APPLICATION NOTE

NEC

μPD7225

PROGRAMMABLE LCD CONTROLLER/DRIVER

Document No. IEA-1254A (2nd edition)
(O. D. No. IEM-745E)
Data Published March 1997 P
Printed in Japan

The application circuits and their parameters are for reference only and are not intended for use in actual design-ins.

Main revisions in this edition

Pages	Description
Chapter 2	• μPD7507 μPD7507B • Deletion of description related to μCOM-43NA

CONTENTS

CHAPTER 1 GENERAL

1.1 General

μPD7225 is an LCD (Liquid Crystal Display) controller/driver which is programmable by software. Figure1-1. shows the pin configurations for μPD7225. Figure 1-2. shows a functional block diagram. The μPD7225 interfaces with the CPU through a serial ports, in a microcomputer application system, for directly controlling LCD by static 2-, 3- or 4-time sharing. It also incorporates a segment decoder for generating specific segment patterns and can also control blinking operations.

Figure1-1. Pin Configurations

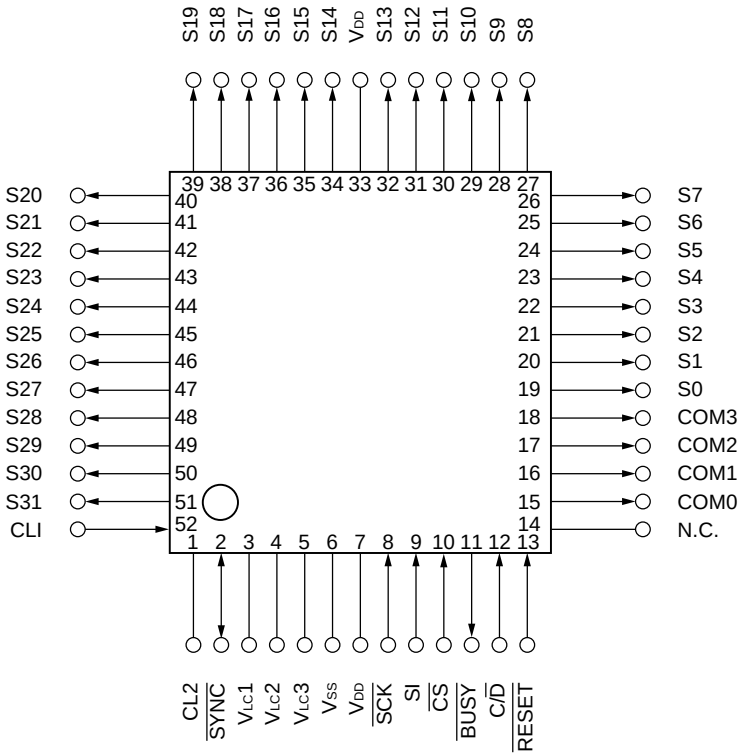
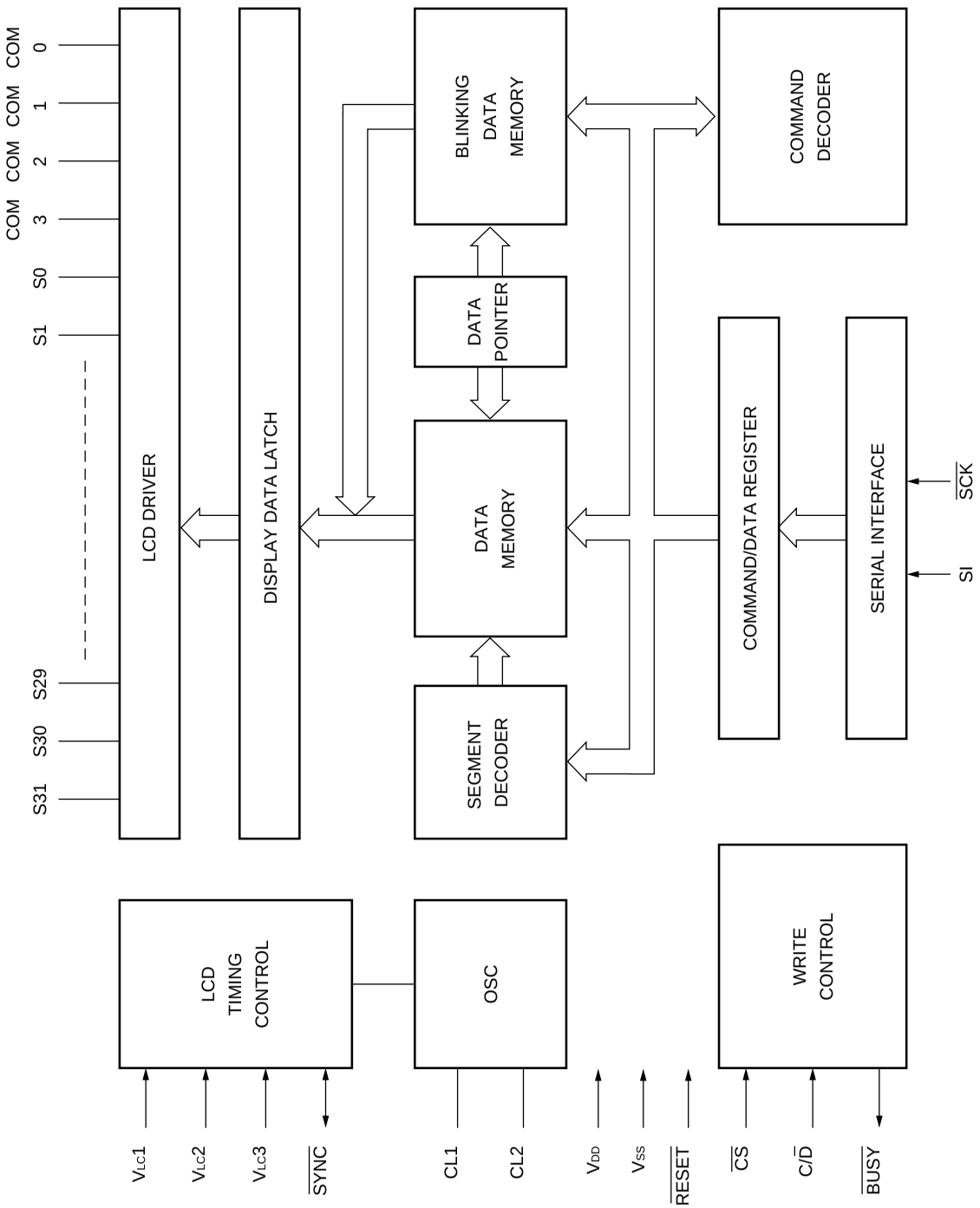


Figure 1-2. Block Diagram



1.2 Internal Segment Decoder

μPD7225 incorporates 7-segment type and 14-segment type decoders, taking in serial data at the SI pin, and generating patters as shown in Figure 1-4. and Figure 1-5.

