**Please note that Cypress is an Infineon Technologies Company.**
The document following this cover page is marked as “Cypress” document as this is the company that originally developed the product. Please note that Infineon will continue to offer the product to new and existing customers as part of the Infineon product portfolio.

**Continuity of document content**
The fact that Infineon offers the following product as part of the Infineon product portfolio does not lead to any changes to this document. Future revisions will occur when appropriate, and any changes will be set out on the document history page.

**Continuity of ordering part numbers**
Infineon continues to support existing part numbers. Please continue to use the ordering part numbers listed in the datasheet for ordering.

FM3

# THERMAL PRINTER DEVELOPMENT KIT 3 INCH USER MANUAL

Doc. No. 002-04440 Rev. *A

Cypress Semiconductor
198 Champion Court
San Jose, CA 95134-1709
http://www.cypress.com

## Copyrights

© Cypress Semiconductor Corporation, 2014-2016. The information contained herein is subject to change without notice. Cypress Semiconductor Corporation assumes no responsibility for the use of any circuitry other than circuitry embodied in a Cypress product. Nor does it convey or imply any license under patent or other rights. Cypress products are not warranted nor intended to be used for medical, life support, life saving, critical control or safety applications, unless pursuant to an express written agreement with Cypress. Furthermore, Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress products in life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

### Trademarks

All other trademarks or registered trademarks referenced herein are property of the respective corporations.

### Source Code

Any Source Code (software and/or firmware) is owned by Cypress Semiconductor Corporation (Cypress) and is protected by and subject to worldwide patent protection (United States and foreign), United States copyright laws and international treaty provisions. Cypress hereby grants to licensee a personal, non-exclusive, non-transferable license to copy, use, modify, create derivative works of, and compile the Cypress Source Code and derivative works for the sole purpose of creating custom software and or firmware in support of licensee product to be used only in conjunction with a Cypress integrated circuit as specified in the applicable agreement. Any reproduction, modification, translation, compilation, or representation of this Source Code except as specified above is prohibited without the express written permission of Cypress.

### Disclaimer

CYPRESS MAKES NO WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Cypress reserves the right to make changes without further notice to the materials described herein. Cypress does not assume any liability arising out of the application or use of any product or circuit described herein. Cypress does not authorize its products for use as critical components in life-support systems where a malfunction or failure may reasonably be expected to result in significant injury to the user. The inclusion of Cypress' product in a life-support systems application implies that the manufacturer assumes all risk of such use and in doing so indemnifies Cypress against all charges.

Use may be limited by and subject to the applicable Cypress software license agreement.

# Contents


**1. Introduction ..... 6**

Scope of the document: ..... 6

1.1 About Document ..... 6

1.2 About Thermal Printer Development Kit 3 Inch ..... 6

1.3 About MB9B310/MB9B510 Series MCU ..... 6

**2. System Overview ..... 7**

2.1 Develop and Work Environment ..... 8

2.1.1 Develop Environment ..... 8

2.1.2 Work Environment ..... 8

**3. Hardware Introduction ..... 9**

3.1 Hardware Overview ..... 9

3.1.1 Features ..... 10

3.1.2 System Block Diagram ..... 10

3.2 Power Supply ..... 11

3.2.1 24V to 5V ..... 11

3.2.2 5V to 3.3V ..... 11

3.3 Connectors ..... 12

3.3.1 USB Connector ..... 12

3.3.2 UART Connector ..... 13

3.3.3 Power Connector ..... 14

3.3.4 Print Head Connector ..... 14

3.3.5 Debug Port and Expand Connector ..... 16

3.4 Configuration ..... 18

3.5 Printing Control Circuit ..... 19

3.6 External Storager ..... 20

**4. Function ..... 21**

4.1 Command List ..... 21

4.2 Character Printing ..... 23

4.2.1 Character Lib ..... 23

4.2.2 User-Define Characters ..... 25

4.2.3 Character Font Style ..... 26

4.3 Bitmap Printing ..... 27
4.4 Barcode Printing ..... 28
4.5 Interface Introduction ..... 30
4.5.1 DIP Switch ..... 30
4.5.2 LED and Button ..... 31
4.6 Self-Test Introduction ..... 32
4.7 Sensor Check ..... 33
4.7.1 Paper Sensors ..... 33
4.7.2 Platen Open Sensor ..... 33
4.7.3 Head Temperature Sensor ..... 33
4.8 Black Mark Function ..... 34
4.9 Error Processing ..... 34
**5. Firmware W Introduction ..... 35**
5.1 Firmware Overview ..... 35
5.2 Communication ..... 39
5.2.1 USB Serial Interface ..... 39
5.2.2 RS-232 Serial Interface ..... 43
5.3 ESC/POS Module ..... 44
5.3.1 Status Check ..... 44
5.3.2 FIFO ..... 44
5.3.3 Task ..... 47
5.3.4 ESC/POS ..... 48
5.4 Data Process ..... 49
5.4.1 Data analysis ..... 49
5.4.2 Page data process ..... 49
5.4.3 Create task ..... 51
5.4.4 Other ..... 52
5.5 Print Header ..... 52
5.5.1 Print Header Driver ..... 52
5.5.2 Print Header Pin ..... 57
5.5.3 Print Header CSIO ..... 59
5.6 Character Module ..... 59
5.6.1 General Overview ..... 59
5.6.2 Function Module Flow ..... 60
5.6.3 Functions Description ..... 62
5.7 Barcode Module ..... 68
5.7.1 General Overview ..... 68
5.7.2 Functions Description ..... 68

5.8 MISC Module ..... 71
5.8.1 ADC ..... 71
5.8.2 Dual Timer ..... 72
5.8.3 Free Run Timer ..... 74
5.8.4 PWM ..... 76
5.8.5 Sys_IF ..... 78
**6. Reference Documents ..... 79**
**7. Appendix ..... 80**
7.1 Commands ..... 80
7.1.1 Command Description Format ..... 80
7.1.2 Terms Description ..... 80
7.1.3 Commands ..... 81
**Revision History ..... 129**
Document Revision History ..... 129

# 1. Introduction

## Scope of the document:

This application note is described about below products;

| Series | Product Number(not included Package suffix) |
|---|---|
| MB9B310 | MB9B316N |
| MB9B510 | MB9B516N |

## 1.1 About Document

This application notes describes how to operate Thermal Printer Development Kit 3 Inch, and describes how to design thermal printer solution by FW resource and HW reference

## 1.2 About Thermal Printer Development Kit 3 Inch

Thermal Printer development Kit 3Inch is about 3 inch thermal printer control solution, this solution developed based on MB9BF316N and MB9BF516N, this solution support high speed (150mm/s) printing, support black mark paper, compatible ESC/POS command, support multiple barcode, Have independent character lib, the communication port include USB and UART.

## 1.3 About MB9B310/MB9B510 Series MCU

MB9B310/MB9B510 series MCU is 32-bit general purpose MCU of FM3 family that features the industry's leading-edge ARM Cortex-M3TM CPU and integrates Fujitsu's highly reliable and high-speed secure embedded flash technology. This MCU can operate at up to 140MHz CPU frequency and work at a wide voltage range (2.7-5.5V), which can be both compatible with 3.3V and 5V system.

It includes a host of robust peripheral features, including motor control timers (MFT), base timer (can be configured to PWM, PPG, Reload, PWC timer), ADCs, on-chip memory (up to 512K Flash, up to 64K SRAM) and a wide range of communication interfaces (USB, I2C, SIO, LIN, CAN).

The size of on-chip memory can be configured according to different part number and the package is available in LQFP and BGA, shown in Table 1

Table 1. FM3 Product List

| Product | Flash | SRAM | Package |
|---|---|---|---|
| MB9BF316N | 512kB+32kB | 64kB | NA: LQFP-100/BGA-112 |
| MB9BF516N | 512kB+32kB | 64kB | NA: LQFP-100/BGA-112 |

# 2. System Overview

This system adopter FM3 MCU MB9B316N, the printer is JX-3R-021AP, the max printing speed design to 150mm/s, and the solution support USB and UART printing, the Printer command is ESC/POS, the PC install Corresponding driver, below is the system architecture:

Figure 1. System Architecture

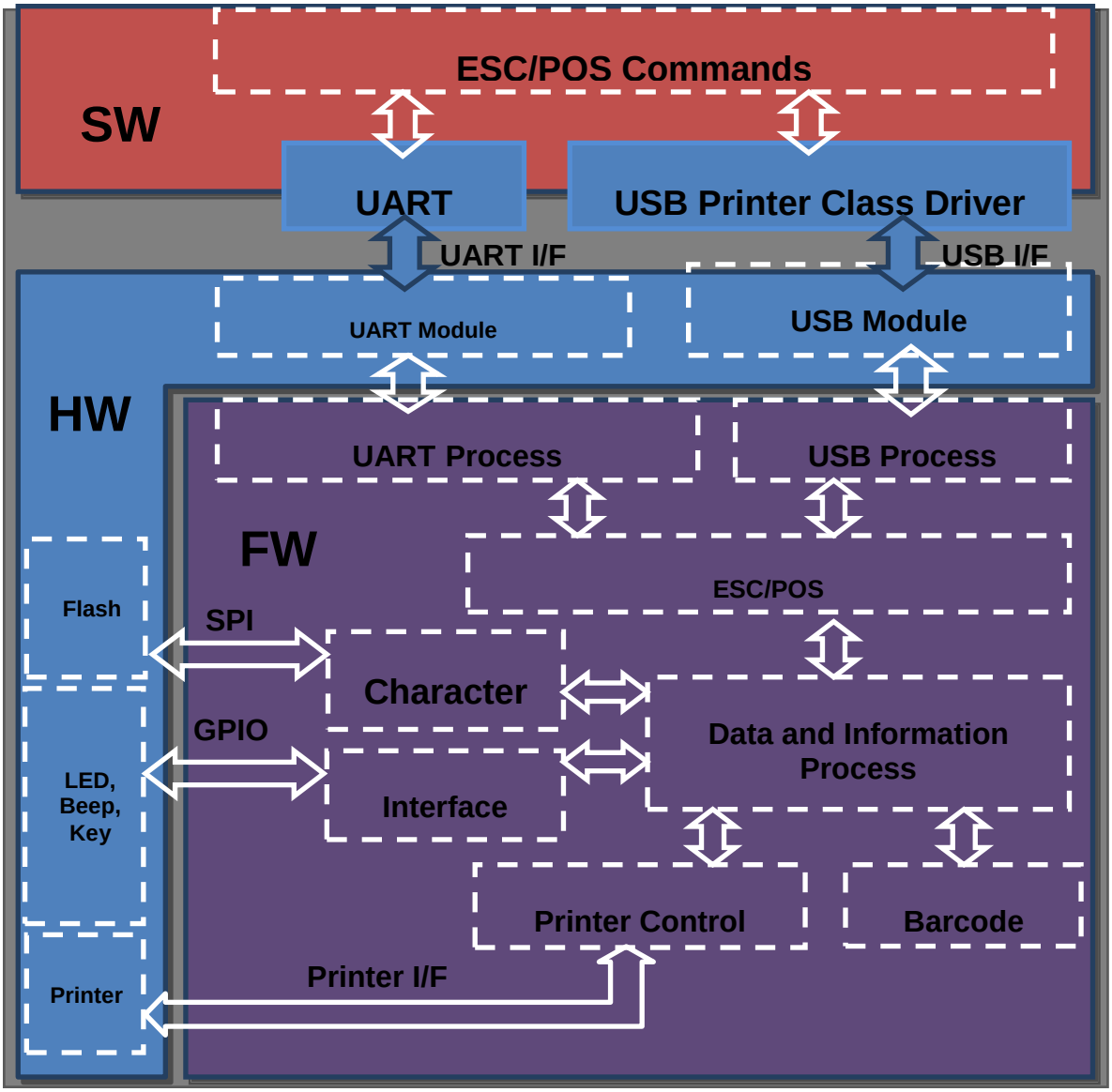
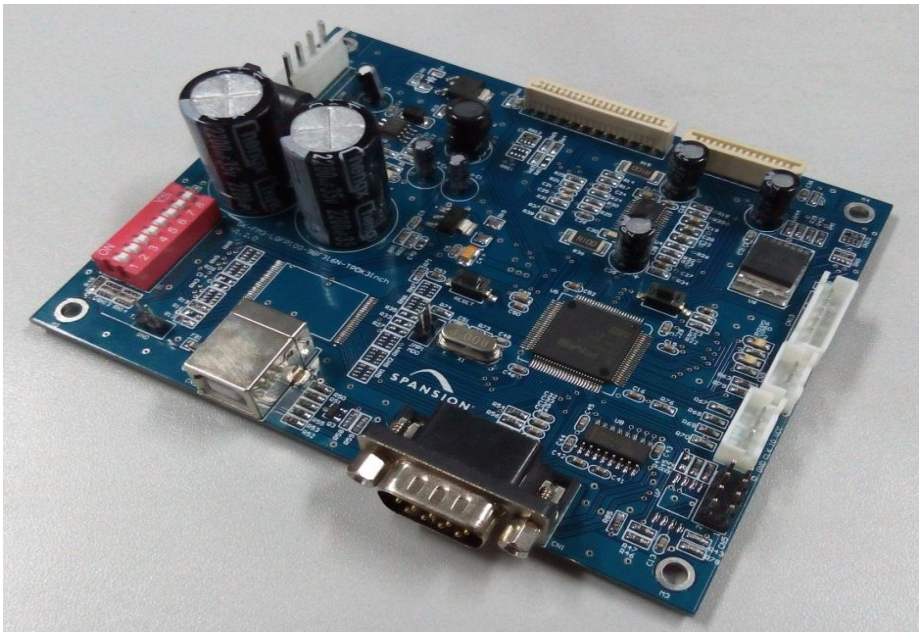


Figure 2. Printer Control Board



## 2.1 Develop and Work Environment

### 2.1.1 Develop Environment

Table 2. HW Develop Environment

| Item | Description |
|---|---|
| OS | Windows 7 Enterprise SP1 64-bit |
| Develop Tool | Cadence<br>IAR 6.5<br>Visual Studio 2013 |
| Develop Accessories | J-link<br>USBA-USBB line<br>USBA-UART line<br>AC-DC Power supply |

### 2.1.2 Work Environment

Table 3. Work Environment

| Item | Description |
|---|---|
| Supply Voltage | DC 24V ± 2.4V |
| Current | Max 2A |
| Temperature | Storage: -40°C ~ 80°C<br>Working: 0°C ~ 55°C |
| Humidity | 10% ~ 70% |

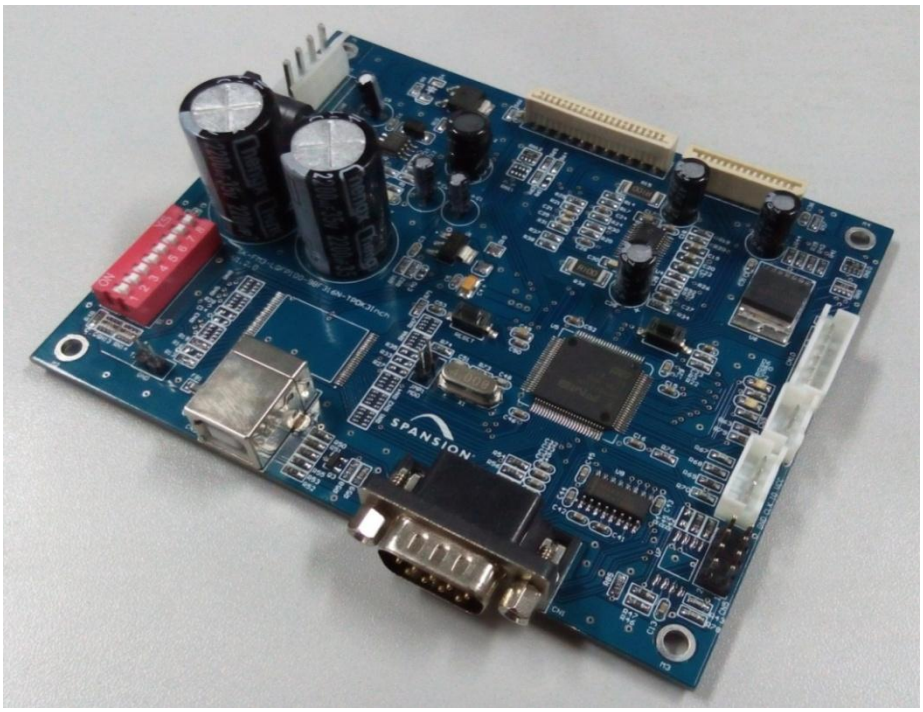
# 3. Hardware Introduction

## 3.1 Hardware Overview

The Thermal printer development is intended to aid the user in the rapid evaluation and development of thermal printer applications using MB9BF316N Serial MCU which are embedded with ARM Cotex-M3 core. This development Board is targeted to control the thermal printer head for printing.

The development Board provides USB,SWD, and UART interface. It supports JX-3R-021 printer header, supports ASCII print, Chinese print, Bitmap print and GUI for command input.

Figure 3. Thermal Printer Development Board



### 3.1.1 Features

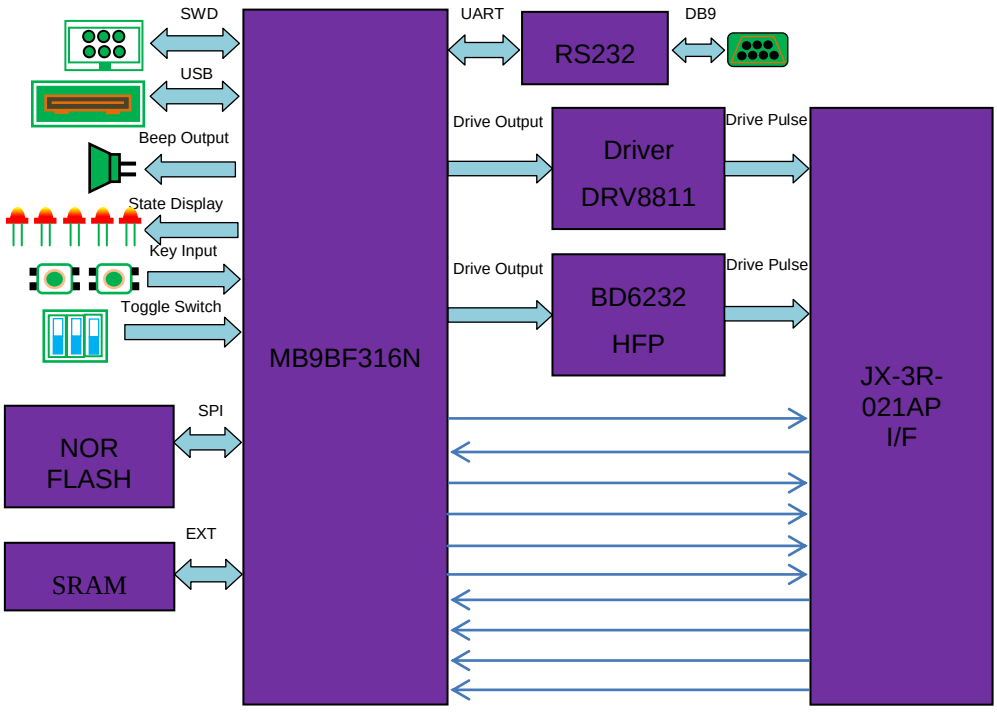
The features of the development Board include:

- Voltage range: 24±10%(DCV);
- Provide basic input/output (2 Keys, 6 LEDs, 8 ways toggle switch);
- Provide multiple interface (1 UART, 1 SWD, 1 USB);
- Provide status check function, just as black label, rubber roller ready, without paper and over temperature etc.;
- Provide a stepping motor driver for printing and feeding paper;
- Provide a DC motor driver for automatic paper cutting function;
- Printing precision: 8dots/mm;
- Support serial communication interface (TTL);
- Maximum print speed is 150mm/s based on the printing voltage;
- Support bar code print, freewill picture print, Chinese print (24*24, GB2312), ASCII print (12*24) and ESC/POS control command;

### 3.1.2 System Block Diagram

Following figure shows overview of the system block diagram.

Figure 4. System Block Diagram



## 3.2 Power Supply

There are two power ICs in this system.

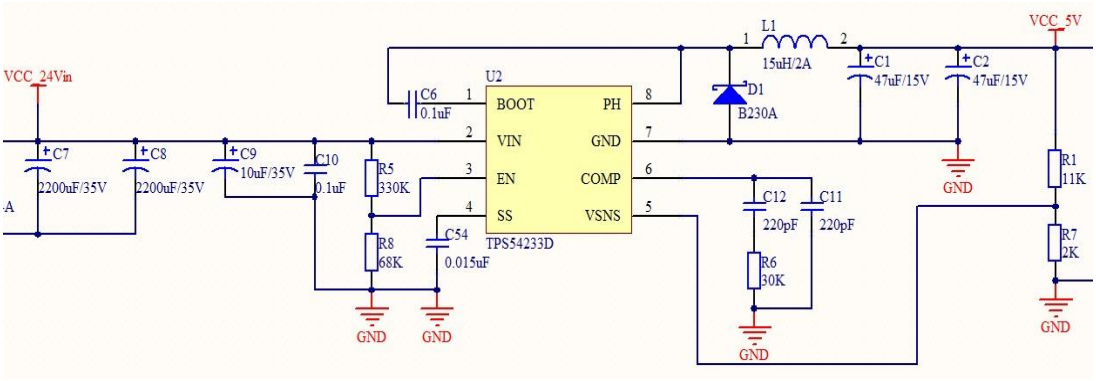
24V to 5V DC/DC converter support the 5V power source for print header logic power.

5V to 3.3V LDO converter support the 3.3V power source for MCU of MB9BF316N, stepper motor driver IC, DC motor driver IC, Nor flash, SRAM etc.

Following chapters show these conversions.

### 3.2.1 24V to 5V

Figure 5. Schematic for 24V to 5V

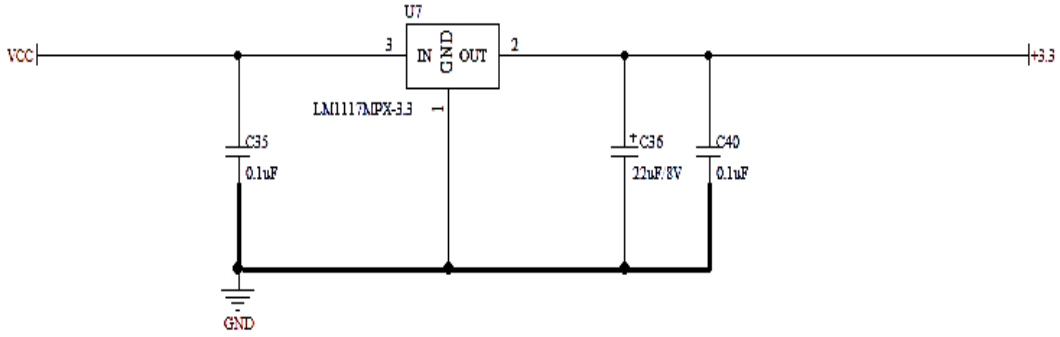



By setting the feedback resistor R1 and R7, can get different output voltage according to the following formula:

Vout = 0.8* (1 + (R1/ R7))

### 3.2.2 5V to 3.3V

Figure 6. Schematic for 5V to 3.3V




As a result of constant voltage output of the power IC, so there is no need to set the feedback resistor.

The system electrical characteristics as below:

Table 4. System Electrical Characteristics

| Item | Condition | Rated Value | | | Units |
|---|---|---|---|---|---|
| | | Min | Type | Max | |
| Supply Power Voltage | | 21.6 | 24 | 26.4 | V |
| Current of Working | Printing ratio of 20%, printing is 150mm/s | -- | 2.09 | -- | A |
| Current of Standby | | -- | 0.3 | -- | A |

## 3.3 Connectors

### 3.3.1 USB Connector

This board uses the USB type B as the USB interface, the detail as below:

- USB Power Supply: 3.3V, 100mA
- USB Standard: 2.0
- USB Port: Type B
- Quantity: 1

Figure 7. USB Connector

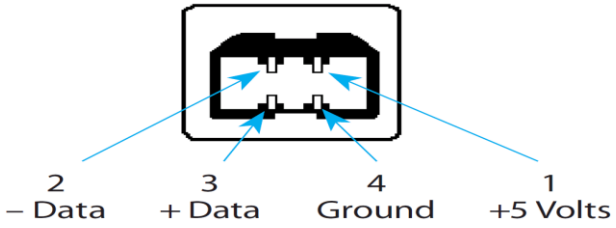
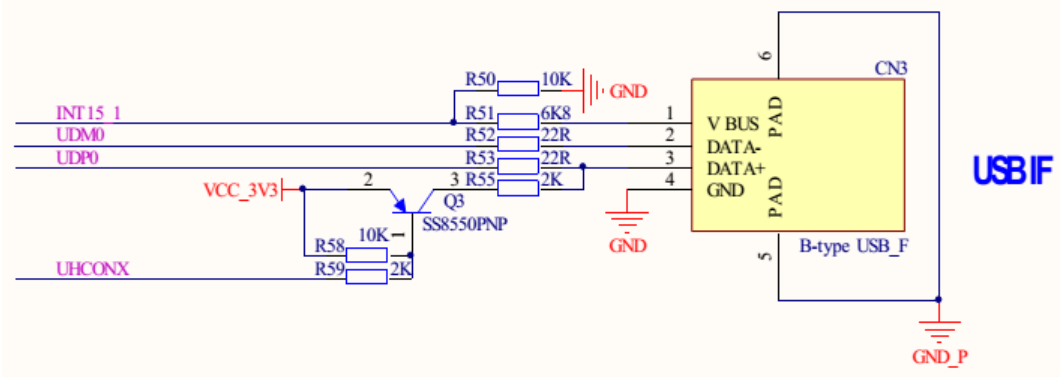



Figure 8. USB Connector Schema



### 3.3.2 UART Connector

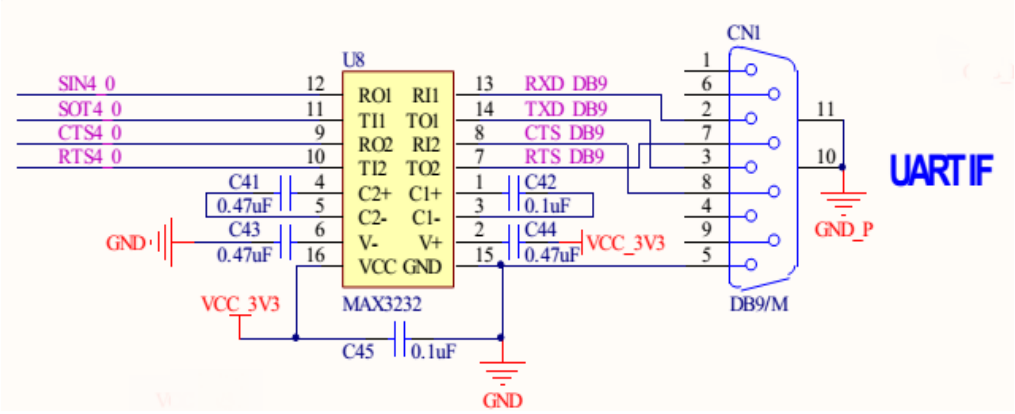
The UART interface is the DB9/M, the UART Handshaking adopts RTS/CTS, and the baud rate is configurable by DIP, as below:

- Connect Port: EIA-RS-232
- UART Port: DB9/M
- Quantity: 1

Figure 9. UART Connector



Figure 10. UART Connector Schema




The UART connection with user side as below:

Table 5. UART Connection

| Board side | | | User side | |
|---|---|---|---|---|
| **Pin No** | **Signal Name** | | **Signal Name** | **Pin No** |
| 1 | NC | | DCD | 1 |
| 2 | RxD |  | RxD | 2 |
| 3 | TxD | | TxD | 3 |
| 4 | DTR | | DTR | 4 |
| 5 | SG | | SG | 5 |
| 6 | DSR | | DSR | 6 |
| 7 | RTS |  | RTS | 7 |
| 8 | CTS | | CTS | 8 |
| 9 | NC | | RI | 9 |

### 3.3.3 Power Connector

Two power interface in this board, CN7 and CN8,

Figure 11. CN8 Connector

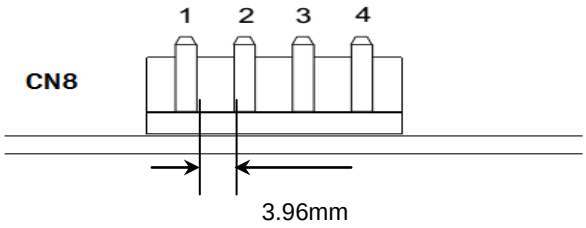
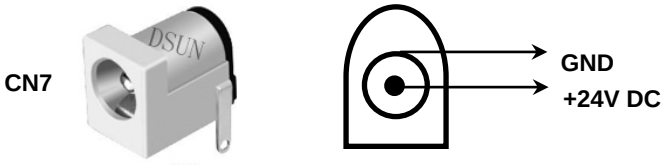



