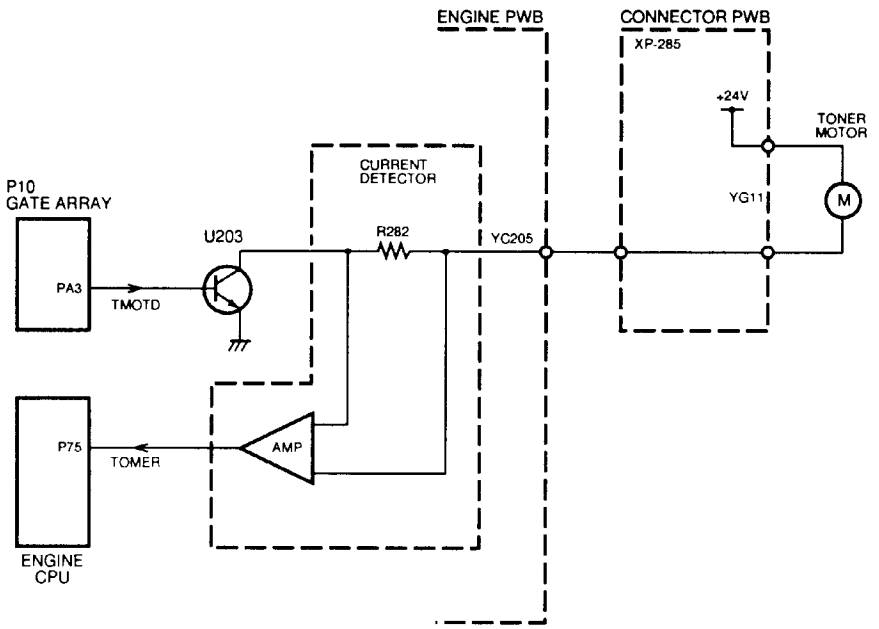
In case of an excessive torque on the motor, for such as the damaged gears inside the developer unit, the over-current detector Q205 turns on and halts the printer, showing message *E9: SERV*. The normal motor current should be 25 to 60mA. The current detector works when the current exceeds approxiamtely 110mA.

Figure 2.17. Toner feeding motor controller



2.5.7. Front Control Panel Controller

The control panel has 7 keys, 2 LEDs, and a TN-type liquid quartz display module of two 8 -character rows which are driven directly by the main logic controller with the assistance of the engine gate array, U202. The gate array function in key scanning and LED driving as shown in figure below. KeysSW1 through SW7 are connected to the data bus.

The data necessary to light the appropriate LED are latched in U202 as the main logic controller selects LD1 and LD2.

The message display has priority in showing messages as tabled in the following tables.

Table 2.3. Indicator priority

Priority	ON-LINE State	OFF-LINE State
High	##: SERV. nnnnnn System error I/F occupied	
	Cover Open	
	Check Process unit	
	Replace Toner Kit TK-11	
	Remove Manual paper	
	Paper jam	
	Cancel ? xxx	
	Size err? xx	
	Memory overflow .. Press ON LINE Print overrun .. Press ON LINE KPDL Error .. Press ON LINE Opt. ROM error .. Press ON LINE	
	Add paper	
	Heating	
	Proc. Waiting TimeOut	Option I/F error
	Option I/F error	Warning Low Toner TK-11 Warning Short memory
	Warning Low Toner TK-11 Warning Low memory	Proc. Waiting TimeOut
Low	Ready	

Key activity to indicators

Table below explains the relevancy of the front panel indicators to the messages on the message display. In this table, ON means the LED turns on; while OFF means the LED does not light. Exceptions are noted at the bottom of the table. Table 2.4. shows key operability in accordance with the current message on the message display.

Table 2.4. Key activity to indicators

Message	Indicators	
	ON LINE	ATTENTION
##: SERV. nnnnnn	Off	—
System error ... Restarting Cover open Check process unit Replace Toner Kit TK-11 Size err? xx Paper jam	Off	On
Heating	Off	Off

Message	Indicators	
	ON LINE	ATTENTION
Memory overflow ... Press ON LINE Print overrun ... Press ON LINE KPDL Error ... press ON LINE Opt. ROM Error ... Press ON LINE	Blinking	On
I/F occupied Remeove Manual paper Add paper	See note 1.	On
Proc. Waiting TimeOut MODE SELECTing	See *1.	See *2.
Cancel ? xxxxxxxxxx	Off	See *3.
Warning Low memory Opt. I/F error	See *1	Blinking.
Warning Toner Kit TK-11	See *1	-
Ready	See *1	Off

*1: On if online; off if offline. *2: Blink if a warning is given; otherwise off. *3: On if an error is indicated; otherwise same as *2.

Table 2.5. Key operability

Message	Key					
	ON LINE	FORM FEED	MODE SELECT	+ and -	ENTER	CANCEL
##: SERV. nnnnnn System error I/F occupied Cover open Check Process unit Replace Toner Kit TK-11 Remeove Manual paper Paper jam Heating	No	No	No	No	No	No
Cancel? xxxxxxxxxx Size err? xx	No	No	No	Yes	Yes	Yes
Add paper	Yes	No	Yes	No	No	Yes
Memory overflow ... Press ONLINE Print overrun ... Press ONLINE KPDL error ... Press ONLINE Opt. ROM Error ... Press ONLINE Proc. TimeOut	Yes	No	No	No	No	Yes
Waiting	Yes	Yes	No	No	No	Yes
Warning Low memory Warning Low Toner TK-11 Opt. I/F error	Same as TimeOut, Proc. or Waiting, if the printer is off-line in printing; otherwise same as Ready.					
Ready	Yes	See *4	Yes	No	No	No
MODE SELECTing	No	Yes	Yes	Yes	Yes	Yes

*4: Enables automatic macro overlay if the current emulation is HP LaserJet III. The PRESCRIBE automatic macro overlay overrides the HP's, however.

2.5.8. Main Motor Controller

The main motor is a PM-type stepping motor driven by the dual phase exciter bipolar driver using a constant-current-chopper-controlled circuit.

The main motor is driven when the 150Hz clock the engine CPU U201 outputs at its MTRCK (pin #16) is half-divided, consequently resulting in two clocks offset in 90 degrees, MTRA and MTRB. The main motor begins rotating as soon as these clocks turn the level of MOTOR*and MTRIN*0 signals, connected to the driver IC via Q201 and Q205 from the engine CPU, to low, giving the maximum driving current.

The slow-starter method is used for revolving the motor: At first the engine CPU sends the 160 pps signal for the pulse duration of 24 counts; then the 300 pps signal in 26 pulse counts (normal revolution). See Figure 2.19.