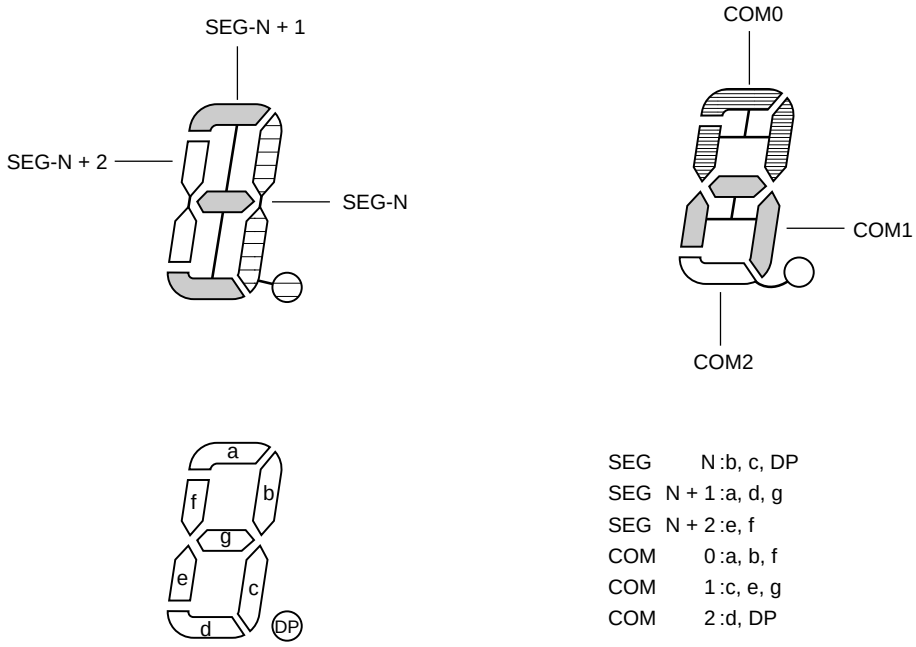
1.2.1 7-segment decoder

A 7-segment decoder, which performs 3- or 4-time sharing drive can generate numeric characters 0 to 9, five kinds of sings and blank display codes.

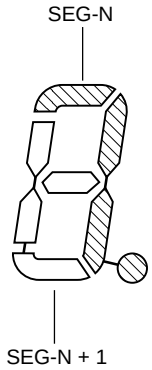
When the LCD display is made with the segment decoder output (display code), the LCD configuration shown in Figure 1-3. should be used. With LCD that is not configured like this, different display patterns are appeared.

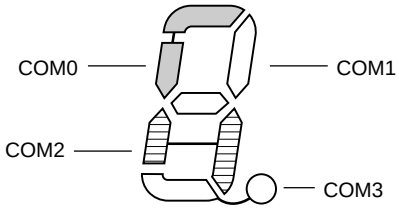
Figure 1-3. 7-segment Type LCD

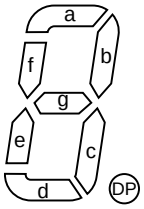
Make the connection in the following configuration for a 3-time sharing type of LCD.



In the case of 4-time sharing, make the following connection.

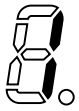


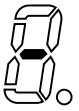
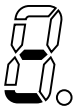
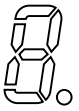




SEG	N	:a, b, c, DP
SEG	N + 1	:d, e, f, g
COM	0	:a, f
COM	1	:b, g
COM	2	:c, e
COM	3	:d, DP

Figure 1-4. 7-Segment

Data (HEX)	Display pattern	Data memory				
		3-time sharing			4-time sharing	
		N + 2	N + 1	N	N + 1	N
00		3	5	3	D	7
01		0	0	3	0	6
02		2	7	1	E	3
03		0	7	3	A	7
04		1	2	3	3	6
05		1	7	2	B	5
06		3	7	2	F	5
07		0	1	3	0	7

Data (HEX)	Display pattern	Data memory				
		3-time sharing			4-time sharing	
		N + 2	N + 1	N	N + 1	N
08		3	7	3	F	7
09		1	7	3	B	7
0A		0	2	0	2	0
0B		3	7	0	F	1
0C		3	5	0	D	1
0D		0	6	0	A	0
0E		2	6	2	E	4
0F		0	0	0	0	0

1.2.2 14-segment decoder

A 14-segment decoder, which performs 4-time sharing drive, can generate 36 kinds of alphanumeric chracters, 12 kinds of signs and blank display codes.

Figure 1-5. 14-Segment Type

In the case of a 14-segment type, only the 4-time sharing can be used. The relation between SEGMENT and COMMON is realized by the following connection.

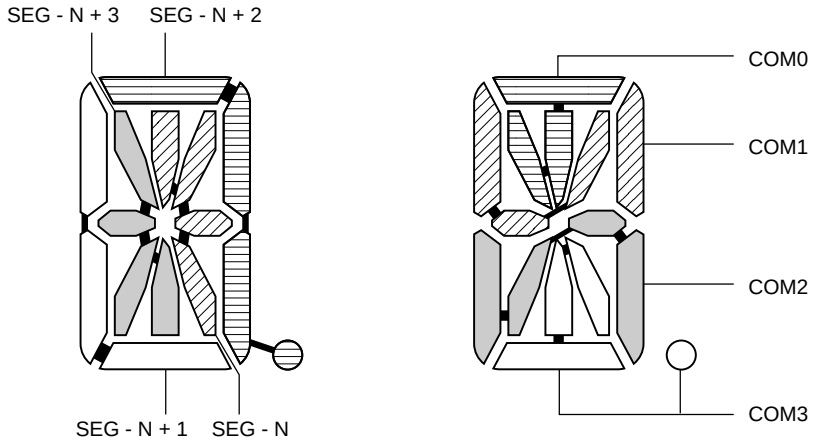
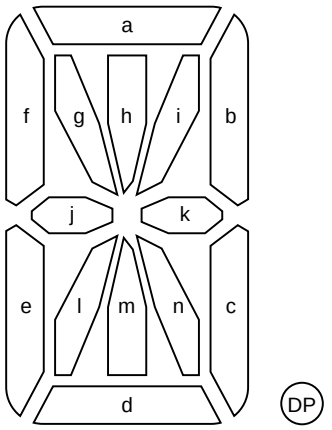