Table 6. CN8 Pin Assignments

| Pin Number | Signal Name |
|---|---|
| 1 | GND |
| 2 | +24V DC |
| 3 | +24V DC |
| 4 | GND |

CN7 Power Port: DC005-F, 5.5mm * 2.1mm

Figure 12. CN7Connect

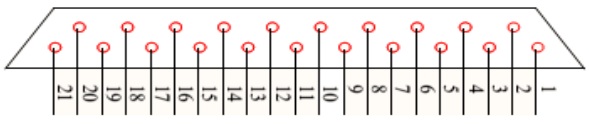



### 3.3.4 Print Head Connector

Connect print head and control board needs 2 FFC connectors, a connector connects motor drive, a connector connect sensor and control signal, the connectors detail description as below:

Sensor and control signal connector:

Figure 13. CN4 Connector:



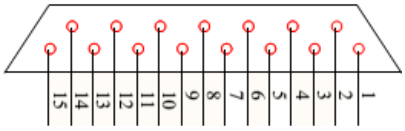

**CN4**

Table 7. CN4 Pin Assignments

| No. | Signal Name | Description |
|---|---|---|
| 1 | COM | Head drive power |
| 2 | COM | Head drive power |
| 3 | COM | Head drive power |
| 4 | NC | Non-contact |
| 5 | $\overline{\text{LAT}}$ | Data latch |
| 6 | NC | Non-contact |
| 7 | GND | Head ground |
| 8 | GND | Head ground |
| 9 | GND | Head ground |
| 10 | TM | Thermostat |
| 11 | $\overline{\text{STR1}}$ | Strobe 1 |
| 12 | VDD | Logic power |
| 13 | $\overline{\text{STR2}}$ | Strobe 2 |
| 14 | GND | Head ground |
| 15 | GND | Head ground |
| 16 | GND | Head ground |
| 17 | CLOCK | Clock |
| 18 | SI | Data in |
| 19 | COM | Head drive power |
| 20 | COM | Head drive power |
| 21 | COM | Head drive power |

Motor drive connector:

Figure 14. CN2 Connector:




**CN2**

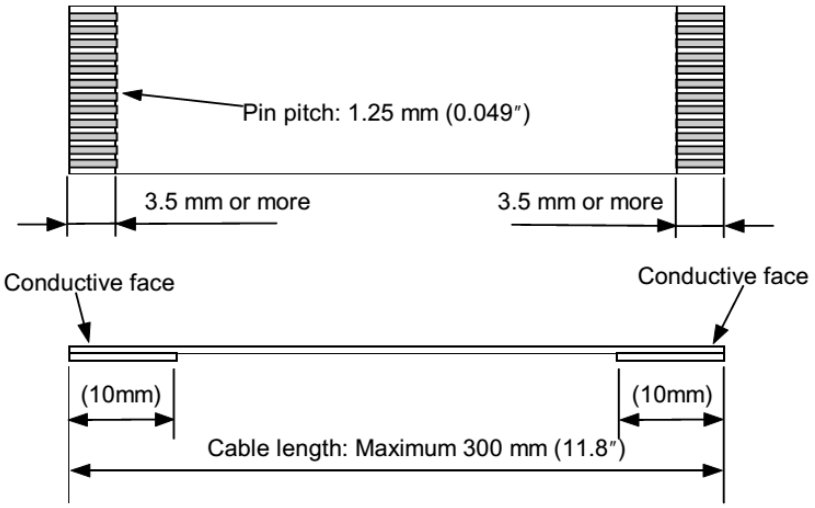
Table 8. CN2 Pin Assignments

| No. | Signal Name | Description |
|---|---|---|
| 1 | Paper feed motor phase /A | Excitation signal /A |
| 2 | Paper feed motor phase B | Excitation signal A |
| 3 | Paper feed motor phase A | Excitation signal A |
| 4 | Paper feed motor phase /B | Excitation signal /B |
| 5 | Auto cutter motor (+) | Excitation signal (+) |
| 6 | Auto cutter motor (−) | Excitation signal (+) |
| 7 | GND | Head ground |

| | | |
|---|---|---|
| 8 | GND | Head ground |
| 9 | Auto cutter reset | The cutter reset signal is generated by recognizing both edges for rising and falling signal of the rectified reset signal from the auto cutter. |
| 10 | GND | Head ground |
| 11 | Paper-end sensor output | A photo sensor is used to detect paper-end. |
| 12 | Black mark sensor output | A photo sensor is used to detect a black mark in the path near the thermal head. |
| 13 | VDD(+5V) | Logic power |
| 14 | Platen open SW2 | A sensor is used to detect whether the platen is loaded or unloaded. |
| 15 | Platen open SW1 | A sensor is used to detect whether the platen is loaded or unloaded. |

FFC cable connects the print head and control board, the FFC standard as below:

Figure 15. FFC Cable




### 3.3.5 Debug Port and Expand Connector

CN6 use to connect the debugger J-Link and board, as below:

Figure 16. CN6 Connector

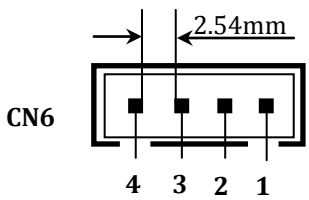


Table 9. CN6 Pin Assignments

| Pin Number | Signal Name |
|---|---|
| 1 | VCC |
| 2 | SWD IO |
| 3 | SWD CLK |
| 4 | GND |

CN9 is the expand front board connector

Figure 17. CN9 Connector

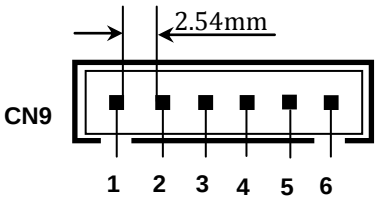



Table 10. CN6 Pin Assignments

| Pin Number | Signal Name |
|---|---|
| 1 | VCC |
| 2 | Power LED |
| 3 | Paper end LED |
| 4 | Error LED |
| 5 | Feed KEY |
| 6 | GND |

CN10 is the paper near-end sensor expands port.

Figure 18. CN10 Connector

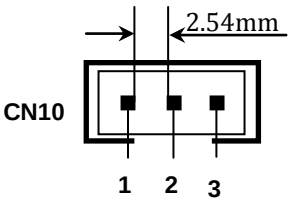



Table 11. CN6 Pin Assignments

| Pin Number | Signal Name |
|---|---|
| 1 | GND |
| 2 | Paper near-end Sensor |
| 3 | VCC |

## 3.4 Configuration

DIP switch is configure the communication mode, enable or disable black mark, select UART baud rate, select print density and select print speed.

Figure 19. DIP Switch




**S1**

Jumped JMP1 to select the MCU MOD0

Figure 20. MOD0 JMP1:

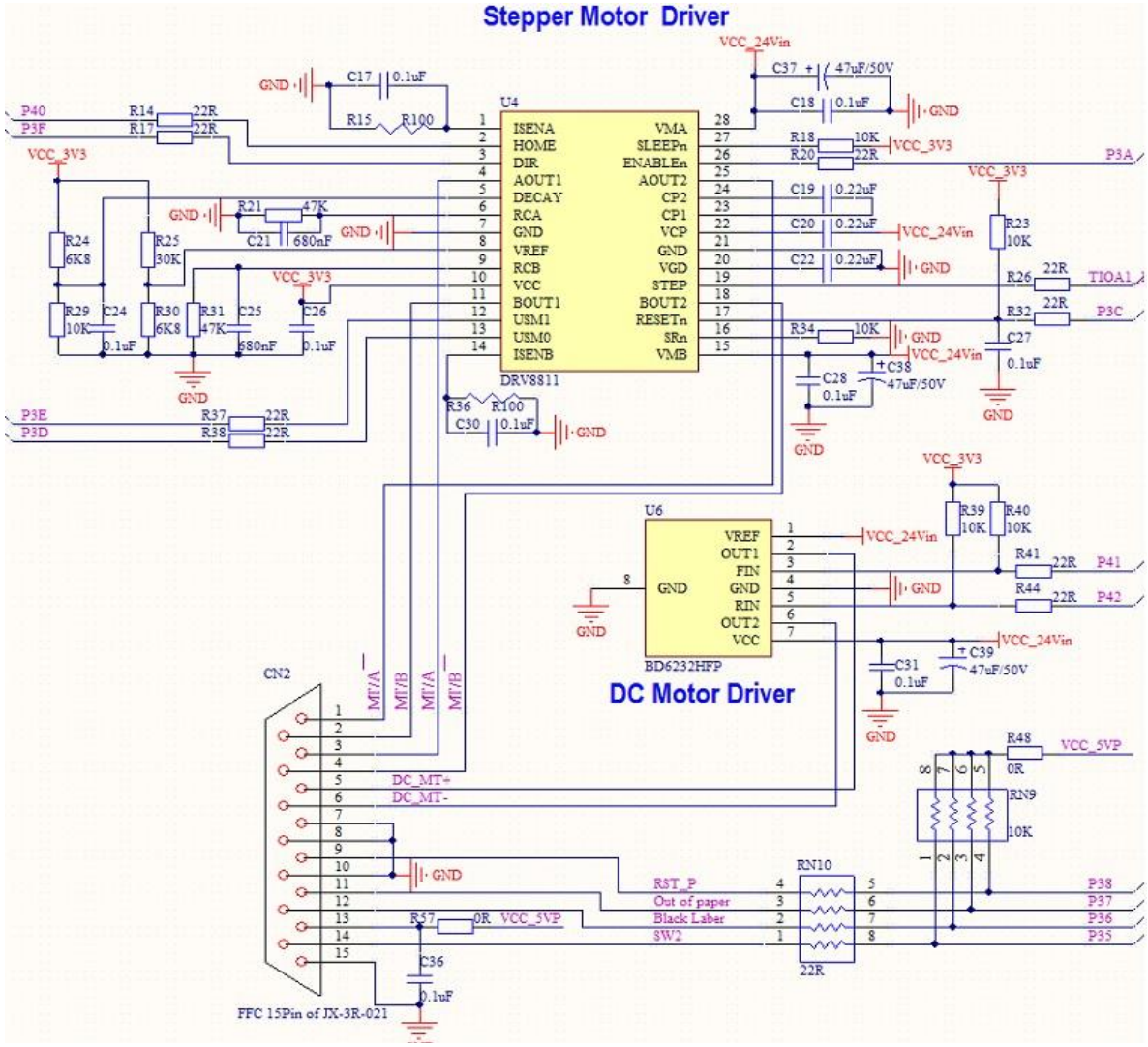


**JMP1**

## 3.5 Printing Control Circuit

DRV8811 was used as stepper motor drive, the logic pulse including the MT_A, MT_/A, MT_B, MT_/B signals, and BD6232HFP was used as DC motor drive, the logic pulse including DC_MT+ and DC_MT-.This connecter also includes four detective signals: the RST signal of automatic cutter, the out of paper signal, the black label signal and the rubber roller ready signal.

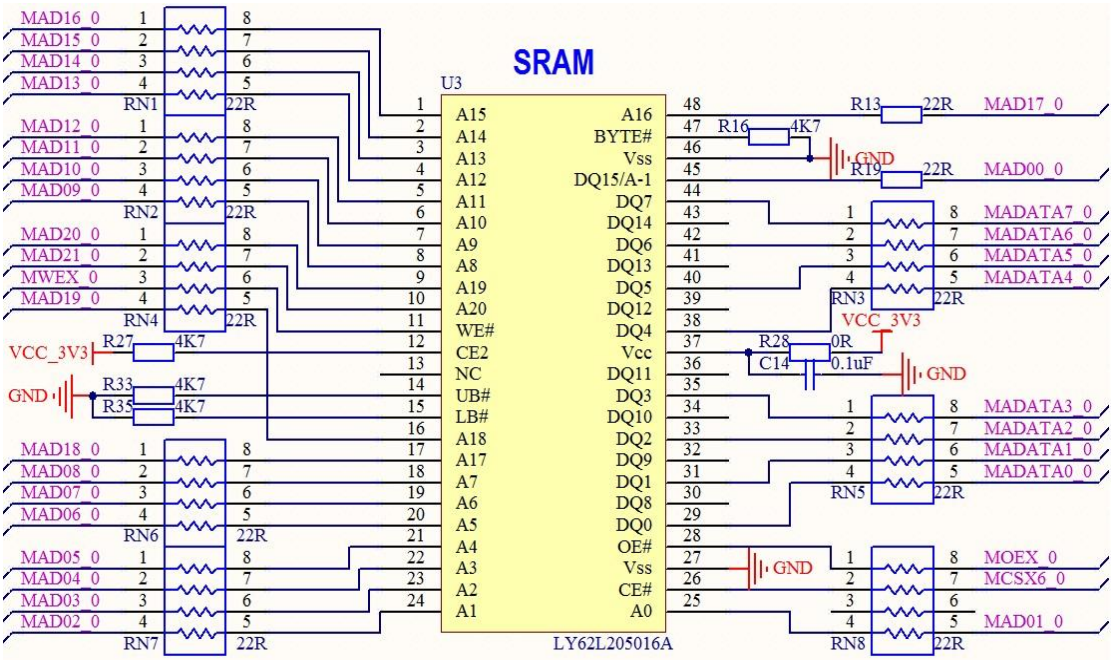
Figure 21. Printing Control Circuit



## 3.6 External Storage

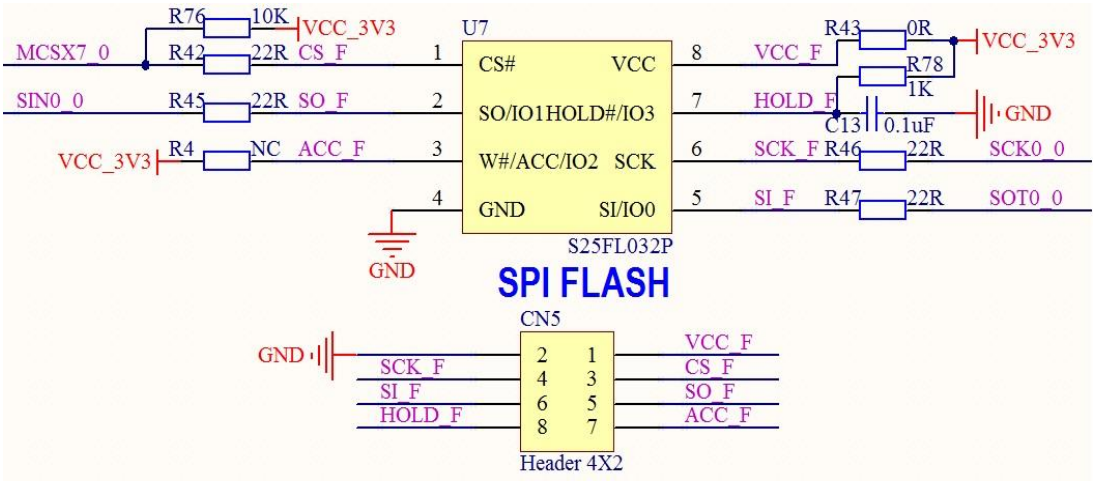
As a 32M Bits (2Mx16 / 4Mx8 Switchable) LOW POWER CMOS SRAM, the LY62L205016A is fabricated using very high performance, high reliability CMOS technology. Its standby current is stable within the range of operating temperature.

Figure 22. Schematic for SRAM




The S25FL032P is a 3.0 Volt (2.7V to 3.6V), single-power-supply Flash memory device. The device consists of 64 uniform 64 KB sectors with the two (Top or Bottom) 64 KB sectors further split up into thirty-two 4KB sub sectors. The S25FL032P device is fully backward compatible with the S25FL032A device. The device accepts data written to SI (Serial Input) and outputs data on SO (Serial Output).

Figure 23. Schematic for SPI Flash



# 4. Function

## 4.1 Command List

This system control commands compatible for ESC/POS command, the command list as Table 12.

Table 12. Command List

| Command | Name | Command Classification | |
|---|---|---|---|
| | | Executing | Setting |
| HT | Horizontal tab | ○ | |
| LF | Print and line feed | ○ | |
| FF | Print and feed label to print starting position (*) | ○ | |
| CR | Print and carriage return | ○ | |
| ESC SP | Set right-side character spacing | | ○ |
| ESC ! | Select print mode(s) | | ○ |
| ESC $ | Set absolute print position | ○ | |
| ESC % | Select/cancel user-defined character set | | ○ |
| ESC & | Define user-defined characters | | ○ |
| ESC * | Select bit-image mode | ○ | |
| ESC - | Turn underline mode on/off | | ○ |
| ESC 2 | Select default line spacing | | ○ |
| ESC 3 | Set line spacing | | ○ |
| ESC ? | Cancel user-defined characters | | ○ |
| ESC @ | Initialize printer | ○ | ○ |
| ESC D | Set horizontal tab positions | | ○ |
| ESC E | Turn emphasized mode on/off | | ○ |
| ESC G | Turn double-strike mode on/off | | ○ |
| ESC J | Print and feed paper | ○ | |
| ESC M | Select character font | | |
| ESC S | Select standard mode | ○ | |
| ESC V | Turn 90°clockwise rotation mode on/off | | ○ |
| ESC \ | Set relative print position | ○ | |
| ESC a | Select justification | | ○ |
| ESC c 3 | Select paper sensor(s) to output paper-end signals | | ○ |

| | | | |
|---|---|---|---|
| ESC c 4 | Select paper sensor(s) to stop printing | | ◯ |
| ESC c 5 | Enable/disable panel buttons | | ◯ |
| ESC d | Print and feed n lines | ◯ | |
| ESC { | Turn upside-down printing mode on/off | | ◯ |
| FS p | Print NV bit image | ◯ | |
| FS q | Define NV bit image | | ◯ |
| GS FF | Feed marked paper to print starting position | ◯ | |
| GS ! | Select character size | | ◯ |
| GS * | Define downloaded bit image | | ◯ |
| GS ( A | Execute test print | ◯ | |
| GS ( F | Set adjustment values(s) | | ◯ |
| GS ( K | Select print control method(s) | | ◯ |
| GS ( M | Customize printer control value(s) | | ◯ |
| GS ( k | Setup and print QR code | ◯ | ◯ |
| GS / | Print downloaded bit image | ◯ | |
| GS : | Start/end macro definition | ◯ | ◯ |
| GS B | Turn white/black reverse printing mode on/off | | ◯ |
| GS C 0 | Select counter print mode | | ◯ |
| GS C 1 | Select count mode (A) | | ◯ |
| GS C 2 | Set counter | | ◯ |
| GS C ; | Select count mode (B) | | ◯ |
| GS E | Select head control method | | ◯ |
| GS H | Select printing position of HRI characters | | ◯ |
| GS I | Transmit printer ID | ◯ | |
| GS L | Set left margin | | ◯ |
| GS T | Set print position to the beginning of print line | ◯ | |
| GS V | Select cut mode and cut paper | ◯ | |
| GS W | Set printing area width | | ◯ |
| GS ^ | Execute macro | ◯ | |
| GS c | Print counter | ◯ | |
| GS f | Select font for HRI characters | | ◯ |
| GS h | Set bar code height | | ◯ |
| GS k | Print bar code | ◯ | |
| GS r | Transmit status | ◯ | |
| GS v 0 | Print raster bit image | ◯ | |
| GS w | Set bar code width | | ◯ |
| FS ! | Set print mode(s) for Chinese characters | | ◯ |
| FS & | Select Chinese character mode | | ◯ |
| FS - | Turn underline mode on/off for Chinese characters | | ◯ |
| FS . | Cancel Chinese character mode | | ◯ |

| | | | |
|---|---|---|---|
| FS 2 | Define user-defined Chinese characters | | ◯ |
| FS S | Set left- and right-side Chinese character spacing | | ◯ |
| FS W | Turn quadruple-size mode on/off for Chinese characters | | ◯ |

Notes:

◯: Enabled.

## 4.2 Character Printing

### 4.2.1 Character Lib

This system support multiple character, it include ASCII and Simplified Chinese as Table 13

Table 13. Character Table

| Support Character | Font Style | Remark |
|---|---|---|
| ASCII | Font A, 12 x 24 | Rang: 0x20-0x7E |
| | Font B, 9 x 17 | Rang: 0x20-0x7E |
| Simplified Chinese | SongTi, 24 x 24 | GB18030-2000 |

Change and select ASCII character by command **ESC !**and **ESC M**, the ASCII include font A and font B, the font A size is 12dots x 24 dots, store one font A ASCII character to memory need 48 bytes; and the font B size is 9dots x 17dots, store one font B ASCII character to memory need 34 bytes.

Figure 24. ASCII Font A

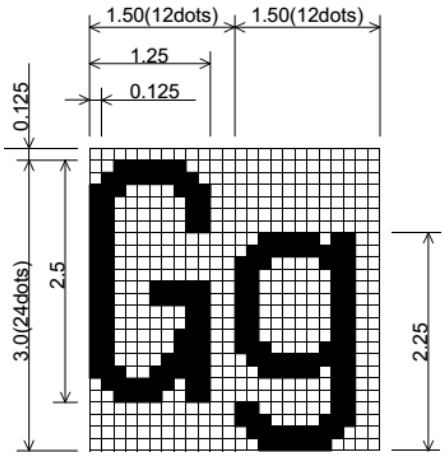
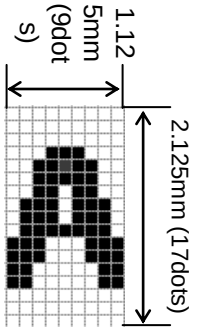
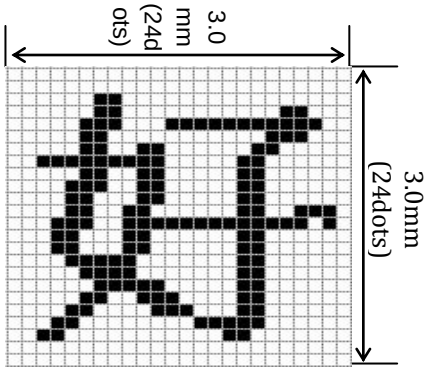


Figure 25. ASCII Font B




Select the simplified Chinese by command **FS &** and cancel the simplified Chinese by command **FS .**, the simplified Chinese font family is **SongTi**, the size is 24dots x 24dots, store one simplified Chinese in memory need 72 bytes.

Figure 26. Simplified Chinese



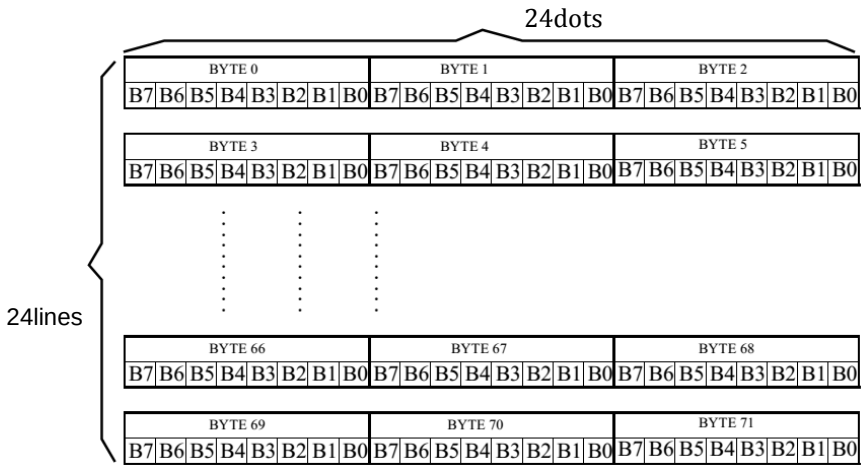

ASCII and simplified Chinese the data get and store is use horizontal direction, as Figure 27 and Figure 28.

Figure 27. ASCII Font A Data

| | MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|---|
| d1 | | | | | | | | |
| d3 | | | | | • | • | • | • |
| d5 | | | • | • | • | • | • | • |
| d7 | | | • | • | | | | |
| d9 | | • | • | | | | | |
| d11 | | • | • | | | | | |
| d13 | | • | • | | | | | |
| d15 | | | | | | | | |
| d17 | | | | | | | | |
| d19 | | | | | | | | |
| d21 | | | | | | | | • |
| d23 | | | | | | | • | • |
| d25 | | | | | | | • | • |
| d27 | | | | | | • | • | |
| d29 | | | | | • | • | • | |
| d31 | | | | • | • | • | | |
| d33 | | | | • | • | | | |
| d35 | | | • | • | | | | |
| d37 | | • | • | • | | | | |
| d39 | | • | • | • | • | • | • | • |
| d41 | | • | • | • | • | • | • | • |
| d43 | | | | | | | | |
| d45 | | | | | | | | |
| d47 | | | | | | | | |

| | MSB | | | | | | | LSB |
|---|---|---|---|---|---|---|---|---|
| d2 | | | | | 0 | 0 | 0 | 0 |
| d4 | | | | | 0 | 0 | 0 | 0 |
| d6 | • | • | | | 0 | 0 | 0 | 0 |
| d8 | • | • | | | 0 | 0 | 0 | 0 |
| d10 | | • | • | | 0 | 0 | 0 | 0 |
| d12 | | • | • | | 0 | 0 | 0 | 0 |
| d14 | | • | • | | 0 | 0 | 0 | 0 |
| d16 | | • | • | | 0 | 0 | 0 | 0 |
| d18 | • | • | | | 0 | 0 | 0 | 0 |
| d20 | • | • | | | 0 | 0 | 0 | 0 |
| d22 | • | | | | 0 | 0 | 0 | 0 |
| d24 | | | | | 0 | 0 | 0 | 0 |
| d26 | | | | | 0 | 0 | 0 | 0 |
| d28 | | | | | 0 | 0 | 0 | 0 |
| d30 | | | | | 0 | 0 | 0 | 0 |
| d32 | | | | | 0 | 0 | 0 | 0 |
| d34 | | | | | 0 | 0 | 0 | 0 |
| d36 | | | | | 0 | 0 | 0 | 0 |
| d38 | | | | | 0 | 0 | 0 | 0 |
| d40 | • | • | • | | 0 | 0 | 0 | 0 |
| d42 | • | • | • | | 0 | 0 | 0 | 0 |
| d44 | | | | | 0 | 0 | 0 | 0 |
| d46 | | | | | 0 | 0 | 0 | 0 |
| d48 | | | | | 0 | 0 | 0 | 0 |

Figure 28. Simplified Chinese Data




## 4.2.2 User-Define Characters

In order to extend and modify the character, system support user define characters, User-Define ASCII and User-Define simplified Chinese, define ASCII User-Define by command **ESC &,**use command **ESC %**to enable or disable the ASCII User-Define, ASCII User-Define characters is use for replace existing ASCII characters. However, the simplified Chinese User-Define are expanding the simplified Chinese, the User-Define rang is from 0xFEA1 to 0xFEFE, use command **FS 2** to define the User-Define simplified Chinese.

The User-Define data is different from existing characters; User-Define data get direction is vertical, as Figure 29, Figure 30 and Figure 31.

Figure 29. ASCII Font A User-Define Character

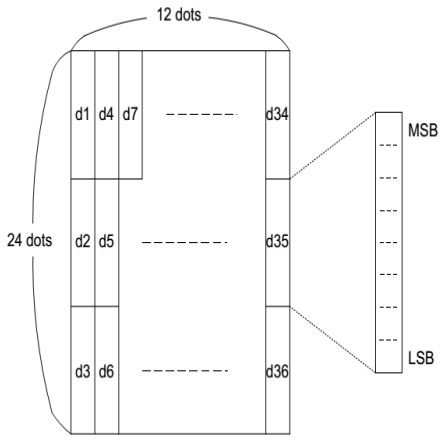


Figure 30. ASCII Font B User-Define Character

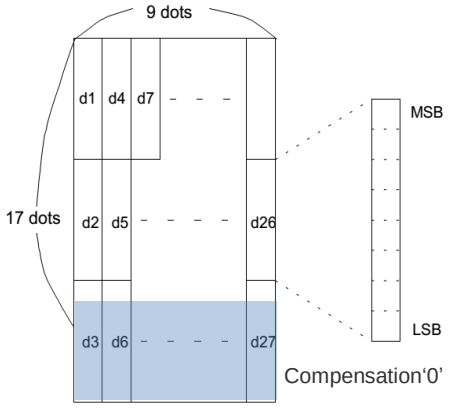
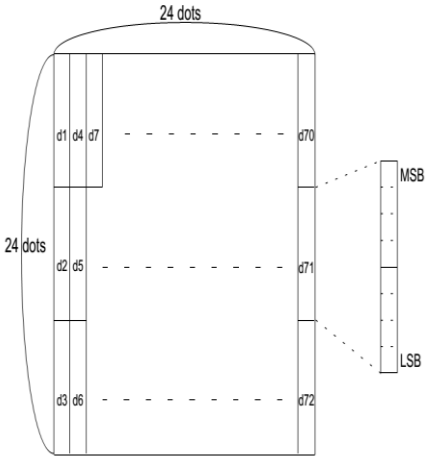



Figure 31. Simplified Chinese User-Define Character




### 4.2.3 Character Font Style

This system support multiple font styles, these font style applicable to ASCII, simplified Chinese, User-Define ASCII and User-Define simplified Chinese, as below:

**Font Style:**

Bold, reference command **ESC !**,**ESC E**,**ESC G**

Underline, the underline include 1 dot underline and 2 dots underline, reference command

**ESC !, ESC -, FS!,FS -**

Rotate 90° Clockwise, reference command **ESC V**.

