Xiamen Rongta

Linux SDK

Reference Guide

Xiamen Rongta Technology Co.,Ltd.

June 2,2017

目录

1. About This Guide

This document applies only to the Linux SDK developed by Xiamen Rongta.This article describes the constants and interfaces used in Linux SDK.The SDK applies only to the large 58mm and 88mm thermal printing Rongta.Including,but not limited to RP58,RPP200 etc.

2. Contant Definition

Linux SDK Contants Definition：

2.1 EALIGN

Select justification:

E_ALIGN_LEFT	Left justification
E_ALIGN_CENTER	Centering
E_ALIGN_RIGHT	Right justification

2.2 ECHARACTER

Select an international character set:

E_CHARACTER_USA	U.S.A.
E_CHARACTER_FRANCE	France
E_CHARACTER_GERMANY	Germany
E_CHARACTER_BRITAIN	U.K.
E_CHARACTER_DENMARK_I	Denmark I
E_CHARACTER_SWEDEN	Sweden
E_CHARACTER_ITALY	Italy
E_CHARACTER_SPAIN_I	Spain I
E_CHARACTER_JAPAN	Japan
E_CHARACTER_NORWAY	Norway
E_CHARACTER_DENMARK_II	Denmark II
E_CHARACTER_SPAIN_II	Spain II
E_CHARACTER_LATIN_AMERICA	Latin America
E_CHARACTER_KOREA	Korea
E_CHARACTER_SLOVENIA	Slovenia/Croatia
E_CHARACTER_CHINA	China

2.3 EUNDERLINE

Turn underline mode on/off:

E_UNDERLINE_CLOSE	Turns off underline mode.
E_UNDERLINE_I	Turns on underline mode, set at 1-dot width.
E_UNDERLINE_II	Turns on underline mode, set at 2-dot width.

2.4 ECHARACTERFORMAT

Select print mode(s):

E_FONT_A	Character font A (12 $\times$ 24) selected.
E_FONT_B	Character font B (9 × 17) selected.

E_CHAR_BOLD	Reserved.
E_CHAR_TIMES_HIGH	Emphasized mode is turned off.
E_CHAR_DOUBLE_WIDTH	Emphasized mode is turned on.
E_CHAR_UNDERLINE	Double-height canceled.

2.5 ECODEPAGE

Select character code table:

E_CODE_PAGE_CP437	[PC437 (USA: Standard Europe)]
E_CODE_PAGE_KATAKANA	[Katakana]
E_CODE_PAGE_CP850	[PC850 (Multilingual)]
E_CODE_PAGE_CP860	[PC860 (Portuguese)]
E_CODE_PAGE_CP863	[PC863 (Canadian-French)
E_CODE_PAGE_CP865	[PC865 (Nordic)]
E_CODE_PAGE_WCP1252	[WPC1252]
E_CODE_PAGE_WCP1253	[WPC1253]
E_CODE_PAGE_CP852	[PC852 (Latin 2)]
E_CODE_PAGE_CP858	[PC858 (Euro)]

2.6 ECHARFONT

Select character font:

E_CHAR_FONT_A	Character font A (12 · 24)
E_CHAR_FONT_B	Character font B (9 · 17)

2.7 ECHARHIGH

Select character hight:

E_CHAR_HIGH_1	1 time (standard)
E_CHAR_HIGH_2	2 times
E_CHAR_HIGH_3	3 times
E_CHAR_HIGH_4	4 times
E_CHAR_HIGH_5	5 times
E_CHAR_HIGH_6	6 times
E_CHAR_HIGH_7	7 times
E_CHAR_HIGH_8	8 times

2.8 ECHARWIDE

Select character width:

E_CHAR_WIDE_1	1 time (standard)
E_CHAR_WIDE_2	2 times
E_CHAR_WIDE_3	3 times
E_CHAR_WIDE_4	4 times
E_CHAR_WIDE_5	5 times
E_CHAR_WIDE_6	6 times
E_CHAR_WIDE_7	7 times

E_CHAR_WIDE_8	8 times

2.9 EHRIPRINTPOSITION

Select print position of HRI characters:

E_HRI_PRINT_COLSE	Not printed.
E_HRI_PRINT_UPPER	Above the bar code.
E_HRI_PRINT_BELOW	Below the bar code.
E_HRI_PRINT_UPPER_AND_BELOW	Both above and below the bar code.