Figure 1-6. 14-Segment Type LCD



SEG N + 2 :a, b, c, DP
SEG N :h, i, j, k, n
SEG N + 1 :d, e, f
SEG N + 3 :g, j, l, m
COM 0 :a, g, h
COM 1 :b, i, j, f
COM 2 :c, e, k, l
COM 3 :d, m, n, DP

Shown next is the configuration of input data, display pattern and the display data written in the data memory. In the case of a 7-segment type, the 4 least significant bits (D3 to D0) are decoded, whereas the 7 least significant bits (D6 to D0) are decoded in the 14-segment type. In this case, the input data and display pattern of the 14-segment type correspond to the 8-bit ASCII code. The beginning write address for the display data should be Address N.

Figure 1-7. 14-Segment

Most significant bits

Least significant bits

Data (HEX)	A		B		C		D	
	Display pattern	Data memory N+3 N+2 N+1 N	Display pattern	Data memory N+3 N+2 N+1 N	Display pattern	Data memory N+3 N+2 N+1 N	Display pattern	Data memory N+3 N+2 N+1 N
0		0 0 0 0		4 7 E 2		A 7 C 0		2 3 6 4
1		0 0 0 0		0 6 0 0		2 7 6 4		0 7 E 8
2		0 0 0 0		2 3 C 4		8 7 8 5		2 3 6 C
3		0 0 0 0		2 7 8 4		0 1 E 0		1 5 8 4
4		0 0 0 0		2 6 2 4		8 7 8 1		8 1 0 1
5		0 0 0 0		2 5 A 4		2 1 E 4		0 6 E 0
6		0 0 0 0		2 5 E 4		2 1 6 4		4 0 6 2
7		0 0 0 2		0 7 0 0		0 5 E 4		4 6 6 8
8		0 0 0 A		2 7 E 4		2 6 6 4		5 0 0 A
9		5 0 0 0		2 7 A 4		8 1 8 1		9 0 0 2
A		F 0 0 F		0 0 0 0		0 6 C 0		4 1 8 2
B		A 0 0 5		0 0 0 0		2 0 6 A		0 0 0 0
C		0 0 0 0		4 0 8 2		0 0 E 0		1 0 0 8
D		2 0 0 4		2 0 8 4		1 6 6 2		0 0 0 0
E		0 0 0 0		1 0 8 8		1 6 6 8		0 0 0 0
F		4 0 0 2		0 0 0 0		0 7 E 0		0 0 0 0

1.2.3 The Input of serial data

Serial data is synchronized by the serial clock in units of 8 bits and inputted to the SI pin at the top of MSB. As $\overline{BUSY}$ becomes low when $\overline{CS}$ becomes low, synchronization is made with SCK when the $\overline{BUSY}$ signal becomes high after internal processing (SCK counter and data pointer are cleared), and the first bit (MSB) is transferred. Serial data is transferred to the serial register in units of 1 bit by the rise of $\overline{SCK}$. Inputting sight serial clock transitions causes all 8-bit data to be transferred to the serial register. Upon the rise of the 8th serial clock $\overline{BUSY}$ becomes low, the state of C/$\overline{D}$ pin is fetched, and it is determined whether the 8-bit data is a command or data.

Then, the contents of the serial register are fetched by the command/data register.

When two bytes or more of serial data are inputted continuously, $\overline{CS}$ should be left low until the input of all the bytes is completed. Every time the input of one byte is completed, $\overline{BUSY}$ becomes low. As $\overline{BUSY}$ becomes high when serial data is fetched by the command/data register from the serial register, the next serial data can be inputted.

When $\overline{CS}$ is raised after the input of all serial data is completed, the contents of the data memory are displayed.

Do not raise $\overline{CS}$ while a byte is being transferred (i. e. in the state where the serial clock is not input through eight transitions).

When it is necessary to make the temporarily stop transfer due to a CPU interruption while several bytes are being transferred, make $\overline{CS}$ high after executing the PAUSE TRANSFER command. IF $\overline{CS}$ is made high in this case, transfer is not made from the data memory to the data display latch.

To start the transfer of the serial data again, $\overline{CS}$ is made low like an ordinary transfer start. In this case, however, only the SCK counter is cleared and the data pointer keeps the contents before interruption. Therefore, when the start of the next serial data starts, the transferred data is processed as the subsequent data.

Figure 1-8. One-byte Input

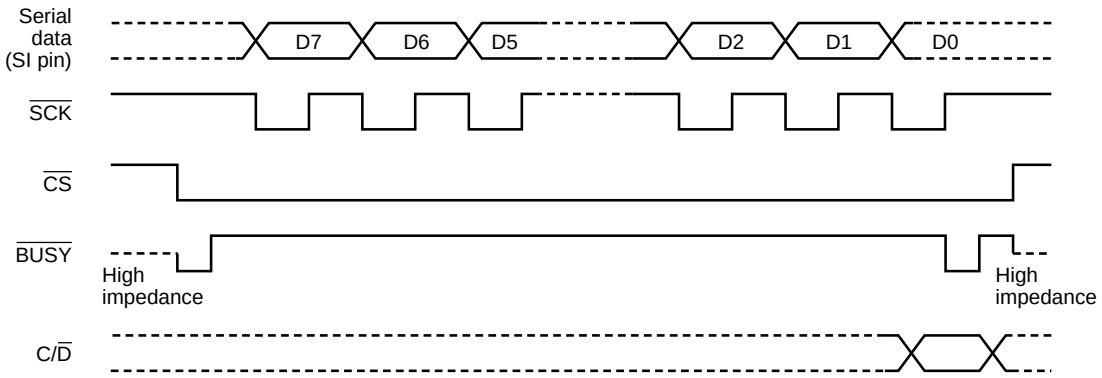
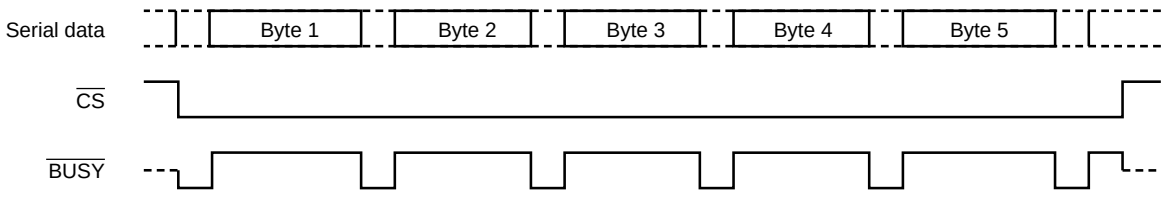


Figure 1-9. The Continuous Input of 5 Bytes



1.3 The Commands of μPD7225

1.3.1 MODE SET

0	1	0	M2	M1	M0	F1	F0

This command sets the time sharings number for LCD display, the bias method and frame frequency.

