

1 ezLCD Commands

The instructions may be fed to ezLCD through the Evaluation Board's RS232 and USB connectors.

The USB-Client port in the USB-RS232 bridge configuration.

Upon installation of the driver, an Operating System of a Personal Computer treats USB port of ezLCD as an additional COM port.

The default parameters of the RS232 and USB are:

Baudrate: 115200 bps

No of Stop Bits: 1

Parity: Off

Handshake: None

The new USB Drivers and software may be downloaded from [USB Drivers & Software](#)

General

CLS

LIGHT ON

LIGHT OFF

SET COLOR

SET XY

Points

PLOT

PLOT XY

Lines

H LINE

V LINE

LINE TO XY

Figures

ARC

CIRCLE R

CIRCLE R FILL

BOX

BOX FILL

Bitmaps

PUT BITMAP

PUT ICON

PICTURE

Text and Fonts

SELECT FONT

SET BG COLOR

TEXT NORTH

TEXT EAST

TEXT SOUTH

TEXT WEST

PRINT CHAR

PRINT CHAR BG

PRINT STRING

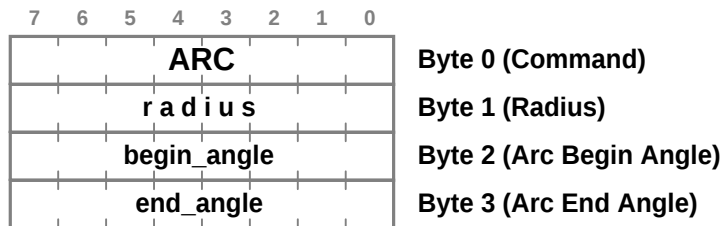
PRINT STRING BG

1.1 ARC

Description: Draws an Arc in Current Color, with the center at Current Position, starting on Begin Angle and ending on the End Angle.

Class: Multi Byte Command

Code: 2Fhex, 47dec, / ASCII



See Also: [SET_XY](#), [SET_COLOR](#), [CIRCLE_R](#)

Angle Coding: The angle range is from 0 to 255.
 To transform degrees to ARC angle units:
 $\text{Angle_lcd} = \text{Angle_deg} \times 32 / 45$

For example:

32 = 45°

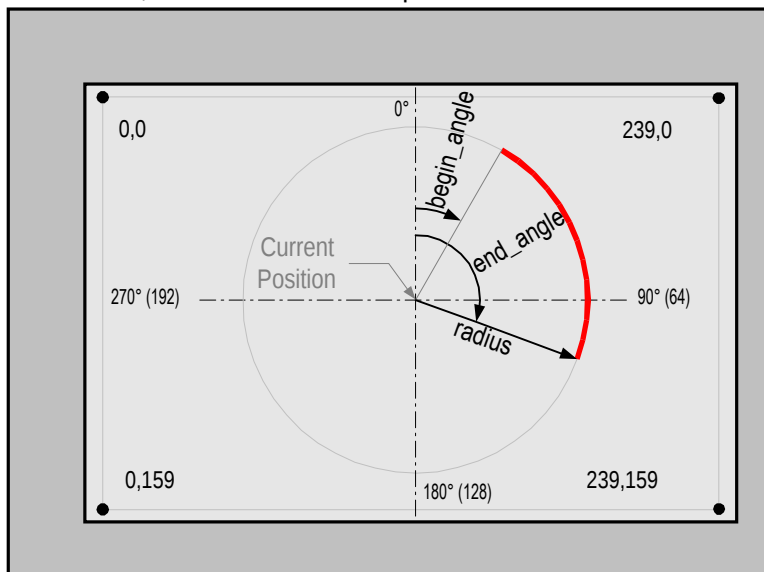
64 = 90°

128 = 180°

192 = 270°

0 = 0° = 360°

The angle is drawn clockwise with the zero positioned at the top of a screen, as it is shown on the picture below



Example:

The following sequence will draw a green arc from 45 to 225 degrees with the center positioned in the middle of a screen.