2.10 EBARTYPE

Bar code system:

E_BAR_TYPE_UPC_A	UPC-A
E_BAR_TYPE_UPC_E	UPC-E
E_BAR_TYPE_JAN13	JAN13
E_BAR_TYPE_JAN8	JAN 8
E_BAR_TYPE_CODE39	CODE39
E_BAR_TYPE_ITF	ITF
E_BAR_TYPE_CODABAR	CODABAR
E_BAR_TYPE_CODE93	CODE93
E_BAR_TYPE_CODE128	CODE128

2.11 E2DBARSIZE

2DBar Size:

E_BAR_SIZE_AUTO	auto size
E_BAR_SIZE_1	size 1
E_BAR_SIZE_2	size 2
E_BAR_SIZE_3	size 3
E_BAR_SIZE_4	size 4
E_BAR_SIZE_5	size 5
E_BAR_SIZE_6	size 6
E_BAR_SIZE_7	size 7
E_BAR_SIZE_8	size 8
E_BAR_SIZE_9	size 9
E_BAR_SIZE_10	size 10

2.12 E2DBARCHECKLEVEL

2DBar EC level:

E_2DBAR_Check_Level_L	L
E_2DBAR_Check_Level_M	M
E_2DBAR_Check_Level_Q	Q
E_2DBAR_Check_Level_H	H

2.13 E2DBARCOMPONENT

2DBar component:

E_BAR_Component_1	Component 1
E_BAR_Component_2	Component 2
E_BAR_Component_3	Component 3
E_BAR_Component_4	Component 4
E_BAR_Component_5	Component 5
E_BAR_Component_6	Component 6
E_BAR_Component_7	Component 7
E_BAR_Component_8	Component 8

2.14   ECUTTINGMODE

Select cut mode and cut paper:

E_CUTTING	Cuts paper.
E_FEED_CUTTING	Feeds paper to and cuts the paper.

2.15   ECHINESEFORMAT

Select chinese format:

E_CHINESE_FORMAT_GBK	GBK
E_CHINESE_FORMAT_UTF_8	UTF-8
E_CHINESE_FORMAT_BIG5	BIG5

2.16   EPANELBUTTONS

Enable/Disable panel buttons:

E_PANEL_BUTTONS_ACTIVATION	Enabled
E_PANEL_BUTTONS_PROHIBIT	Disable

2.17   EPRINTERINFORMATION

Printer information:

E_PRINTER_INFORMATION_ID_MODEL	Printer model ID
E_PRINTER_INFORMATION_ID_NUMBER	Type ID
E_PRINTER_INFORMATION_HARDWARE_VERSION	Firmware version
E_PRINTER_INFORMATION_MANUFACTURER	Manufacturer
E_PRINTER_INFORMATION_PRINTER_NAME	Printer name
E_PRINTER_INFORMATION_COM_NAME	Serial number
E_PRINTER_INFORMATION_CHARACTER	Type of mounted additional fonts

2.18   ECUTTERBOXPIN

Cutter-Box pin:

E_CUTTER_BOX_PIN2	pin 2
E_CUTTER_BOX_PIN5	pin 5

2.19   EPRINTBITMAP

Prints a raster bit image using the mode:

E_PRINT_BITMAP_COMMON_MODE	Normal
E_PRINT_BITMAP_DOUBLE_WIDTH	Double-width
E_PRINT_BITMAP_TIMES_HIGHER	Double-height
E_PRINT_BITMAP_FOURFOLD	Quadruple