(a) M1 and M0 specify time sharing.

M1	M0	
0	0	---------- 4-time sharing drive
1	0	---------- 3-time sharing drive
1	1	---------- 2-time sharing drive
0	1	---------- Static drive

(b) M2 specifies the bias method.

M2

0 ------- 1/3 bias method

1 ------- 1/2 bias method

1/1 ------ Static

(c) F1 and F0 sets the frame frequency.

F1	F0		Scale ratio
0	0	----------	2^7
0	1	----------	2^8
1	0	----------	2^9
1	1	----------	2^{11}

1.3.2 SYNCHRONIZED TRANSFER

0	0	1	1	0	0	0	1

This command controls the re-writing of display data.

Raising the $\overline{CS}$ signal usually re-writes the display data (i. e. the transfer of display data from the data memory to the display data latch). After this command is executed, the display data is re-written at the beginning of the alternating drive period (frame frequency times the number of time sharings) when the $\overline{CS}$ signal is raised.

1.3.3 UNSYNCHRONIZED TRANSFER

0	0	1	1	0	0	0	0

This command controls the re-writing of display data.

After this command is executed, the display data is re-written upon raising the CS pin.

1.3.4 PAUSE TRANSFER

0	0	1	1	1	0	0	0

This commands inhibits the re-writing of display data.

After this command is executed, data displayed at the rise of the first $\overline{CS}$ pin is not re-written, but it is retained until the second $\overline{CS}$ pin is raised. Furthermore, raising the first $\overline{CS}$ pin does not clear the data pointer.

This command is used when the CS pin must be raised temporarily due to a CPU interrupt that occurred while the serial data is being inputted.

1.3.5 BLINKING ON

0	0	0	1	1	0	1	K0

This command sets the blinking state. The least significant bit K0 is used to set the blinking frequency.

K0
0 ---------- fosc/2^{17} (Hz)
1 ---------- fosc/2^{16} (Hz) fosc: Oscillation frequency

1.3.6 BLINKING OFF

0	0	0	1	1	0	0	0

Execution of this command stops the blinking operation.

1.3.7 DISPLAY ON

0	0	0	1	0	0	0	1

After this command is executed, the LCD display begins according to the display data of the display data latch.

1.3.8 DISPLAY OFF

0	0	0	1	0	0	0	0

Execution of this command makes non-selective the relation between all the common drive signals and the segment drive signals.

As a result, the display is extinguished. This command does not affect the transfer of display data from the data memory to the display data latch.

1.3.9 WITH SEGMENT DECODER

0	0	0	1	0	1	0	1

Data inputted after this command is executed is sent to the segment decoder, and the code obtained by the segment decoder is written into the data memory.

1.3.10 WITHOUT SEGMENT DECODER

0	0	0	1	0	1	0	0

Data inputted after this command is executed is written into the data memory without passing the segment decoder.

1.3.11 LOAD DATA POINTER

1	1	1	D4	D3	D2	D1	D0

This command sets the immediate data D4-D0 in the data pointer.

1.3.12 WRITE DATA MEMORY

1	1	0	1	D3	D2	D1	D0

This command stores the immediate data D3-D0 in the data memory addressed by the data pointer, incrementing by (+1) the contents of the data pointer.

1.3.13 OR DATA MEMORY

1	0	1	1	D3	D2	D1	D0

The contents of the data memory addressed by the data pointer and the immediate data D3-D0 are ORed, and the result is stored in the data memory. The contents of the data pointer are incremented by (+1).

1.3.14 AND DATA MEMORY

1	0	0	1	D3	D2	D1	D0

The contents of the data memory addressed by the data pointer and the immediate data D3-D0 are ANDed; the result is stored in the data memory. The contents of the data pointer are incremented by (+1).

1.3.15 CLEAR DATA MEMORY

0	0	1	0	0	0	0	0

This command clears the contents of the data memory and the data pointer.

1.3.16 WRITE BLINKING DATA MEMORY

1	1	0	0	D3	D2	D1	D0

This command stores the immediate data D3-D0 in the blinking data memory addressed by the data pointer, incrementing the contents of the data pointer by (+1).

1.3.17 OR BLINKING DATA MEMORY

1	0	1	0	D3	D2	D1	D0

The contents of the blinking data memory addressed by the data pointer and the immediate data D3-D0 are ORed, the result is stored in the blinking data memory. The contents of the data pointer are incremented by (+1).

1.3.18 AND BLINKING DATA MEMORY

1	0	0	0	D3	D2	D1	D

The contents of the blinking data memory addressed by the data pointer and the immediate data D3-D0 are ANDed, the result is stored in the blinking data memory. The contents of the data pointer are incremented by (+1).

1.3.19 CLEAR BLINKING DATA MEMORY

0	0	0	0	0	0	0	0

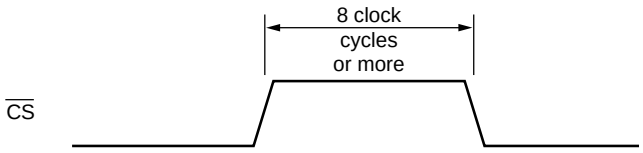
This command clears the contents of the blinking data memory and the data pointer.

[MEMO]

CHAPTER 2 EXAMPLES OF APPLICATIONS

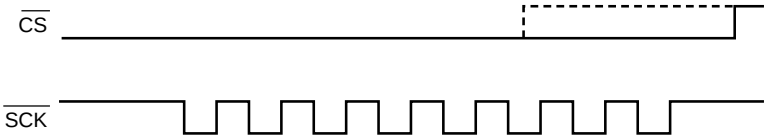
2.1 Matters Attended to in a Program

(a) The high-level signal to the $\overline{CS}$ pin must have 8 clock cycles or more at the clock frequency.