Upside-Down, reference command **ESC {**.

White/Black Reverse, reference command **GS B**

**Font Size:**

Double Width, reference command **ESC !**, **GS !**, **FS !**, **FS S**.

Double Height, reference command **ESC !**, **GS !**, **FS !**, **FS S**.

## 4.3 Bitmap Printing

This system support multiple bitmap mode, as Table 14

Table 14. Bitmap Mode

| Bitmap Mode | Description | Data Mode | Commands |
|---|---|---|---|
| Bit-image mode | Using number of dots and dot length specified by length | Vertical get data | ESC * |
| NV Bitmap | Store the bitmap to external flash, and print the bitmap from external flash. | Vertical get data | FS p, FS q |
| Downloaded Bitmap | Downloaded bitmap to internal RAM, print the bitmap from internal RAM | Vertical get data | GS *, GS / |
| Raster Bitmap | Real time to print bitmap | Horizontal get data | GS v 0 |

Figure 32. Horizontal Get Data

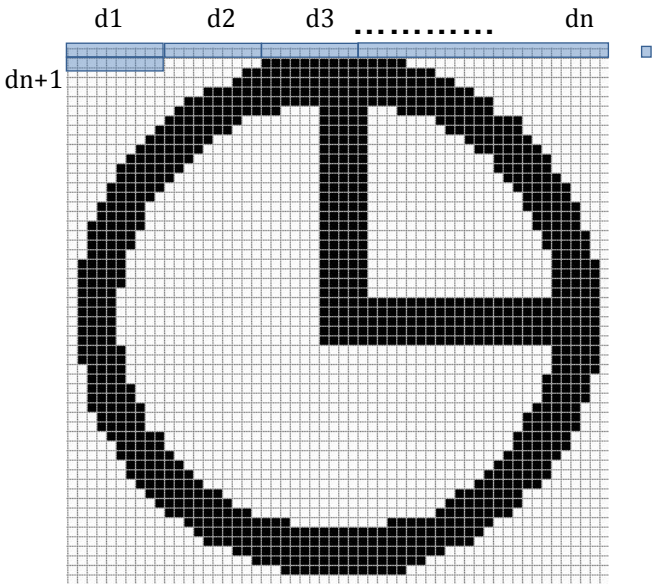
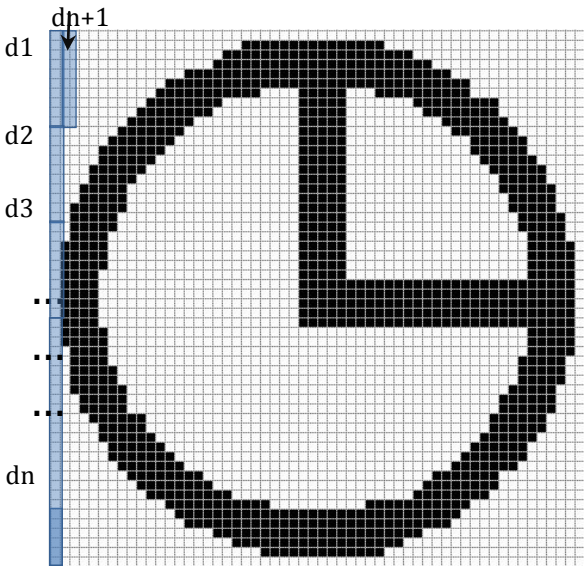


Figure 33. Vertical Get Data




**Bitmap Format:**

- Double Width
- Double Height
- Upside-Down

## 4.4 Barcode Printing

This system include 4 barcode, EAN 8, EAN 13, Code 128 and QR code

Print EAN 8 barcode by **GS k.**

Figure 34. EAN 8




Print EAN 13 barcode by **GS k.**

Figure 35. EAN 13




Print Code 128 barcode by **GS k.**

Figure 36. Code 128




The QR code max size is 250dots x 250dots in this system, so the QR code support under version 20, print EAN 13 barcode by command **GS ( k.**

Figure 37. QR Code



## 4.5 Interface Introduction

### 4.5.1 DIP Switch

Configure system mode and function by DIP switch S1, as shown in Figure 38

Figure 38. DIP Switch Layout




**S1**

DIP Switch configures description:

Table 15. DIP S1

| S1 No. | Description | ON | OFF | Default |
|---|---|---|---|---|
| 1 | Communication Selection | UART | USB | OFF |
| 2 | BM Sensor | Enable | Disable | OFF |
| 3 | Serial interface baud rate selection | Refer to Table 16 | | OFF |
| 4 | | | | OFF |
| 5 | Print density selection | Refer to Table 17 | | OFF |
| 6 | | | | OFF |
| 7 | Printing selection | Refer to Table 18 | | OFF |
| 8 | | | | OFF |

Baud rate configure:

Table 16. DIP S1

| Baud rate (bps) | Switch Number | |
|---|---|---|
| | 3 | 4 |
| 115200 | OFF | OFF |
| 38400 | OFF | ON |
| 19200 | ON | OFF |
| 9600 | ON | ON |

Print Density configures:

Table 17. DIP S1

| Print Density | Switch Number | |
|---|---|---|
| | 5 | 6 |
| Normal | OFF | OFF |
| Dark | OFF | ON |
| Slightly dark | ON | OFF |
| Slightly Light | ON | ON |

Print Speed configures:

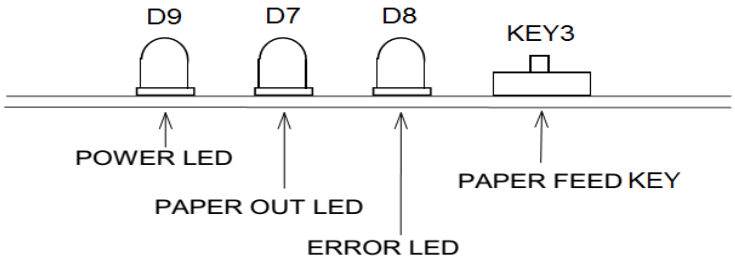
Table 18. DIP S1

| Print Speed(mm/s) | Switch Number | |
|---|---|---|
| | 7 | 8 |
| 150 | OFF | OFF |
| 100 | OFF | ON |
| 50 | ON | OFF |
| Reserved | ON | ON |

### 4.5.2 LED and Button

This system has 6 LEDs and 2 buttons, LED description as shown in Table 19

Figure 39: LED and Button




LED Indication:

Table 19. LED Indication

| LED Indicators | Mark | Color | Status | |
|---|---|---|---|---|
| | | | OFF | ON |
| Power LED | D2, D9 | Green | Power OFF | Power ON |
| Paper End LED | D6, D7 | Red | Paper loaded | Paper end or near end |
| Error LED | D5, D8 | Red | Normal | Error status, refer to 4.9 |

**Buttons Function:**

- When the BM (black mark) sensor is disabled, the printer feeds paper one line, based on the line spacing set by **ESC 2**and **ESC 3**. When the BM sensor is enabled, the printer will feed paper by mark paper unit. Paper feeding using the paper FEED button cannot be performed under the following conditions:

a. The panel buttons are disabled by **ESC c 5**.

b. The paper roll end sensor detects a paper end.

c. When the platen cover is open.

- If you press this button when the printer is in the macro execution standby state, the defined macro is executed.
- During self-test printing, you can stop the self-test temporarily by pressing this button and restart it by pressing the button again.

## 4.6 Self-Test Introduction

Self-test include startup mode and command mode, command mode refer to command GS ( A, and the startup mode as below:

1. The printer has a self-test function that checks the following:
   - Status Printing

   a. Firmware version

   b. USB interface status

   c. UART interface status

   d. Buffer capacity

   e. DIP switch settings

   f. Logo printing

   g. Barcode printing
   - Character Printing
2. Starting the self-test

   To start the self-test on a paper roll, Power on or reset, and hold down the FEED button then turn on the printer. Then the current printer status is printed.
3. Self-test standby state

   After printing the current printer status, the printer prints the message "Self-test printing. Please press FEED button". Press the FEED button to start the rolling character printing.
4. Ending the self-test

   After a number of lines are printed, the printer indicates the end of the self-test by printing"### completed ###," initializes, and goes into the standard mode.

## 4.7 Sensor Check

### 4.7.1 Paper Sensors

The printer has 2 paper sensors as follows:

1. Paper roll end sensor

   The sensor that detects whether paper is present or not.

   When the sensor detects a paper-end, the printer stops printing.

2. Paper roll near-end sensor

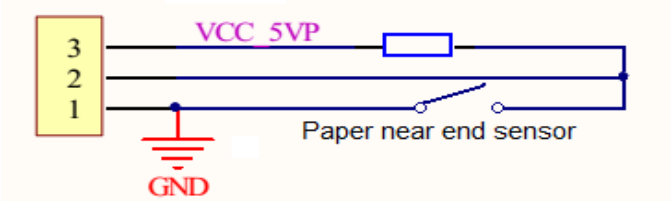
   The sensor that detects a near-end of a paper roll.

   When the paper roll diameter becomes sufficiently small, the sensor detects a near-end of the paper roll and the PAPER OUT LED indicator lights. If the sensor is enabled by **ESC c4**, the printer stops printing.

NOTES:

- After installing new paper roll, close the platen cover; then the printer restarts printing.
- The near-end sensor will be provided by user.

Figure 40. Paper Near End Sensor




### 4.7.2 Platen Open Sensor

The platen open sensor monitors the open/close status of the platen cover which covers the paper feeding rubber rollers. When the sensor detects a platen open while the printer is in the standby status, the printer goes offline. The printer recovers when the platen is closed.

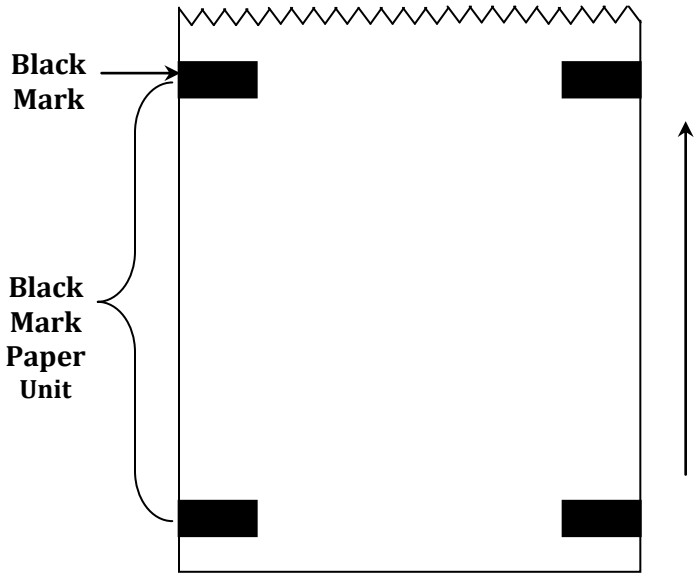
### 4.7.3 Head Temperature Sensor

The temperature sensor is a thermostat, the resistance value of thermostat increasing with the increasing of the temperature, sampling the voltage to calculate the temperature, this system temperature threshold is 65°C, when the temperature over the threshold temperature, the printer stop to save the print head, until the temperature decrease to 45°C, the printer resume.

## 4.8 Black Mark Function

The black mark sensor must be installed in either of the following- back of the paper. The standard position of the black mark is on the right side on back of the paper, enter to Black Mark mode need setting DIP Switch before system startup, more detail please refer to 4.5.1 The Black Mark mode operate unit is the Black Mark Paper Unit, not a line, feed paper and print every operation is a Black Mark Paper Unit.

Figure 41. Black Mark Paper




## 4.9 Error Processing

- Automatically Recover Errors
  - ☐ Print head high temperature error: The temperature of the print head exceeds 65°C, when the Error Occurs stops all printer operations, goes BUSY and blinks the ERROR LED.
- Possibility of Recovery Errors
  - ☐ Platen open error: Printing is not performed due to a platen open, when the Error Occurs stops all printer operations, goes BUSY and blinks the ERROR LED.
  - ☐ Auto cutter error: The auto cutter does not work correctly, when the Error Occurs stops all printer operations, goes BUSY and blinks the ERROR LED.
  - ☐ BM sensor detection error: No black mark is detected even though the paper roll is marked correctly, when the Error Occurs stops all printer operations, goes BUSY and blinks the ERROR LED.
- Impossible to Recover Errors
  - ☐ CPU execution error: The CPU executes an incorrect address or the I/F board is not connected, when the Error Occurs stops all printer operations, goes BUSY and blinks the ERROR LED.

# 5. Firmware W Introduction

## 5.1 Firmware Overview

There are 6 folders in the FW structure of IAR, which is shown in Figure 42 and Table 20

Figure 42. Structure of FW

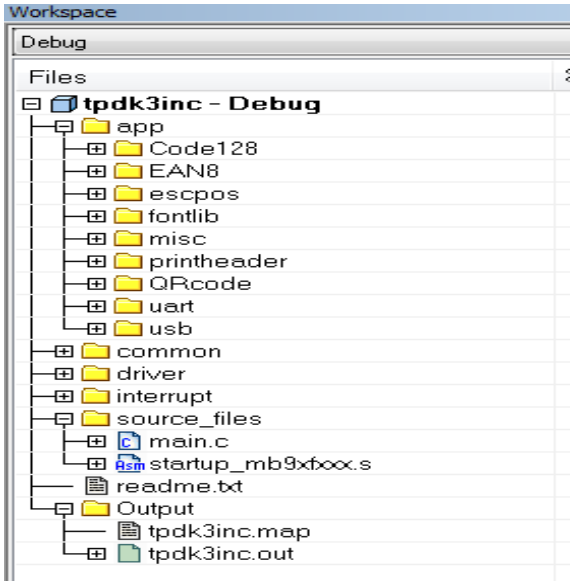



Table 20. Directory Description of Project

| Folder | Description |
|---|---|
| app | It includes escpos, font lib, print head, communication, data process and qr code and barcode module |
| common | It includes system file and global macro define, register define and FM3 system initialization functions |
| driver | It includes FM3 MCU peripheral driver library |
| interrupt | It includes interrupt handle and copy back function |
| tpdk project | It includes IAR project file and system main function |
| readme.txt | It records the version log and upgrade information of firmware |

The detail file description is shown in Table 21

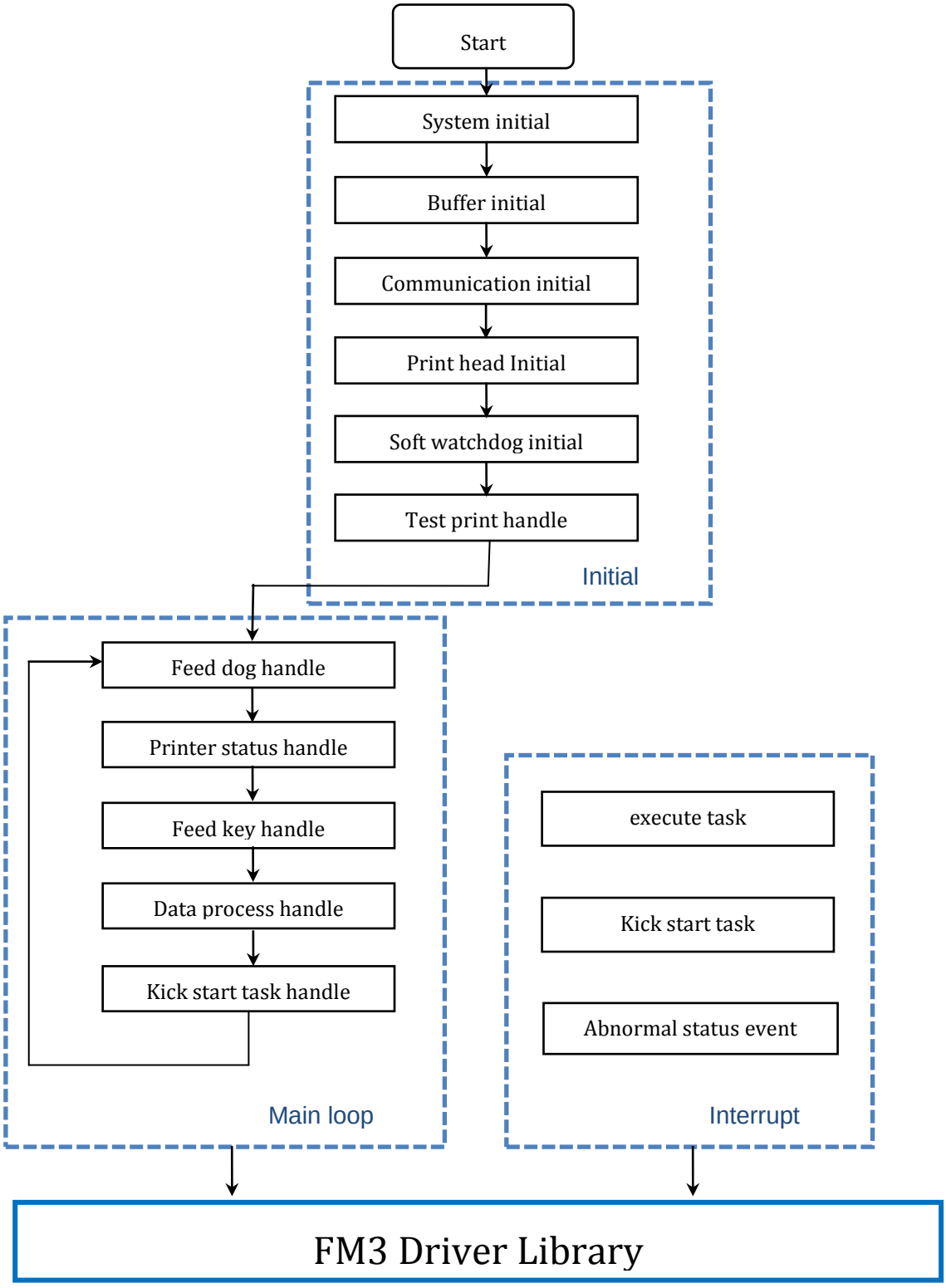
Table 21. File Description of Project

| Folder | File | Description | Mark |
|---|---|---|---|
| app\Code128 | Code128.c | Barcode Code128 encode | N/A |
| | Code128.h | Include Barcode Code128 encode functions and global variables | N/A |
| app\EAN8 | EAN8.c | Barcode EAN8 and EAN13 encode | N/A |
| | EAN8.h | Include Barcode EAN8 and EAN13 encode functions and global variables | N/A |
| app\escpos | dataprocess.c | analysis printer data and page info data process | N/A |
| | dataprocess.h | Include data process functions and global variables | N/A |
| | escpos.c | Analysis the printer data base on ESC/POS standard | N/A |
| | escpos.h | Include ESC/POS analysis functions and global variables | N/A |
| | fifo.c | Definition and operation for buffer | N/A |
| | fifo.h | Include buffer control functions and global variables | N/A |
| | logodata.c | define logo data for test print | N/A |
| | logodata.h | Include definition for logo data | N/A |
| | statuscheck.c | Definition and operation for printer status | N/A |
| | statuscheck.h | Include status check functions and global variables | N/A |
| | task.c | Definition and operation for task FIFO | N/A |
| | task.h | Include task functions and global variables | N/A |
| app\fontlib | ascii1224.c | ASCII font library | N/A |
| | ascii1224.h | Include ASCII font library global variable | N/A |
| | character.c | character and bitmap function | N/A |
| | character.h | Include character and bitmap function and global variables | N/A |
| | chinese_songti2424.c | Definition font of Chinese character for test | N/A |
| | chinese_songti2424.h | Include Chinese character font library global variable | N/A |
| | csio_int_s25fl127s.c | Provides basic operation APIs to access to S25FL127S | N/A |
| | csio_int_s25fl127s.h | Include character and bitmap function and global variables | N/A |
| | Style_convert.c | convert the style for font library | N/A |
| | Style_convert.h | Include convert the style function and global variables | N/A |
| app\misc | adc.c | Initial and operate ADC | N/A |
| | adc.h | Include Initial and operate ADC functions, and global variable | N/A |
| | dt.c | define and operation timer | N/A |
| | dt.h | Include operation timer functions, and global | N/A |

| | | | |
|---|---|---|---|
| | | variable | |
| | frt.c | define and operation timer | N/A |
| | frt.h | Include operation timer functions, and global variable | N/A |
| | misc.c | Include LED, beep, key operations and configuration. | N/A |
| | misc.h | Include operations and configuration functions and global variable | N/A |
| | pwm.c | definition and operation for PWM module | N/A |
| | pwm.h | Include operations for PWM functions and global variable | N/A |
| | sys_if.c | System interrupt handle | N/A |
| | sys_if.h | System interrupt handle functions | N/A |
| app\printheader | printheadcsio.c | Initial and operate CSIO functions, and global variable. | N/A |
| | printheadcsio.h | Include CSIO operations and configuration functions and global variable. | N/A |
| | printheaddriver.c | Initial and operate print head functions, and global variable. | N/A |
| | printheaddriver.h | Include print head operations and configuration functions and global variable. | N/A |
| | printheadpin.c | Initial and operate print head pin functions, and global variable. | N/A |
| | printheadpin.h | Include print head pin operations and configuration functions and global variable. | N/A |
| app\QRcode | GenericGF.c | | N/A |
| | GenericGF.h | | N/A |
| | Qrcode.c | | N/A |
| | Qrcode.h | | N/A |
| app\uart | uart.c | Definition and operation for UART module | N/A |
| | uart.h | Include Definition and operation for UARTmodule functions and global variable | N/A |
| app\usb | UsbConfig.c | Initial USB module and switch USB status | N/A |
| | UsbConfig.h | Include Initial USB module and switch status functions and global variable | N/A |
| | UsbDevicePrinter.c | Read and write operation via USB printer device | N/A |
| | UsbDevicePrinter.h | Include operation via USB printer device functions and global variable | N/A |
| common | System_mb9xfxxx.s | system initialization functions | N/A |
| driver | Slide over | Slide over it, please take details in document of FM3 PDL | N/A |

Main routine of the thermal printer development FW as shown in Figure 43

Figure 43. Flow Chart of Main Function



## 5.2 Communication

### 5.2.1 USB Serial Interface

**Specifications:**

General specifications: USB 2.0 Full-speed mode (12Mbps)

Communication method: USB bulk transfer

Power: USB self-powered function device

USB packet size:

USB bulk OUT 64Byte

USB bulk IN 64Byte

USB device class: USB printer class

USB descriptor:

Vendor ID:

Product ID:

**Interface description:**

The USB device is described as a standard USB Printer device in the TPDK project. The hot plug function is supported, the enumeration process on after USB cable plug in or after power on. The printer data can send to printer device when enumeration process completed correctly.

Figure 44. Device Descriptor

```
const uint8_t au8DeviceDescriptor[18] =
{
    ///DEVICE DESCRIPTOR
    0x12,    ///bLength: Length of this descriptor
    0x01,    ///bDescriptorType: Device Descriptor Type
    0x00,    ///bcdUSB: USB Version
    0x02,    ///bcdUSB: USB Version
    0x00,    ///bDeviceClass: Class Code: MISCELLANEOUS
    0x00,    ///bDeviceSubClass: Sub Class Code
    0x00,    ///bDeviceProtocol: Protocol Code
    0x40,    ///bMaxPacketSize0: Maximum size of endpoint 0
    0xb8,//0x6A,    ///idVendor: Vendor ID
    0x04,//0x1A,    ///idVendor: Vendor ID
    0x05,    ///idProduct: Product ID
    0x0e,    ///idProduct: Product ID
    0x00,    ///bcdDevice: Release Number
    0x00,    ///bcdDevice: Release Number
    0x01,    ///iManufacture: String-Index of Manufacture
    0x02,    ///iProduct: String-Index of Product
    0x03,    ///iSerialNumber: String-Index of Serial Number
    0x01     ///bNumConfigurations: Number of possible configurations
};
```

Figure 45. Configure Descriptor

```
const uint8_t au8ConfigDescriptor[84] =
{
  ///NEW CONFIG DESCRIPTOR(1)
    0x09,    ///bLength: Length of this descriptor
    0x02,    ///bDescriptorType: Config Descriptor Type
    0x20,0x00,//0x54, 0x00,  ///wTotalLength: Total Length with all interface- and endpoint descriptors
    0x01,//0x03,    ///bNumInterfaces: Number of interfaces
    0x01,    ///iConfigurationValue: Number of this configuration
    0x00,    ///iConfiguration: String index of this configuration
    0xa0,//0xC0,    ///bmAttributes: Bus-Powered, Remote-Wakeup not supported
    0x32,//0xFA,    ///MaxPower: (in 2mA)
       ///NEW INTERFACE DESCRIPTOR(0)
       0x09,    ///bLength: Length of this descriptor
       0x04,    ///bDescriptorType: Interface Descriptor Type
       0x00,    ///bInterfaceNumber: Interface Number
       0x00,    ///bAlternateSetting: Alternate setting for this interface
       0x02,    ///bNumEndpoints: Number of endpoints in this interface excluding endpoint 0
       0x07,    ///iInterfaceClass: Class Code: PRINTER
       0x01,    ///iInterfaceSubClass: SubClass Code
       0x02,    ///bInterfaceProtocol: Protocol Code
       0x00,    ///iInterface: String index
          ///NEW ENDPOINT DESCRIPTOR(1)
          0x07,    ///bLength: Length of this descriptor
          0x05,    ///bDescriptorType: Endpoint Descriptor Type
          0x81,    ///bEndpointAddress: Endpoint address (IN,EP1)
          0x02,    ///bmAttributes: Transfer Type: BULK_TRANSFER
          0x40,    ///wMaxPacketSize: Endpoint Size
          0x00,    ///wMaxPacketSize: Endpoint Size
          0x00,    ///bIntervall: Polling Intervall
          ///NEW ENDPOINT DESCRIPTOR(0)
          0x07,    ///bLength: Length of this descriptor
          0x05,    ///bDescriptorType: Endpoint Descriptor Type
          0x02,    ///bEndpointAddress: Endpoint address (OUT,EP2)
          0x02,    ///bmAttributes: Transfer Type: BULK_TRANSFER
          0x40,    ///wMaxPacketSize: Endpoint Size
          0x00,    ///wMaxPacketSize: Endpoint Size
          0x00,    ///bIntervall: Polling Intervall

};
```

**API Definition**

| | |
|---|---|
| Name | UsbConfig_UsbInit |
| Parameter | None |
| Return | None |

Description:

This function Initializes USB functions for USB printer device, which trigger the enumeration process. It should be called at system initialization after receive buffer initialized.

| | |
|---|---|
| Name | Usb_PrinterReceive |
| Parameter | pstcEpHandle: Pointer to instance of endpoint<br>pu8Data: Pointer to data buffer<br>pu16ByteCount: The length of data which should be received<br>bClearDrq: If need clear DRQ flag. |
| Return | ok: Receive data successful<br>Error Operation InProgress: Device is in offline mode or receive buffer full |

Description:

This function receives data from USB endpoint FIFO to receive buffer. Need to notice that, this function is called from interrupt process in the project.

The parameter pstcEpHandle is pointed to the instance of endpoint which should receive data. It must be endpoint 2 in the TPDK project.

| | |
|---|---|
| Name | UsbDevice_ReceiveData |
| Parameter | pstcUsb: Pointer to USB instance<br>pstcEndpoint: Pointer to endpoint instance<br>pu8Buffer: Pointer to data buffer<br>u32DataSize: The Size of data to be received.<br>enMode: Send data mode |
| Return | |

Description:

This function Initiate receiving data (OUT direction).

The Parameteren Mode Mode how to send data (polled, irq driven or dma driven, the dma mode is not support in the project).

| | |
|---|---|
| Name | UsbDevice_SendData |
| Parameter | pstcUsb: Pointer to USB instance<br>pstcEndpoint: Pointer to endpoint instance<br>pu8Buffer: Pointer to data buffer<br>u32DataSize: The Size of data to be received<br>enMode: Send data mode |
| Return | |

Description:

This function Initiate sending data (IN direction)

The Parameteren Mode Mode how to send data (polled, irq driven or dma driven, the dma mode is not support in the project)

| | |
|---|---|
| Name | SetEPNS_DRQIE |
| Parameter | None |
| Return | None |

Description:

The function recovered the USB communication in the project. It will be called when printer device is exit from the offline mode or the receive buffer have enough space to storage the data.