Figure 2.18. Main motor controller

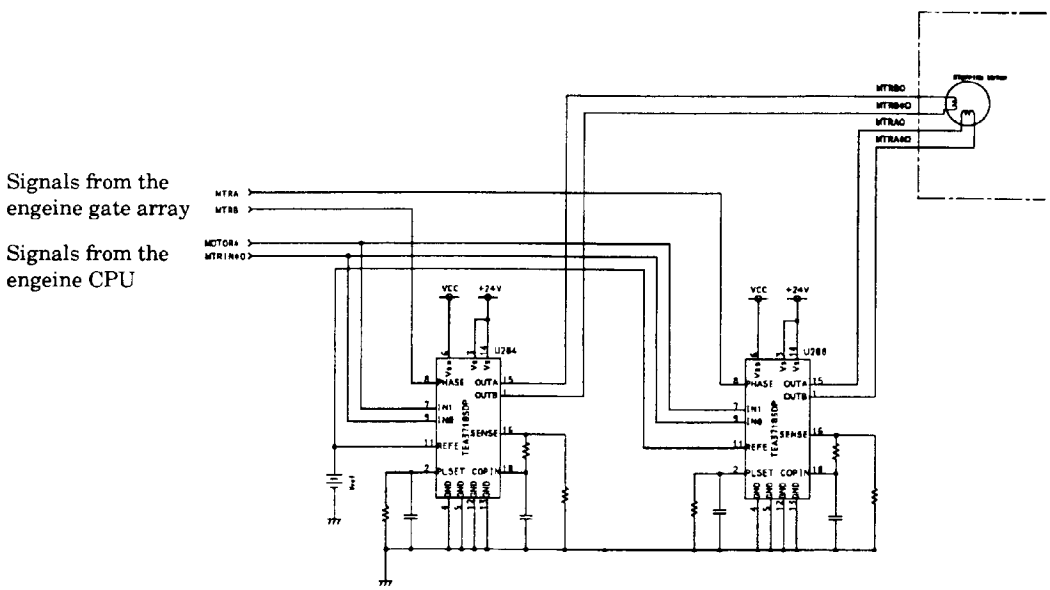
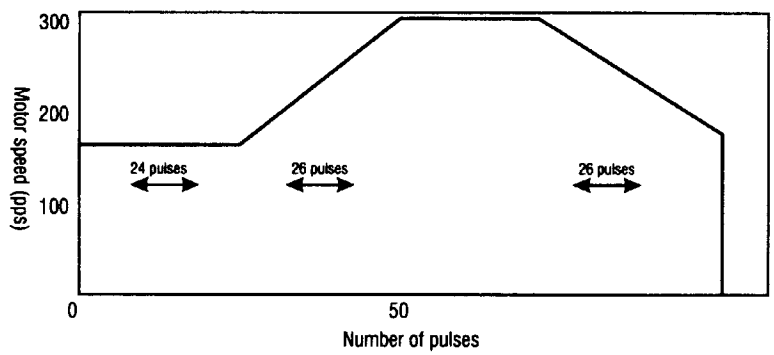


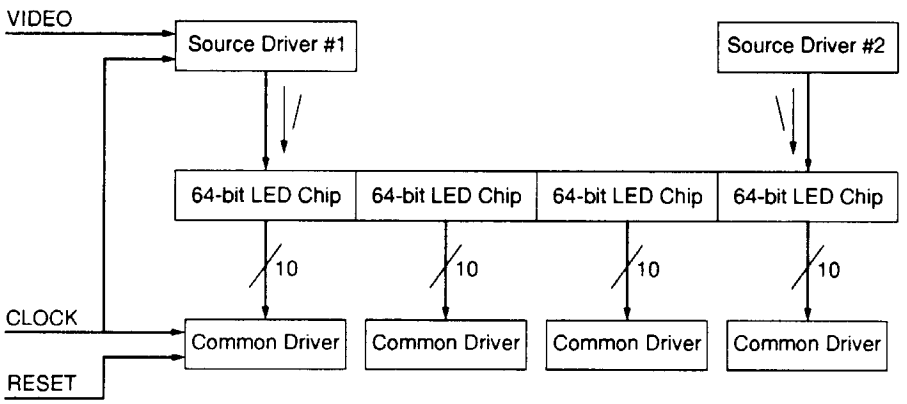
Figure 2.19. Main motor slow-start



2.5.9. LED Head Driver

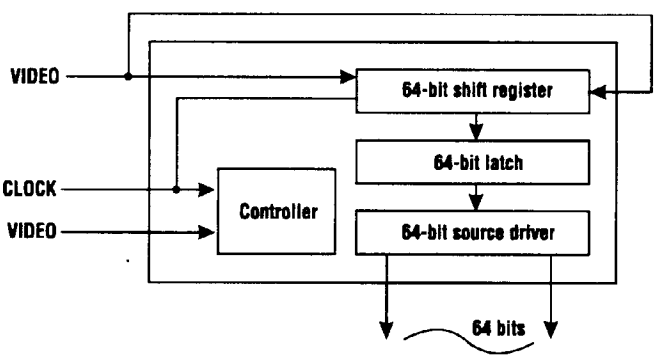
Figure below shows the block diagram of the LED head drivers mounted on the glass-printed -circuit board. The LED head uses forty (40) one-chop 64-bit LED chips (64 bits × 40 blocks = 2560 bits). In this figure, the driving current for the LED chips flows from both source driver #1 and source driver #2 to ensure an even current supply to overcome the resistance of the glass -printed-circuit board.

Figure 2.20. LED head driver



The source driver has the internal segments as shown in the figure below. Each time the video data come into the 64-bit shift register, the driver moves it into a 64-bit latch, then drives the source driver.

Figure 2.21. Source driver for LED head



The common driver IC includes 10-bit common drivers as shown on next page. Four ICs including this common driver are subordinately connected to drive the 40 LED chips.

The least significant bit of the common driver shift register is given a 1 as a RESET signal arrives the controller in the IC. (The RESET signal can be referred to as the synchronization signal in laser printers.) An LED segment is selectively driven when the 1 in the shift register is read in 64-bit configuration. The LED segment does not flash if a 64-bit input is made right after a RESET; the first block of the LED chips flashes when the next 64-bit input is being made.

Figure 2.22. Common driver

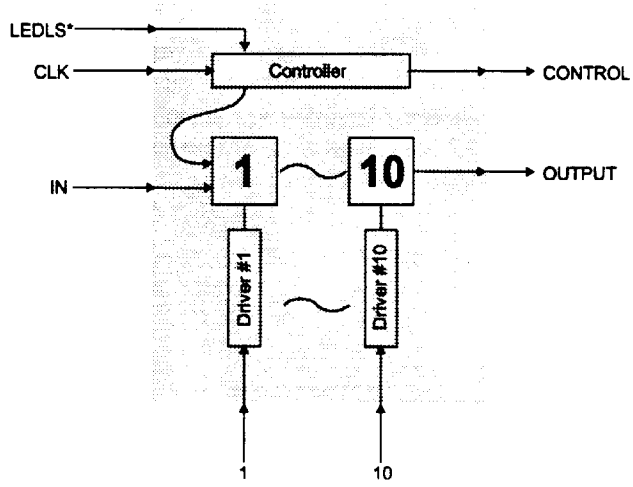
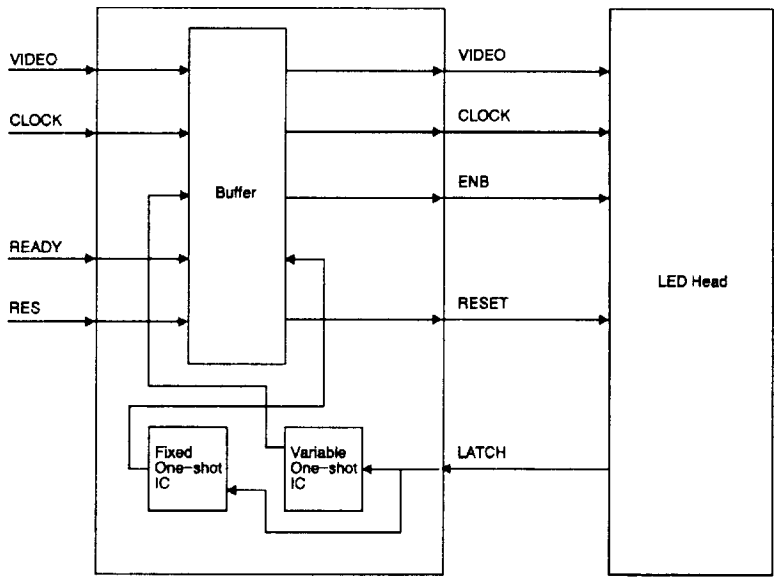