(b) When the data is transferred, be sure to check the $\overline{BUSY}$ pin to confirm whether or not the data is transferable.

(c) Be sure not to raise the chip select signal while the data is being transferred. Raising it during transfer may cause a malfunction.

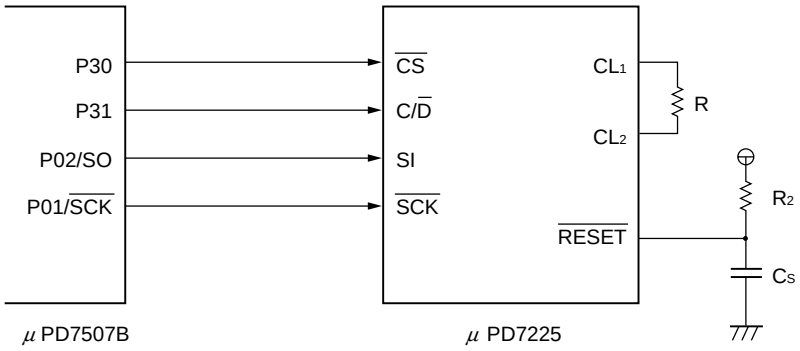


2.2 System Using μPD7507B

This section introduces an example where a CMOS 4-bit one-chip microcomputer (μPD7507B) is used as the control system for the μPD7225. In this example, it is presumed that the display panel which uses a 14-segment 8-digit LCD panel writes the data in the mode to use the segment decoder. In addition, it is presumed that this program does not test the $\overline{BUSY}$ signal outputted by μPD7225, and that the serial data is sent.

2.2.1 Interface with μPD7507B and μPD7225

Figure 2-1. Interface with μPD7507B



Shown below are the pin functions of μPD7507B in this interface.

P30: Used for the chip Select signal.

P31: Specifies command/data for the serial data to be written.

P02/SO: Used for serial data output (command/data).

P01/$\overline{SCK}$: Used for the serial clock output.

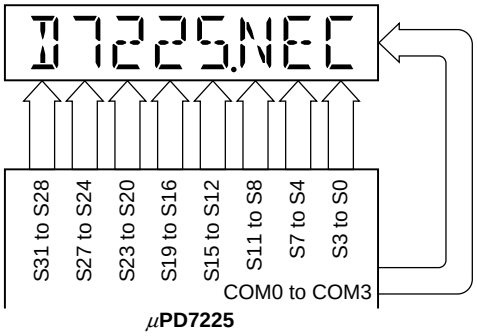
The clock for μPD7225 can be easily made only by connecting resistance to clock the pins (CL1, CL2).

Example: R = 180kΩ – f_{CLK} ≃ 130 kHz (TYP.)
(V_{DD} = 5.0 V)

When it is necessary to adjust the clock with a variable resistance, use R= 180 kΩ ± 5 %.

2.2.2 The connection of μPD7225 and LCD

Figure 2-2. Example of Connection for LCD Display



2.2.3 Structure of the display data within the program memory of μPD7507B

This section shows the structure of the data written in the program memory of μPD7507B to display “D7225NEC”, as shown in the example.

Program memory

Address		
150H	C4	; “D”
151H	B7	; “7”
152H	B2	; “2”
153H	B2	; “2”
154H	B5	; “5”
155H	CE	; “N”
156H	C5	; “E”
157H	C3	; “C”

Write the data in μPD7225 in the order 157H ⟶ 150H of the program memory.

2.2.4 Program example

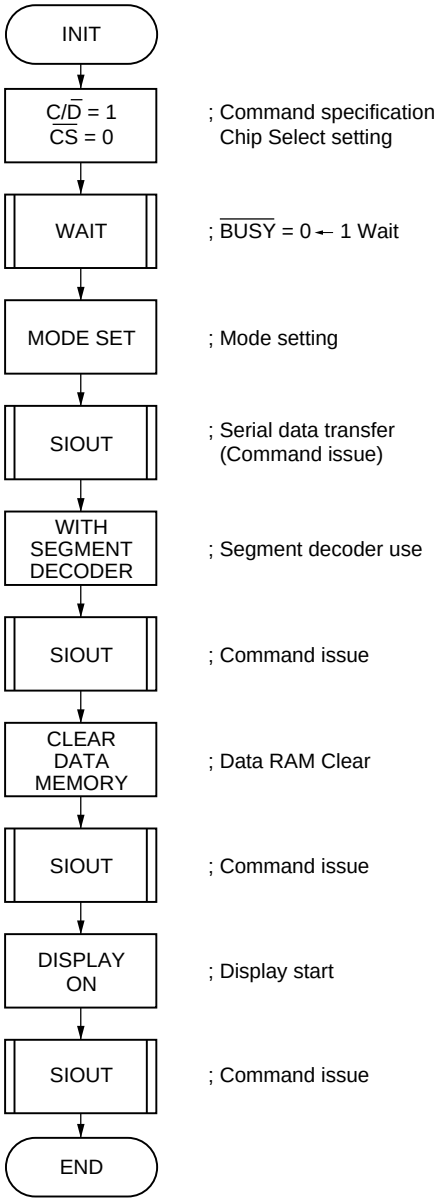
This section shows a program to control the μPD7225 by dividing one into the INITIALIZE routine, data routine, the display routine of DP, weight routine and serial data transfer routine.

This section shows a program which only controls μPD7225. It is assumed, therefore, that the data memory of μPD7507B has been initialized or the stack pointer has already been set.

(1) INITIALIZE Routine

In the INITIALIZE routine, each mode is set after the RESET for μPD7225 is removed, the display RAM is cleared and the command of display start is trnasferred.