2.20 EPRINTDIRECTION

Select print direction in page mode:

E_PRINT_DIRECTION_LEFT_TO_RIGHT	Left to right
E_PRINT_DIRECTION_BOTTOM_TO_TOP	Bottom to top
E_PRINT_DIRECTION_RIGHT_TO_LEFT	Right to left
E_PRINT_DIRECTION_TOP_TO_BOTTOM	Top to bottom

2.21 ETESTPRINT

Execute test print mode:

E_TEST_PRINT_DUMP	Hexadecimal dump print
E_TEST_PRINT_PRINTER_STATE	Printer status print
E_TEST_PRINT_ROLL_OF_PAPER	Rolling pattern print

3. Function Interface Definition

3.1 print_text(char * str)

Description: The string str to be printed is sent to the print buffer.

3.2 set_config_state(p_config_state pconfig)

Description: Configure port information.

p_config_state is struct pointer. The struct is as follows：

```
typedef struct
{
    E_CONFIG_STATE eState;        ----Port：COM，LPT，ENET，USB，Printer Driver
    char name[MAX_PATH];          ----Port Name
    union
    {
        p_enet_info net_info;     ----ENET Information
        int baudrate;             ----Baud Rate
    }context;
}config_state,*p_config_state;
```

3.3 LF()

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

3.4 CR()

Description: Prints the data in the print buffer and carriage return.

3.5 HT()
Description: Moves the print position to the next horizontal tab position.

3.6 set_feed_line(unsigned int line)
Description: Prints the data in the print buffer and feeds the paper [line×0.125mm)]. (0 < line < 256)

3.7 set_feed_points(unsigned int points)
Description: Prints the data in the print buffer and feeds the paper [points×0.125mm]. (0 < points< 256)

3.8 set_level_location(unsigned int loation)
Description: Sets horizontal tab positions.(0 < loation< 256)

3.9 set_printer_status(bool bset)
Description:Set printer is Enables/Disable.

3.10   self_test()
Description: Print self test page.

3.11   set_line_spacing(unsigned int spacing)
Description: Sets the line spacing to [spacing×0.125mm]. (0 < spacing< 255)

3.12   set_blank_amount(unsigned int amount)
Description: In standard mode, sets the left margin to [amount×0.125mm]. (0 < amount< 65535)

3.13   set_print_position(unsigned int position)
Description: Moves the print position to [position×0.125mm] from the current position. (0<position< 65535)

3.14   set_print_area(unsigned int area)
Description: In standard mode, sets the print area width to [area×0.125mm]. (0 < area < 65535)

3.15   set_align_mode(EALIGN align)
Description: In standard mode, Set aligns mode.

3.16   set_absolute_print_position(unsigned int position)
Description: Moves the print position to [position×0.125mm] from the left edge of the print area. (0 < position< 65535)

3.17   defalt_line_spacing()

Description: Sets the line spacing to approximately 4.23 mm {30/180"}.

3.18   set_character_format(unsigned int mode)
Description: Selects the character font and styles (emphasized, double-height, double-width, and underlined) together.

3.19   set_inverse_mode(bool bset)
Description: Turns white/black reverse print mode on or off.

3.20   set_rotate_mode(bool bset)
Description: In standard mode, turns 90° clockwise rotation mode on or off for characters.

3.21   select_chinese_mode(bool bset)
Description: Chinese mode on or off.

3.22   set_overlap(bool bset)
Description: Turns double-strike mode on or off.

3.23   set_bold(bool bset)
Description: Turns emphasized mode on or off.

3.24   select_user_character(bool bset)
Description: Selects or cancels the user-defined character set.

3.25   set_resupinate_print(bool bset)
Description: In standard mode, turns upside-down print mode on or off.

3.26   set_charact(ECHARACTER character)
Description: Select an international character set.

3.27   set_under_line(EUNDERLINE underline)
Description: Turns underline mode on or off.