SET_COLOR 24 hex

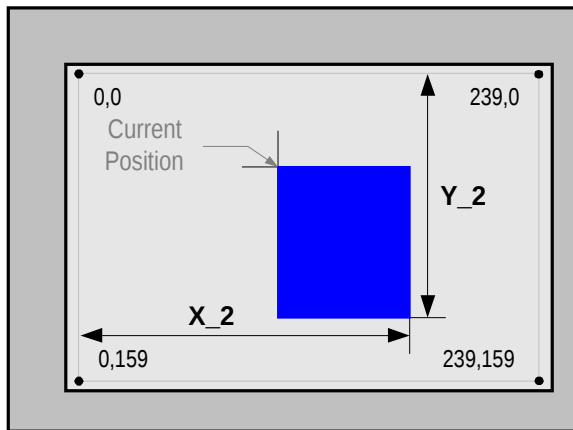
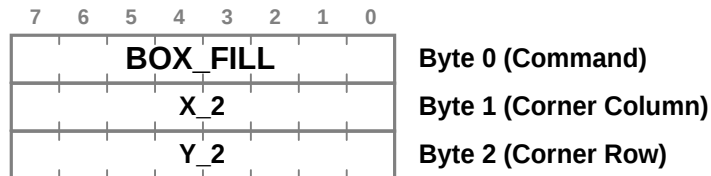
GREEN	00111000	bin
SET_XY	25	hex
120	120	dec
80	80	dec
ARC	2F	hex
60	60	dec (radius)
32	32	dec (begin_angle = 45 degrees)
160	160	dec (end_angle = 225 degrees)

1.3 BOX_FILL

Description: Draws a rectangle filled with Current Color

Class: Multi Byte Command

Code: 43hex, 67dec, C ASCII



See Also: [SET_XY](#), [BOX](#)

Example:

The following sequence will draw the rectangle filled with blue color

SET_COLOR 24 hex

RED 11000000 bin

SET_XY 25 hex

95 95 dec

40 10 dec

BOX_FILL 43 hex

180 180 dec (X_2)

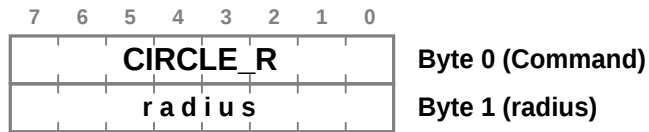
120 120 dec (Y_2)

1.4 CIRCLE_R

Description: Draws a circle in Current Color at Current Position

Class: Double Byte Command

Code: 29hex, 41dec,) ASCII



See Also: [SET_XY](#), [SET_COLOR](#)

Example:

The following sequence will draw a green circle in the middle of the screen.

```
SET_COLOR  24 hex
GREEN      00111000 bin
SET_XY     25 hex
120        120 dec
80          80 dec
CIRCLE_R   29 hex
60         60 dec
```

1.5 CIRCLE_R_FILL

Description: Draws a circle in Current Color at Current Position, filled with Current Color
Class: Double Byte Command
Code: 39hex, 57dec, 9 ASCII



See Also: [SET_XY](#), [SET_COLOR](#)

Example:

The following sequence will draw a red filled circle in the middle of the screen.

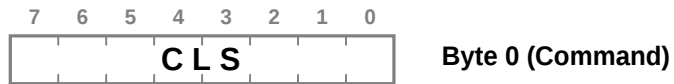
```
\SET_COLOR      24 hex
RED             00000111 bin
SET_XY          25 hex
120             120 dec
80              80 dec
CIRCLE_R_FILL   39 hex
60              60 dec
```

1.6 CLS

Description: Clears screen by filling it with the Current Color

Class: Single Byte Command

Code: 21hex, 33dec, ! ASCII



See Also: [SET_COLOR](#)

Example:

The following sequence will clear the screen

SET_COLOR 24 hex

WHITE 11111111 bin

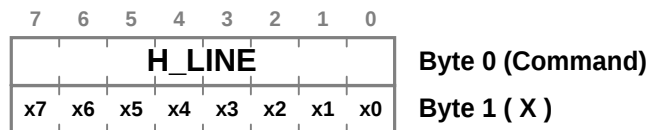
CLS 21 hex

1.7 H_LINE

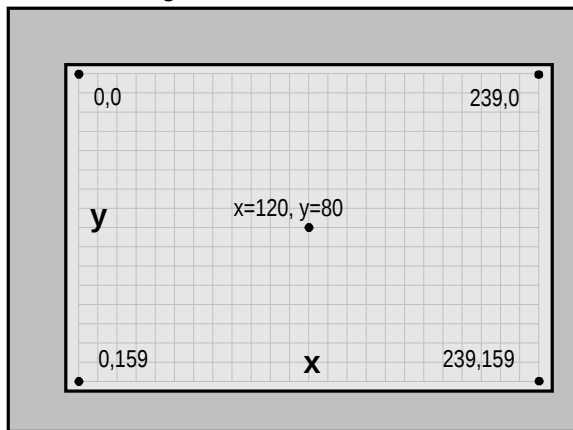
Description: Fast draws a horizontal line from Current Position, to the column specified by the parameter.