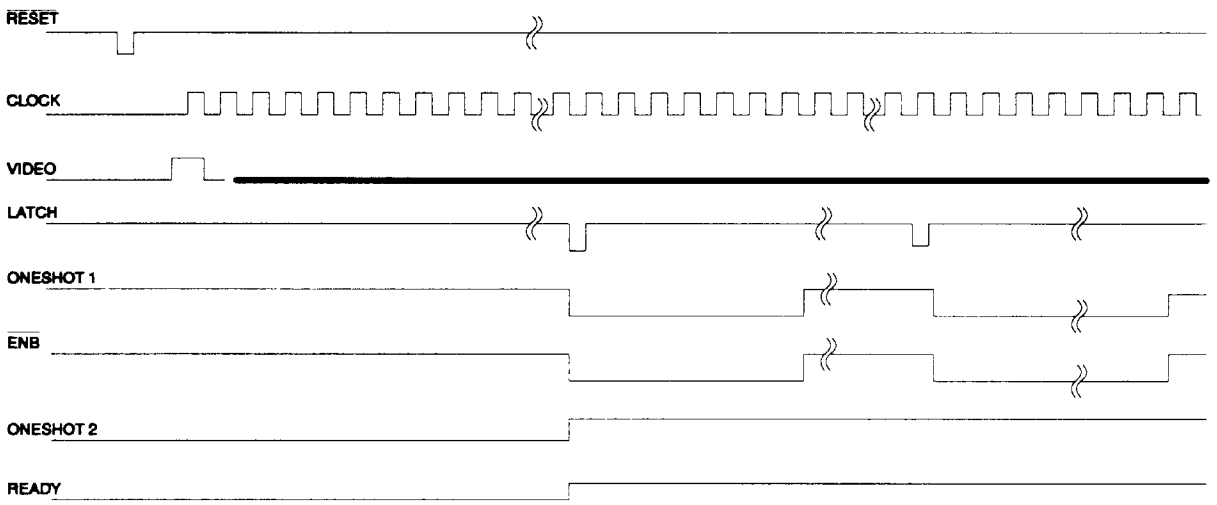
Figure below shows all singlas used for the LED head driver. Figure 3.24 on next page is the timing table for driving the LED head.

Figure 2.23. LED driving signals



The meaning of theses signals and timings are explained in the following table.

Figure 2.24. LED Head Signal Timings



Signal	Meaning
$\overline{\text{RESET}}$	A horizontal-synchronized signal used as the instruction signal for starting a horizontal scanning.
CLOCK	Synchronizing clock for video data.
VIDEO	Video data.
$\overline{\text{LATCH}}$	Used within the LED head; transfers the contents of the 64-bit shift register into the 64-bit latch.
$\overline{\text{ENB}}$	An enabling signal to determine the period of time during which the LED is driven. This signal is delivered from the one-shot IC that accepts the LATCH signal as an input. The output width of this signal is adjustable with a potentiometer so that the luminosity of the LED head varies.
READY	An acknowledging signal that means the LED head is working normally. This is given as the output signal from the retriggering one-shot IC which uses the LATCH signal ash the input signal. The width of this pulse signal is adjusted to be longer than the period of time required to transfer the 64-bit video signal.

2.5.10. Engine Timing Charts

The following are engine timing charts at the different five paper sizes. Drum speed is 25.335 mm/s and paper feeding speed is 71.471 mm/s.