### 5.2.2 RS-232 Serial Interface

**Specifications:**

Synchronization: Asynchronous

Handshaking: CTS/RTS

Transmission speed: 9600, 19200, 38400, 115200(bps)

Bit length: 8 bit

Parity Settings: NONE

Stop bit: 1 bit

**Interface description:**

| | |
|---|---|
| Name | MFS_UartInitial |
| Parameter | u8Ch: The channel of UART which should initialed<br>pstcRecBuf: Pointer to the receive data buffer |
| Return | None |

Description:

This function Initializes UART functions for printer device. It should be called at system initialization after receive buffer initialized.

| | |
|---|---|
| Name | MFS_UartPollTx |
| Parameter | u8Ch: The channel of UART which should initialed<br>pu8Data: Pointer to the data buffer with send data<br>u16Size: Send data size |
| Return | None |

Description:

This function Initializes UART functions for printer device. It should be called at system initialization after receive buffer initialized.

| | |
|---|---|
| Name | Mfs0RxIrqHandler |
| Parameter | None |
| Return | None |

Description:

This function store the UART data into the receive buffer. If receive buffer have enough space and the printer device is not in offline mode, the data will be received successfully. Otherwise the UART interrupt request will be disabled until recover the communication module.

## 5.3 ESC/POS Module

### 5.3.1 Status Check

| | |
|---|---|
| Name | Sc_PrintHeadStartUp |
| Parameter | None |
| Return | None |

Description:

Initialize sensor.

| | |
|---|---|
| Name | Sc_StatusCheck |
| Parameter | None |
| Return | None |

Description:

Check printer current status and recover printer operation

### 5.3.2 FIFO

| | |
|---|---|
| Name | FIFO_Create |
| Parameter | u16Size :buffer size |
| Return | None |

Description:

Create FIFO buffer.

| | |
|---|---|
| Name | FIFO_Free |
| Parameter | pstcFifoBuf :Pointer to FIFO buffer |
| Return | None |

Description:

Free FIFO.

| | |
|---|---|
| Name | FIFO_Empty |
| Parameter | pstcFifoBuf :Pointer to FIFObuffer |
| Return | None |

Description:

Empty FIFO

| | |
|---|---|
| Name | FIFO_Write |
| Parameter | pstcFifoBuf :Pointer to FIFObuffer<br>u8char:To be write |
| Return | TRUE: success to write<br>FALSE: fail to write |

Description:

Write data to FIFO buffer

| | |
|---|---|
| Name | FIFO_Read |
| Parameter | pstcFifoBuf : Pointer to FIFObuffer<br>u8char:To be read |
| Return | TRUE: success to read<br>FALSE: fail to read |

Description:

Read data to FIFO buffer.

| | |
|---|---|
| Name | FIFO_Read_NByte |
| Parameter | pstcFifoBuf : Pointer to FIFO buffer<br>u16Num : how many data to read |
| Return | TRUE: success to set<br>FALSE: fail to set |

Description:

Set FIFOout index.

| | |
|---|---|
| Name | FIFO_ReadOnly |
| Parameter | pstcFifoBuf : Pointer to FIFO buffer<br>u16Offset: position for Read |
| Return | None |

Description:

Read data from FIFO without change the data index for FIFO.

| | |
|---|---|
| Name | FIFO_WriteOnly |
| Parameter | pstcFifoBuf : Pointer to FIFO buffer<br>u16Offset: position for writing |
| Return | None |

Description:

Write data from FIFO without change the data index for FIFO.

| | |
|---|---|
| Name | FIFO_CancelReadByte |
| Parameter | pstcFifoBuf : Pointer to FIFO buffer |
| Return | None |

Description:

Cancel reading byte

| | |
|---|---|
| Name | FIFO_GetFreeSpaceSize |
| Parameter | pstcFifoBuf : Pointer to FIFO buffer |
| Return | free space size |

Description:

Get free space size of FIFO buffer.

| Name | FIFO_GetOutIndex |
|---|---|
| Parameter | pstcFifoBuf : Pointer to FIFO buffer |
| Return | Out index |

Description:

Get out index of FIFO buffer

| Name | FIFO_GetBufferSize |
|---|---|
| Parameter | pstcFifoBuf : Pointer to FIFO buffer |
| Return | total space size of FIFO buffer |

Description:

Get total space size of FIFObuffer.

| Name | FIFO_Release |
|---|---|
| Parameter | pstcFifoBuf : Pointer to FIFO buffer<br>uint16_t: release count |
| Return | TRUE: success to release<br>FALSE: fail to release |

Description:

Release FIFO buffer.

| Name | Pwm_StartSoft |
|---|---|
| Parameter | u8Ch:Requested channel |
| Return | None |

Description:

Start PWM

| Name | Pwm_Stop |
|---|---|
| Parameter | u8Ch:Requested channel |
| Return | None |

Description:

Stop PWM.

| Name | Pwm_WriteCycleVal |
|---|---|
| Parameter | u8Ch: requested channel<br>u16Cycle: cycle value |
| Return | None |

Description:

Write cycle value of PWM

| Name | Pwm_WriteDutyVal |
|---|---|
| Parameter | u8Ch: requested channel<br>u16Duty: duty value |
| Return | None |

Description:

Write duty value of PWM.

| Name | Pwm_ReadCurCnt |
|---|---|
| Parameter | u8Ch: requested channel |
| Return | count value |

Description:

Read current count value of PWM timer.

### 5.3.3 Task

| Name | Task_FifoFull |
|---|---|
| Parameter | None |
| Return | 1: FIFO is full<br>0: FIFO is not full |

Description:

Justify whether task FIFO status is full.

| Name | Task_FifoEmpty |
|---|---|
| Parameter | None |
| Return | 1: FIFO is EMPTY<br>0: FIFO is not EMPTY |

Description:

Empty task FIFO.

| Name | Task_Init |
|---|---|
| Parameter | None |
| Return | 1: success to initialize |

Description:

Initialize task FIFO.

### 5.3.4 ESC/POS

ESC/POS module in the TPDK project support part of ESC/POS standard command.

| | |
|---|---|
| Name | EscposAnalysis |
| Parameter | u8EscCmd: The first data in receive buffer |
| Return | SUCCEED: Analysis and execute successfully<br>UNDONE: Data not enough in receive buffer<br>FAILED: Analysis failed |

Description:

This function analysis the data in receive buffer. The function is called when the first data is not in ASCII and character range in data process. It is the main function in ESC POS module in the project.

| | |
|---|---|
| Name | WaitRevBuf |
| Parameter | u16Width: The width of data length(in byte)<br>u16High: The height of data length(in bit) |
| Return | 0x00: Receive data successfully<br>0xFF: Can't receive enough data in time, failed |

Description:

This function wait the data enough in receive buffer. It must be called before read some data from buffer.

| | |
|---|---|
| Name | WaitPrintAndTaskBuf |
| Parameter | u16Width: The width of data length(in byte)<br>u16High: The height of data length(in bit) |
| Return | 0x00: Wait enough space successfully<br>0xFF: Can't receive enough data in time |

Description:

This function wait for the enough space in print buffer and one space in task FIFO. It must be called before write a manageable number of data in print buffer or before write a task in to task FIFO.

## 5.4 Data Process

### 5.4.1 Data analysis

| | |
|---|---|
| Name | BufDataProcess |
| Parameter | None |
| Return | None |

Description:

This function analysis the data in the receive buffer.

| | |
|---|---|
| Name | BufDataProcessMain |
| Parameter | None |
| Return | None |

Description:

This function is the main process of data process. It will switch to execute macro command or analysis receive data in receive buffer.

| | |
|---|---|
| Name | DataProcessIni |
| Parameter | None |
| Return | None |

Description:

These functions initial the status of data process and ESC/POS process.

### 5.4.2 Page data process

| | |
|---|---|
| Name | PrintCount |
| Parameter | u16Counter: The number printed<br>u16PosX: The position of the number printed in X direction<br>u16PosY: The position of the number printed in Y direction |
| Return | None |

Description:

This function is printing a number.

The Parameteru16Counter is.

The Parameteru16PosY is the position of the number printed in Y direction, now this parameter is not take effect because of the page mode is not supported.

| | |
|---|---|
| Name | PutFontToBuf |
| Parameter | u16CurPos: The position of the font print<br>character output: Store the font information and the font bit map data |
| Return | None |

Description:

This function should put a font data into print buffer in the relevant position.

| Name | FillCharLine |
|---|---|
| Parameter | u16FillPos: The fill start position(in bit)<br>u16FillLen: Fill length(in bit) |
| Return | None |

Description:

This function fills the print buffer with data 0 in relevant position.

| Name | JustificationPro |
|---|---|
| Parameter | u16Height: The height  of character which should be printed |
| Return | None |

Description:

This function justification the font data in print buffer which should be printed.

| Name | AddUnderLine |
|---|---|
| Parameter | u16StartPos: The start position<br>u16Lenth: Underline length(in bit)<br>u8Thick: The thick of underline<br>character output: The font information which least recently used |
| Return | None |

Description:

This function adds underline in relevant position.

| Name | ReverseData |
|---|---|
| Parameter | u16StartPos: The start position<br>u16Lenth: The reverse length(in bit)<br>character output: The font information which least recently used |
| Return | None |

Description:

This function reverses the data in the print buffer which will be printed.

| Name | PrintString |
|---|---|
| Parameter | u8String: pointer to the string<br>u16Len: The length of the string<br>u16Position: The position of the string printed |
| Return | None |

Description:

This function will print a string.

| Name | ClrCharLine |
|---|---|
| Parameter | u8Height: The height of the character line<br>u16Width: The width of the character line. |
| Return | None |

Description:

This function will cleat the one character line in print buffer.

### 5.4.3 Create task

| Name | CreateIniPrinterTask |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function created an initial printer task.

| Name | CreateCutTask |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function created a cut paper task.

| Name | CreatFeedNLineTask |
|---|---|
| Parameter | u16LineCnt: The length of the paper feed<br>u8BMEn: The BM mode enable flag<br>u8BMStopEn: Indicate that if need stop paper feed when detected black mark<br>u8Dir: The direction of paper feed |
| Return | None |

Description:

This function created a feed paper task.

| Name | CreatPrintLineTask |
|---|---|
| Parameter | u16LineNum: The print length in bit of this task<br>u16leftMargin: The left margin of the print task<br>u16RightMargin: The right margin of the print task |
| Return | None |

Description:

This function created a print task.

### 5.4.4 Other

| | |
|---|---|
| Name | RecoverCommFunc |
| Parameter | None |
| Return | None |

Description:

This function recover the communication function after the communication is disable when receive buffer full or device enter offline mode.

| | |
|---|---|
| Name | ClrOfflineMode |
| Parameter | None |
| Return | None |

Description:

This function clears the offline mode of system status.

| | |
|---|---|
| Name | ReadDIP |
| Parameter | None |
| Return | None |

Description:

This function reads the status of DIP and stores it in relevant variable.

| | |
|---|---|
| Name | DP_StartTestPrinting |
| Parameter | None |
| Return | None |

Description:

This function starts the test print.

## 5.5 Print Header

### 5.5.1 Print Header Driver

| | |
|---|---|
| Name | Phd_SetAutoFeedAttribute |
| Parameter | stcAutoFeedAttribute: auto feed attribute |
| Return | None |

Description:

This function sets auto feed attribute.

| | |
|---|---|
| Name | Phd_SetSensorConfigure |
| Parameter | unSensorConfig: sensor configure parameters |
| Return | None |

Description:

This function sets sensor parameters of configure.

| Name | Phd_PrintDriverInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function initializes print head.

| Name | Phd_StepNextpixel |
|---|---|
| Parameter | None |
| Return | 0: No phase<br>Other: step next phase |

Description:

This function switches next motor phase.

| Name | Phd_GetMotorPhases |
|---|---|
| Parameter | None |
| Return | phases number |

Description:

This function gets motor phases.

| Name | Phd_ResetPrintControlParam |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function resets control parameters.

| Name | Phd_ResetPrintMotorParam |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function resets motor parameters.

| Name | Phd_WriteDotLineData |
|---|---|
| Parameter | u16LeftMargin:left margin<br>u16RightMargin: right margin |
| Return | None |

Description:

Write dot line data to head buffer.

| Name | Phd_WriteDotLineHistoryData |
|---|---|
| Parameter | u16LeftMargin:left margin<br>u16RightMargin: right margin |
| Return | None |

Description:

Write dot line history data to head buffer.

| Name | Phd_StartFeed |
|---|---|
| Parameter | punTaskParam:pointer to task parameter structure |
| Return | None |

Description:

Start feed paper.

| Name | Phd_StartPrint |
|---|---|
| Parameter | punTaskParam:pointer to task parameter structure |
| Return | None |

Description:

Start printing.

| Name | Phd_StartAutocut |
|---|---|
| Parameter | punTaskParam:pointer to task parameter structure |
| Return | None |

Description:

Start auto feed.

| Name | Phd_AutoFeed |
|---|---|
| Parameter | None |
| Return | None |

Description:

Start auto feed.

| Name | Phd_AutoCutInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize auto cutter.

| Name | Phd_AutoCutClearTask |
|---|---|
| Parameter | None |
| Return | None |

Description:

Clear auto cut task.

| Name | Phd_AutoCutStop |
|---|---|
| Parameter | None |
| Return | None |

Description:

Stop auto cut.

| Name | Phd_AutoCutTimeOut200ms |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when auto cut time out 200ms.

| Name | Phd_AutoCutTimeOut1s |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when auto cut time out 1s.

| Name | Phd_AutoCutInitTimeOut1s |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when initializing auto cutter time out 1s.

| Name | Phd_RecoverPaper |
|---|---|
| Parameter | None |
| Return | 0: No error to recover<br>1: There is error to recover |

Description:

This function will execute when recovering normal after paper errors.

| Name | Phd_TimerForHeat |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when heating.

| Name | Phd_PWMIsrForMotor |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when heating or step motor.

| Name | Phd_PaperEndStop |
|---|---|
| Parameter | None |
| Return | None |

Description:

Stop print and store print head status when errors happen.

| Name | Phd_OverHeat |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when overheating.

| Name | Phd_CutterResetCallback |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when cutter reset signal is trigger.

| Name | Phd_OutPaperCallback |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when paper is end.

| Name | Phd_PaperNearEndCallback |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when paper is near end.

| Name | Phd_BlackMarkCallback |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when detect black mark if black mark is enabling.

| Name | Phd_PlatenOpenCallback |
|---|---|
| Parameter | None |
| Return | None |

Description:

This function will execute when platen is opening.

### 5.5.2 Print Header Pin

| | |
|---|---|
| Name | Php_MsftOn |
| Parameter | None |
| Return | None |

Description:

Turn on Msft.

| | |
|---|---|
| Name | Php_MsftOff |
| Parameter | None |
| Return | None |

Description:

Turn off Msft.

| | |
|---|---|
| Name | Php_HeatStb1 |
| Parameter | None |
| Return | None |

Description:

Enable Stb1 pin to heat.

| | |
|---|---|
| Name | Php_HeatStb2 |
| Parameter | None |
| Return | None |

Description:

Enable Stb2 pin to heat.

| | |
|---|---|
| Name | Php_HeatStb12 |
| Parameter | None |
| Return | None |

Description:

Enable Stb1 2 pin to heat.

| | |
|---|---|
| Name | Php_StopStb12 |
| Parameter | None |
| Return | None |

Description:

Disable Stb1 2 pin.

| | |
|---|---|
| Name | Php_SetMicrostepENABLEn |
| Parameter | u8PinLevel: Pin level |
| Return | None |

Description:

This function sets ENABLE n pin.

| Name | Php_SetMicrostepDir |
|---|---|
| Parameter | enRotationDir: step direction |
| Return | None |

Description:

This function setsmicro step direction

| Name | Php_SetMicrostepMode |
|---|---|
| Parameter | enMicrostepMode: step mode |
| Return | None |

Description:

This function sets micro step mode.

| Name | Php_SetMicrostepRESETn |
|---|---|
| Parameter | u8PinLevel: Pin Level |
| Return | None |

Description:

This function sets RESET n pin

| Name | Php_GetMicrostepHOMEn |
|---|---|
| Parameter | None |
| Return | Pin level |

Description:

Get micro step HOME n pin level.

| Name | Php_DcMotorForward |
|---|---|
| Parameter | None |
| Return | None |

Description:

Forward DC motor.

| Name | Php_DcMotorReverse |
|---|---|
| Parameter | None |
| Return | None |

Description:

Reverse DC motor

| Name | Php_DcMotorStandby |
|---|---|
| Parameter | u8PinLevel: Pin level |
| Return | None |

Description:

Standby DC motor

| Name | Php_DcMotorBrake |
|---|---|
| Parameter | None |
| Return | None |

Description:

Brake DC motor.

| Name | Php_PinInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initial print head pin.

### 5.5.3 Print Header CSIO

| Name | Phc_CsioInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize CSIO for print head.

## 5.6 Character Module

### 5.6.1 General Overview

Character module includes 3 C files, the detail as shown in Table 22

Table 22. Character Module Overview

| File | Function | Description |
|---|---|---|
| character.c | CHAR_CharacterRange | Confirm the character belong to ASCII or Chinese |
| | CHAR_CharacterService | Service function for character |
| | Char_EraseNVBitmapArea | Erase NV bitmap area in flash |
| | Char_WriteNVBitmapTable | Write NV bitmap index table in flash |
| | Char_WriteNVBitmapTotal | Write NV bitmap total number in flash |
| | Char_WriteNVBitmapData | Write NV bitmap data in flash |
| | Char_ReadNVBitmapTotal | Read NV bitmap total number from flash |
| | Char_ReadNVBitmapTable | Read NV bitmap index table from flash |
| | Char_ReadNVBitmapData | Read NV bitmap data from flash |
| | Char_ReadDownLoadBitmapData | Read download bitmap data |
| | Char_ReadSoftDIP | Read soft DIP from flash |
| | Char_WriteSoftDIP | Write soft DIP to flash |
| | Char_WriteAutomaticLoadControlValueSetting | Write automatic load control value setting in flash |
| | Char_ReadAutomaticLoadControlValueSetting | Read automatic load control value setting from flash |
| | Char_WritePrinterControlValue | Write printer control value |

| | | |
|---|---|---|
| | Char_ReadPrinterControlValue | Read printer control value |
| | Char_NVInit | Initialize NV bitmap |
| csio_int_s25fl127s.c | S25fl127s_Init | Initialize flash process |
| | S25fl127s_ReadId | Read external flash ID |
| | S25fl127s_BulkErase | Flash bulk erase |
| | S25fl127s_4kSectorErase | Flash 4K sector erase |
| | S25fl127s_SectorErase | Flash sector erase |
| | S25fl127s_PageProgram | Write data to external flash |
| | S25fl127s_Read | Read data from external flash |
| Style_convert.c | CSC_FontStlyeBold | Set or cancel font bold style |
| | CSC_FontStlyeUnderline | Set or cancel font underline style |
| | CSC_FontStlyeStriketrougth | Set or cancel font strikethrough style |
| | CSC_FontStlyeUpline | Set or cancel font up line style |
| | CSC_FontStlyeRotate90 | Rotate font 90° |
| | CSC_FontStlyeRotate180 | Rotate font 180° |
| | CSC_FontStlyeRotate270 | Rotate font 270° |
| | CSC_FontStlyeDoubleWidth | Zoom out the font by double width |
| | CSC_FontStlyeDoubleHeight | Zoom out the font by double height |

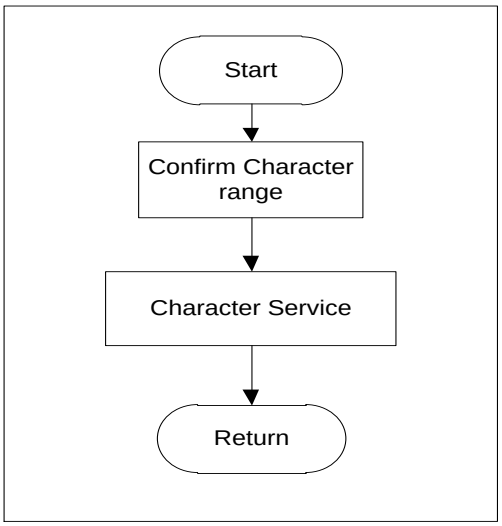
### 5.6.2 Function Module Flow

This module includes 5 function modules, Character process, NV Bitmap process, Download Bitmap process, Soft DIP process and Printer Control Value process.

1. Character Process

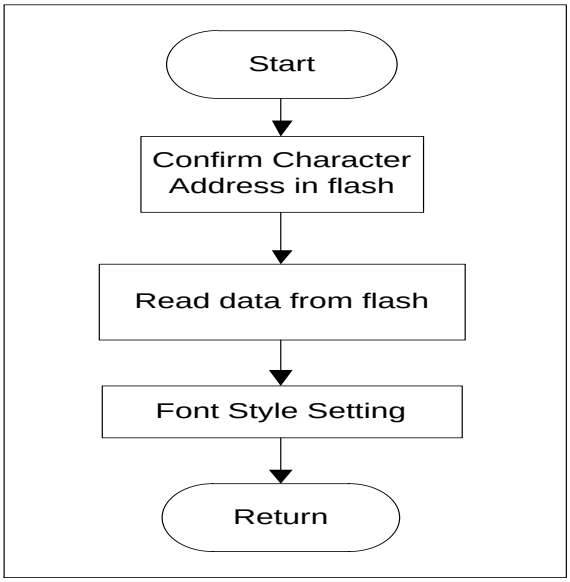
Main process flow:

Figure 46. Character Process Main Flow



CHAR_CharacterService flow:

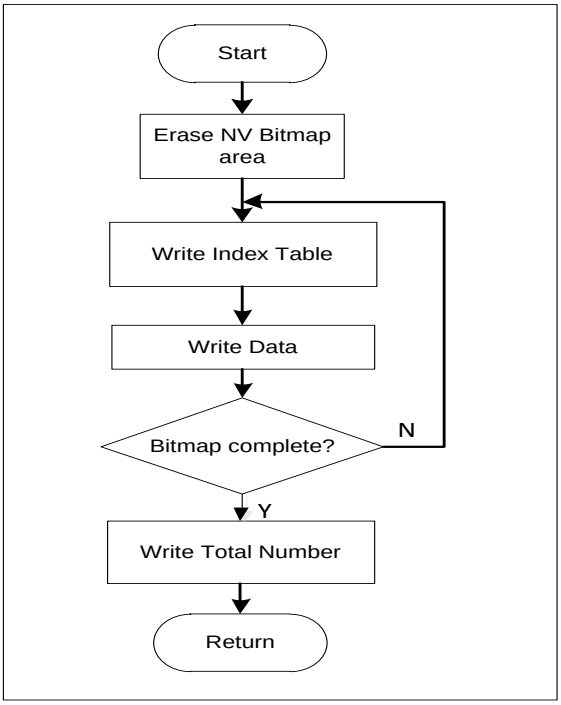
Figure 47. CHAR_CharacterServiceProcess Flow




2. NV Bitmap Process

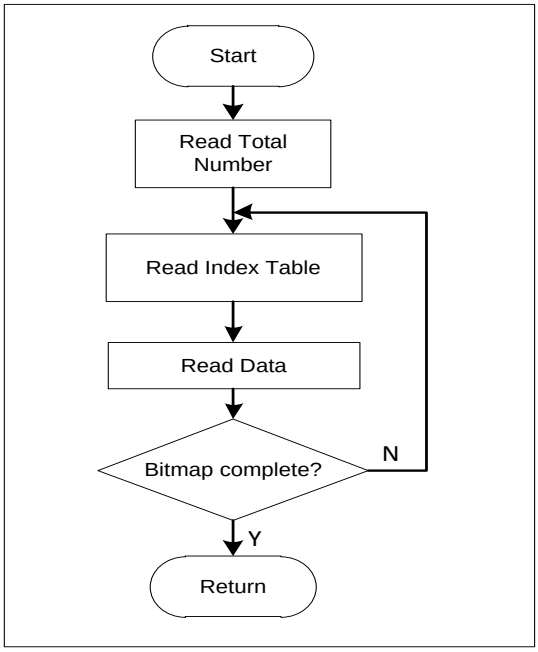
Write NV Bitmap Flow:

Figure 48. Write NV Bitmap Flow



Read NV Bitmap Flow:

Figure 49. Read NV Bitmap Flow




### 5.6.3 Functions Description

| | |
|---|---|
| Name | CHAR_CharacterRange |
| Parameter | content: Character content<br>content length: Content length |
| Return | Character range |

Description:

Confirm the character belong to ASCII or Chinese.

| | |
|---|---|
| Name | CHAR_CharacterService |
| Parameter | content: Character content<br>content length: Content length<br>character output: Processed data |
| Return | 0: Success<br>Other: Fail |

Description:

Character service function, read the character data from external flash and process the font style.

| | |
|---|---|
| Name | Char_EraseNVBitmapArea |
| Parameter | None |
| Return | None |

Description:

Erase the flash of NV bitmap area.

| Name | Char_WriteNVBitmapTable |
|---|---|
| Parameter | u8Index: Current NV bitmap index number<br>u16Width: Current NV bitmap width<br>u16Height: Current NV bitmap height |
| Return | 1: Success<br>0: Fail |

Description:

Write the NV bitmap index table.

| Name | Char_WriteNVBitmapTotal |
|---|---|
| Parameter | u8total: All NV bitmap quantity |
| Return | 1: Success<br>0: Fail |

Description:

Write the NV bitmap quantity in flash.

| Name | Char_WriteNVBitmapData |
|---|---|
| Parameter | u8index: NV bitmap index<br>u16height: NV bitmap height<br>u8data: processed data store<br>u16widthindex: NV bitmap width index<br>u16length: process sector length |
| Return | 1: Success<br>0: Fail |

Description:

Write the NV bitmap data in flash.

| Name | Char_ReadNVBitmapTotal |
|---|---|
| Parameter | u8total: All NV bitmap quantity |
| Return | 1: Success<br>0: Fail |

Description:

Read the NV bitmap quantity from flash.

| Name | Char_ReadNVBitmapTable |
|---|---|
| Parameter | u8Index: Current NV bitmap index number<br>u16Width: Current NV bitmap width<br>u16Height: Current NV bitmap height |
| Return | 1: Success<br>0: Fail |

Description:

Read the NV bitmap index table.

| Name | Char_ReadNVBitmapData |
|---|---|
| Parameter | u8index: NV bitmap index<br>u16width: NV bitmap width<br>u16height: NV bitmap height<br>u8data: processed data store<br>u16heightindex: NV bitmap height index<br>u16length: process sector length<br>u8mode: NV bitmap zoom out mode |
| Return | 1: Success<br>0: Fail |

Description:

Read the NV bitmap data from flash.

| Name | Char_ReadDownLoadBitmapData |
|---|---|
| Parameter | u16width: Download bitmap width<br>u16height: Download bitmap height<br>u8data: processed data store<br>u16heightindex: Download bitmap height index<br>u16length: process sector length<br>u8mode: Download bitmap zoom out mode<br>u8bitmapbuf: Download bitmap store base address |
| Return | 1: Success<br>0: Fail |

Description:

Read the download bitmap data fromRAM.

| Name | Char_ReadSoftDIP |
|---|---|
| Parameter | u8DipNumIndex: DIP number index<br>u8DipNumLength: Read DIP quantity<br>u8DipData: store DIP data |
| Return | None |

Description:

Read DIP value.

| Name | Char_WriteSoftDIP |
|---|---|
| Parameter | u8DipData: DIP value<br>u8DipNumLength: Write DIP quantity |
| Return | None |

Description:

Write DIP value.

| Name | Char_WriteAutomaticLoadControlValueSetting |
|---|---|
| Parameter | u8Setting: Setting |
| Return | None |

Description:

Write automatic load control value setting.

| Name | Char_ReadAutomaticLoadControlValueSetting |
|---|---|
| Parameter | u8Setting: Store setting |
| Return | None |

Description:

Read automatic load control value setting.

| Name | Char_WritePrinterControlValue |
|---|---|
| Parameter | None |
| Return | None |

Description:

Write printer control value to flash.

| Name | Char_ReadPrinterControlValue |
|---|---|
| Parameter | None |
| Return | None |

Description:

Read printer control value from flash.

| Name | Char_NVInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize NV bitmap.

| Name | S25fl127s_Init |
|---|---|
| Parameter | None |
| Return | Generic error codes |

Description:

Initialize SPI.

| Name | S25fl127s_ReadId |
|---|---|
| Parameter | pManufactureId: Pointer to manufacture ID<br>pDeviceId: Pointer to device ID |
| Return | Generic error codes |

Description:

Read external flash ID.

| Name | S25fl127s_BulkErase |
|---|---|
| Parameter | None |
| Return | Generic error codes |

Description:

Chip erases the whole Flash memory.

| Name | S25fl127s_4kSectorErase |
|---|---|
| Parameter | u32Addr: The address within the 4k area |
| Return | Generic error codes |

Description:

Sector erase 4KB sector.

| Name | S25fl127s_SectorErase |
|---|---|
| Parameter | u32Addr: The address within the 64kB or 256kB area |
| Return | Generic error codes |

Description:

Sector erase 64kB or 256kB sector.

| Name | S25fl127s_PageProgram |
|---|---|
| Parameter | u32Addr: Address<br>pu8Data: Pointer to data address<br>u16Size: Data size |
| Return | Generic error codes |

Description:

Page program S25FL127S.

| Name | S25fl127s_Read |
|---|---|
| Parameter | u32Addr: Address<br>pu8Data: Pointer to data address<br>u16Size: Data size |
| Return | Generic error codes |

Description:

Read data from S25FL127S.

| Name | CSC_FontStlyeBold |
|---|---|
| Parameter | Character array: Character dot data |
| Return | None |

Description:

Bold style process.

| Name | CSC_FontStlyeUnderline |
|---|---|
| Parameter | Character array: Character dot data |
| Return | None |

Description:

Underlinestyle process.

| Name | CSC_FontStlyeStriketrougth |
|---|---|
| Parameter | Character array: Character dot data |
| Return | None |

Description:

Strikethroughstyle process.

| Name | CSC_FontStlyeUpline |
|---|---|
| Parameter | Character array: Character dot data |
| Return | None |

Description:

Uplinestyle process.

| Name | CSC_FontStlyeRotate90 |
|---|---|
| Parameter | Character array: Character dot data<br>Target array: Processed data |
| Return | None |

Description:

Character rotates 90 degree.

| Name | CSC_FontStlyeRotate180 |
|---|---|
| Parameter | Character array: Character dot data<br>Target array: Processed data |
| Return | None |

Description:

Character rotates 180 degree.

| Name | CSC_FontStlyeRotate270 |
|---|---|
| Parameter | Character array: Character dot data<br>Target array: Processed data |
| Return | None |

Description:

Character rotates 270 degree.

| Name | CSC_FontStlyeDoubleWidth |
|---|---|
| Parameter | Character array: Character dot data<br>Target array: Processed data |
| Return | None |

Description:

Character double width.

| Name | CSC_FontStlyeDoubleHeight |
|---|---|
| Parameter | Character array: Character dot data<br>Target array: Processed data |
| Return | None |

Description:

Character double height.