Class: Double Byte Command

Code: 40hex, 64dec, @ ASCII



Note: The screen size is 240x160. However, the valid X range is 0 - 255



See Also: [V_LINE](#), [SET_XY](#)

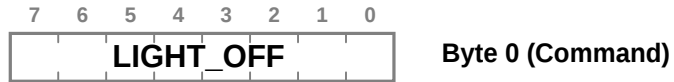
Example:

The following sequence will draw the horizontal green line from (20, 60) to (170, 60)

```
SET_COLOR  24 hex
GREEN      00111000 bin
SET_XY     25 hex
20         20 dec
60         60 dec
H_LINE     40 hex
170        170 dec
```

1.8 LIGHT_OFF

Description: Turns off the screen light
Class: Single Byte Command
Code: 23hex, 35dec, # ASCII



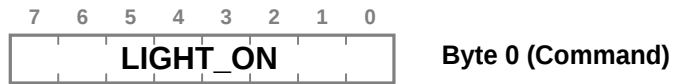
See Also: [LIGHT_ON](#)

Example:

The following sequence will turn off the screen light
LIGHT_OFF 23 hex

1.9 LIGHT_ON

Description: Turns on the screen light
Class: Single Byte Command
Code: 22hex, 34dec, " ASCII



See Also: [LIGHT_OFF](#)

Example:

The following sequence will turn on the screen light
LIGHT_ON 22 hex

1.10 LINE_TO_XY

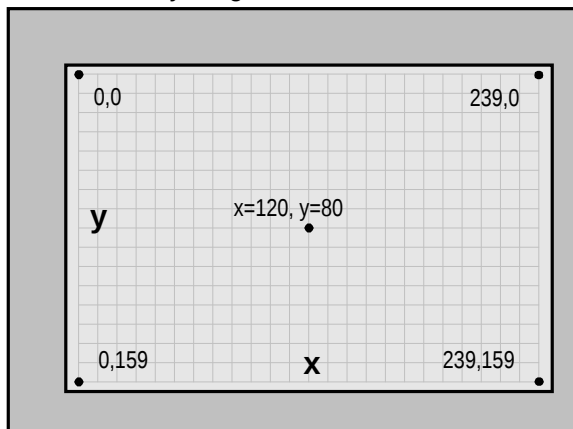
Description: Draws a line in Current Color, from the Current Position to the to specified position

Class: Multi Byte Command

Code: 28hex, 40dec, (ASCII

7	6	5	4	3	2	1	0	
LINE_TO_XY								Byte 0 (Command)
x7	x6	x5	x4	x3	x2	x1	x0	Byte 1 (x)
y7	y6	y5	y4	y3	y2	y1	y0	Byte 2 (y)

Note: The screen size is 240x160. However, the valid x and y ranges are 0 - 255



See Also: [SET_XY](#), [SET_COLOR](#), [PLOT](#)

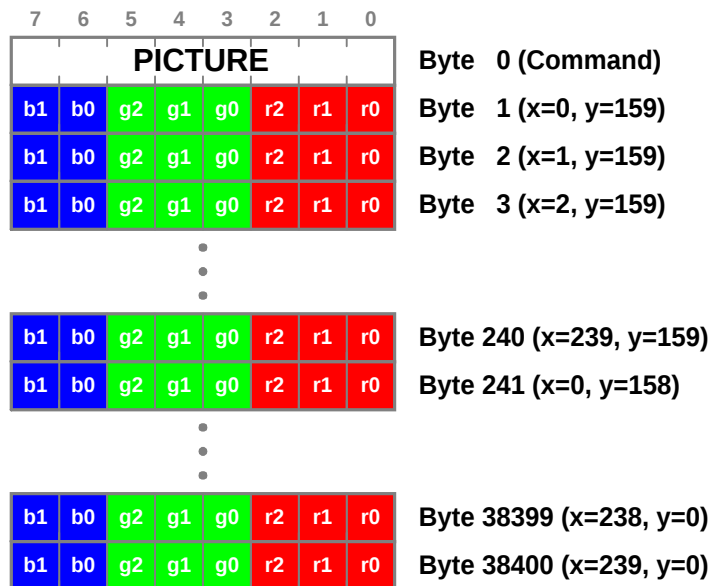
Example:

The following sequence will draw a red line across the screen.

```
SET_COLOR    24 hex
RED          00000111 bin
SET_XY       25 hex
0            0 dec
0            0 dec
LINE_TO_XY   28 hex
239          239 dec
159          159 dec
```

1.11 PICTURE

Description: Puts a bitmap picture over the entire screen
Class: Multi Byte Command
Code: 2Ahex, 42dec, * ASCII



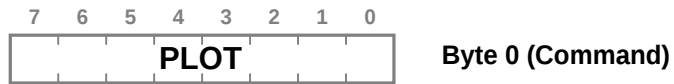
See Also: [SET_XY](#), [SET_COLOR](#), [PUT_BITMAP](#)

1.12 PLOT

Description: Plots a point at Current Position in Current Color

Class: Single Byte Command

Code: 26hex, 38dec, & ASCII



See Also: [SET_XY](#), [SET_COLOR](#)

Example:

The following sequence will put the blue point in the middle of the screen.

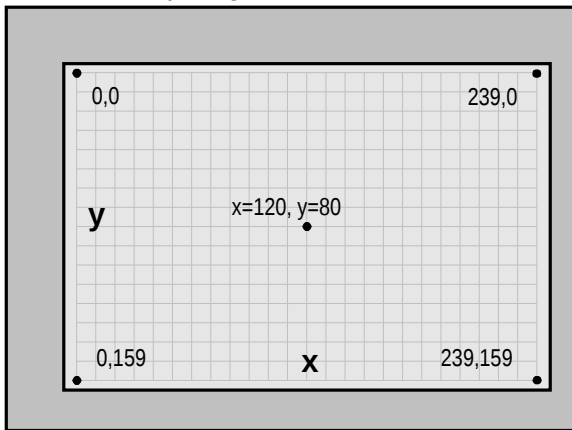
```
SET_COLOR  24 hex
BLUE       11000000 bin
SET_XY     25 hex
120        120 dec
80         80 dec
PLOT       26 hex
```

1.13 PLOT_XY

Description: Plots a point in Current Color, at specified position.
Class: Multi Byte Command
Code: 27hex, 39dec, ' ASCII

7	6	5	4	3	2	1	0	
PLOT_XY								Byte 0 (Command)
x7	x6	x5	x4	x3	x2	x1	x0	Byte 1 (x)
y7	y6	y5	y4	y3	y2	y1	y0	Byte 2 (y)

Note: The screen size is 240x160. However, the valid x and y ranges are 0 - 255



See Also: [SET_XY](#), [SET_COLOR](#), [PLOT](#)

Example:

The following sequence will put the red point in the middle of the screen.

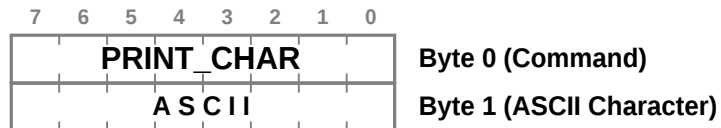
```
SET_COLOR 24 hex
RED 00000111 bin
PLOT_XY 27 hex
120 120 dec
80 80 dec
```

1.14 PRINT_CHAR

Description: Prints a character at Current Position

Class: Double Byte Command

Code: 2Chex, 44dec, , ASCII



See Also: [SELECT_FONT](#), [PRINT_STRING](#)

Example:

The following sequence will print black character 'M' in the middle of the screen, using font number 2

```

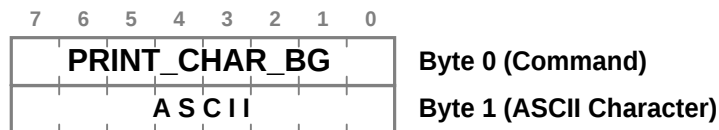
SELECT_FONT  2B hex
2            2 dec
SET_COLOR    24 hex
BLACK        00000000 bin
SET_XY       25 hex
120          120 dec
80           80 dec
PRINT_CHAR   2C hex
'M'          4D hex
  