Flowchart



Program list

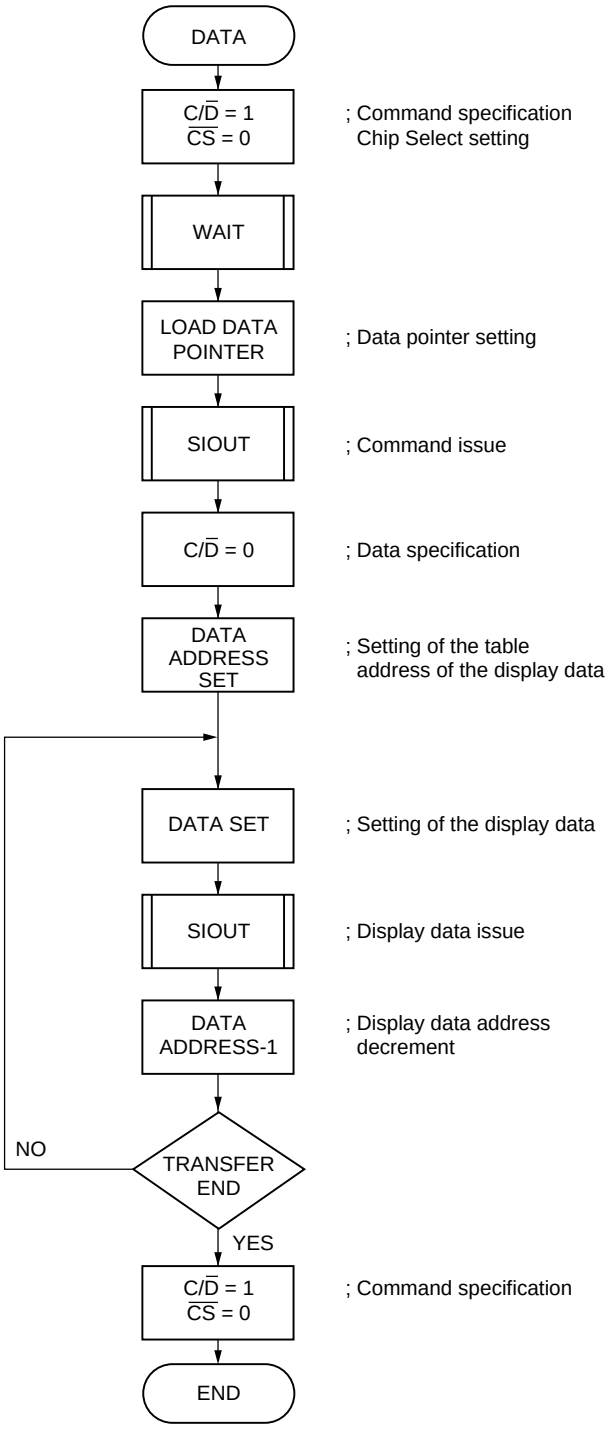
```
INIT:   LAI     0EH
        OP      3       ;   C/D = 1   CS = 0
        CALL    WAIT    ;   WAIT   BUSY   0 ⟶ 1
        LHLI    7FH
        LAI     2       ;
        ST              ;   MODE SET
        LAI     4       ;   COMMAND SET
        CALL    SIOUT   ;   COMMAND OUT
        LAI     5       ;
        ST              ;   WITH SEGMENT DECODER
        LAI     1       ;
        CALL    SIOUT   ;
        LAI     0       ;
        ST              ;   CLEAR DATA MEMORY
        LAI     2       ;
        CALL    SIOUT
        LAI     1       ;
        ST              ;   DISPLAY ON
        CALL    SIOUT
; INIT END
```

In this program, $\overline{CS}$ is not raised after issuing the DISPLAY ON command. If it is necessary to raise $\overline{CS}$, be sure to hold $\overline{CS}$ = 1 for 48 clocks or more. (The μPD7225 clock) (In the case of UNSYNCHRONIZED TRANSFER MODE)

(2) Data routine

The data routine is used to write display data.

Flowchart



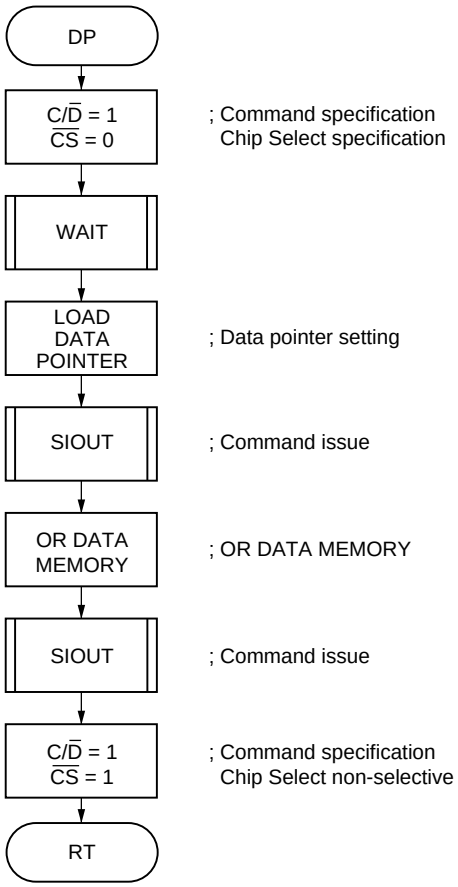
Program list

```
DATA:   LAI     0EH
        OP      3       ;   C/D = 1, CS = 0
        CALL    WAIT
        LHLI    7FH     ;
        LAI     0       ;   LOAD DATA POINTER
        ST              ;         (DP = 00)
        LAI     0EH     ;
        CALL    SIOUT   ;   COMMAND OUT
        LAI     0CH
        OP      3       ;   C/D = 0, CS = 0
        LAI     7       ;   DATA NO.
        LHLI    71H     ;   (71) DATA ADDRESS
        ST
        DLS
TABL:   LADR    71H     ;
        ST              ;  DATA TABLE ADDRESS SET
        LAI     5       ;
        LAMTL           ;   TABLE LOOK UP
        CALL    SIOUT   ;   DATA OUT
        DDRS    71H     ;   DATA ADDRESS-1
        JCP     TABL
        LAI     0FH
        OP      3       ;   C/D = 1, CS = 1
;DATA END
```

(3) DP display routine

The DP display routine displays "." between "5" and "N" as shown in Figure 2-12. The use of this routine is not restricted to DP, but it can be applied to the change of each character.