## 5.7 Barcode Module

### 5.7.1 General Overview

Character module includes4 C files, the detail as shown in Table 23.

Table 23. Character Module Overview

| File | Function | Description |
|---|---|---|
| EAN8.c | EAN8_BarcodeCheckSum | Calculate EAN 8 and EAN 13 checksum |
| | EAN8_BarcodeEncode | EAN 8 and EAN 13 encode |
| Code128.c | C128_BitmapHorizontalzoom | Bitmap Horizontal zoom out |
| | C128_Code128Barcode | Code 128 encode |
| Qrcode.c | Qrcode | QR code encode |
| GenericGF.c | encode | Error correcting code encode |
| | ArrayCopy | Array copy for word type |
| | ArrayCopyByte | Array copy for byte type |

### 5.7.2 Functions Description

| Name | EAN8_BarcodeCheckSum |
|---|---|
| Parameter | au8CheckData: Check data<br>u8Length: data length |
| Return | Checksum |

Description:

Calculate EAN 8 and EAN 13 checksum.

| Name | EAN8_BarcodeEncode |
|---|---|
| Parameter | au8BarcodeString: Character content<br>u8codelength: Character length<br>pu8Target: Store the processed data<br>u8StandWidth: A stand width |
| Return | Barcode dot width |

Description:

EAN 8 and EAN 13 encode.

| Name | C128_BitmapHorizontalzoom |
|---|---|
| Parameter | au8SourceData: Source data<br>u16CodeLength: Data length<br>u8ZoomSize: Zoom out size<br>pu8Target: Store process data |
| Return | None |

Description:

Bitmap Horizontal zooms out.

| Name | C128_Code128Barcode |
|---|---|
| Parameter | pu8Barcode: Character content<br>pu8BarcodeLength: Content length<br>pu8TempArray: Store processed data<br>u8StandWidth: Stand width |
| Return | Barcode width |

Description:

Code 128 encode.

| Name | Qrcode |
|---|---|
| Parameter | content: Character content<br>content length: Content length<br>cell size: Cell size<br>level: Error correcting level |
| Return | 1: Success<br>0: Fail |

Description:

QR code encode.

| Name | encode |
|---|---|
| Parameter | Source data: Source data<br>toEncode: Store processed data<br>sourcedatalen: Source data length<br>ecBytes: Error correcting length |
| Return | None |

Description:

Error correcting code encode.

| Name | ArrayCopy |
|---|---|
| Parameter | SourceArray: Source array<br>SourceArrayIndex: Source array index<br>Destination Array: Destination array<br>Destination Array index: Destination array index<br>Copy Length: Copy length |
| Return | None |

Description:

Array copy for word type.

| Name | ArrayCopyByte |
|---|---|
| Parameter | SourceArray: Source array<br>SourceArrayIndex: Source array index<br>DestinationArray: Destination array<br>Destination Array index: Destination array index<br>Copy Length: Copy length |
| Return | None |

Description:

Array copy for byte type.

## 5.8 MISC Module

### 5.8.1 ADC

| | |
|---|---|
| Name | Adc_TempScanIntCb |
| Parameter | None |
| Return | None |

Description:

This function will execute when scan interrupt for temperature happens.

| | |
|---|---|
| Name | Adc_SetScanIntCb |
| Parameter | pfnScanIntCb: pointer to scan interrupt callback |
| Return | None |

Description:

Set Scan interrupt callback function.

| | |
|---|---|
| Name | Adc_ScanIntCb |
| Parameter | None |
| Return | None |

Description:

This functionwill execute when scan interrupt happens.

| | |
|---|---|
| Name | Adc_SetCompIntCb |
| Parameter | pfnCompIntCb: pointer to compare interrupt callback function |
| Return | None |

Description:

Set compare interrupt callback function.

| | |
|---|---|
| Name | Adc_CompIntCb |
| Parameter | None |
| Return | None |

Description:

This function will execute when compare interrupt happen.

| | |
|---|---|
| Name | Adc_DisableTempCompInt |
| Parameter | None |
| Return | None |

Description:

Disable compare interrupt.

| Name | Adc_EnableTempCompInt |
|---|---|
| Parameter | None |
| Return | None |

Description:

Enable compare interrupt.

| Name | Adc_Init |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize ADC for print head.

| Name | Adc_StartScan |
|---|---|
| Parameter | u8AdcUnit:adc unit |
| Return | None |

Description:

Start ADC scan.

| Name | Adc_StopScan |
|---|---|
| Parameter | u8AdcUnit:adc unit |
| Return | None |

Description:

Stop ADC scan.

| Name | Adc_TmpSampleVal |
|---|---|
| Parameter | None |
| Return | Sample value |

Description:

Sample the voltage of thermistor resistance.

### 5.8.2 Dual Timer

| Name | Dt_SetHeatCount |
|---|---|
| Parameter | u32HeatCount: heat time |
| Return | None |

Description:

Set timer count.

| Name | Dt_Start |
|---|---|
| Parameter | pstcUnitCh: Pointer to structure of timer channel and unit |
| Return | None |

Description:

Start DT.

| Name | Dt_Stop |
|---|---|
| Parameter | pstcUnitCh: Pointer to  structure of timer channel and unit |
| Return | None |

Description:

Stop DT.

| Name | Dt_StartHeatTimer |
|---|---|
| Parameter | None |
| Return | None |

Description:

Start DT for heating.

| Name | Dt_StopHeatTimer |
|---|---|
| Parameter | None |
| Return | None |

Description:

Stop DT for heating.

| Name | Dt_Configure |
|---|---|
| Parameter | pstcTimerConfig: Pointer to timer parameter structure<br>TimerCallBack: timer interrupt call back function<br>u8DtChCnt: Pointer to dual timer channel and unit |
| Return | None |

Description:

Configure DT.

| Name | Dt_GetTimerValue |
|---|---|
| Parameter | u8Ch: timer channel |
| Return | Dual timer current count value |

Description:

Get DT value.

| Name | Dt_GetTimerValueHeat |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize ADC for print head.

| Name | Adc_StartScan |
|---|---|
| Parameter | u8Ch: timer channel |
| Return | Dual timer current count value |

Description:

Get DT value for heat.

| Name | Dt_DisableTimerHeat |
|---|---|
| Parameter | None |
| Return | None |

Description:

Disable DT for heat.

| Name | Dt_SetIntCbHeat |
|---|---|
| Parameter | pfnDtCb: pointer to DT interrupt callback function |
| Return | None |

Description:

Set DT interrupt callback for heat.

| Name | Dt_Init |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize DT.

### 5.8.3 Free Run Timer

| Name | Frt_Start |
|---|---|
| Parameter | pstcUnitCh: Pointer to structure of timer channel and unit |
| Return | None |

Description:

Start FRT.

| Name | Frt_Stop |
|---|---|
| Parameter | pstcUnitCh: Pointer to structure of timer channel and unit |
| Return | None |

Description:

StopFRT.

| Name | Frt_StartAutoFeedCount |
|---|---|
| Parameter | None |
| Return | None |

Description:

Start auto feed count.

| Name | Frt_StopAutoFeedCount |
|---|---|
| Parameter | None |
| Return | None |

Description:

Stop auto feed count

| Name | Frt_SetAutoFeedCb |
|---|---|
| Parameter | pfnAutoFeedCb: Pointer to auto feed callback function |
| Return | None |

Description:

Set auto feed callback.

| Name | Frt_SetAutoFeedCount |
|---|---|
| Parameter | u16CntCycle: timer cycle |
| Return | None |

Description:

Set FRTcycle for auto feed.

| Name | Frt_StartTimeOutCount |
|---|---|
| Parameter | None |
| Return | None |

Description:

Start FRT for timeout.

| Name | Frt_StopTimeOutCount |
|---|---|
| Parameter | None |
| Return | None |

Description:

Stop FRT for timeout.

| Name | Frt_SetTimeOutCount |
|---|---|
| Parameter | u16CntCycle: time out cycle |
| Return | None |

Description:

Set FRT cycle for timeout

| Name | Frt_SetTimeOutCb |
|---|---|
| Parameter | pfnTimeOutCb: pointer to FRT interrupt callback |
| Return | None |

Description:

Set FRT callback for timeout

| Name | Frt_Config |
|---|---|
| Parameter | pstcFrtParam:Pointer to frt parameter structure |
| Return | None |

Description:

Configure FRT.

| Name | Frt_Init |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize FRT.

### 5.8.4 PWM

| Name | Pwm_Init |
|---|---|
| Parameter | pstcIoBtTioa: pointer to PWM IO structure<br>pstcBtPwmInitReg: pointer to PWM timer initialization data structure<br>pstcPwmIntCb: pointer to PWM interrupt callback structure |
| Return | None |

Description:

Initialize PWM.

| Name | Pwm_OutputHighLevel |
|---|---|
| Parameter | None |
| Return | None |

Description:

Output high.

| Name | Pwm_OutputLowLevel |
|---|---|
| Parameter | None |
| Return | None |

Description:

Output low.

| Name | Pwm_Configure |
|---|---|
| Parameter | None |
| Return | None |

Description:

Configure PWM

| Name | Pwm_SetTrigIntCb |
|---|---|
| Parameter | pfnPwmTrigIntCb: Trig interrupt callback function |
| Return | None |

Description:

Set trig interrupt callback.

| Name | Pwm_SetDutyMatchIntCb |
|---|---|
| Parameter | pfnPwmDutyMatchIntCb: Pointer to duty match interrupt callback |
| Return | None |

Description:

Set duty match interrupt callback.

| Name | Pwm_SetUnderflowIntCb |
|---|---|
| Parameter | pfnPwmUnderflowIntCb: Pointer to underflow interrupt callback function |
| Return | None |

Description:

Set underflows interrupt callback.

| Name | Pwm_StartSoft |
|---|---|
| Parameter | u8Ch:Requested channel |
| Return | None |

Description:

Start PWM

| Name | Pwm_Stop |
|---|---|
| Parameter | u8Ch:Requested channel |
| Return | None |

Description:

Stop PWM.

| Name | Pwm_WriteCycleVal |
|---|---|
| Parameter | u8Ch: requested channel<br>u16Cycle: cycle value |
| Return | None |

Description:

Write cycle value of PWM

| Name | Pwm_WriteDutyVal |
|---|---|
| Parameter | u8Ch: requested channel<br>u16Duty: duty value |
| Return | None |

Description:

Write duty value of PWM.

| Name | Pwm_ReadCurCnt |
|---|---|
| Parameter | u8Ch: requested channel |
| Return | count value |

Description:

Read current count value of PWM timer.

### 5.8.5 Sys_IF

| Name | System_SystemIntInit |
|---|---|
| Parameter | None |
| Return | None |

Description:

Initialize system interrupt callback.

# 6. Reference Documents



1. 厦门精芯-打印指令集
2. DRV8811_datasheet
3. Thermal line printerM-T530A/T540ASpecification
4. MB9BF516R-MN706-00016-1v0-E(MB9B500 Data Sheet)
5. MN706-00002-7v0-E(MB9Axxx/MB9Bxxx Series Periphery Manual)

# 7. Appendix

## 7.1 Commands

### 7.1.1 Command Description Format

| | |
|---|---|
| [Name] | The name of the command |
| [Format] | The code sequence |
| [ ]k | indicates the contents of [ ] should be repeated k times |
| [Range] | Gives the allowable ranges for the arguments |
| [Description] | Describes the functions of the command |
| [Details] | Describes the usage of the command in detail |
| [Notes] | Provides important information on setting and using the printer command if necessary |
| [Default] | Gives the default values, if any, for the command parameters |
| [Reference] | Lists related commands |

The numbers denoted by <>H are hexadecimal

The numbers denoted by <>B are binary

### 7.1.2 Terms Description

1. Receive buffer

   The receive buffer is a buffer that stores, as is, the data received from the host computer (the reception data). The receive data is stored in the receive buffer temporarily, and is then processed sequentially.

2. Print buffer

   The print buffer is a buffer that stores the image data to be printed.

3. Print buffer full

   This is the state where there is no more room in the print buffer. If new print data is input while the print buffer is full, the data in the print buffer is printed out and a line feed is executed. This is the same operation as the LF operation.

4. Start of line

   The start of line state satisfies the following conditions:

   - There is no print data (including spaces and portions of data skipped due to HT currently in the print buffer.
   - The print position is not specified by the **ESC $** or **ESC ** command.

5. Printable area

   The maximum range within which printing is possible under the printer specifications. The printable area for this printer is as follows:

   - (1) The length of the horizontal direction in standard mode:

   For 82.5 mm paper-width model: approximately 80 mm {640/203.2"}

6. Printing area

   Printing range is set by the command. The printing area must be ≤ the printable area.

7. Ignore

   The state in which all codes including parameters, are read in and discarded, and nothing happens.

8. Inch

   One inch is 25.4 mm.

9. MSB

   Most Significant Bit

10. LSB

    Least Significant Bit

### 7.1.3 Commands

**HT**

| | | |
|---|---|---|
| [Name] | Horizontal tab | |
| [Format] | ASCII | HT |
| | Hex | 09 |
| | Decimal | 9 |

[Description] Moves the print position to the next horizontal tab position.

[Notes]

- This command is ignored unless the next horizontal tab position has been set.
- If the next horizontal tab position exceeds the printing area, the printer sets the printing position to [printing area width + 1].
- Horizontal tab positions are set with **ESC D**.
- If this command is received when the printing position is at [printing area width + 1], the printer executes print buffer-full printing of the current line and horizontal tab processing from the beginning of the next line.

[Reference] **ESC D**

**LF**

[Name] Print and line feed

[Format] ASCII LF

Hex 0A

Decimal 10

[Description] Prints the data in the print buffer and feeds one line, based on the current line spacing.

[Note] This command sets the print position to the beginning of the line.

[Reference] ESC 2, ESC 3, APPENDIX A.1

**FF**

[Name] (1) Print and return to standard mode in page mode

(2) Print and feed marked paper to print starting position

[Format] ASCII FF

Hex 0C

Decimal 12

(1) When page mode is selected:

[Description] Prints the data in the print buffer collectively and returns to standard mode.

[Notes]

- This command is enabled only in page mode.
- The buffer data is deleted after being printed.
- The printing area set by ESC W is reset to the default setting.
- This command sets the print position to the beginning of the line.

[Reference] ESC FF, ESC L, ESC S

(2) When BM sensor is effective:

[Description] Prints the data in the print buffer and feeds marked paper to the print starting position.

[Notes]

- This command is enabled only when the BM sensor is set to be effective using with DIP SW1-1.
- This command sets the print position to the beginning of the line.
- If this command is executed at the print starting position of the marked paper, the printer feeds the marked paper to the next print starting position.

[Reference] GS ( F, GS FF, 3.3.2.1, DIP Switch 1

**CR**

[Name] Print and carriage return

[Format] ASCII CR

Hex 0D

Decimal 13

[Description] When automatic line feed is enabled, this command functions the same as LF; when automatic line feed is disabled, this command is ignored.

[Details]

- Auto line feed is enabled / disabled by the setting of memory switch 1-5.
- Sets the print starting position to the beginning of the line.

[Reference] **LF**

**ESC SP n**

[Name] Set right-side character spacing

[Format] ASCII ESC SP n

Hex 1B 20 n

Decimal 27 32 n

[Range] 0 ≤n≤ 255

[Description] Set the character spacing for the right side of the character to [n× 0.125 mm].

[Details]

- When characters are enlarged, the right-side character spacing is as n times as normal value.
- This command sets values independently in each mode (standard and page modes).

[Default] n = 0

**ESC ! n**

[Name] Select print mode(s)

[Format] ASCII ESC ! n

Hex 1B 21 n

Decimal 27 33 n

[Range] 0 ≤n≤ 255

[Description] Selects print mode(s) using n as follows:

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0 | Off | 0 | 0 | Character Font A (12 × 24) selected. |
| | On | 1 | 1 | Character Font B (9 × 17) selected.. |
| 1 | — | — | — | Undefined |
| 2 | — | — | — | Undefined |
| 3 | Off | 0 | 0 | Emphasized mode is turned off. |
| | On | 8 | 8 | Emphasized mode is turned on. |
| 4 | Off | 0 | 0 | Double-height canceled. |
| | On | 10 | 16 | Double-height selected. |
| 5 | Off | 0 | 0 | Double-width canceled. |

| | | | | |
|---|---|---|---|---|
| | On | 20 | 32 | Double-width selected. |
| 6 | — | — | — | Undefined.th |
| 7 | Off | 0 | 0 | Underline mode is turned off. |
| | On | 80 | 128 | Underline mode is turned on. |

[Details]

- When both double-height and double-width modes are selected, quadruple-size characters are printed.
- The printer can underline all characters, but cannot underline the space set by **HT** or 90° clockwise rotated characters.
- The thickness of the underline is that selected by **ESC −**, regardless of the character size.
- When some characters in different height magnification are in a line, all the characters in the line are aligned at the baseline.
- **ESC M** can also select the character font type. However, the setting of the last received command becomes effective.
- **ESC E** can also turn on or off emphasized mode. However, the setting of the last received command becomes effective.
- **ESC −** can also turn on or off underline mode. However, the setting of the last received command becomes effective.
- **GS !** can also select character size. However, the setting of the last received command becomes effective.
- Turning emphasized mode on/off is effective only with alphanumeric and Kanji characters. Other print modes are effective only with alphanumeric and Kanji characters.

[Default] n = 0

[Reference] **ESC -**, **ESC E**, **GS !**

**ESC $ nL nH**

[Name] Set absolute print position

| [Format] | ASCII | ESC | $ | nL | nH |
|---|---|---|---|---|---|
| | Hex | 1B | 24 | nL | nH |
| | Decimal | 27 | 36 | nL | nH |

[Range] 0 ≤nL≤ 255

0 ≤nH≤ 255

[Description] Sets the distance from the beginning of the line to the position at which subsequent characters are to be printed.

The distance from the beginning of the line to the print position is

[(nL + nH× 256) × 0.125 mm].

[Details]

- Settings outside the specified printable area are ignored.
- In standard mode, the horizontal motion unit (x) is used.
- In page mode, horizontal or vertical motion units differ depending on the starting position of the printable area, as follows:

  1. When the starting position is set to the upper left or lower right of the printable area using **ESC T**, the horizontal motion unit (x) is used.

2. When the starting position is set to the upper right or lower left of the printable area using **ESC T**, the vertical motion unit (y) is used.

[Reference] **ESC **, **GS $**, **GS **

**ESC % n**

[Name] Select/cancel user-defined character set

| [Format] | ASCII | ESC | % | n |
|---|---|---|---|---|
| | Hex | 1B | 25 | n |
| | Decimal | 27 | 37 | n |

[Range] 0 ≤n≤ 255

[Description] Selects or cancels the user-defined character set.

- When the LSB of n is 0, cancels the user-defined character set.
- When the LSB of n is 1, selects the user-defined character set.

[Details]

- n is available only for the least significant bit.
- When the user-defined character set is canceled, the built-in character set is automatically selected.

[Default] n = 0

[Reference] **ESC &**, **ESC ?**

***ESC &*y c1 c2[x1 d1...d(y × x1)]...[xk d1...d(y × xk)]**

[Name] Define user-defined characters

| [Format] | ASCII | ESC | & | *y c1 c2[x1 d1...d(y × x1)]...[xk d1...d(y × xk)]* |
|---|---|---|---|---|
| | Hex | 1B | 26 | *y c1 c2[x1 d1...d(y × x1)]...[xk d1...d(y × xk)]* |
| | Decimal | 27 | 38 | *y c1 c2[x1 d1...d(y × x1)]...[xk d1...d(y × xk)]* |

[Range] y = 3

32 ≤*c1*≤*c2*≤ 126

0 ≤*x*≤ 12 (when Font A (12 × 24) is selected)

0 ≤*x*≤ 9 (when Font B (9 × 17) is selected)

0 ≤*d1 ... d(y × xk)*≤ 255

[Description] Defines user-defined characters.

- y specifies the number of bytes in the vertical direction.
- c1 specifies the character code that starts the definition, and c2 specifies the character code that ends the definition.
- x specifies the number of dots in the horizontal direction.

[Details]

- 
- The allowable character code range is from ASCII code <20>H to <7E>H.
- It is possible to define multiple characters for consecutive character codes. To define only one character, use *c1 = c2*.
- d is the dot data for the characters. The dot pattern is in the horizontal direction from the left side. If value of x is less than the dot number to configure the character, any remaining dots on the right side become space (white).
- The data to define a user-defined character is (y × x) bytes.

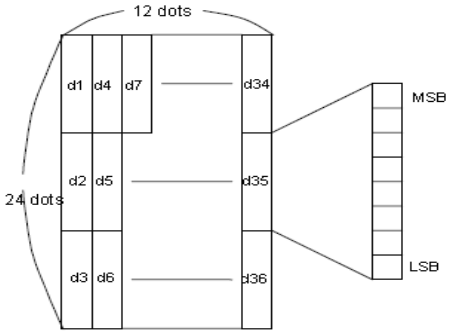
- Set a corresponding bit to 1 to print a dot or 0 not to print a dot.
- This command can define different user-defined character patterns for each font. To select a font, use **ESC !** or **ESC M**.
- User-defined characters and a downloaded bit image cannot be defined simultaneously. When this command is executed, the downloaded bit image is cleared.
- The user-defined character definition is cleared when:
  1. **ESC** @ is executed.
  2. **GS** ∗ is executed.
  3. **ESC ?** is executed.
  4. The printer is reset or the power is turned off.
- When user-defined characters are defined in Font B (9 × 17), only the most significant bit of the 3rd byte of data in vertical direction is effective.

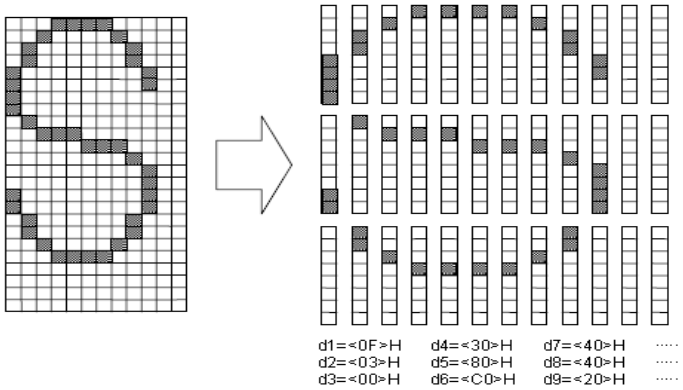
[Default] The embedded character set

[Reference] **ESC %**, **ESC ?**

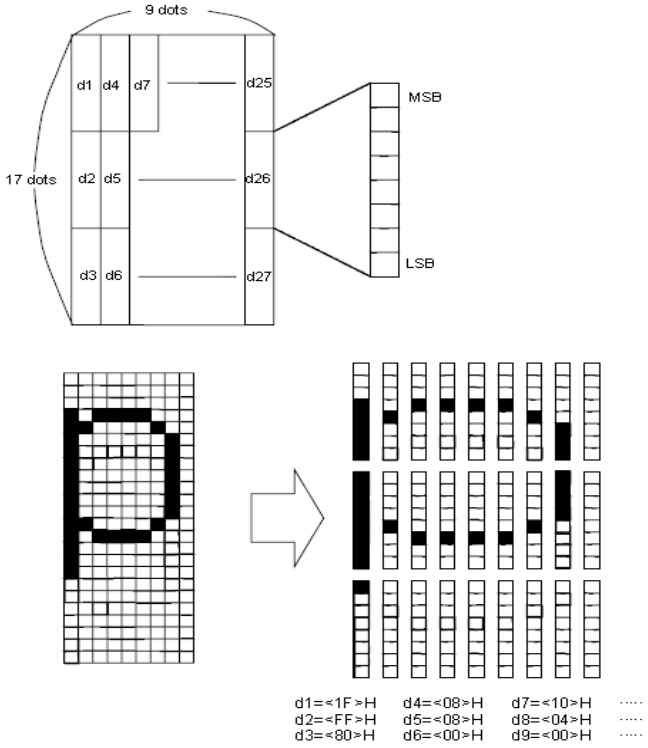
[Example]

- When Font A (12 × 24) is selected.



- When font B (9 × 17) is selected.




**ESC ∗mnL nH d1...dk**

[Name] Select bit-image mode

[Format]

| | | | |
|---|---|---|---|
| ASCII | ESC | ∗ | *m nL nH d1...dk* |
| Hex | 1B | 2A | *m nL nH d1...dk* |
| Decimal | 27 | 42 | *m nL nH d1...dk* |

[Range] *m* = 0, 1, 32, 33

0 ≤*nL*≤ 255

0 ≤*nH*≤ 3

0 ≤*d*≤ 255

[Description] Selects a bit-image mode using m for the number of dots specified by nL and nH, as follows:

| m | Mode | Vertical Direction | | Horizontal Direction | |
|---|---|---|---|---|---|
| | | Number of dots | Dot density | Dot density | Number of data (k) |
| 0 | 8-dot single-density | 8 | 67.7 dpi | 101.6 dpi | *nL* + *nH* × 256 |
| 1 | 8-dot double-density | 8 | 67.7 dpi | 203.2 dpi | *nL* + *nH* × 256 |
| 32 | 24-dot single-density | 24 | 203.2 dpi | 101.6 dpi | (*nL* + *nH* × 256) ×3 |
| 33 | 24-dot double-density | 24 | 203.2 dpi | 203.2 dpi | (*nL* + *nH* × 256) ×3 |

[dpi: dots per 25.4 mm {1"}]

[Notes]

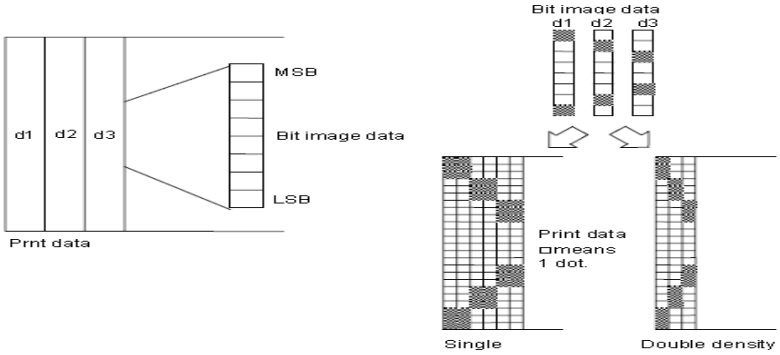
- If the value of m is out of the specified range, nL and the data following are processed as normal data.
- The nL and nH indicate the number of dots in the bit image in the horizontal direction. The number of dots is calculated by *nL* + *nH*× 256.
- If the bit-image data input exceeds the number of dots to be printed on a line, the excess data is ignored.
- d indicates the bit-image data. Set a corresponding bit to 1 to print a dot or to 0 not to print a dot.
- If the width of the printing area set by **GS L** and **GS W** less than the width required by the data sent with the **ESC** * command, the following will be performed on the line in question (but the printing cannot exceed the maximum printable area):

  1. The width of the printing area is extended to the right to accommodate the amount of data.
  2. If step (1) does not provide sufficient width for the data, the left margin is reduced to accommodate the data.