3.28   set_code_page(ECODEPAGE codepage)
Description: Select character code table.

3.29   set_font(ECHARFONT font)
Description: Selects a character font.

3.30   set_char_size(p_char_size pcharsize)
Description: Selects character size (height magnification and width magnification). p_char_size is struct pointer. The struct is as follows：
typedef struct _t_char_size
{

    ECHARHIGH high;   ----character height
    ECHARWIDE wide;   ----character width
}char_size,*p_char_size;

3.31   set_right_char_spacing(unsigned int spacing)

Description: Sets the right-side character spacing to [spacing×0.125mm].(0 < spacing< 256)

3.32   cancel_user_char(unsigned int cancel)

Description: Deletes the user-defined character pattern specified by character code cancel. (32≤cancel≤126)

3.33   set_chinese_spacing(unsigned int left,unsigned int right)

Description: Sets the left-side character spacing of the multi-byte code character to [left× 0.125mm]; sets the right-side character spacing of the multi-byte code character to [right× 0.125]. (0 < left < 256 , 0 < right < 256)

3.34   hri_print_position(EHRIPRINTPOSITION eposition)

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

3.35   set_bar_width(unsigned int wide)

Description: Sets the horizontal size of the bar code.(2≤wide≤6)

3.36   set_hri_font(ECHARFONT font)

Description: Selects a font for the HRI characters when printing a bar code.

3.37   set_barcode_height(unsigned int high)

Description: Sets the height of the bar code to high dots. (1≤high≤255);

3.38   set_bar_left_spacing(unsigned int spacing)

Description: Sets the left-side spacing of the bar code.(1≤spacing≤255);

3.39   print_bar(p_bar_print pbarprint)

Description: Print bar code.

p_bar_print is struct pointer. The struct is as follows：

typedef struct _t_bar_print

{

    EBARTYPE type;   ----bar code type
    char *data;   ----bar code data

}bar_print,*p_bar_print;

The bar code data length is different according to the bar code type:

Bar Code Type	Length
UPC-A	11 digit characters
UPC-E	11 digit characters

JAN13	12 characters
JAN 8	7 characters
CODE39	1~30 characters
TIF	2~30 even number characters
CODEBAR	1~30 characters
CODE93	1~30 characters
CODE128	1~30 characters

3.40   print_2dbar(p_t2dbar_print pt2dbar)

Description: Print 2DBar code.

p_t2dbar_print is struct pointer. The struct is as follows：

typedef struct _t_2dbar_print

{

    E2DBARSIZE size;   ----2DBar code size

    E2DBARCHECKLEVEL level;   ----2DBar code EC level

    E2DBARCOMPONENT component;   ----2DBar code component

    char *data;   ----2DBar code data

}t2dbar_print,*p_t2dbar_print;

3.41   select_cutting_mode(p_cutting_mode pcutter)

Description: Select cut mode and cut paper.

p_cutting_mode is struct pointer. The struct is as follows：

typedef struct _t_cutting_mode

{

    ECUTTINGMODE mode;   ----cut mode

    unsigned int line;   ----Feed (0 < line < 255)

}cutting_mode,*p_cutting_mode;

3.42   buzzer_set(p_buzzer pbuzzerset)

Description: Set times of the buzzer and time of the buzzer.

p_buzzer is struct pointer. The struct is as follows：

typedef struct _t_buzzer_set

{

    unsigned int times;   ----Times of buzzer(1<times<9)

    unsigned int time;   ----Time of buzzer(1<times<9)

}buzzer,*p_buzzer;

3.43   set_chinese_format(ECHINESEFORMAT format)

Description: Select Chinese code format.

3.44   set_panel_button(EPANELBUTTONS set)

Description: Enables or disables the panel buttons.

3.45   full_cut()
Description: Executes a partial cut of the roll paper.

3.46   half_cut()
Description: Executes a partial cut of the roll paper.

