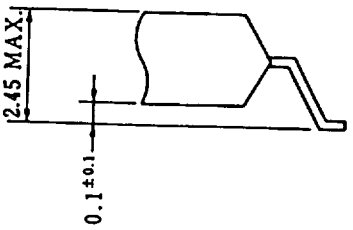
Notes: All inputs must be set below V_{DDDR} in the data retention mode.

12. PACKAGE DRAWINGS

80-pin plastic QFP (14 x 20) (Unit: mm)



Enlarged detail of lead end



P80G-80-12-1

REFERENCE DOCUMENTS

Document Name	No.
Semiconductor device mounting technology manual	IEI-1207
NEC semiconductor device reliability/quality control system	IEI-1212

MEMO