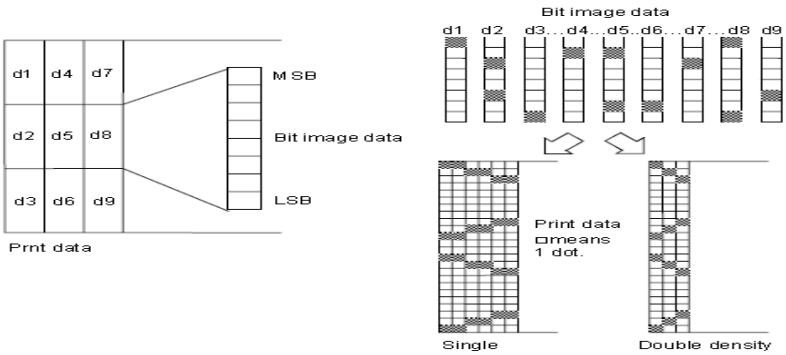
For each bit of data in single-density mode (*m* = 0, 32), the printer prints two dots: for each bit of data in double-density mode (*m* = 1, 33), the printer prints one dot. This must be considered in calculating the amount of data that can be printed in one line.

- After printing a bit image, the printer returns to normal data processing mode.
- This command is not affected by print modes (emphasized, double-strike, underline, character size, or white/black reverse printing), except upside-down printing mode.
- The relationship between the image data and the dots to be printed is described in Figure below.

(When 8-dot bit image is selected)




- When 24-dot bit image is selected:



**ESC - n**

[Name] Turn underline mode on/off

[Format]

| | | | |
|---|---|---|---|
| ASCII | ESC | - | n |
| Hex | 1B | 2D | n |
| Decimal | 27 | 45 | n |

[Range] 0 ≤n≤ 2, 48 ≤n≤ 50

[Description] Turns underline mode on or off, based on the following values of n:

| n | Function |
|---|---|
| 0, 48 | Turns off underline mode |
| 1, 49 | Turns on underline mode (1-dot thick) |
| 2, 50 | Turns on underline mode (2-dots thick) |

[Notes]

- The printer can underline all characters (including right-side character spacing), but cannot underline the space set by **HT**.
- The printer cannot underline 90° clockwise rotated characters and white/black inverted characters.
- When underline mode is turned off by setting the value of n to 0 or 48, the following data is not underlined, and the underline thickness set before the mode is turned off does not change. The default underline thickness is 1 dot.
- Changing the character size does not affect the current underline thickness.
- Underline mode can also be turned on or off by using **ESC !**. Note, however, that the last received command is effective.

[Default] n = 0

[Reference] **ESC !**

**ESC 2**

[Name] Select default line spacing

[Format]

| | | |
|---|---|---|
| ASCII | ESC | 2 |
| Hex | 1B | 32 |
| Decimal | 27 | 50 |

[Description] Selects 3.75 mm (30 × 0.125 mm) line spacing.

[Notes] The line spacing can be set independently in standard mode and in page mode.

[Reference] **ESC 3**

**ESC 3 n**

[Name] Set line spacing

[Format] ASCII ESC 3 n

Hex 1B 33 n

Decimal 27 51 n

[Range] 0 ≤*n*≤ 255

[Description] Sets the line spacing to [n× 0.125 mm].

[Notes]

- The line spacing can be set independently in standard mode and in page mode.
- In standard mode, the vertical motion unit (y) is used.
- In page mode, this command functions as follows, depending on the starting position of the printable area:
- When the starting position is set to the upper left or lower right of the printable area using **ESC T**, the vertical motion unit (y) is used.
- When the starting position is set to the upper right or lower left of the print able area using **ESC T**, the horizontal motion unit (x) is used.

[Default] *n* = 30

[Reference] **ESC 2**

**ESC ? n**

[Name] Cancel user-defined characters

| [Format] | ASCII | ESC | ? | n |
|---|---|---|---|---|
| | Hex | 1B | 3F | n |
| | Decimal | 27 | 63 | n |

[Range] 32 ≤n≤ 126

[Description] Cancels user-defined characters.

[Notes]

- This command cancels the patterns defined for the character codes specified by n. After the user-defined characters are canceled, the corresponding patterns for the embedded characters are printed.
- This command deletes the pattern defined for the specified code in the font selectedby **ESC !**.
- If a user-defined characters have not been defined, the printer ignores this command.

[Reference] **ESC &, ESC %**

**ESC @**

[Name] Initialize printer

| [Format] | ASCII | ESC | @ |
|---|---|---|---|
| | Hex | 1B | 40 |
| | Decimal | 27 | 64 |

[Description] Clears the data in the print buffer and resets the printer mode to the mode that was in effect when the power was turned on.

[Notes]

- The DIP switch and memory switch settings are not checked again.
- The data in the receive buffer is not cleared.
- The macro definition is not cleared.

**ESC D n1...nk NUL**

[Name] Set horizontal tab positions

| [Format] | ASCII | ESC | D | n1...nk | NUL |
|---|---|---|---|---|---|
| | Hex | 1B | 44 | n1...nk | 00 |
| | Decimal | 27 | 68 | n1...nk | 0 |

[Range] 1 ≤n≤ 255

0 ≤k≤ 32

[Description] Sets horizontal tab positions.

- n specifies the column number for setting a horizontal tab position from the beginning of the line.
- k indicates the total number of horizontal tab positions to be set.

[Notes]

- The horizontal tab position is stored as a value of [character width ×n] measured from the beginning of the line. The character width includes the right-side character spacing, and double-width characters are set with twice the width of normal characters.
- This command cancels the previous horizontal tab settings.
- When setting n = 8, the print position is moved to column 9 by sending **HT**.
- Up to 32 tab positions (k = 32) can be set. Data exceeding 32 tab positions is processed as normal data.
- Transmit [n]k in ascending order and place a NUL code 0 at the end. When [n]k is less than or equal to the preceding value [n]k-1, tab setting is finished and
- The following data is processed as normal data.
- **ESC D NUL** cancels all horizontal tab positions.
- The previously specified horizontal tab positions do not change, even if the character width changes.
- The character width is memorized for each standard and page mode.

[Default] The default tab positions are at intervals of 8 characters (columns 9, 17, 25,...)

for Font A (12 × 24).

[Reference] **HT**

**ESC E n**

[Name] Turn emphasized mode on/off

| [Format] | | | | |
|---|---|---|---|---|
| | ASCII | ESC | E | n |
| | Hex | 1B | 45 | n |
| | Decimal | 27 | 69 | n |

[Range] 0 ≤n≤ 255

[Description] Turns emphasized mode on or off.

- When the LSB of n is 0, turns emphasized mode off.
- When the LSB of n is 1, turns emphasized mode on.

[Notes]

- Only the least significant bit of n is enabled.
- **ESC !** also turns on and off emphasized mode. However, the setting of the last received command is effective.

[Default] *n* = 0

[Reference] **ESC !**

**ESC G n**

| | | | | |
|---|---|---|---|---|
| [Name] | Turn on/off double-strike mode | | | |
| [Format] | ASCII | ESC | G | n |
| | Hex | 1B | 47 | n |
| | Decimal | 27 | 71 | n |
| [Range] | 0 ≤n≤ 255 | | | |

[Description] Turns double-strike mode on or off.

- When the LSB of n is 0, turns double-strike mode off.
- When the LSB of n is 1, turns double-strike mode on.

[Notes]

- Only the lowest bit of n is enabled.
- Since this printer does not support double-strike mode, printer output is the same in double-strike mode and in emphasized mode.

[Default] n = 0

[Reference] **ESC E**

**ESC J n**

| | | | | |
|---|---|---|---|---|
| [Name] | Print and feed paper | | | |
| [Format] | ASCII | ESC | J | n |
| | Hex | 1B | 4A | n |
| | Decimal | 27 | 74 | n |
| [Range] | 0 ≤n≤ 255 | | | |

[Description] Prints the data in the print buffer and feeds the paper [n× 0.125 mm].

[Notes]

- After printing is completed, this command sets the print starting position to the beginning of the line.
- The paper feed amount set by this command does not affect the values set by **ESC 2** or **ESC 3.**
- In standard mode, the printer uses the vertical motion unit (y).
- In page mode, this command functions as follows, depending on the starting position of the printable area:
  1. When the starting position is set to the upper left or lower right of the printable area using **ESC T**, the vertical motion unit (y) is used.
  2. When the starting position is set to the upper right or lower left of the print able area using **ESC T**, the horizontal motion unit (x) is used.
- When the BM sensor is enabled and in standard mode, this command is effective even if the set value exceeds the lavel length,.(BM =black mark.)

**ESC M n**

| | | | | |
|---|---|---|---|---|
| [Name] | Select character font | | | |
| [Format] | ASCII | ESC | M | n |
| | Hex | 1B | 4D | n |

| | | | | |
|---|---|---|---|---|
| | Decimal | 27 | 77 | n |

[Range] n = 0, 1, 48, 49

[Description] Selects the character font.

| n | Function |
|---|---|
| 0, 48 | Selects character Font A (12 × 24). |
| 1, 49 | Selects character Font B (9 × 17). |

[Details] **ESC !** can also select character font types. However, the setting of the last received command is effective.

[Reference] **ESC !**

**ESC S**

[Name] Select standard mode

| | | | |
|---|---|---|---|
| [Format] | ASCII | ESC | S |
| | Hex | 1B | 53 |
| | Decimal | 27 | 83 |

[Description] Switches from page mode to standard mode.

[Notes]

- This command is effective only in page mode.
- Data buffered in page mode is cleared.
- This command sets the print position to the beginning of the line.
- The printing area set by **ESC W** is initialized.
- This command switches the settings for the following commands (in which the values can be set independently in standard mode and page mode) to those for standard mode:
- Set right-side character spacing: **ESC SP**
- Select default line spacing: **ESC 2, ESC 3**

[Reference] **FF, ESC FF, ESC L**

**ESC V n**

[Name] Turn 90° clockwise rotation mode on/off

| | | | | |
|---|---|---|---|---|
| [Format] | ASCII | ESC | V | n |
| | Hex | 1B | 56 | n |
| | Decimal | 27 | 86 | n |

[Range] 0 ≤ n ≤ 1, 48 ≤ n ≤ 49

[Description] Turns 90° clockwise rotation mode on/off

n is used as follows:

| n | Function |
|---|---|
| 0, 48 | Turns off 90° clockwise rotation mode |
| 1, 49 | Turns on 90° clockwise rotation mode |

[Notes]

- This command affects printing in standard mode. However, the setting is always effective.

- When underline mode is turned on, the printer does not underline 90°clockwise-rotated characters.
- Double-width and double-height commands in 90° rotation mode enlarge characters in the opposite directions from double-height and double- width commands in normal mode.
- If this command is input in page mode, the printer performs only internal flag operations.

[Default] *n* = 0

[Reference] **ESC !, ESC −**

***ESC* \nL nH**

[Name] Set relative print position

| [Format] | ASCII | ESC | \ | *nL* | *nH* |
|---|---|---|---|---|---|
| | Hex | 1B | 5C | *nL* | *nH* |
| | Decimal | 27 | 92 | *nL* | *nH* |

[Range] 0 ≤*nL*≤ 255

0 ≤*nH*≤ 255

[Description]

- Sets the print starting position based on the current position using horizontal or vertical motion units.
- This command sets the next print starting position away from the current position by [(*nL* + *nH*× 256) × 0.125 mm]

[Notes]

- Any setting that exceeds the printable area is ignored.
- When pitch N is specified to the right:

*nL*+ *nH*× 256 = N

When pitch N is specified to the left (the negative direction), use the complement of

65536.

When pitch N is specified to the left:

*nL*+ *nH*× 256 = 65536 - N

- In standard mode, the horizontal motion unit is used.
- In page mode, the horizontal or vertical motion unit differs as follows, depending on the starting point of the printing area:
  1. When the starting position is set to the upper left or lower right of the printable area using **ESC T**, the horizontal motion unit (x) is used.
  2. When the starting position is set to the upper right or lower left of the printable area using **ESC T**, the vertical motion unit (y) is used.

[Reference] **ESC $**

**ESC a n**

[Name] Select justification

| [Format] | ASCII | ESC | a | *n* |
|---|---|---|---|---|
| | Hex | 1B | 61 | *n* |
| | Decimal | 27 | 97 | *n* |

[Range] 0 ≤*n*≤ 2, 48 ≤*n*≤ 50

[Description] Aligns all the data in one line to the specified position. n selects the justification as follows:

| n | Justification |
|---|---|
| 0, 48 | Left justification |
| 1, 49 | Centering |
| 2, 50 | Right justification |

[Notes]

- The command is enabled only when processed at the beginning of the line in standard mode.
- If this command is input in page mode, the printer performs only internal flag operations.
- This command has no effect in page mode.
- This command executes justification in the set print area.
- This command justifies the space area according to **HT, ESC $,** or **ESC \.**

[Default] *n* = 0

[Example]

Left justification

```
ASC
ABCD
ABCDE
```

Centering

```
ASC
ABCD
ABCDE
```

Right justification

```
ASC
ABCD
ABCDE
```

**ESC c 3 n**

[Name] Select paper-end sensor(s) to output paper-end signals

[Format]

| | | | | |
|---|---|---|---|---|
| ASCII | ESC | c | 3 | *n* |
| Hex | 1B | 63 | 33 | *n* |
| Decimal | 27 | 99 | 51 | *n* |

[Range] 0 ≤*n*≤ 255

[Description] Selects whether the specified paper sensor(s) to output paper end signals when a paper end is detected.

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0 | Off | — | — | Undefineddo |
| 1 | Off | 0 | 0 | Disables roll paper near-end sensor. |
| | On | 2 | 2 | Enables roll paper near-end sensor. |
| 2 | Off | — | — | Undefinedl |
| 3 | Off | 0 | 0 | Disables roll paper end sensor. |
| | On | 8 | 8 | Enables roll paper end sensor. |
| 4-7 | Off | — | — | Undefined |

[Notes]

- Multiple paper sensors can be selected. If multiple paper sensors are made valid, a paper-end signal is output when one of them detects a paper out.
- This command is enabled only with the parallel interface type.

[Default] *n* = 0

**ESC c 4 n**

[Name] Select paper sensor(s) to stop printing

[Format]

| | | | | |
|---|---|---|---|---|
| ASCII | ESC | c | 4 | *n* |
| Hex | 1B | 63 | 34 | *n* |
| Decimal | 27 | 99 | 52 | *n* |

[Range] 0 ≤*n*≤ 255

[Description] Selects the paper sensor(s) used to stop printing when a paper-end is detected, using n as follows:

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0 | Off | — | — | Undefineddo |
| 1 | Off | 00 | 0 | Roll paper near-end sensor disabled. |
| | On | 02 | 2 | Roll paper near-end sensor enabled. |
| 2-7 | — | — | — | Undefinedl |

[Notes]

- When a paper sensor is enabled with this command, printing stops after printing of the current line and paper feeding has finished.
- When a paper-end is detected by the roll paper sensor, the printer goes offline after printing stops.
- When either bit 1 is on, the printer selects the roll paper near-end sensor for the paper sensor to stop printing.

[Default] *n* = 0

**ESC c 5 n**

[Name] Enable/disable panel button

[Format]

| | | | | |
|---|---|---|---|---|
| ASCII | ESC | c | 5 | *n* |
| Hex | 1B | 63 | 35 | *n* |
| Decimal | 27 | 99 | 53 | *n* |

[Range] 0 ≤*n*≤ 255

[Description] Enables or disables the panel button.

- When the LSB of n is 0, enables the panel button.
- When the LSB of n is 1, disables the panel button.

[Notes]

- Only the lowest bit of n is valid.
- When the panel buttons are disabled, none of them are usable.
- With this printer, the panel buttons means the paper FEED button.
- When in macro execution standby, the FEED button is enabled regardless of thesetting of this command. However, the paper cannot be fed.

[Default] *n* = 0

**ESC d n**

[Name] Print and feed n lines

[Format]

| | | | |
|---|---|---|---|
| ASCII | ESC | d | *n* |
| Hex | 1B | 64 | *n* |

| | | | | |
|---|---|---|---|---|
| | Decimal | 27 | 100 | *n* |

[Range] 0 ≤ n ≤ 255

[Description] Prints the data in the print buffer and feeds n lines.

[Notes]

- This command sets the print starting position to the beginning of the line.
- This command does not affect the line spacing set by **ESC 2** or **ESC 3**.
- The maximum paper feed amount is 1016 mm {40"}. If the paper feed amount (n × line spacing) of more than 1016 mm {40"} is specified, the printer feeds the paper only 1016 mm {40"}.
- Even when the set value exceeds the maximum with the BM sensor enabled in standard mode, this command is effective. (BM = black mark.)

[Reference] **ESC 2**, **ESC 3**

**ESC { n**

[Name] Turns on/off upside-down printing mode

| | | | | |
|---|---|---|---|---|
| [Format] | ASCII | ESC | { | *n* |
| | Hex | 1B | 7B | *n* |
| | Decimal | 27 | 123 | *n* |

[Range] 0 ≤*n*≤ 255

[Description] Turns upside-down printing mode on or off.

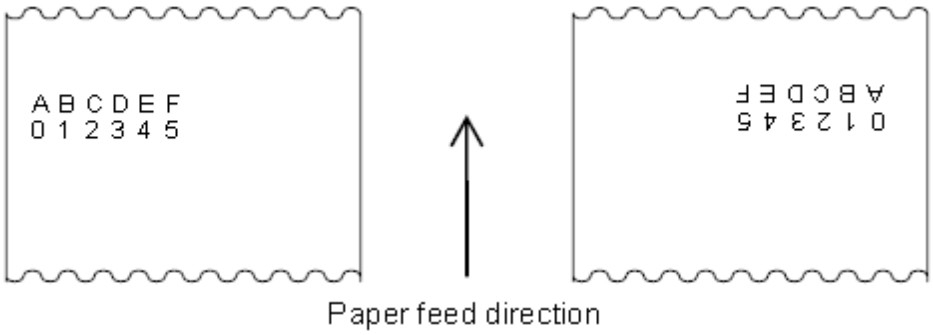
- When the LSB of *n* is 0, turns upside-down printing mode off.
- When the LSB of *n* is 1, turns upside-down printing mode on.

[Notes]

- Only the lowest bit of n is valid.
- This command is enabled only when processed at the beginning of a line in standard mode.
- This command does not affect printing in page mode.
- In upside-down printing mode, the printer rotates the line to be printed by 180° and then prints it.

[Default] *n* = 0

[Example] When upside-down printing mode is off. When upside-down printing mode is on




**FS p n m**

| | | | | | |
|---|---|---|---|---|---|
| [Name] | Print NV bit image | | | | |
| [Format] | ASCII | FS | p | *n* | *m* |

| | | | | | |
|---|---|---|---|---|---|
| | Hex | 1C | 70 | *n* | *m* |
| | Decimal | 28 | 112 | *n* | *m* |

[Range] 1 ≤*n*≤255

0 ≤*m*≤3 , 48 ≤*m*≤51

[Description] Prints NV bit image n using the mode specified by m.

| m | Mode | Vertical Dot Density | Horizontal Dot Density |
|---|---|---|---|
| 0, 48 | Normal | 203.2 dpi | 203.2 dpi |
| 1, 49 | Double-width | 203.2 dpi | 101.6 dpi |
| 2, 50 | Double-height | 101.6 dpi | 203.2 dpi |
| 3, 51 | Quadruple | 101.6 dpi | 101.6 dpi |

- *n* is the number of the NV bit image (defined using the **FS q** command).
- *m* specifies the bit image mode.

[Detail]

- NV bit image is a bit image defined in non-volatile memory by
- **FS q** and printed by **FS p**.
- This command is not effective when the specified NV bit image has not been defined.
- In standard mode, this command is effective only when there is no data in the print buffer.
- In page mode, this command is not effective.
- This command is not affected by print modes (emphasized, double-strike, underline, character size, white/black reverse printing, or 90°rotated characters, etc.), except upside-down printing mode.
- If the printing area width set by **GS L** and **GS W** for the NV bit image is less than one vertical line, the following processing is performed only on the line in question. However, in NV bit image mode, one vertical line means 1 dot in normal mode (m = 0, 48) and in double-height mode (m = 2, 50), and it means 2 dots in double-width mode (m = 1, 49) and in quadruple mode (m = 3, 51).
  1. The printing area width is extended to the right in NV bit image mode up to one line vertically. In this case, printing does not exceed the printable area.
  2. If the printing area width cannot be extended by one line vertically, the left margin is reduced to accommodate one line vertically.
- If the downloaded bit-image to be printed exceeds one line, the excess data is not printed.
- This command feeds dots (for the height n of the NV bit image) in normal and double-width modes, and (for the height n X 2 of the NV bit image) in doubleheight and quadruple modes, regardless of the line spacing specified by **ESC 2** or **ESC 3**.
- After printing the bit image, this command sets the print position to the beginning of the line and processes the data that follows as normal data.

[References] **ESC *, FS q, GS *l*, GS v 0**

**FS q n [xL xH yL yH d1...dk]1...[xL xH yL yH d1...dk]n**

| | | | | | |
|---|---|---|---|---|---|
| [Name] | Define NV bit image | | | | |
| [Format] | ASCII | FS | q | n | [xL xH yL yH d1...dk]1...[ xL xH yL yH d1...dk]n |
| | Hex | 1C | 71 | n | [xL xH yL yH d1...dk]1...[ xL xH yL yH d1...dk]n |
| | Decimal | 28 | 113 | n | [xL xH yL yH d1...dk]1...[ xL xH yL yH d1...dk]n |

[Range] 1 ≤*n*≤255

0 ≤*xL*≤255

0 ≤*xH*≤3 (when 1 ≤(*xL* + *xH* x 256) ≤1023

0 ≤yL ≤255

0 ≤yL ≤1 (when 1 ≤(yL + yH x 256) ≤288

0 ≤*d*≤255

k = (*xL* + *xH* x 256) x (*yL* + *yH* x 256) x 8

Total defined data area = 192K bytes

[Description]

- Define the NV bit image specified by n.
- n specifies the number of the defined NV bit image.
- *xL, xH* specifies (*xL* + *xH* x 256) x 8 dots in the horizontal direction for the NV bit image you are defining.
- *yL, yH* specifies (*yL* + *yH* x 256) x 8 dots in the vertical direction for the NV bit image you are defining.

[Notes]

- Frequent write command executions may damage the NV memory. Therefore, it is recommended to write the NV memory 10 times or less a day.
- The printer performs a hardware reset after the procedure to place the image into the NV memory. Therefore, user-defined characters, downloaded bit images, and macros should be defined only after completing this command. The printer clears the receive and print buffers and resets the mode to the mode that was in effect at power on. At this time, DIP switch settings are checked again.
- This command cancels all NV bit images that have already been defined by this command. The printer cannot redefine only one of several data definitions previously defined. In this case, all data needs to be sent again.
- From the beginning of the processing of this command till the finish of hardware reset, mechanical operations (including initializing the position of the print head when the cover is open, paper feeding using the FEED button, etc.) cannot be performed.
- During processing of this command, the printer is BUSY when writing data to the user NV memory and stops receiving data. Therefore it is prohibited to transmit the data, including real-time commands, during the execution of this command.
- NV bit image is a bit image defined in non-volatile memory by **FS q** and printed by **FS p**.

**GS FF**

| | | | |
|---|---|---|---|
| [Name] | Feed marked paper to print starting position | | |
| [Format] | ASCII | GS | FF |
| | Hex | 1D | 0C |
| | Decimal | 29 | 12 |

[Description] Feeds the marked paper to the print starting position.

[Notes:]

- This command is enabled only when the BM sensor is set to be effective using with DIP switch.
- This command sets the next print starting position to the beginning of the line.
- Even if this command is executed at the print starting position of the marked paper, the printer does not feed the marked paper to the next print starting position.

[Reference] **GS ( F, FF)**

**GS ! n**

[Name] Select character size

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | ! | n |
| Hex | 1D | 21 | n |
| Decimal | 29 | 33 | n |

[Range] 0 ≤n≤ 255

(1 ≤ vertical number of times ≤ 8, 1 ≤ horizontal number of times ≤ 8)

[Description] Selects the character height using bits 0 to 2 and selects the character width using bits 4 to 7, as follows:

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0 | See Table 2 [Height magnification] | | | Specification of height magnification. |
| 1 | | | | |
| 2 | | | | |
| 3 | | | | |
| 4 | See Table 1 [width magnification] | | | Specification of width magnification |
| 5 | | | | |
| 6 | | | | |
| 7 | | | | |

Table 1 Width magnification

| Hex | Decimal | Width magnification |
|---|---|---|
| 00 | 0 | 1 (normal) |
| 10 | 16 | 2 (double-width) |
| 20 | 32 | 3 |
| 30 | 48 | 4 |
| 40 | 64 | 5 |
| 50 | 80 | 6 |
| 60 | 96 | 7 |
| 70 | 112 | 8 |

Table 2 Width magnification

| Hex | Decimal | Height magnification |
|---|---|---|
| 00 | 0 | 1 (normal) |
| 01 | 1 | 2 (double-height) |
| 02 | 2 | 3 |
| 03 | 3 | 4 |
| 04 | 4 | 5 |
| 05 | 5 | 6 |
| 06 | 6 | 7 |
| 07 | 7 | 8 |

[Notes]

- This command is effective for all characters (alphanumeric and Kanji), except for HRI characters.
- If n is outside the defined range, this command is ignored.
- In standard mode, the vertical direction is the paper feed direction, and the horizontal direction is perpendicular to the paper feed direction. However, when character orientation changes in

90° clockwise-rotation mode, the relationship between vertical and horizontal directions is reversed.

- In page mode, vertical and horizontal directions are based on the character orientation.
- When characters are enlarged with different sizes on one line, all the characters on the line are aligned at the baseline.
- The **ESC !** command can also turn double-width and double-height modes on or off. However, the setting of the last received command is effective.

[Default] n = 0

[Reference] **ESC !**

**GS ∗x y d1...d(x × y × 8)**

[Name] Define downloaded bit image

| [Format] | | | | | | |
|---|---|---|---|---|---|---|
| | ASCII | GS | ∗ | x | y | d1...d(x × y × 8) |
| | Hex | 1D | 2A | x | y | d1...d(x × y × 8) |
| | Decimal | 29 | 42 | x | y | d1 ...d(x × y × 8) |

[Range] 1 ≤x≤ 255

1 ≤y≤ 48 (where x×y≤ 1536)

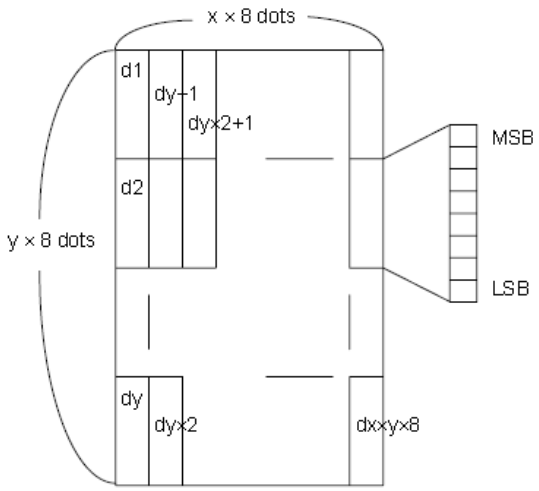
0 ≤d≤ 255

[Description]

- Defines a downloaded bit image using the number of dots specified by x and y.
- x specifies the number of dots in the horizontal direction.
- y specifies the number of dots in the vertical direction.