Figure 2.25. A5 size

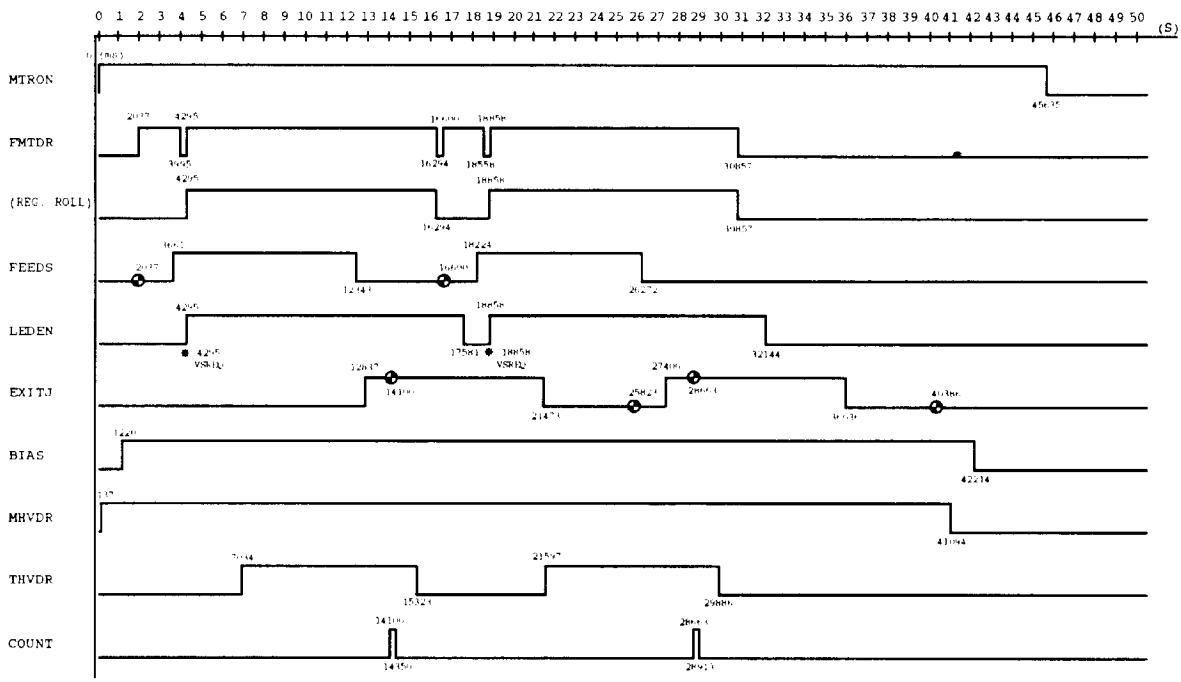


Figure 2.26. B5 size

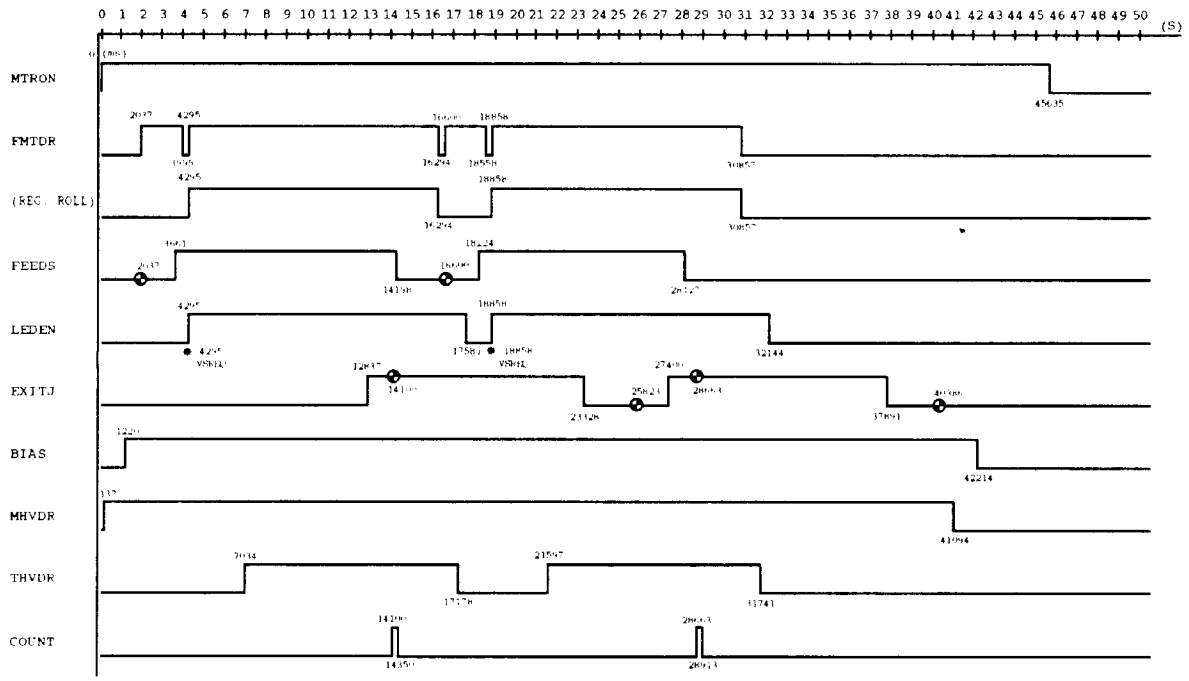


Figure 2.27. Letter size

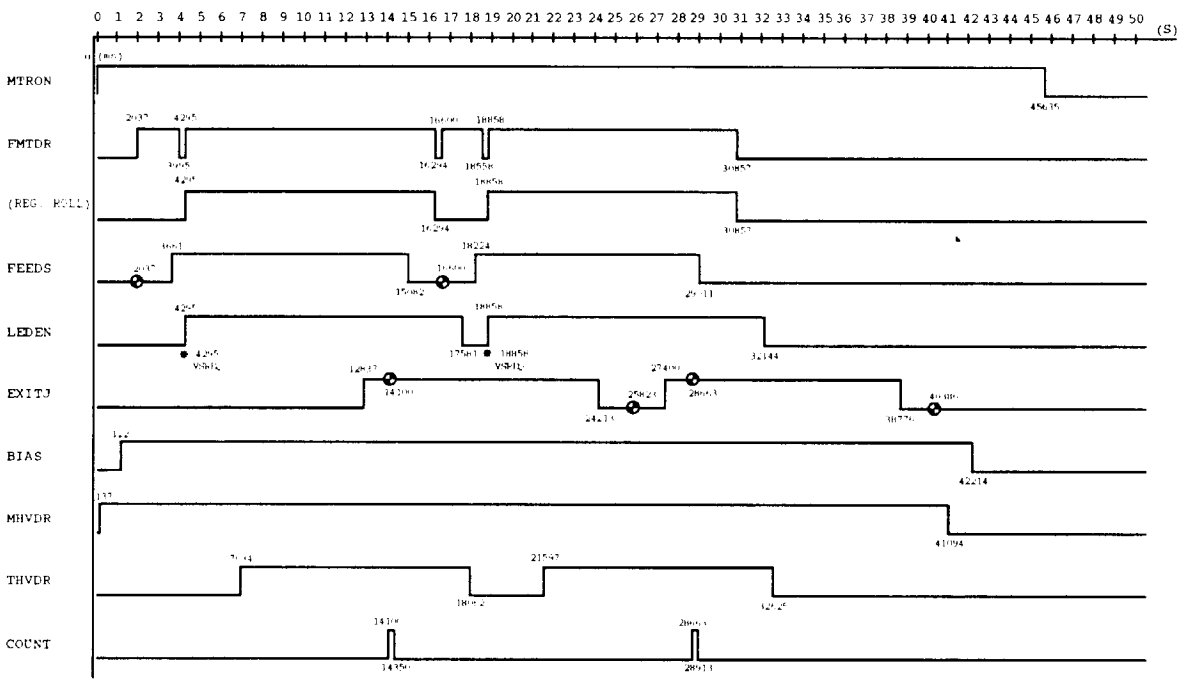
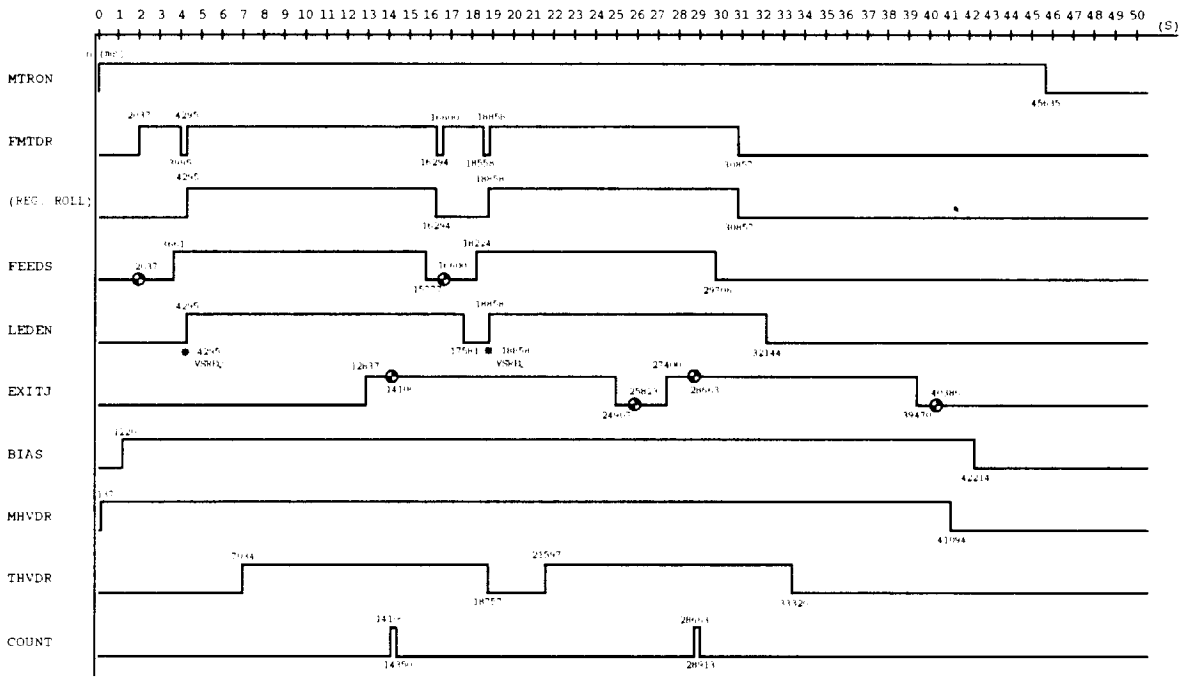