Flowchart



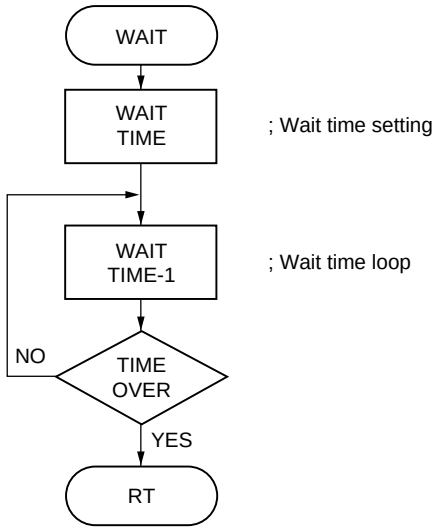
Program list

```
DP:     LAI     0EH
        OP      3       ;   C/D = 1, CS = 0
        CALL    WAIT
        LAI     0CH     ;
        ST              ;   LOAD DATA POINTER
        LAI     0EH     ;       (DP = 0C)
        CALL    SIOUT
        LAI     8       ;
        ST              ;   OR DATA MEMORY
        LAI     0BH     ;
        CALL    SIOUT   ;
        LAI     0FH
        OP      3       ;   C/D = 1, CS = 1
        RT
;DP END
```

(4) Wait routine

Because the $\overline{\text{BUSY}}$ signal outputted by μPD7225 is not tested here, it is necessary to allow enough time until $\overline{\text{BUSY}}$ = 1 is obtained before starting serial data transfer.

Flowchart



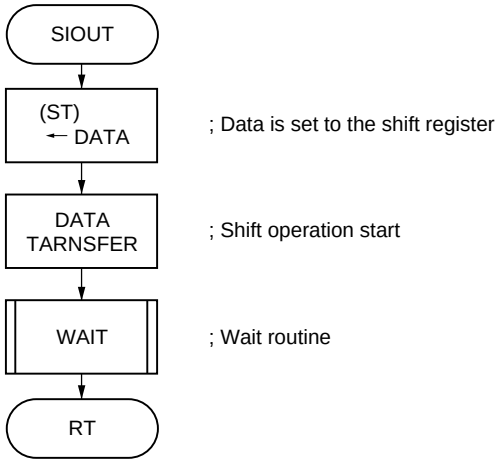
Program list

```
WAIT:   LAI     05H     ;   WAIT TIME SET
        TAL
LOOP:   LAI     0FH     ;   WAIT TIME SET
        TAE
        DES             ;   WAIT
        JCP     $-1
        DLS             ;   WAIT
        JCP     LOOP
        JHLT    7FH     ;   “HL” REG SET
        RT
```

(5) Serial data transfer routine

This routine transfers 8- bit serial data to the μPD7225.

Flowchart



Program list

```
SIOUT:  TAMSIO      ;  (ST) ←—— DATA
        SIO         ;  DATA TRANSFER
        CALL   WAIT
        RT
```

2.2.5 How to set wait time

The $\overline{BUSY}$ low level time (t_{WLB}) for the μPD7225 is shown in the table below.

	MIN.	MAX.	Unit
t_{WLB}	4	44 (57)*	1/fc

Where:
- fc is the clock frequency of the μPD7225.
- 14-segment decoder is used.
- $\overline{CS}$ = 1 ≥ 48/fc : (UNSYNCHRONIZED TRANSFER MODE)
  $\overline{CS}$ = 1 ≥ (48 + Alternating drive frequency)/fc : (SYNCHRONIZED TRANSFER MODE)
  In this case, alternating drive frequency = frame frequency x the number of time-shares
- ∗signifies the case where BLINKING is used.

When serial data is sent from CPU to μPD7225, the $\overline{BUSY}$ signal need not be tested if the data is sent after Wait time of t_{WLB} MAX.

The example below shows the setting of the actual Wait time in the above-mentioned Wait routine.

Although μPD7225 generates the clock through an internal oscillator (i. e. R connected to CL1 and CL2), there is a drift in the clock frequency. (See Figure 2-11.)

	Condition	MIN.	MAX.	Unit
Oscillation frequency	R = 180 kΩ ± 5% V_{DD} = 5 V ± 10 %	85	175	kHz
	R = 180 kΩ ± 5% V_{DD} = 3 V ± 10 %	50	140	kHz

In considering the MAX. time of $\overline{BUSY}$ = 0, it is necessary to consider the case where the frequency is the lowest.

Therefore the MAX. time of $\overline{BUSY}$ = 0 under the conditions

- R = 180 kΩ + 5 %
- V_{DD} = 5 V + 10 %
- BLINKING unused
- 14-segment decoder used
- UNSYNCHRONIZED TRANSFER mode

is obtained according to the following expression:

$$t_{WLB}\ (MAX.) \times 1/fc\ (MIN.) = 44 \times 1/85 \times 10^3$$
$$= 520\ (\mu s)$$

On the other hand, when Wait time is set without using a timer by the software of μPD7507B, it is necessary to consider the case where the clock is the fastest.

On the assumption that the system clock is generated by CR in μPD7507B, the oscillation frequency has the following deviations.

	Condition	MIN.	MAX.	Unit
f_{CC}	R = 82 kΩ ± 2 % C = 33pF ± 5 % (V_{DD} = 5 V ± 10 %)	120	280	kHz

It is therefore necessary to set the Wait routine with respect to f_{CC} = 280 kHz.

Because one machine cycle is 7 [μs] at fc = 280 kHz in μPD7507B, Wait of 75 (machine cycles) becomes necessary in order to pression indicates:

520 [μs] / 7 [μs] = 74.3 = 75 (machine cycles)

In order to set Wait time, the lowest clock of the μPD7225 is combined with the fastest system clock of the CPU.

APPENDIX A BIAS OF LCD AND THE NUMBER OF TIME SHARING DRIVES

Figure A-1. shows the relationship between LCD bias and the number of time sharing drives.

Figure A-1. Relation between the Bias and the Number of Time Sharing Drives

Bias \ Number of time-shares	Static	2-time shares	3-time shares	4-time shares
Static	○			
1/2		○	○	
1/3			○	○

[MEMO]

APPENDIX B TIME SHARING DRIVE AND THE MAXIMUM NUMBER OF DISPLAY ELEMENTS

Shown below are each time sharing drive for the μPD7225 and the maximum number of elements displayable in the static display.

Number of time-sharing drives	Maximum number of displayable elements	Structure of LCD
Static drive	32	COM0, S0-S31
2-time sharing drive	64	COM0, COM1, S0-S31
3-time sharing drive	96	COM0-COM2, S0-S31
4-time sharing drive	128	COM0-COM3, S0-S31

Shown below is the maximum number of display digits for a 7- segment type or a 14-segment type.

Number of time-sharing drives	7-segment LCD	14-segment LCD
Static drive	4 digits + 4 elements*	2 digits + 4 elements*
2-time sharing drive	8 digits + 16 elements*	4 digits + 8 elements*
3-time sharing drive	10 digits + 26 elements*	6 digits + 12 elements*
4-time sharing drive	16 digits + 16 elements*	8 digits + 16 elements*

Remarks: * These elements can be used as decimal points or indicators.