```

1.15 PRINT_CHAR_BG

Description: Prints a character at Current Position on the background specified by [SET_BG_COLOR](#) command

Class: Double Byte Command

Code: 3Chex, 60dec, < ASCII



See Also: [SELECT_FONT](#), [SET_BG_COLOR](#), [PRINT_STRING_BG](#)

Example:

The following sequence will print white character 'M', on a black background in the middle of the screen, using font number 2

```

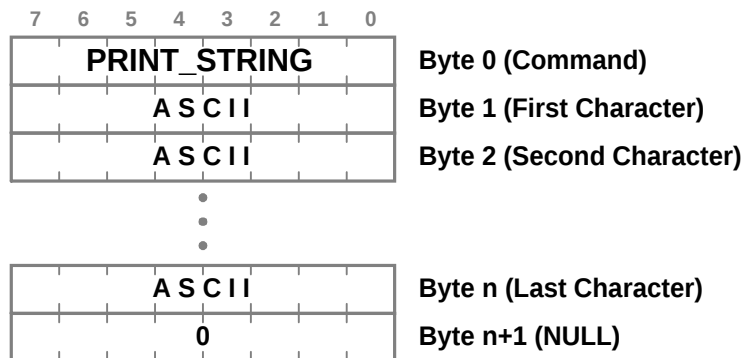
SELECT_FONT  2B hex
2            2 dec
SET_BG_COLOR 34 hex
BLACK        00000000 bin
SET_COLOR    24 hex
WHITE        11111111 bin
SET_XY       25 hex
120          120 dec
80           80 dec
PRINT_CHAR_BG 3C hex
'M'          4D hex
  
```

1.16 PRINT_STRING

Description: Prints null-terminated String starting at Current Position

Class: Multi Byte Command

Code: 2Dhex, 45dec, - ASCII



See Also: [SELECT_FONT](#), [PRINT_CHAR](#)

Example:

The following sequence will print violet sign "LCD" in the middle of the screen, using font number 1

```

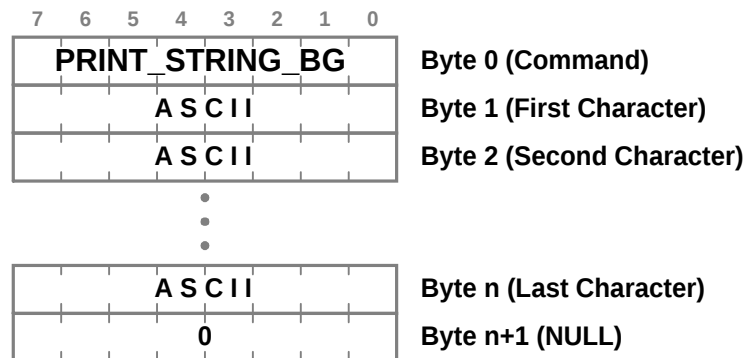
SELECT_FONT  2B hex
1            1 dec
SET_COLOR    24 hex
VIOLET       11000100 bin
SET_XY       25 hex
120          120 dec
80           80 dec
PRINT_STRING 2D hex
'L'          4C hex
'C'          43 hex
'D'          44 hex
NULL         0 hex
  
```

1.17 PRINT_STRING_BG

Description: Prints null-terminated String starting at Current Position on the background specified by [SET_BG_COLOR](#) command

Class: Multi Byte Command

Code: 3Dhex, 61dec, = ASCII



See Also: [SELECT_FONT](#), [SET_BG_COLOR](#), [PRINT_CHAR_BG](#)