Figure 2.28. A4 size



2.6. Data Processing System Overview

This section contains detailed discussion on the operating features of the printer's main logic controller circuitry. Figure 2.28. illustrates the component layout of the KP-400 board. Figure 2.29. on next page shows all component circuitries in the printer's main logic controller.

The printer uses a 16-bit microprocessor **68EC000** as the main controller CPU. Function of each segment in the controller system will discussed in the following sections. In the following descriptions, the number with a letter-U prefixed means the symbol number of the chip or device which is also printed on the printed circuit board. Major chips and their symbol numbers on the board are tabled below.

Figure 2.29. Main controller board layout

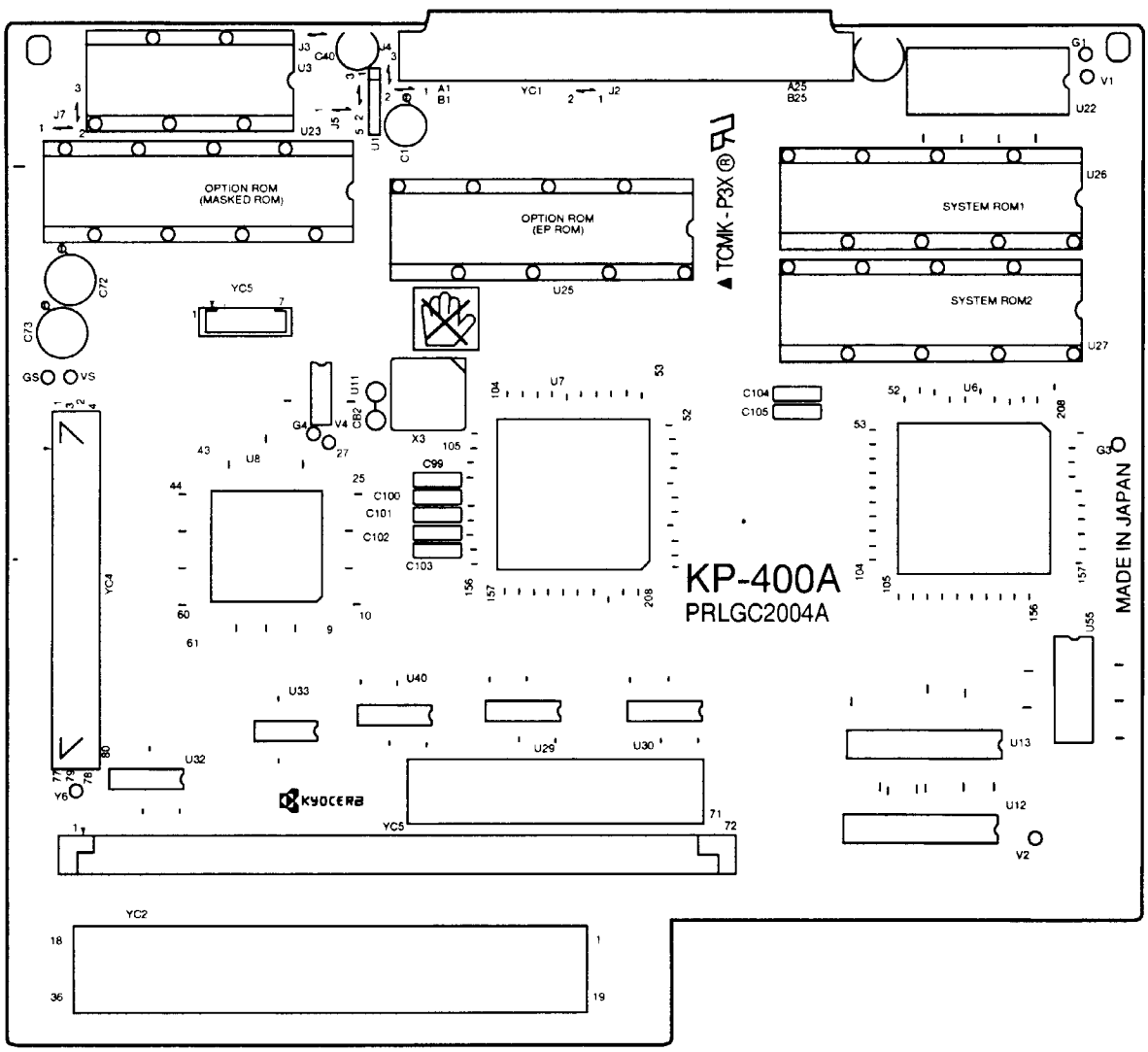
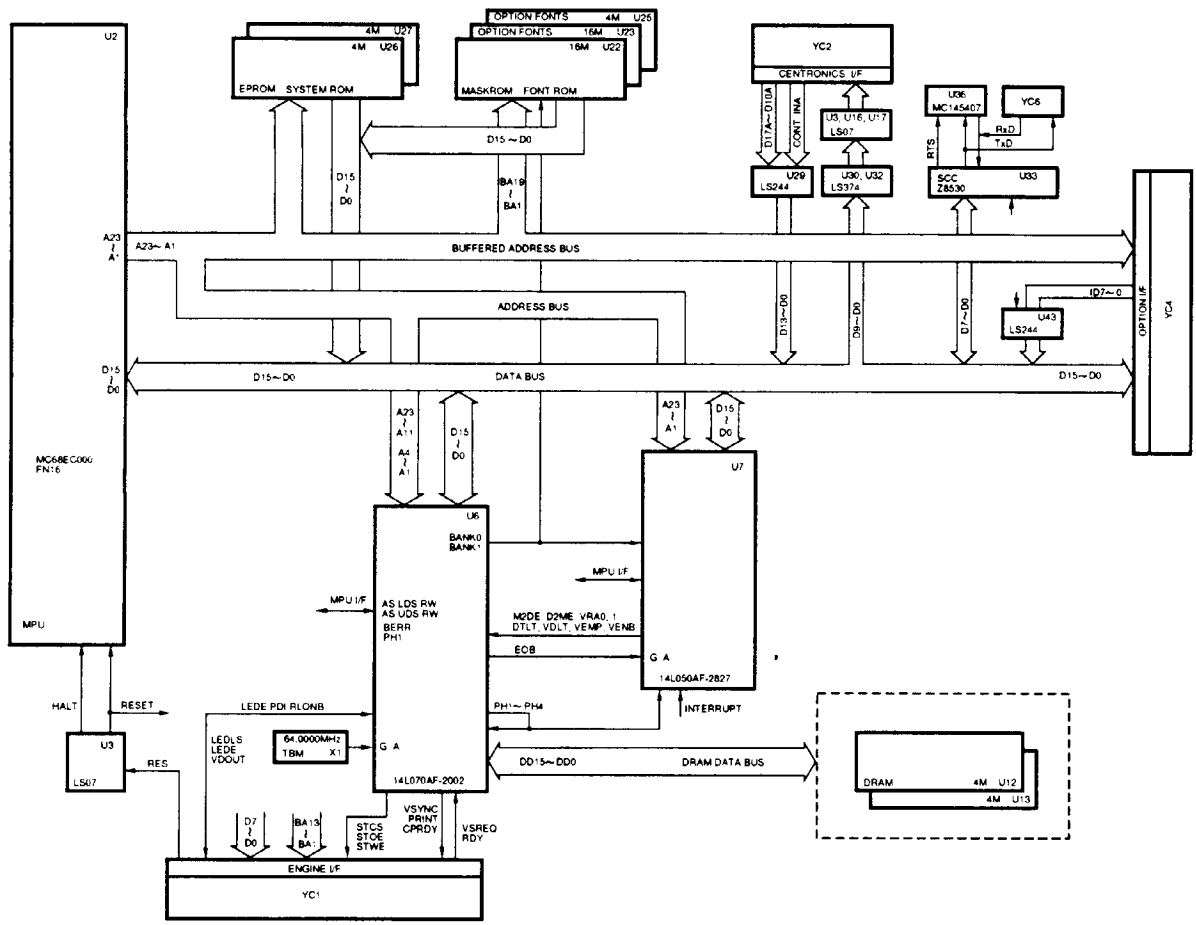