As shown above, an increase in the number of time shares results in an increase in the number of controllable elements.

Therefore, select the number of time shares according to what display is made in the application system and how many display elements are required. An insufficient number of display elements can be coped with by using several μPD7225s.

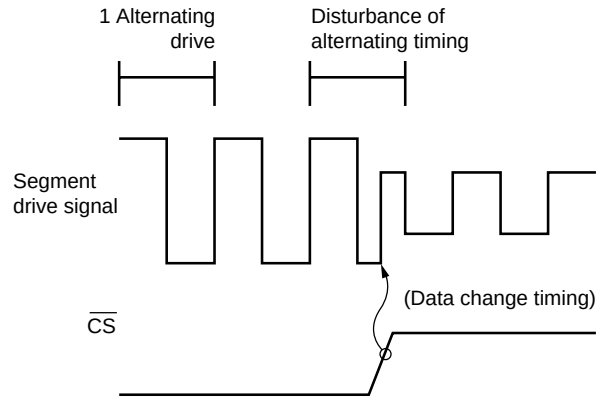
In the case of a static drive, there is no problem of cross-talk in a time sharing drive, and better display quality can be obtained.

[MEMO]

APPENDIX C DISPLAY TIMING AND SEGMENT DRIVE SIGNALS

An LCD requires alternating drive by its nature. However, because an alternating drive loses balance temporarily the instant the display state changes due to a change in timing, the relation may be disturbed between a common drive signal and a segment drive signal. Therefore, a small direct portion may be imposed. (See Figure C-1.)

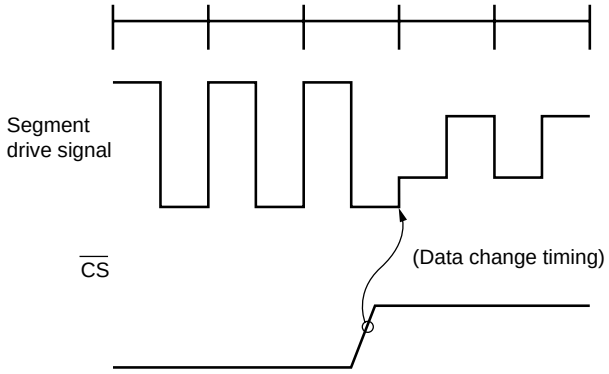
Figure C-1. Segment Drive Signal Waveform



Although an LCD's life shortens when DC voltage is imposed for a long time, the life of LCD's being used today is not significantly affected by the instantaneous imbalance in an alternating drive as shown in Figure C-1. Therefore, the LCD can be driven by the UNSYNCHRONIZED TRANSFER command.

To synchronize a change in the display state with alternating timing, the SYNCHRONIZED TRANSFER command is used. After $\overline{CS}$ is raised and atter this command is executed, a change is made in the display data at the beginning of the next alternating timing.

Figure C-2. Segment Drive Signal Waveform



[MEMO]

APPENDIX D LCD POWER SOURCE CIRCUITS

Figure D-1(a), (b) and (c) show LCD power source circuits in Static, ½ bias and 1/3 bias.

Figure. D-1 (a)

Static

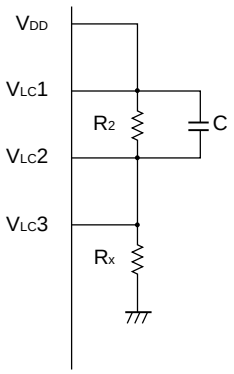
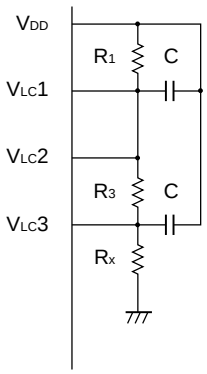


Figure. D-1 (b)

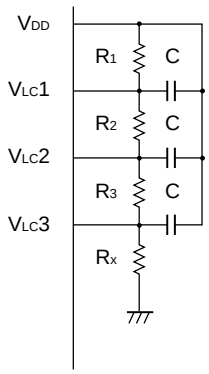
1/2 Bias



$R_1 = R_3$

Figure. D-1 (c)

1/3 Bias



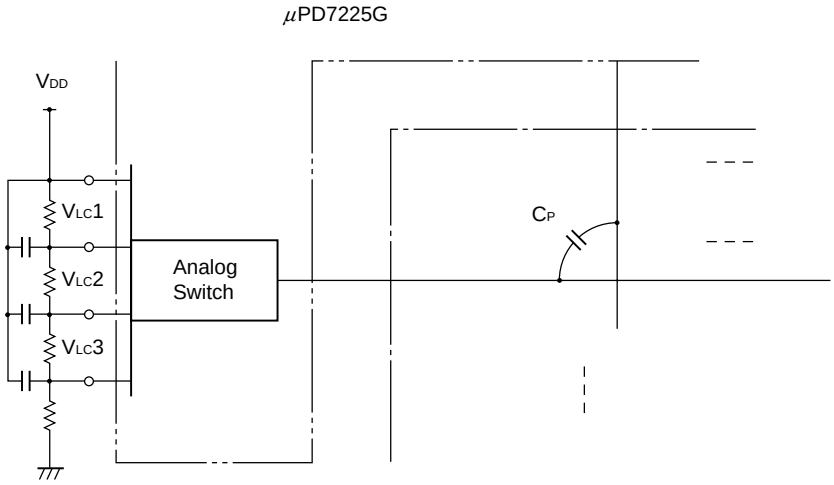
$R_1 = R_2 = R_3$

Remark: $V_{LCD} = V_{DD} - V_{LC}3$

$V_{LCD} : V_{LC}3 = R_{1+2+3} : R_X$

Example: In case of R1 = R2 = R3 = 5 kΩ,bias in 1.3 of 3-V drive and 4-time sharing drive becomes R_X = 10 kΩ

Figure D-2. LCD Equivalent Circuit



Remark: C_P = pattern capacity

When represented in an equivalent circuit, LCD drive can be compared with the charging/discharging of C_P as shown in Figure D-2.

Where drive voltage $V_{LC}0$ to 3 is made from an external split resistance, the charging/discharging waveform becomes dulled when this split resistance is large. Therefore, the effective value of voltage drops and contrast becomes poor.

To better the contrast of LCD and improve visibility in such a case, connect a condenser, lower the impedance of the circuit and adjust the waveform.