[Notes]

- The number of dots in the horizontal direction is x× 8; in the vertical direction it is y× 8.
- If x × y is out of the specified range, this command is disabled.
- The d indicates bit-image data. Data (d) specifies a bit printed as 1 and not printed as 0.
- The downloaded bit image definition is cleared when:
  1. **ESC** @ is executed.
  2. **ESC &** is executed.
  3. Printer is reset or the power is turned off.
- The following figure shows the relationship between the downloaded bit image and the printed data.




[Reference] **GS /**

**GS ( A pL pH n m**

[Name] Execute test print

[Format]

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| ASCII | GS | ( | A | pL | pH | n | m |
| Hex | 1D | 28 | 41 | pL | pH | n | m |
| Decimal | 29 | 40 | 65 | pL | pH | n | m |

[Range] (pL + pH× 256)=2 (where pL=2, pH=0)

0 ≤n≤ 2, 48 ≤n≤ 50

1 ≤m≤ 3, 49 ≤m≤ 51

[Description]

- Executes a test print with a specified test pattern on a specified paper.
- pL and pH set the number of parameters so that (pL + pH× 256) bytes. n specifies the paper to be tested.

n specifies the paper to be tested.

| n | Paper type |
|---|---|
| 0, 48 | Basic sheet (roll paper) |
| 1, 49<br>2, 50 | Roll paper |

m specifies a test pattern

| n | Test pattern |
|---|---|
| 1, 49 | Hexadecimal dump |
| 2, 50 | Printer status print |
| 3, 51 | Rolling pattern print |

[Details]

- This command has enabled only when processed at the beginning of a line in standard mode.
- This command is no effect in page mode.

- When this command is received during macro definition, the printer ends macro definition and begins performing this command.
- After the test print is finished, the printer resets itself automatically. Therefore, data already defined before this command is executed, such as user-defined characters, downloaded bit image, and macro, becomes undefined; the receive buffer and print buffer are cleared; and each setting returns to the default value. The printer also re-reads the DIP switch settings.
- The printer cuts the paper at the end of the test print.
- The printer goes BUSY while this command is executed.

**GS ( F pL pH a m nL nH**

[Name] Set adjustment value(s)

| [Format] | ASCII | GS | ( | F | pL | pH | a | m | nL | Nh |
|---|---|---|---|---|---|---|---|---|---|---|
| | Hex | 1D | 28 | 46 | pL | pH | a | m | nL | nH |
| | Decimal | 29 | 40 | 70 | pL | pH | a | m | nL | nH |

[Range] (pL + pH× 256) = 4 (where pL = 4, pH = 0)

1 ≤ a ≤ 2

m = 0, 48 or 1, 49

0 ≤( nL + nH× 256) ≤ 65535 (where 0 ≤nL≤ 255, 0 ≤nH≤ 255)

[Description]

This command is effective only when the BM sensor is enabled.

Sets adjustment values(s) for the printer operations specified by a.

- pL and pH specifies the number of the parameter such as a to (pL + pH× 256) bytes.
- a specifies setting values for the positions to start printing and cutting.

| a | Function |
|---|---|
| 1 | Sets value for the positions to start the printing. |
| 2 | Sets value for the positions to start the cutting. |

- m specifies the direction of the adjustment

| m | Function |
|---|---|
| 0, 48 | Specifies a forward paper feeding direction |
| 1, 49 | Specifies a backward paper feeding direction. |

- nL and nH specifies the setting value to [(nL + nH × 256) × 0.125 mm].

[Details]

- If this command (three bytes for **GS ( F**) is executed during a macro definition, the printer stops a macro definition and starts the execution process of this command.
- The adjustment value for the print starting position (a = 1) is affected with the following commands: **FF, GS FF**
- The adjustment value for the paper cutting position (a = 2) is affected with the following commands:**GS V m n**
- This command is stored in the receive buffer first from the host, then executed in the execution process of other normal commands. Therefore, there may occur time delay for the execution of this command after the printer receives this command. The delay time depends on the status of the receive buffer.

[Default]

All adjustment values are set to “0”.

(At the factory setting, the print starting position and the cutting position are set to the head position and the cutter position respectively when the BM sensor detects the BM.)

[Reference] **FF, GS FF, GS V**

[Notes]

- Frequent write command executions by a NV memory write command (**FS q, GS ( C**, **GS ( E, GS ( F, or GS ( M**) may damage the NV memory. Therefore, it is recommended to limit writing data with the write command into the NV memory to 10 times or fewer a day.
- The printer may go BUSY during processing of this command. However, be sure not to transmit data from the host computer while the printer is BUSY.

**GS ( K pL pH n m**

[Name] Select print control method(s)

| [Format] | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | ASCII | GS | ( | K | pL | pH | n | m |
| | Hex | 1D | 28 | 4B | pL | pH | n | m |
| | Decimal | 29 | 40 | 75 | pL | pH | n | m |

[Parameters] pL, pH specifies the number of bytes after n to (pL + pH× 256).

n specifies the function code.

m depends on the function code (n) See functional descriptions of this command.

[Range] (pL + pH× 256) = 2 (where pL = 2, pH = 0)

1 ≤n≤ 255

For m, see each functional descriptions of this command.

[Description] •Specifies various settings for print control or operations of print mechanism.

| n | Function no. | Function |
|---|---|---|
| 48 | Function 48 | Selects the print control mode. |
| 49 | Function 49. | Sets the print density. |

[Details]

- If a not-supported parameter’s value is processed under the following conditions, this command is ignored:
  1. In case of (pL + pH × 256) < 2
  2. In case that n is specified if n does not correspond to any functions of the printer.
  3. In case that m is out of range in each functions.
- The printer starts to process the specified function if all specified parameters are satisfied.
- If the printer is in an offline state, this command is not executed because the printer does not read the data.
- This command is stored in the receive buffer first from the host computer, then executed in the execution process of other normal commands. Therefore, there may be a time delay for the execution of this command after the Printer receives this command. The delay time depends on the status of the receive buffer.

**GS ( M pL pH a n m**

[Name] Customize printer control value(s)

| [Format] | ASCII | GS | ( | M | pL | pH | n | m |
|---|---|---|---|---|---|---|---|---|
| | Hex | 1D | 28 | 4D | pL | pH | n | m |
| | Decimal | 29 | 40 | 77 | pL | pH | n | m |

[Range] (pL + pH × 256) = 2 (pL = 2, pH = 0)

1 ≤ n ≤ 3, 49 ≤ n ≤ 51

0 ≤ m ≤ 1, 48 ≤ m ≤ 49

[Description] Saves or loads the data which are defined with the commands.

| n | Function |
|---|---|
| 1, 49 | Saves the data which are set by **GS ( F** to the user NV memory |
| 2, 50 | Loads the data which are set by **GS ( F** from the user NV memory |
| 3, 51 | Specifies to disable or enable the automatic-data-loading process at the initial setting. |

- m specifies the data as follows:

  m = 0, 48: The same with the initial setting value of GS ( F command described in this specification.

  m = 1, 49: Memory area to be stored.

[Default] Memory area to be stored (at the initial setting):

The same with the initial setting value of **GS ( F** command described in this specification.

[Reference] **ESC** @

[Notes]

- Frequent write command (**FS q, GS ( C, GS ( E, GS ( F, or GS ( M**) executions may damage the NV memory. Therefore, it is recommended to limit writing data with the write command into the NV memory to 10 times or fewer a day.
- If the printer becomes BUSY while processing this command, be sure not to transmit data from the host computer when the printer is BUSY.
- If the power is turned off or the printer is reset via an interface while this command is being executed, the printer may go into an abnormal condition. Be sure not to turn the power off or let the printer be reset via an interface while this command is being executed.

<Function1> **GS ( M pL pH n m ((n) = 1, 49)**

| [Format] | ASCII | GS | ( | M | pL | pH | n | m |
|---|---|---|---|---|---|---|---|---|
| | Hex | 1D | 28 | 4D | pL | pH | n | m |
| | Decimal | 29 | 40 | 77 | pL | pH | n | m |

[Range] (pL + pH × 256) = 2 (pL = 2, pH = 0)

n = 1, 49

m = 1, 49

[Description]

- Saves the data which are set by **GS ( F** command to the user NV memory.If the data will be written in the user NV memory which are already written with thesame data, the data saving process is not executed.
- If an error will occur in writing the data, the printer executes the memory error process.

[Details]

- The Printer executes the following process:
- Before saving the data to the NV memory, the printer sets BUSY for the interface. In this case, the printer becomes BUSY regardless of the memory switch settings.
- Even if the ASB function is enabled, the printer does not transmit the ASB status. However, if the status change occurs during the data transmission, the printer transmits the ASB status after transmitting the data.

[Default] None

[Reference] <Function 2> and <Function 3> of this command

<Function 2>**GS ( M pL pH n m ((n) = 2, 50)**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| [Format] | ASCII | GS | ( | M | pL | pH | n | m |
| | Hex | 1D | 28 | 4D | pL | pH | n | m |
| | Decimal | 29 | 40 | 77 | pL | pH | n | m |

[Range] (pL + pH × 256) = 2 (pL = 2, pH = 0)

n = 2, 50

0 ≤ m ≤ 1, 48 ≤ m ≤ 49

[Description]

- When m = 0 or 48, the setting value of **GS ( F** command is set to the default valuewhich is described in this specification.
- When m ≠ 0 or 48, the setting values are stored in area m of the memory.

[Details]

- This command is enabled only when processed at the beginning of a line in standardmode.
- This command has no effect in page mode.
- If this command is received during macro definition, the printer ends the macrodefinition, and begins performing this command.
- See <Function 1> of this command for the setting values for this function.

[Default] None

[Reference] <Function 1>

<Function 3>**GS ( M pL pH n m ((n) = 3, 51)**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| [Format] | ASCII | GS | ( | M | pL | pH | n | m |
| | Hex | 1D | 28 | 4D | pL | pH | n | m |
| | Decimal | 29 | 40 | 77 | pL | pH | n | m |

[Range] (pL + pH × 256) = 2 (pL = 2, pH = 0)

n = 3, 51

0 ≤ m ≤ 1, 48 ≤ m ≤ 49

[Description]

- When m = 0 or 48, the printer does not load the data from the user NV memory at theinitialization.After the initialization, the setting value of **GS ( F** becomes the initial value which isdescribed in this specification.
- When m ≠ 0 or 48, the printer loads the data from the user NV memory at theinitialization.After the initialization, the setting value of **GS ( F** becomes the setting values which arestored in area m of the memory.
- The specified data by this command are stored in the NV memory.If the data will be written in the NV memory which are already written with the samedata, the data saving process is not executed.
- If an error will occur in writing the data, the printer executes the memory error process.

[Details]

- When the data is automatically loaded, the initialization process is executed when anyof the following process is executed.
  - ☐ Power-on process by the power switch
  - ☐ Power-on process when the hardware reset is executed by the interface reset
  - ☐ Execution of **ESC** @

[Default] m = 0

[Reference] **ESC** @, <Function 1> of this command.

**GS ( k pL pH cn fn [parameters]**

[Name] Setup and print symbol

[Format]

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| ASCII | GS | ( | k | pL | pH | cn | fn | [parameter] |
| Hex | 1D | 28 | 6B | pL | pH | cn | fn | [parameter] |
| Decimal | 29 | 40 | 107 | pL | pH | cn | fn | [parameter] |

[Description] •Various processes are performed to the symbol specified with cn.

| cn | Type of symbol |
|---|---|
| 48 | PDF417 (two-dimensional codes) |
| 49 | QR Code (two-dimensional symbols) |
| 50 | MaxiCode (two-dimensional symbols) |
| 51 | Two-dimensional GS1 DataBar<br>(GS1 DataBar Stacked, GS1 DataBar Stacked Omnidirectional,<br>GS1 DataBar Expanded Stacked) |
| 52 | Composite Symbology (two-dimensional symbols) |

| cn | fn | Code | Function | Description |
|---|---|---|---|---|
| 48 | 65 | GS ( k pL pH cn fn n | Function 65 | Sets the number of columns for PDF417. |
| | 66 | GS ( k pL pH cn fn n | Function 66 | Sets the number of rows for PDF417. |
| | 67 | GS ( k pL pH cn fn n | Function 67 | Sets the module width for PDF417. |
| | 68 | GS ( k pL pH cn fn n | Function 68 | Sets the module height for PDF417. |

| | | | | |
|---|---|---|---|---|
| | 69 | GS ( k pL pH cn fn m n | Function 69 | Sets the error correction level for PDF417. |
| | 70 | GS ( k pL pH cn fn m | Function 70 | Specifies the options for PDF417. |
| | 80 | GS ( k pL pH cn fn m d1...dk | Function 80 | Stores received data in the symbol storage area. |
| | 81 | GS ( k pL pH cn fn m | Function 81 | Prints symbol data in the symbol storage area. |
| | 82 | GS ( k pL pH cn fn m | Function 82 | Transmits the size of information for the symbol data in the symbol storage area. |
| 49 | 65 | GS ( k pL pH cn fn n1 n2 | Function 165 | QR Code: Selects the model. |
| | 67 | GS ( k pL pH cn fn n | Function 167 | QR Code: Sets the size of module. |
| | 69 | GS ( k pL pH cn fn n | Function 169 | QR Code: Selects the error correction level. |
| | 80 | GS ( k pL pH cn fn md1...dk | Function 180 | QR Code: Stores the data into the symbol storage area. |
| | 81 | GS ( k pL pH cn fn m | Function 181 | QR Code: Prints the symbol data in the symbol storage area. |
| | 82 | GS ( k pL pH cn fn m | Function 182 | QR Code: Transmits the size information of the symbol data in the symbol storage area. |
| 50 | 65 | GS ( k pL pH cn fn n | Function 265 | MaxiCode: Selects the print mode. |
| | 80 | GS ( k pL pH cn fn m d1...dk | Function 280 | MaxiCode: Stores data in the symbol storage area. |
| | 81 | GS ( k pL pH cn fn m | Function 281 | MaxiCode: Prints symbol data in the symbol storage area. |
| | 82 | GS ( k pL pH cn fn m | Function 282 | MaxiCode: Transmits size information of the symbol data in the symbol storage area. |
| 51 | 67 | GS ( k pL pH cn fn n | Function 367 | Two-dimensional GS1 DataBar: Sets the module width. |
| | 71 | GS ( k pL pH cn fn nL nH | Function 371 | Two-dimensional GS1 DataBar: Sets the maximum width of GS1 DataBar Expanded Stacked. |
| | 80 | GS ( k pL pH cn fn m n d1...dk | Function 380 | Two-dimensional GS1 DataBar: Stores data in the symbol storage area. |
| | 81 | GS ( k pL pH cn fn m | Function 381 | Two-dimensional GS1 DataBar: Prints symbol data in the symbol storage area. |
| | 82 | GS ( k pL pH cn fn m | Function 382 | Two-dimensional GS1 DataBar: Transmits size information of the symbol data in the symbol storage area. |
| 52 | 67 | GS ( k pL pH cn fn n | Function 467 | Composite Symbology: Sets the module width. |
| | 71 | GS ( k pL pH cn | Function | Composite Symbology: Sets the maximum |

| | | | | |
|---|---|---|---|---|
| | | fn nL nH | 471 | width of GS1 DataBar Expanded Stacked. |
| | 72 | GS ( k pL pH cn fn n | Function 472 | Composite Symbology: Selects an HRI font. |
| | 80 | GS ( k pL pH cn fn m a b d1...dk | Function 480 | Composite Symbology: Stores data in the symbol storage area. |
| | 81 | GS ( k pL pH cn fn m | Function 481 | Composite Symbology: Prints symbol data in the symbol storage area. |
| | 82 | GS ( k pL pH cn fn m | Function 482 | Composite Symbology: Transmits size information of symbol data in the symbol storage area. |

- "Symbol data" refers to the data (d1…dk) received with <Function 080, 180, 280, 380,480>.
- "Symbol storage area" refers to the range for storing data received with <Function 080,180, 280, 380, 480> before encoding.

[Notes]

- After transmitting <Function 082, 182, 282, 382, or 482>, do not transmit the other datauntil its corresponding data is received.
- Only the ANK model has PDF417 (cn = 48).

[Reference] APPENDIX F

**GS / m**

[Name] Print downloaded bit image

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | / | m |
| Hex | 1D | 2F | m |
| Decimal | 29 | 47 | m |

[Range] 0 ≤m≤ 3, 48 ≤m≤ 51

[Description]

Prints a downloaded bit image using the mode specified by m.

m selects a mode from the table below:

| m | Mode | Vertical density | Horizontal density |
|---|---|---|---|
| 0, 48 | Normal | 203.2 dpi | 203.2 dpi |
| 1, 49 | Double-width | 203.2 dpi | 101.6 dpi |
| 2, 50 | Double-height | 101.6 dpi | 203.2 dpi |
| 3, 51 | Quadruple | 101.6 dpi | 101.6 dpi |

[dpi: dots per 25.4 mm {1"}]

[Notes]

- This command is ignored if a downloaded bit image has not been defined.
- In standard mode, this command is effective only when there is no data in the print buffer.
- This command has no effect in the print modes (emphasized, double-strike, underline, character size, or white/black reverse printing), except for upside-down printing mode.
- If the downloaded bit-image to be printed exceeds the printable area, the excess data is not printed.
- See Figure 3.10.4 for the downloaded bit image development position in page mode.

- If the width of the printing area set by **GS L** and **GS W** is less than the width required by the data sent with the **GS /** command; the following will be performed on the line in question (but the printing cannot exceed the maximum printable area)
  1. The width of the printing area is extended to the right to accommodate the amount of data.
  2. If step (1) does not provide sufficient width for the data, the left margin is reduced to accommodate the data.

For each bit of data in normal mode (m = 0, 48) and double-height mode (m = 2, 50),

the printer prints one dot: for each bit of data in double-width mode (m = 1, 49) and

quadruple mode (m = 3, 51), the printer prints two dots.

[Reference] **GS ∗**

**GS :**

[Name] Start/end macro definition

| [Format] | ASCII | GS | : |
|---|---|---|---|
| | Hex | 1D | 3A |
| | Decimal | 29 | 58 |

[Description] Starts or ends macro definition.

[Notes]

- Macro definition starts when this command is received during normal operation.
- Macro definition ends when this command is received during macro definition.
- When **GS ^** is received during macro definition, the printer ends macro definition and clears the definition.
- Macro is not defined when the power is turned on.
- The defined contents of the macro are not cleared by **ESC** @. Therefore, **ESC** @ can be included in the contents of the macro definition.
- If the printer receives **GS :** again immediately after previously receiving GS : the printer remains in the macro undefined state.
- The contents of the macro can be defined up to 2048 bytes. If the macro definition exceeds 2048 bytes, excess data is not stored.

[Reference] **GS ^**

**GS B n**

[Name] Turn white/black reverse printing mode

| [Format] | ASCII | GS | B | n |
|---|---|---|---|---|
| | Hex | 1D | 42 | n |
| | Decimal | 29 | 66 | n |

[Range] 0 ≤n≤ 255

[Description] Turns on or off white/black reverse printing mode.

- When the LSB of n is 0, white/black reverse mode is turned off.
- When the LSB of n is 1, white/black reverse mode is turned on.

[Notes]

- Only the lowest bit of n is valid.
- This command is available for built-in characters and user-defined characters.

- When white/black reverse printing mode is on, it also applies to character spacing set by **ESC SP.**
- This command does not affect bit images, user-defined bit images, bar codes, HRI characters, and spacing skipped by **HT, ESC $,** and **ESC **.
- This command does not affect the space between lines.
- White/black reverse mode has a higher priority than underline mode. Even if underline mode is on, it is disabled (but not canceled) when white/black reverse mode is selected.

[Default] n = 0

**GS C 0 n m [obsolete command]**

[Name] Select counter print mode

| [Format] | | | | | | |
|---|---|---|---|---|---|---|
| | ASCII | GS | C | 0 | n | m |
| | Hex | 1D | 43 | 30 | n | m |
| | Decimal | 29 | 67 | 48 | n | m |

[Range] 0 ≤n≤ 5

0 ≤m≤ 2, 48 ≤m≤ 50

[Description] Selects a print mode for the serial number counter.

- n specifies the number of digits to be printed as follows:
  - ☐ When n = 0, the printer prints the actual digits indicated by the number value.
  - ☐ When n≠ 0, this command sets the number of digits to be printed to n.
- m specifies the printing position within the entire range of printed digits, as follows:

| m | Printing position | Processing of digits less than those specified |
|---|---|---|
| 0, 48 | Align right | Adds spaces to the left. |
| 1, 49 | Align right | Adds 0 to the left. |
| 2, 50 | Align left | Adds spaces to the right. |

[Notes]

- When the digits of the actual counter exceeds the set digits for n, prints the last n degits of the counter.
- When n=0, the value of m has no meaning.

[Default] n = 0, m = 0

[Reference] **GS C 1, GS C 2, GS C ;, GS c**

[Example]

| n = 3, m = 0 | n = 3, m = 1 | n = 3, m = 2 |
|---|---|---|
| ΔΔ1 | 001 | 1ΔΔ |

Δ indicates a space.

**GS C 1 aL aH bL bH n r**

| | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| [Name] | Select count mode (A) | | | | | | | | | |
| [Format] | ASCII | GS | C | 1 | aL | aH | bL | bH | n | r |
| | Hex | 1D | 43 | 31 | aL | aH | bL | bH | n | r |
| | Decimal | 29 | 67 | 49 | aL | aH | bL | bH | n | r |
| [Range] | 0 ≤aL≤ 255 | | | | | | | | | |
| | 0 ≤aH≤ 255 | | | | | | | | | |
| | 0 ≤bL≤ 255 | | | | | | | | | |
| | 0 ≤bH≤ 255 | | | | | | | | | |
| | 0 ≤n≤ 255 | | | | | | | | | |
| | 0 ≤r≤ 255 | | | | | | | | | |

[Description] Selects a count mode for the serial number counter.

- aL, aH or bL, bH specify the counter range.
- n indicates the stepping amount for counting up or down.
- r indicates the repetition number with the counter value fixed.

[Notes]

- When [aL + aH × 256 < bL + bH× 256 and n≠ 0 and r≠ 0], sets count up mode.
- When [aL + aH × 256 > bL + bH× 256 and n≠ 0 and r≠ 0], sets count down mode.
- [aL + aH × 256 > bL + bH× 256 or n = 0 or r = 0], sets count stop.
- When in count up mode,[aL + aH× 256] is the minimum counter value, and [bL + bH×256] is the maximum counter value. When the counter value exceeds the maximum value, restarts count up from the minmum count value.
- When in count down mode, [aL + aH× 256] is the maximum counter value, and [bL + bH×256] is the mumimum counter value. When the counter value is less than the minimum value, restarts count down from the maximum value.
- When this command is executed, the internal couonter that indicates the number of times of repeated printing is cleared.

[Default] aL = 1, aH = 0, bL = 255, bH = 255, n = 1, r = 1

[Reference] **GS C 0, GS C 2, GS C ;, GS c**

**GS C 2 nL nH**

| | | | | | | |
|---|---|---|---|---|---|---|
| [Name] | Set counter | | | | | |
| [Format] | ASCII | GS | C | 2 | nL | nH |
| | Hex | 1D | 43 | 32 | nL | nH |
| | Decimal | 29 | 67 | 50 | nL | nH |
| [Range] | 0 ≤nL≤ 255 | | | | | |
| | 0 ≤nH≤ 255 | | | | | |

[Description] Sets the serial number counter value to (nL + nH× 256).

[Notes]

- When in count up mode, if the counter value set with this command is outside the counter range set with **GS C** 1or **GS C ;,**the next execution of **GS c** forcively changes the counter value to the minimum value.

- When in count down mode, if the counter value set with this command is outside the counter range set with **GS C 1**or **GS C ;**, the next execution of **GS c** forcively changes the counter value to the maximum value.

[Default] nL = 1, nH = 0

[Reference] **GS C 0, GS C 1, GS C ; , GS c**

**GS C ; sa ; sb ; sn; sr; sc;**

[Name] Select count mode (B)

[Format]

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ASCII | GS | C | ; | sa | ; | sb | ; | sn | ; | sr | ; | sc | ; |
| Hex | 1D | 43 | 3B | sa | 3B | sb | 3B | sn | 3B | sr | 3B | sc | 3B |
| Decimal | 29 | 67 | 59 | sa | 59 | sb | 59 | sn | 59 | sr | 59 | sc | 59 |

[Range] “0”≤ sa ≤“65535”

“0”≤ sb ≤“65535”

“0”≤ sn ≤“255”

“0”≤ sr ≤“255”

“0”≤ sc ≤“65535”

[Description] Selects a count mode for the serial number counter and specifies the value of the counter.

- sa, sb, sn, sr, and sc are all displayed in ASCII characters, using the codes for “0” to “9.”
- sa and sb specify the counter range.
- sn indicates the stepping amount for counting up or down.
- sr indicates the repetition number with the counter value fixed.
- sc indicates the counter value.

[Notes]

- Count-up mode is specified when: sa < sb and sn ≠“0” and sr ≠“0”
- Count-down mode is specified when: sa > sb and sn ≠“0” and sr ≠“0”
- Counting stops when: sa = sb or sn = “0” or sr = “0”
- When count-up mode is specified, sa is the minimum counter value and sb is the maximum counter value. If counting up reaches a value exceeding the maximum, it is resumed with the minimum value. If the counter value set by sc is outside the counter operation range, the counter value is forced to convert to the minimum value by executing **GS c**.
- When count-down mode is specified, sa is the maximum counter value and sb is the minimum counter value. If counting down reaches a value less than the minimum, it is resumed with the maximum value. If the counter value set by sc is outside the counter operation range, the counter value is forced to convert to the maximum value by executing **GS c.**
- Parameters sa to sc can be omitted. If omitted, these argument values are unchanged.
- Parameters sa to sc must not contain characters, except 0 to 9.
- If an incorrect syntax is used, the corresponding parameter setting has no effect, and the data after that is processed as normal data.

[Default] sa = “1,” sb = “65535,” sn = “1,” sr = “1,” sc = “1”

[Reference] **GS C 0, GS C 1, GS C 2, GS c**

**GS E n**

[Name] Select head control method

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | E | n |
| Hex | 1D | 45 | n |
| Decimal | 29 | 69 | n |

[Range] 0 ≤n≤ 255

[Description] Selects the head control method.

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0-3 | Off | 00 | 0 | Undefined |
| 4 | (See Table [Print speed].) | | | Specifies the print speed. |
| 5 | | | | |
| 6-7 | Off | 00 | 0 | Undefined |

Table [Print speed]

| Speed level | Bit 5 | Bit 4 | |
|---|---|---|---|
| Speed 1 (250 mm/s max.) | 0 | 0 | High speed |
| Speed 2 (200 mm/s max.) | 0 | 1 | |
| Speed 3 (150 mm/s max.) | 1 | 0 | |
| Speed 4 (100 mm/s max.) | 1 | 1 | Low speed |

[Notes]

- In standard mode, this command is effective only at the starting position of the line.
- In two-part energization at speed level 1 is 126 mm/s maximum.
- In four-part energization, the print speed is always 59 mm/s maximum.

[Default] Speed level 1.

**GS H n**

[Name] Select print position of HRI characters

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | H | n |
| Hex | 1D | 48 | n |
| Decimal | 29 | 72 | n |

[Range] 0 ≤n≤ 3, 48 ≤n≤ 51

[Default] n = 0

[Description] Selects the print position of HRI characters when printing a bar code.

| n | Print position |
|---|---|
| 0, 48 | Not printed. |
| 1, 49 | Above the bar code. |
| 2, 50 | Below the bar code. |
| 3, 51 | Both above and below the bar code. |

[Notes]

- HRI stands for Human Readable Interpretation.

- HRI characters are printed using the font specified by **GS f**.

[Default] n = 0

[Reference] **GS f, GS k**

**GS I n**

[Name] Transmit printer ID

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | I | n |
| Hex | 1D | 49 | n |
| Decimal | 29 | 73 | n |

[Range] 1 ≤n≤ 3, 49 ≤n≤ 51

[Description]

- Transmits the ID of the printer specified.
- The printer IDs that can be specified are as follows

| n | Type of printer ID | Value (Hex) |
|---|---|---|
| 1, 49 | Printer model ID | 07 |
| 2, 50 | Type ID | See Table [Type ID]. |
| 3, 51 | ROM version ID | Depends on the ROM version |

[Type ID]

| Bit | Off/On | Hex | Decimal | Contents |
|---|---|---|---|---|
| 0 | Off | 0 | 0 | Multi-byte code characters not supported. |
| | On | 1 | 1 | Multi-byte code characters supported. |
| 1 | On | 0 | 0 | Autocutter not installed. |
| | On | 2 | 2 | Autocutter Installed. |
| 2 | Off | 0 | 0 | BM sensor disabled. |
| | On | 4 | 4 | BM sensor enabled. |
| 3 | Off | 0 | 0 | Not used. |
| 4 | Off | 0 | 0 | Not used. |
| 5 | — | — | — | Not defined.abl |
| 6 | — | — | — | Not defined.abl |
| 7 | Off | 0 | 0 | Not used. |

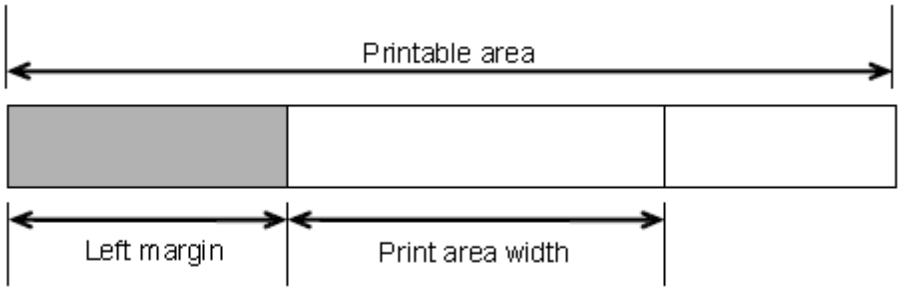
[Details] When ASB is enabled by GS a, it is necessary to identify the printer ID set by GS a and the priner status obtained by ASB. See APPENDIX C for how to identify them.