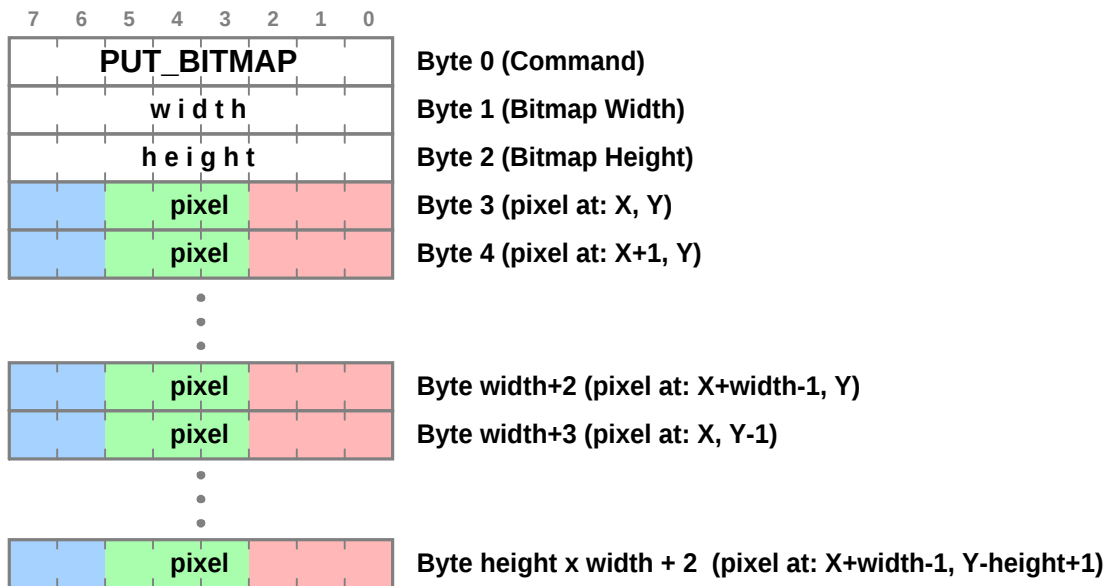
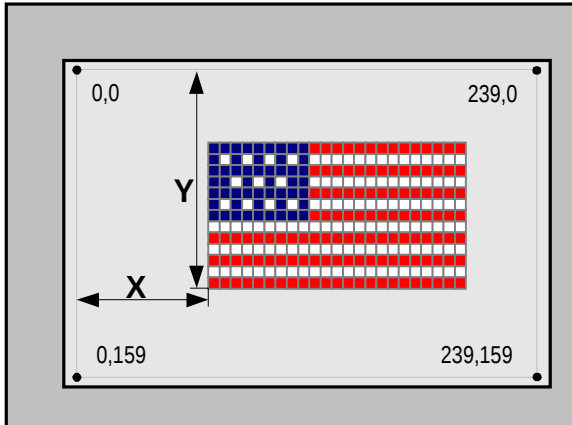
Example:

The following sequence print Yellow "LCD" on the Navy background, in the middle of a screen, using font no 0.

```
SET_BG_COLOR    34 hex
NAVY            10000000 bin
SET_COLOR      24 hex
YELLOW         00111111 bin
SET_XY         25 hex
120            120 dec
80             80 dec
SELECT_FONT    2B hex
0             0 dec
PRINT_STRING_BG 3D hex
'L'           4C hex
'C'           43 hex
'D'           44 hex
NULL          0 hex
```

1.18 PUT_BITMAP

Description: Puts Bitmap on the screen starting at Current Position, then UP and RIGHT
Class: Multi Byte Command
Code: 2Ehex, 46dec, . ASCII



Note: The total number of bytes is: width x height + 3

See Also: [SET_XY](#), [SET_COLOR](#), [PICTURE](#)

Example:

The following sequence will put 4x3 bitmap at x = 60, y = 80

SET_XY	25	hex	
x	60	dec	
y	80	dec	
PUT_BITMAP	2E	hex	-----+--
width	4	dec	
height	3	dec	

```

pixel (x = 60, y = 80)
pixel (x = 61, y = 80)
pixel (x = 62, y = 80)
pixel (x = 63, y = 80)
pixel (x = 60, y = 79)
pixel (x = 61, y = 79)
pixel (x = 62, y = 79)
pixel (x = 63, y = 79)
pixel (x = 60, y = 78)
pixel (x = 61, y = 78)
pixel (x = 62, y = 78)
pixel (x = 63, y = 78)

```

TOTAL:
4 x 3 + 3 = 15 bytes

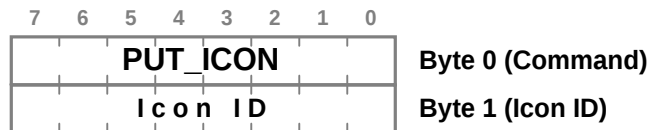
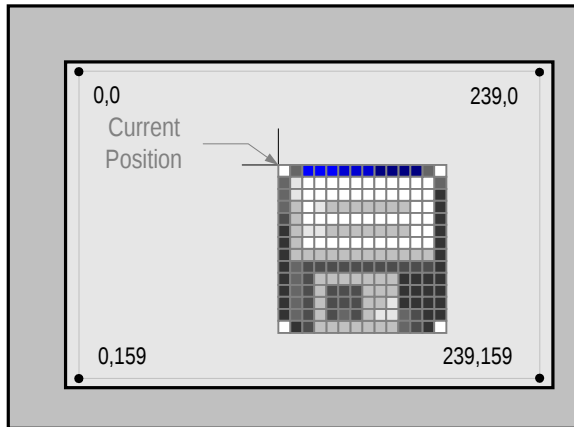
-----+