Figure 2.30. Main controller block diagram



2.6.1. Controller CPU: 68EC000 (U8)

The 68EC000, operating at the 16MHz clock, takes care of the timings among the data and commands coming through the interfaces to the gate arrays [which are the main functionality of the controller system], dynamic RAMs, program ROMs, mask ROMs, etc. The 68EC000 can be allocated with up to 16 MB of address memory. The controller system employs the bank manager for allocating a large area for the address memory.

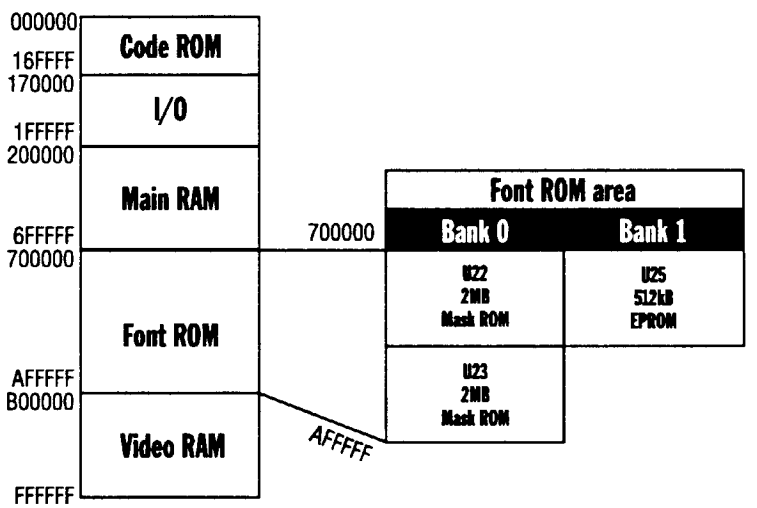
2.6.2. System ROMs (U26/U27)

The system ROMs includes the control programs for driving CPU in the repeated routines during printing. Physical allocation is 512 kB for both U26 and U27.

2.6.3. Font ROMs (U22)

These are font data ROMs that retain bitmap data and algorithms for the printer's resident fonts. The physical area is 4MB which is expandable by means of the internal bank management as shown in the diagram below. Bank 0 is a 2MB (2M by 16) area provided by U22, as well as the 2MB (2M by 16) area provided by the masked ROM U23. Bank 1 is 512kB (256k by 16) and optionally used for U25.

Figure 2.31. Memory mapping



2.6.4. Dynamic RAMs (U106, U107)

The DRAM area is referred to as *raster memory* or video RAM for page printers. Two 512-kB (256 k × 16 bits) DRAMs are used to constitute the memory area of 2 Mbytes. This area is expandable using expansion SIMMs. Generation of the data to be printed is done in this area.

2.6.5. Gate Array M65315AFP (U7)

This gate array implements the following functions:

Address decoding: Decodes address information that 68EC000 produces in order to generate select signals for memory and I/O.

Clock divisor: Generates the quadruple-phase 16 MHz clock by division of the time-base-base module 64 MHz.

Pattern data processing: Computes data for implementation of the required print model.

Video data output control: Converts the video data constituted in the raster memory (video RAM) into serial data. The resultant serial data is used to control the engine system.

System data bus control: Switches the data buses to control the direction of data among the following:

- 68EC000 data bus
- Data bus connected to DRAMs
- Data bus connected to the pattern operator

EEPROM control: Manages EEPROM in which the permanent parameters are stored.

Engine interface: Manages communication between the controller and the engine system for the following:

- Interrupt control (Manages the interrupting signals 68EC000 utilizes)
- Front control panel control (Controls the signals input on the key pad, delivers the output signals for driving LED indicators, and writes data for display of messages)

2.6.6. Gate Array M65316AFP (U6)

This is the second gate array that implements the following functions:

DRAM control: Provides control over the signals used during writing/reading DRAMs and the signals used during refreshing.

Timer control: Provides the refreshing interval for the DRAMs and programmable timer. The proghrammable timer is used for excution of various commands of the internal firmware.

68EC000 bus control: Determines whether to use 68EC000 or an external device as the current bus master.

2.6.7. SRAM (U55)

The SRAM is used for buffering data during the KIR processing.

2.7. Printing Data Processing

This section describes the brief discussion on how the video data are generated and used to drive the LED head and how a printing job completes. It starts with the introduction of data on the printer's interface(s) and concludes the LED head driven to write image on the drum.

Discussion on the electrophotography process, such as development and transferring image, including paper transportation, are given in details in the Electrophotography section and this is not repeated here.

1. Printing data arrive at the parallel or option interface (if installed) when the host computer starts sending through its output port (and through the network). These data are temporarily stored in the printer's interface buffers corresponding to each interface. The most printing data normally include software commands which instruct the printer to supplementary jobs needed to accomplish printing such as line-feeding, page-breaking, changing fonts, etc.