**GS L nL nH**

| | | | | | |
|---|---|---|---|---|---|
| [Name] | Set left margin | | | | |
| [Format] | ASCII | GS | L | nL | nH |
| | Hex | 1D | 4C | nL | nH |
| | Decimal | 29 | 76 | nL | nH |
| [Range] | 0 ≤nL≤ 255 | | | | |
| | 0 ≤nH≤ 255 | | | | |

[Description]

- Sets the left margin using nL and nH.
- The left margin is set to [(nL + nH× 256) × 0.125 mm].




[Notes]

- This command is effective only when processed at the beginning of the line in standard mode.
- If this command is input in page mode, the printer performs only internal flag operations.
- This command does not affect printing in page mode.
- The maximum left margin equals the printable area in the horizontal direction. If the setting exceeds the printable area, the maximum value of the printable area is used.
- The left margin can be set only with each 8 bits by this command when the raster bit image com and (**GS v 0**) is executed. If the left margin to set cannot be divided by eight, the remainder is omitted.

Example)

When (nL + nH× 256) = 20, the setting value is 16.

[Default] nL = 0, nH = 0

[Reference] **GS W**

**GS T n**

| | | | | |
|---|---|---|---|---|
| [Name] | Set print position to the beginning of print line | | | |
| [Format] | ASCII | GS | T | n |
| | Hex | 1D | 54 | n |
| | Decimal | 29 | 84 | n |
| [Range] | n = 0, 1, 48, 49 | | | |

[Description]

- Sets the print position to the beginning of print line.
- n specifies the data processing method in the print buffer.

| n | Printing position |
|---|---|
| 0, 48 | Sets the print position to the beginning of print line after deleting all data in the print buffer. |
| 1, 49 | Set the print position to the beginning of print line after printing all data in the print buffer |

**(1) GS V m (2) GS V m n**

[Name] Select cut mode and cut paper

[Format]

| | | | | | |
|---|---|---|---|---|---|
| (1) | ASCII | GS | V | m | |
| | Hex | 1D | 56 | m | |
| | Decimal | 29 | 86 | m | |
| (2) | ASCII | GS | V | m | n |
| | Hex | 1D | 56 | m | n |
| | Decimal | 29 | 86 | m | n |

[Range] (1) m = 1, 49

(2) m = 66, 0 ≤n≤ 255

[Description] Selects a mode for cutting paper and executes paper cutting. The value of m selects the mode as follows:

| m | Print mode |
|---|---|
| 1, 49 | Cuts paper |
| 66 | Feeds paper (cutting position + [n × 0.125 mm]), and cuts the paper. |

[Details]

- The cutting depends on the autocutter unit installed.
- In standard mode, cutting is enabled only when curring at the beginning of the line is enabled.

[Details of (1) and (2)]

- Cuts paper.

[Details of (2)]

- When n = 0, the printer feeds the paper to the cutting position and cuts it.
- When n ≠ 0, the printer feeds the paper to (cutting position + [n × 0.125 mm {0.0049”}]) and cuts it.
- When the BM sensor is set to be effective with DIP switch 1-1, [(Value which is set by **GS ( F**) + 0.125 mm] is applied.

**GS W nL nH**

[Name] Set printing area width

[Format]

| | | | | |
|---|---|---|---|---|
| ASCII | GS | W | nL | nH |
| Hex | 1D | 57 | nL | nH |
| Decimal | 29 | 87 | nL | nH |

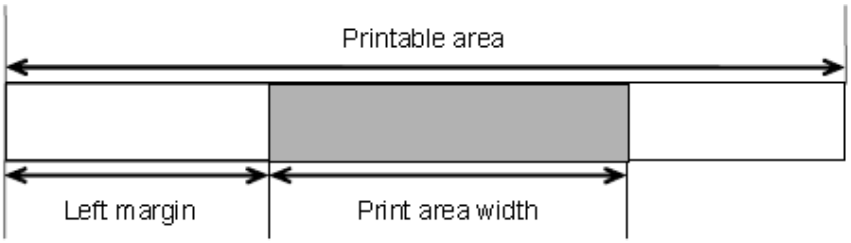
[Range] 0 ≤nL≤ 255

0 ≤nH≤ 255

[Description]

- Sets the printing area width to the area specified by nL and nH.

- The printing area width is set to [(nL + nH× 256) × 0.125 mm].




[Notes]

- This command is effective only when processed at the beginning of the line.
- If this command is input in page mode, the printer performs only internal flag operations.
- This command does not affect printing in page mode.
- If the setting exceeds the printable area, the maximum value of the printable area is used.
- The setting by **GS L** takes precedence over the setting by **GS W**. If the [left margin + printing area width] exceeds the printable area, the printer uses [Printable area width - left margin]. However, the setting by **GS W** is still reserved, even when it is not used in the current printing.
- If the width set for the printing area is less than the width of one character, when the character data is developed, the following processing is performed:

[Reference] **GS L**

**GS ^ r t m**

| | | | | | | |
|---|---|---|---|---|---|---|
| [Name] | Execute macro | | | | | |
| [Format] | ASCII | GS | ^ | r | t | m |
| | Hex | 1D | 5E | r | t | m |
| | Decimal | 29 | 94 | r | t | m |
| [Range] | 0 ≤r≤ 255 | | | | | |
| | 0 ≤t≤ 255 | | | | | |
| | 0 ≤m≤ 1 | | | | | |

[Description] Executes a macro.

- r specifies the number of times to execute the macro.
- t specifies the waiting time for executing the macro.
- m specifies macro executing mode.

When the LSB of m = 0:

The macro executes r times continuously at the interval specified by t.

When the LSB of m = 1:

After waiting for the period specified by t, the PAPER OUT LED indicators flash and the printer waits for the FEED button to be pressed. After the button is pressed, the printer executes the macro once. The printer repeats the operation r times.

[Notes]

- The waiting time is approximately (t × 100 ms) for every macro execution.
- If this command is received while a macro is being defined, the macro definition is aborted and the definition is cleared.
- If the macro is not defined or if r is 0, this command is ignored.

- When the macro is executed (m = 1), paper cannot be fed by using the FEED button.

[Reference] **GS :**

**GS c**

[Name] Print counter

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | c | |
| Hex | 1D | 63 | |
| Decimal | 29 | 99 | |

[Description] Sets the serial counter value in the print buffer and increments or decrements the counter value.

[Notes]

- After setting the current counter value in the print buffer as print data (a character string), the printer counts up or down based on the count mode set. The counter value in the print buffer is printed when the printer receives a print command or is in the buffer-full state.
- The counter print mode is set by **GS C 0**.
- The counter mode is set by **GS C 1** or **GS C**.
- In count-up mode, if the counter value set by this command goes out of the counter operation range set by **GS C 1** or **GS C**, it is forced to convert to the minimum value.
- In count-down mode, if the counter value set by this command goes out of the counter operation range set by **GS C 1** or **GS C**, it is forced to convert to the maximum value.

[Reference] **GS C 0, GS C 1, GS C 2, GS C**;

**GS f n**

[Name] Select font for Human Readable Interpretation (HRI) characters

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | f | n |
| Hex | 1D | 66 | n |
| Decimal | 29 | 102 | n |

[Range] n = 0, 1, 48, 49

[Description] Selects a font for the HRI characters used when printing a bar code.

n selects a font from the following table:

| n | Font |
|---|---|
| 0, 48 | Font A (12 × 24) |
| 1, 49 | Font B (9 × 17) |

[Notes]

- HRI stands for Human Readable Interpretation.
- HRI characters are printed on the position specified by **GS H**.

[Default] n = 0

[Reference] **GS H, GS k**

**GS h n**

[Name] Select bar code height

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | h | n |
| Hex | 1D | 68 | n |

| | | | | |
|---|---|---|---|---|
| | Decimal | 29 | 104 | n |

[Range] 1 ≤n≤ 255

[Description] Selects the height of the bar code.

n specifies the number of dots in the vertical direction.

[Default] n = 162

[Reference] **GS k**

**<A> GS k md1...dk NUL**

**<B> GS k m nd1...dn**

[Name] Print bar code

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| [Format] | <A> ASCII | GS | k | m | d1 | dk | NUL |
| | Hex | 1D | 6B | m | d1 | dk | 00 |
| | Decimal | 29 | 107 | m | d1 | dk | 0 |
| | <B> ASCII | GS | k | m | n | d1...dn | |
| | Hex | 1D | 6B | m | n | d1...dn | |
| | Decimal | 29 | 107 | m | n | d1...dn | |

[Range]

<A> 0 ≤m≤ 6 (k and d depend on the bar code system used)

<B> 65 ≤m≤ 73 (n and d depend on the bar code system used)

[Description] Selects a bar code system and prints the bar code.

m selects a bar code system as follows:

<Function A>

| m | Bar code system | Range of k | Range of d |
|---|---|---|---|
| 0 | UPC-A | k =11, 12 | 48 ≤ d ≤ 57 |
| 1 | UPC-E | 6 ≤ k ≤ 8, k =11, 12 | 48 ≤ d ≤ 57[Where k=7,8,11,12, d1=48] |
| 2 | JAN13 / EAN13 | k = 12, 13 | 48 ≤ d ≤ 57 |
| 3 | JAN 8 / EAN8 | k = 7, 8 | 48 ≤ d ≤ 57 |
| 4 | CODE39 | 1 ≤ k | 48 ≤ d ≤ 57, 65 ≤ d ≤ 90, d = 32, 36, 37, 42,43, 45, 46, 47 |
| 5 | ITF | 2 ≤ k (even number) | 48 ≤ d ≤ 57 |
| 6 | CODABAR (NW-7) | 2 ≤ k | 48 ≤ d ≤ 57, 65 ≤ d ≤ 68, 97 ≤ d ≤ 100,d=36, 43, 45, 46, 47, 58[Where 65 ≤ d1 ≤ 68, 65≤ dk ≤68,97 ≤ d1 ≤ 100, 97≤ dk ≤ 100] |

- k of <Function A> indicates the number of bar code data.
- d specifies bar code data.

| m | Bar code system | Range of n | Range of d |
|---|---|---|---|
| 65 | UPC-A | n = 11, 12 | 48 ≤ d ≤ 57 |
| 66 | UPC-E | 6 ≤ n ≤ 8,<br>n =11, 12 | 48 ≤ d ≤ 57 [Where n=7,8,11,12, d1=48] |
| 67 | JAN13 (EAN13) | n = 12, 13 | 48 ≤ d ≤ 57 |
| 68 | JAN 8 (EAN8) | n = 7, 8 | 48 ≤ d ≤ 57 |

| | | | |
|---|---|---|---|
| 69 | CODE39 | 1 ≤ n ≤ 255 | 48 ≤ d ≤ 57, 65 ≤ d ≤ 90, d=32, 36, 37, 42, 45,46, 47 |
| 70 | ITF | 2 ≤ n ≤ 254 (even number) | 48≤ d ≤ 57 |
| 71 | CODABAR (NW-7) | 2 ≤ n ≤ 255 | 48 ≤ d ≤ 57, 65 ≤ d ≤ 68, 97 ≤ d ≤ 100,d=36, 43, 45, 46, 47, 58 [Where 65 ≤ d1 ≤ 68, 65 ≤ dn ≤ 68,97 ≤d1 ≤100, 97 ≤dn ≤100] |
| 72 | CODE93 | 1 ≤ n ≤ 255 | 0 ≤ d ≤ 127 |
| 73 | CODE128 | 2 ≤ n ≤ 255 | 0 ≤ d ≤ 127 [Where d1=123, 65 ≤ d2 ≤ 67] |
| 74 | GS1-128 | 2 ≤ n ≤ 255 | 0 ≤ d ≤ 127 |
| 75 | GS1 DataBar Omnidirectional | n=13 | 48 ≤ d ≤ 57 |
| 76 | GS1 DataBar Truncated | n=13 | 48 ≤ d ≤ 57 |
| 77 | GS1 DataBar Limited | n=13 | 48 ≤ d ≤ 57 [Where 48 ≤ d1 ≤ 49] |
| 78 | GS1 DataBar Expanded | 2 ≤ n ≤255 | 32 ≤ d ≤ 34, 37 ≤ d ≤ 63, 65 ≤ d ≤ 90, d=95,97 ≤ d ≤ 122, d=123[Where d1=40, 48 ≤ d2 ≤ 57, 48 ≤ d3 ≤ 57, or 48 or d1 ≤57, 48 ≤d2 ≤57] |

- n of <Function B> specifies the number of bytes of bar code data.
- d specifies bar code data.

[Notes] Users must secure the quiet zone (left or right side space area defined by the bar code standard) for bar code printing.

<Notes on Function A>

- This command ends with a NUL code.
- When the bar code system used is UPC-A or UPC-E, the printer prints the bar code data after receiving 12 bytes of bar code data and processes the following data as normal data.
- When the bar code system used is JAN13 (EAN13), the printer prints the bar code after receiving 13 bytes of bar code data and processes the following data as normal data.
- When the bar code system used is JAN8 (EAN8), the printer prints the bar code after receiving 8 bytes of bar code data and processes the following data as normal data.
- The number of data for the ITF bar code must be even numbers. When an odd number of bytes of data is input, the printer ignores the last received data.

<Notes on Function B>

- n indicates the number of bar code data bytes, and the printer processes n bytes from the next character data as bar code data.
- If n is outside the specified range, the printer stops command processing and processes the following data as normal data.

[Notes on standard mode]

- If d is outside the specified range, the printer only feeds paper and processes the following data as normal data.
- If the horizontal size exceeds printing area, the printer only feeds the paper.
- This command feeds as much paper as is required to print the bar code, regardless of the line spacing specified by ESC 2 or ESC 3.

- This command is enabled only when no data exists in the print buffer. When data exists in the print buffer, the printer processes the data following m as normal data.
- After printing the bar code, this command sets the print position to the beginning of the line.
- This command is not affected by print modes (emphasized, double-strike underline, character size, white/black reverse printing, or 90° rotated character, etc.), except for upside-down printing mode.

[Notes on page mode]

- This command develops bar code data in the print buffer, but does not print it. After processing bar code data, this command moves the print position to the right side dot of the bar code.
- If d is out of the specified range, the printer stops command processing and processes the following data as normal data. In this case the data buffer position does not change.
- If bar code width exceeds the printing area, the printer does not print the bar code, but moves the data buffer position to the left side out of the printing area.
- See Section 3.10, Page Mode.

[When CODE93 (m = 72) is used]

- The printer prints an HRI character ( ) as the start character at the beginning of the HRI character string.
- The printer prints an HRI character ( ) as a stop character at the end of the HRI character string.
- The printer prints HRI characters (■+ an alphabetic character) as a control character(<00>H to <1F>H and <7F>H):

| Control character | | | HRI character | Control character | | | HRI character |
|---|---|---|---|---|---|---|---|
| ASCII | Hex | Decimal | | ASCII | Hex | Decimal | |
| NUL | 0 | 0 | ■U | DLE | 10 | 16 | ■P |
| SOH | 1 | 1 | ■A | DC1 | 11 | 17 | ■Q |
| STX | 2 | 2 | ■B | DC2 | 12 | 18 | ■R |
| ETX | 3 | 3 | ■C | DC3 | 13 | 19 | ■S |
| EOT | 4 | 4 | ■D | DC4 | 14 | 20 | ■T |
| ENQ | 5 | 5 | ■E | NAK | 15 | 21 | ■U |
| ACK | 6 | 6 | ■F | SYN | 16 | 22 | ■V |
| BEL | 7 | 7 | ■G | ETB | 17 | 23 | ■W |
| BS | 8 | 8 | ■H | CAN | 18 | 24 | ■X |
| HT | 9 | 9 | ■I | EM | 19 | 25 | ■Y |
| LF | 0A | 10 | ■J | SUB | 1A | 26 | ■Z |
| VT | 0B | 11 | ■K | ESC | 1B | 27 | ■A |
| FF | 0C | 12 | ■L | FS | 1C | 28 | ■B |
| CR | 0D | 13 | ■M | GS | 1D | 29 | ■C |
| SO | 0E | 14 | ■N | RS | 1E | 30 | ■D |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| SI | 0F | 15 | ■O | US | 1F | 31 | ■E |
| | | | | DEL | 7F | 127 | ■T |

<Example> Printing GS k 72 7 67 111 100 101 13 57 51




When CODE128 (m = 73) is used:

- See APPENDIX E for the information for the CODE128 bar code and its code table.
- When using CODE128 in this printer, take the following points into account for data transmission:

  1. The top of the bar code data string must be the code set selection character CODE A, CODE B, or CODE C), which selects the first code set.
  2. Special characters are defined by combining two characters "{" and one character. The ASCII character "{" is defined by transmitting "{" twice consecutively.

| Specific character | Transmit data | | |
|---|---|---|---|
| | **ASCII** | **Hex** | **Decimal** |
| SHIFT | {S | 7B, 53 | 123, 83 |
| CODE A | {A | 7B, 41 | 123, 65 |
| CODE B | {B | 7B, 42 | 123, 66 |
| CODE C | {C | 7B, 43 | 123, 67 |
| FNC1 | {1 | 7B, 31 | 123, 49 |
| FNC2 | {2 | 7B, 32 | 123, 50 |
| FNC3 | {3 | 7B, 33 | 123, 51 |
| FNC4 | {4 | 7B, 34 | 123, 52 |
| "{" | {{ | 7B, 7B | 123, 123 |

<Example> Example data for printing "No. 123456"

This example, the printer first prints "No." using CODE B, then prints the following numbers using CODE C.

GS k 73 10 123 66 78 111 46 123 67 12 34 56




- If the top of the bar code data is not the code set selection character, the printer stops command processing and processes the following data as normal data.
- If the combination of "{" and the following character does not apply any special character, the printer stops command processing and processes the following data as normal data.
- If the printer receives characters that cannot be used in the special code set, the printer stops command processing and processes the following data as normal data.

- The printer does not print HRI characters that correspond to the shift characters or code set selection characters.
- HRI character for the function character is space.
- HRI characters for the control character (<00>H to <1F>H and <7F>H) are space.
- <Others> Be sure to keep spaces on both right and left sides of a bar code.(Spaces are different depending on the types of the bar code.)

[Reference] **GS H, GS f, GS h, GS w**

**GS r n**

[Name] Transmit status

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | r | n |
| Hex | 1D | 72 | n |
| Decimal | 29 | 114 | n |

[Range] n = 1, 49

[Description] Transmits the status specified by n as follows:

| n | Function |
|---|---|
| 1, 49 | Transmits paper sensor status |

[Notes]

- When using a serial interface
- When DTR/DSR control is selected, the printer transmits only 1 byte after confirming the host is ready to receive data (DSR signal is SPACE). If the host computer is not ready to receive data (DSR signal is MARK), the printer waits until the host is ready.
- When XON/XOFF control is selected, the printer transmits only 1 byte without confirming the condition of the DSR signal.
- This command is executed when the data in the receive buffer is developed.
- Therefore, there may be a time lag between receiving this command and transmitting the status, depending on the receive buffer status.
- When Auto Status Back (ASB) is enabled using GS a, the status transmitted by **GS r** and the ASB status must be differentiated using the table in APPENDIX C.
- The status types to be transmitted are shown below:

Paper sensor status (n = 1, 49):

| Bit | Off/On | Hex | Decimal | Status for ASB |
|---|---|---|---|---|
| 0, 1 | Off | 00 | 0 | Roll paper near-end sensor: paper adequate. |
| | On | 03 | 3 | Roll paper near-end sensor: paper near end. |
| 2, 3 | Off | 00 | 0 | Roll paper end sensor: paper present. |
| | On | (0C) | (12) | Roll paper end sensor: paper not present. |
| 4 | Off | 00 | 0 | Not used. |
| 5,6 | - | - | - | Undefined. |
| 7 | Off | 00 | 0 | Not used. |

Bits 2 and 3: When the paper end sensor detects a paper end, the printer goes offline and does not execute this command. Therefore, bits 2 and 3 do not transmit the status of paper end.

[Reference] **DLE EOT, GS a**

**GS v 0 m xL xH yL yH d1...dk** **[obsolete command]**

**GS ( L**<Function 112 and 50>, which is the upward-compatible command replacing **GS v 0**, is recommended to use, since **GS v 0** is an obsolete command in the ESC/POS command system.

[Name] Print raster bit image

[Format]

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| ASCII | GS | v | 0 | m | xL | xH | yL | yH | d1...dk |
| Hex | 1D | 76 | 30 | m | xL | xH | yL | yH | d1...dk |
| Decimal | 29 | 118 | 48 | m | xL | xH | yL | yH | d1...dk |

[Range] 0 ≤ m ≤ 3, 48 ≤ m ≤ 51

1 ≤ xL ≤ 255

1 ≤ xH ≤ 255 [where 1 ≤ (xL+xH × 256) ≤ 128]

0 ≤ yL ≤ 255

0 ≤ yH ≤ 15 [where 1 ≤ (yL+yH × 256) ≤ 4095]

0 ≤ d ≤ 255

k = (xL+xH × 256) × (yL+yH × 256) [where k ≠ 0]

[Description] Prints a raster bit image using the mode specified by m.

| m | Mode | Vertical direction | Horizontal direction |
|---|---|---|---|
| 0, 48 | Normal | 203.2 dpi | 203.2 dpi |
| 1, 49 | Double-width | 203.2 dpi | 101.6 dpi |
| 2, 50 | Double-height | 101.6 dpi | 203.2 dpi |
| 3, 51 | Quadruple | 101.6 dpi | 101.6 dpi |

[dpi: dots per inch (25.4 mm)]

- xL, xH specify the number of bytes in the horizontal direction as (xL + xH × 256).
- yL, yH specify the number of dots in the vertical direction as (yL + yH × 256).

**GS w n**

[Name] Set bar code width

[Format]

| | | | |
|---|---|---|---|
| ASCII | GS | w | n |
| Hex | 1D | 77 | n |
| Decimal | 29 | 119 | n |

[Range] 2 ≤n≤ 6

[Description] Sets the horizontal size of the bar code.

n specifies the bar code width as follows:

| n | Module width (mm) for Multi-level bar code | Binary-level bar code | |
|---|---|---|---|
| | | Thin element width (mm) | Thick element width (mm) |
| 2 | 0.250 | 0.250 | 0.625 |
| 3 | 0.375 | 0.375 | 1.000 |
| 4 | 0.560 | 0.500 | 1.250 |
| 5 | 0.625 | 0.625 | 1.625 |
| 6 | 0.750 | 0.750 | 2.000 |

- Multi-level bar codes are as follows:

  UPC-A, UPC-E, JAN13, JAN8, CODE93, CODE128

- Binary-level bar codes are as follows:

  CODE39, ITF, CODABAR

[Default] n = 3

[Reference] **GS k**

## Multi-byte Code Character Control Commands

## (Supported by Japanese/Simple Chinese/Traditional Chinese/Korean models)

**FS ! n**

[Name] Set print mode(s) for Kanji characters

[Format]

| | | | |
|---|---|---|---|
| ASCII | FS | ! | n |
| Hex | 1C | 21 | n |
| Decimal | 28 | 33 | n |

[Range] 0 ≤ n ≤ 255

[Description] Sets the print mode (Double-height, double-width, Kanji underline) for multi-byte code characters, using n as follows:

| Bit | Off/On | Hex | Decimal | Function |
|---|---|---|---|---|
| 0, 1 | - | 00 | 0 | Reserved |
| 2 | Off | 00 | 0 | Double-width mode is OFF. |
| | On | 04 | 4 | Double-width mode is ON. |
| 3 | Off | 00 | 0 | Double-height mode is OFF. |
| | On | 08 | 8 | Double-height mode is ON. |
| 4-6 | - | 00 | 0 | Reserved |
| 7 | Off | 00 | 0 | Underline mode for Kanji characters is OFF. |
| | On | 08 | 128 | Underline mode for Kanji characters is ON. |

**FS &**

[Name] Select Kanji character mode

[Format]

| | | | |
|---|---|---|---|
| ASCII | FS | & | |
| Hex | 1C | 26 | |
| Decimal | 28 | 38 | |

[Description] Selects Kanji character mode.

**FS - n**

[Name] Turn underline mode on/off for Kanji characters

[Format]

| | | | |
|---|---|---|---|
| ASCII | FS | - | n |
| Hex | 1C | 2D | n |
| Decimal | 28 | 45 | n |

[Range] 0 ≤n≤ 2, 48 ≤n≤ 50

[Default] n = 0

[Description] Turns underline mode for Kanji characters on or off, based on the following values of n for both receipt and slip.

| n | Function |
|---|---|
| 0, 48 | Turns off underline mode for Kanji characters |
| 1, 49 | Turns on underline mode for Kanji characters (1-dot thick) |
| 2, 50 | Turns on underline mode for Kanji characters (2-dot thick) |

**FS .**

[Name] Cancel Kanji character mode

[Format]

| | | | |
|---|---|---|---|
| ASCII | FS | . |
| Hex | 1C | 2E |
| Decimal | 28 | 46 |

[Description] Cancels Kanji character mode.

**FS 2 c1 c2 d1...dk**

[Name] Define user-defined Kanji characters

[Format]

| | | | | | |
|---|---|---|---|---|---|
| ASCII | FS | 2 | c1 | c2 | d1...dk |
| Hex | 1C | 32 | c1 | c2 | d1...dk |
| Decimal | 28 | 50 | c1 | c2 | d1...dk |

[Range] c1 and c2 indicate character codes for the defined characters. The range of values for c1 and c2 differ depending on the character code system used.

| Model type | c1 | c2 |
|---|---|---|
| Japanese model (JIS code system) | c1 = 77 | 21 ≤c2≤7E |
| Japanese model (SHIFT-JIS code system) | c1 = EC | 40 ≤ c2≤ 7E<br>80 ≤c2≤ 9E |
| Simplified Chinese model (GB18030)<br>Traditional Chinese model | c1 = FE | A1 ≤c2≤ FE |

0 ≤d≤ 255

k = 72

[Description] Defines user-defined Kanji characters (di…dk) for the character codes specified by c1 and c2.

**FS S n1 n2**

[Name] Set left- and right Kanji character spacing

[Format]

| | | | | |
|---|---|---|---|---|
| ASCII | FS | S | n1 | n2 |
| Hex | 1C | 53 | n1 | n2 |
| Decimal | 28 | 83 | n1 | n2 |

[Range] 0 ≤n1≤ 255

0 ≤n2≤ 255

[Default] n1 = 0, n2 = 0

[Description] Sets left and right Kanji character spacing to n1 and n2, respectively.

- The left-side character spacing is [n1 × 0.125 mm], and the right-side character spacing is [n2 × 0.125 mm].

[Note] Settable maximum left and right space is both 31.91 mm {255/203"}.

**FS W n**

[Name] Turn on/off quadruple-size mode for Kanji characters

[Format]

| | | | |
|---|---|---|---|
| ASCII | FS | W | n |
| Hex | 1C | 57 | n |
| Decimal | 28 | 87 | n |

[Range] 0 ≤n≤ 255

[Default] n = 0

[Description]

- Turn on or off quadruple-size mode for Kanji characters.
- When the LSB of n is 0, quadruple-size mode for Kanji characters is turned off.
- When the LSB of n is 1, quadruple-size mode for Kanji characters is turned on.

# Revision History

## Document Revision History

| Document Title: FM3, Thermal Printer Development Kit 3 Inch User Manual<br>Document Number:002-04440 | | | |
|---|---|---|---|
| Revision | Issue Date | Origin of Change | Description of Change |
| ** | 09/18/2014 | RLIU | Initial Release |
| *A | 02/16/2016 | RLIU | Migrated Spansion guide “AN706-00093-1v0-E” to Cypress format |