11	12	13	14
7	8	9	10
3	4	5	6

1.19 PUT_ICON

Description: Displays the icon with it's upper-left corner positioned at the Current Position.
The icon is read from the ezLCD ROM.
Use ezLCDrom.exe utility to store icons in the ezLCD ROM

Class: Double Byte Command
Code: **57**hex, **87**dec, **W** ASCII



See Also: [SET_XY](#)

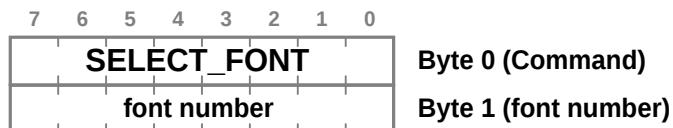
Example:

The following sequence will display an icon no 3 with it's upper-left corner positioned at X = 60, Y = 43

```
SET_XY    25 hex
60        60 dec
43        43 dec
PUT_ICON  57 hex
3         3 dec
```

1.20 SELECT_FONT

Description: Sets the Current Font
Class: Double Byte Command
Code: 2Bhex, 43dec, + ASCII



Note: The following fonts are implemented

Font 0: ezLCD-001

Font 1: ezLCD-001

Font 2: ezLCD-001

Font 3: ezLCD-001

Font 4: ezLCD-001

Font 5: ezLCD-001

See Also: [PRINT_STRING](#), [PRINT_CHAR](#)

Example:

The following sequence will print black character 'M' in the middle of the screen, using font number 2

```
SELECT_FONT  2B hex
2            2 dec
SET_COLOR    24 hex
BLACK        00000000 bin
SET_XY       25 hex
120          120 dec
80           80 dec
PRINT_CHAR   2C hex
'M'          4D hex
```

1.21 SET_BG_COLOR

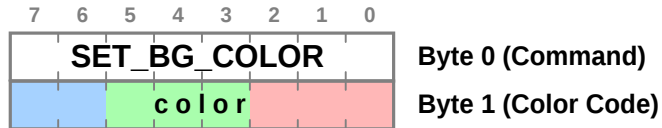
Description: Sets the Background Color for the following instructions:

[PRINT_CHAR_BG](#)

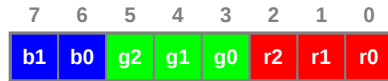
[PRINT_STRING_BG](#)

Class: Double Byte Command

Code: 34hex, 52dec, 4 ASCII



Note: The default NATURAL palette has the following color coding:



See Also: [PRINT_CHAR_BG](#), [PRINT_STRING_BG](#), [PALETTE](#)

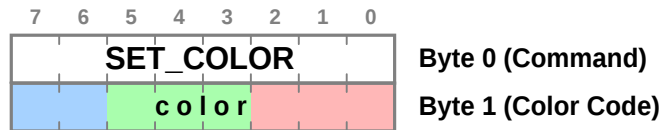
Example:

The following sequence print Yellow "LCD" on the Navy background, in the middle of a screen, using font no 0.

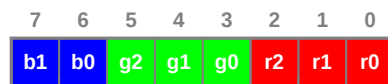
```
SET_BG_COLOR    34 hex
NAVY            10000000 bin
SET_COLOR       24 hex
YELLOW         00111111 bin
SET_XY          25 hex
120            120 dec
80             80 dec
SELECT_FONT     2B hex
0              0 dec
PRINT_STRING_BG 3D hex
'L'            4C hex
'C'            43 hex
'D'            44 hex
NULL           0 hex
```

1.22 SET_COLOR

Description: Sets the Current Color
Class: Double Byte Command
Code: 24hex, 36dec, \$ ASCII



Note: The default NATURAL palette has the following color coding:



See Also: [CLS](#), [PLOT](#), [PALETTE](#)

Example:

The following sequence will fill the whole display with green

```
SET_COLOR 24 hex
GREEN 00111000 bin
CLS 21 hex
```