2. The main controller analyzes the data in sequence, translates them into dots according to the original image, and depicts the image resultant of the dot data in the raster memory (video RAM). Operations additionally required during processing data is accomplished by the internal circuit of the gate arrays which supplement the main CPU.

3. While data processing is in course, on the other hand, the main CPU talks to the engine CPU, via the common RAM and the engine interface (U105), to discern the readiness of the printer's engine for printing.

4. If the main logic controller determines that the engine is ready to start, it triggers the engine CPU and, according to the vertical synchronization signal, requests to feed the paper.

5. In synchronization with the succession of the paper, U6 releases video data in sequence. The video data are then applied to the LED head driver together with the horizontal synchronization signal and the video clock.

6. On reception of video data, the LED head driver switches the LED segments of the head on and off and constitutes the image over the drum. The image on the drum, referred to as *static latent* image, is applied with toner, transferred onto the paper, and finally fused permanently on the paper by means of heat and pressure.

2.8. Kyocera Image Refinement (KIR) System

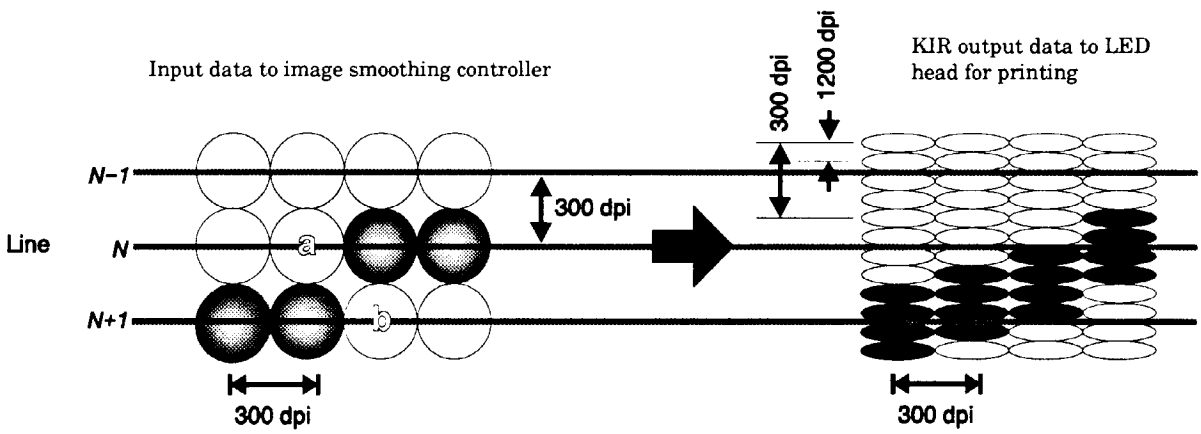
The printer employs the KIR technology which provides refinement of printed image by smoothing stairsteps constituting the outlines of the image.

This section describes how image refinement is obtained using KIR. Also, the electrical operation theory on the LED head driver that is provided in subsection 2.4.11. will help easy understanding of the KIR system.

2.8.1. Operating Principle of KIR

Figure below shows how the normal 300 dpi pixels are processed with KIR by increasing the speed of a scan line four times the normal scanning speed. This means that the vertical resolution of the printer becomes virtually four times finer, namely 1200 dpi, than when KIR is turned off.

Figure 2.32. Principle of KIR



Pixels are divided into subpixels by the circuit shown in next figure: *Smoothing controller*. The circuit consists of a pattern memory, a decoder, and the image smoothing controller. The pattern memory receives in serial manner the incoming video data for a scan line (from the printer's video RAM) and stores the data. The decoder includes a boolean operation unit. The video data and the pattern memory are subjected to a boolean operation upon receiving an SL signal fedback from the image smoothing controller.

A desired smoothing of stairsteps in the image is obtained by adding extra *black* subpixels, or, in other words, reversing a *white* pixel to a *black* pixel, using the proprietary stairstep recognition algorithm to be detailed in the following. Figure 2.50. shows a typical smoothing of adding subpixels.

This algorithm involves generating *border data* for each pixel of the image, then using the border data to detect the existence and type of stairsteps to be smoothed. The length of the pixels

Figure 2.34. Image Smoothing Controller

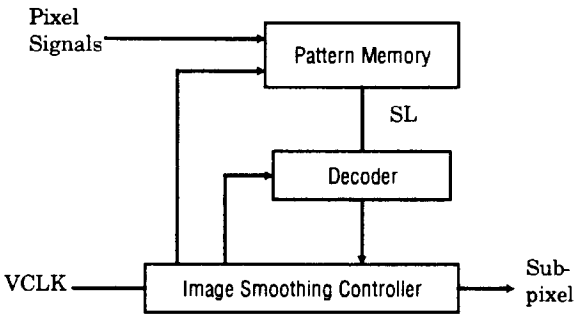
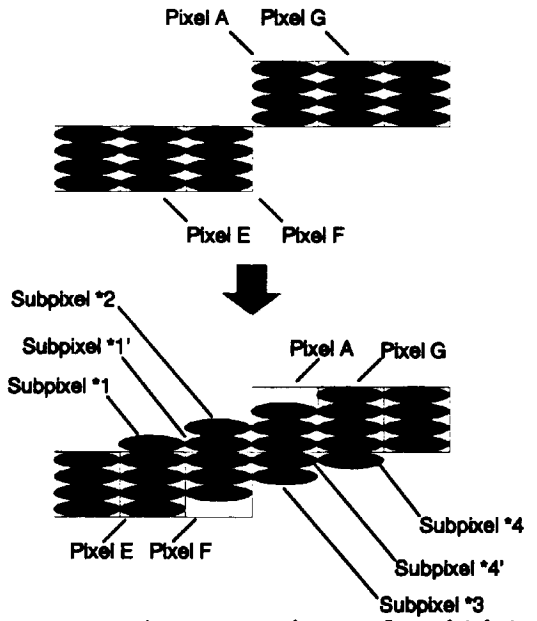


Figure 2.33. Added subpixels



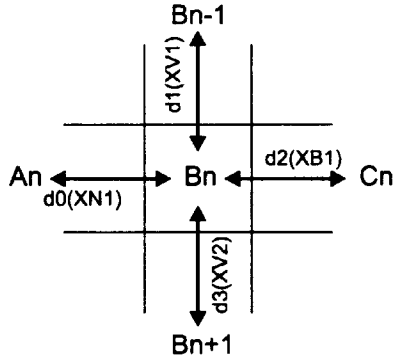
on a given scan line are then counted to obtain an appropriate **reversion code**, which is used to determine the appropriate subpixel to be added to effectuate the refinement.

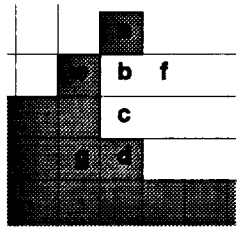
To simplify the discussion, the steps for this theory are divided into the following, different sections.