3.47   printer_state(unsigned char *receive)
Description: Get the cutter-box status information.

3.48   paper_state(unsigned char *receive)
Description: Get the printing paper status information.

3.49   off_line_state(unsigned char *receive)
Description: Get the off line status information.

3.50   error_state(unsigned char *receive)
Description: Get the error status information.

3.51   print_paper_state(unsigned char *receive)
Description: This printer transmits the paper sensor status.

3.52   get_printer_information(p_printer_information pinfo)
Description: Get the printer information.
p_printer_information is struct pointer. The struct is as follows：
typedef struct _t_printer_information
{
       EPRINTERINFORMATION information;   ----Printer information type
       unsigned char receive[PRINTER_INFO_LEN];   ----Printer information
}printer_information,*p_printer_information;

3.53   error_recovery()
Description: Recovers from a recoverable error and restarts printing from the line where
the error occurred.

3.54   clear_buffer()
Description: Recovers from a recoverable error after clearing the receive and print
buffers.

3.55   printer_initialization()
Description: Clears the data in the print buffer and resets the printer modes to the modes
that were in effect when the power was turned on.

3.56   send_pulse(p_pulse_level_set plevelset)
Description: Sets the pulse level time and pulses to the specified pin.

p_pulse_level_set is struct pointer. The struct is as follows：

typedef struct _t_pulse_level_set

{

    ECUTTERBOXPIN pin;   ----cutter-box pin

    unsigned int time;   ----pulse level time

}pulse_level_set,*p_pulse_level_set;

3.57　　send_pulse_time(p_pulse_time_set ptimeset)

Description: Sets the boottime and the off time,pulses to the specified pin.

p_pulse_time_set is struct pointer. The struct is as follows：

typedef struct _t_pulse_time_set

{

    ECUTTERBOXPIN pin;   ---- cutter-box pin

    unsigned int boottime;   ----boottime

    unsigned int offtime;   ----offtime

}pulse_time_set,*p_pulse_time_set;

3.58　　page_mode()

Description: Switches from standard mode to page mode.

3.59　　standard_mode()

Description: Switches from page mode to standard mode.

3.60　　page_mode_print()

Description: Print data in page mode.

3.61　　print_and_standard()

Description: Print and return to standard mode (in page mode).

3.62　　print_area(p_page_print_area parea)

Description: Set print area in page mode.

3.63　　absolute_vertical_position(unsigned int position)

Description: Set absolute vertical print position in page mode .(0 < position < 65535)

3.64　　relative_vertical_position(unsigned int position)

Description: Set relative vertical print position in page mode .(0 < position < 65535)

3.65　　print_direction(EPRINTDIRECTION direction)

Description: In page mode, selects the print direction and starting position.

3.66　　fourfold_chinese(bool bset)

Description: Turns quadruple-size mode on or off for multi-byte code character.

3.67   test_print(ETESTPRINT print)

Description: Execute test print.

3.68   print_raster_bmp(char * filename)

Description: Print raster bit image.filename is bitmap file path.

3.69   download_bitmap(char *bitmap_name[],int count)

Description: Download NV bit image. bitmap_name is the bitmap file path to download . count is the number of bitmaps to download.

3.70   print_nv_bitmap(int index)

Description: Print NV bit image.

4. Linux SDK Using Steps

The specific steps for using Linux SDK in Linux are as follows:

First steps：Open port access;Specific instructions are as follows:

COM: sudo chmod a+rw /dev/ttyS0

USB: sudo chmod a+rw /dev/usb/lp0

ENET: The network port needs to be configured with the correct IP address.

Second steps：Using the SDK interface to compile .c files,such as test.c.

Third steps：Using instructions in 32-bit system:gcc –o test test.c ./Linux58_80SDK.so

Using instructions in 64-bit system:gcc –o test test.c ./Linux58_80SDK_x64.so

Fourth steps：Using instructions:./test

Specific use of the SDK interface,please refer to SDK Example.