2.8.2. Generation of Border Data

Refer to Figure 2.51. For **all** pixels, represented by **Bn** , a four-bit code **d0, d1, d2,** and **d3** is generated. For each **di** (***i*=0, 1, 2,** or **3**), if the adjacent pixel on the corresponding side differs from pixel **Bn** , **di** is assigned a value of **1** , while if the adjacent pixel on the corresponding side is the same as pixel **Bn** , **di** is assigned a value of **0** . The reversion code, thus generated, describes the relationships between pixels.

Figure 2.35. Generation of Reversion Code



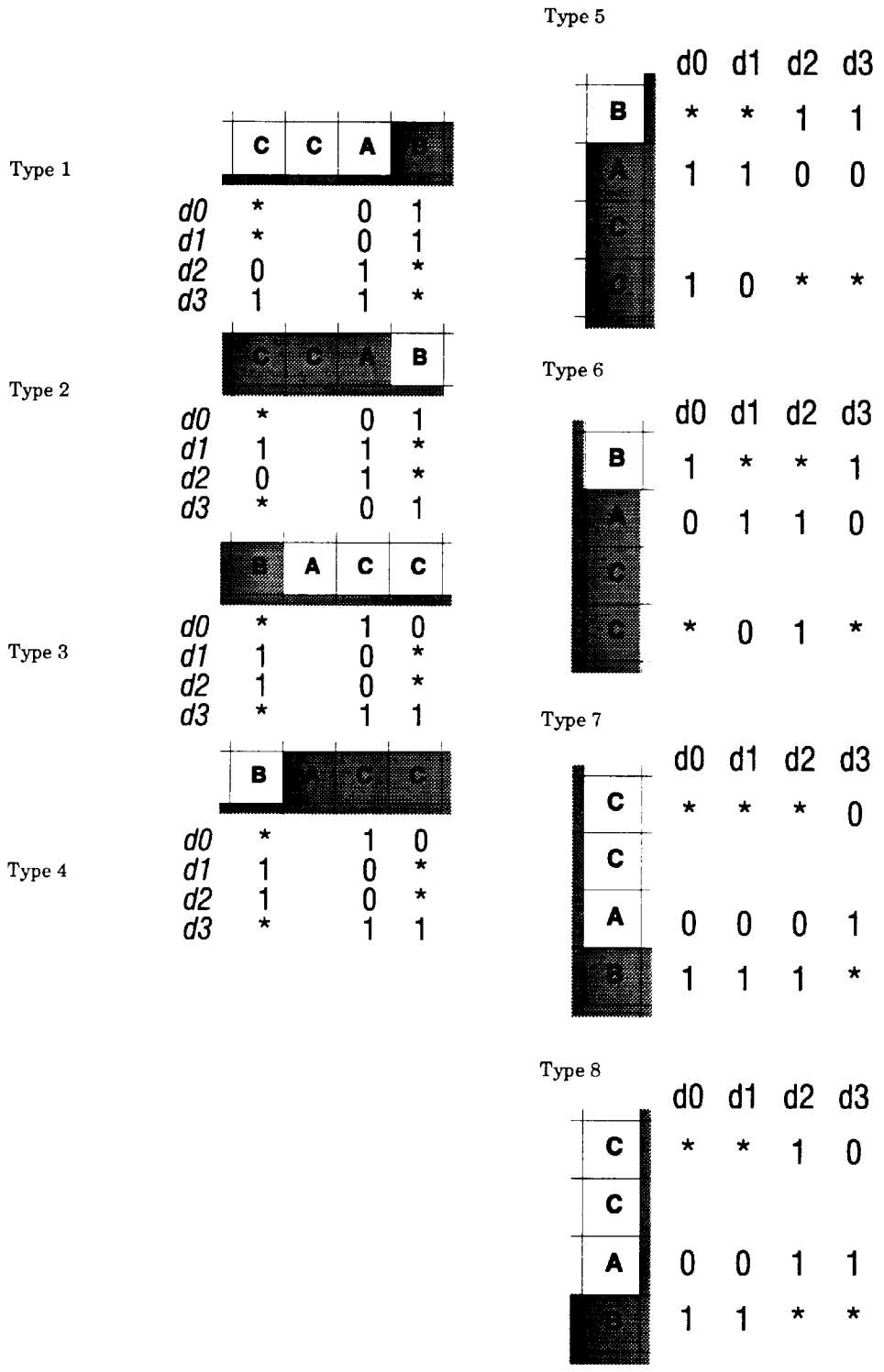


	a	b	c	d	e	f	g
d0	1	1	1	0	1	0	0
d1	1	1	0	1	1	0	0
d2	1	0	0	1	1	0	0
d3	1	0	1	0	0	0	0

2.8.3. Detection and Style Recognition of Stairsteps

Figure 2.52. shows eight different types of stairsteps 1 through 8 that are used as templates for style recognition of stairsteps. The type of stairsteps is determined according to the four-bit code. For example, in Figure 2.51., if transition pixel A has a 0011 code, and the bits **d0** and **d1** of pixel B both have a value of **1** , and bits **d2** and **d3** of pixel C have values **0** and **1** , respectively, then the stairstep is of type 1. The existance of a stairstep is determined by boolean operation on the pixels C and B adjacent to a specific pixel A (which may be a transition pixel).

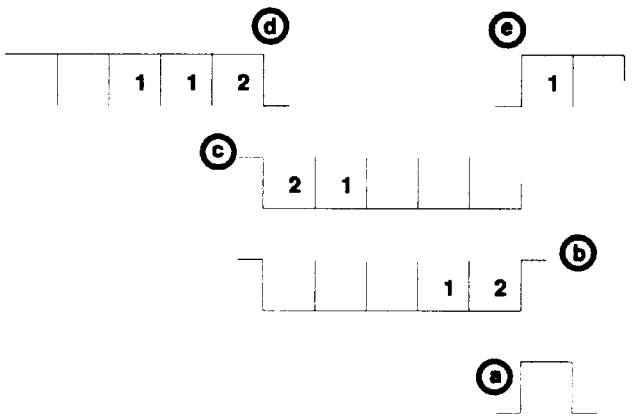
Figure 2.36. Smoothing Templates



2.8.4. Assignment of Reversion Code

The **length** of the pixels on a given scan line are counted (i.e., pixels succeeding or preceding the stairstep), and a reversion code of either 1 or 2 is assigned. Figure 2.53. shows an example.

Figure 2.38. Assigning Reversion Code



Number of pixels succeeding or preceeding the stairstep	Length of level pixels				
0	1	1	2	2	2
1			1	1	1
2					1

Figure 2.37. Reversion Data Table

For example, the **length** for the stairsteps **e** in the above figure is 1 (the pixel at the stairstep is not counted) and the pixel adjacent to the stairstep is assigned a reversion code of 1 (See the table in Figure 2.53. above). Likewise, the **length** for the stairstep **d** in the figure above is greater than 5 and the pixel adjacent to the stair step is assigned a reversion code of 2 (It has zero distance from the stairstep.), while the pixel which is on pixel from the stairstep is assigned a reversion code of 1, and so on.

Depending on the type of stairsteps 1 through 8 as shown in Figure 2.52. and the reversion code assignd as 1 or 2, the corresponding subpixel is selected as shown in Figure 2.54.