1.23 SET_XY

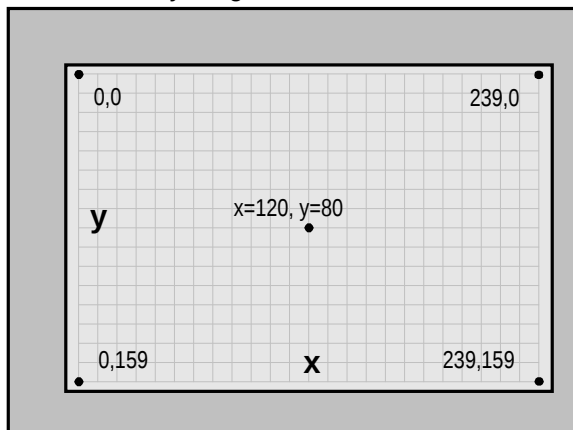
Description: Sets the Current Position

Class: Multi Byte Command

Code: 25hex, 37dec, % ASCII

7	6	5	4	3	2	1	0	
SET_XY								Byte 0 (Command)
x7	x6	x5	x4	x3	x2	x1	x0	Byte 1 (x)
y7	y6	y5	y4	y3	y2	y1	y0	Byte 2 (y)

Note: The screen size is 240x160. However, the valid x and y ranges are 0 - 255



See Also: [PLOT](#), [LINE TO XY](#), [CIRCLE R](#)

Example:

The following sequence will put the blue point in the middle of the screen.

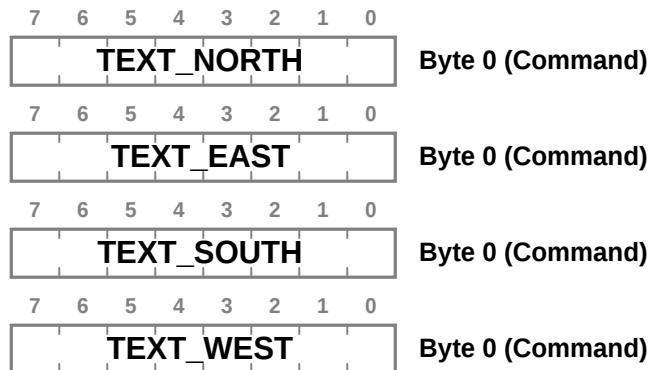
```
SET_COLOR 24 hex
BLUE      11000000 bin
SET_XY    25 hex
120       120 dec
80        80 dec
PLOT      26 hex
```

1.24 TEXT_EAST

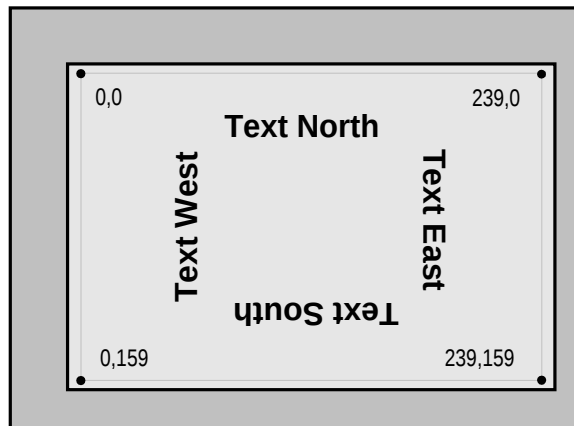
Description: Set the orientation of the text, as shown on the picture below

Class: Single Byte Commands

Code:
TEXT_NORTH : 60hex, 96dec, ' ASCII
TEXT_EAST : 61hex, 97dec, a ASCII
TEXT_SOUTH : 62hex, 98dec, b ASCII
TEXT_WEST : 2Fhex, 99dec, c ASCII



Note: TEXT_NORTH is the default text orientation



See Also: [PRINT_CHAR](#), [PRINT_STRING](#), [SELECT_FONT](#)

Example:

The following sequence will print the text pattern similar to the one on the picture above.

```
SET_XY      25 hex
60          60 dec
10          10 dec
SELECT_FONT 2B hex
0           0 dec
TEXT_NORTH  60 hex
PRINT_STRING 2D hex
"Text North "
```

```
NULL          0 hex
TEXT_EAST     61 hex
PRINT_STRING  2D hex
" Text East  "
NULL          0 hex
TEXT_SOUTH    62 hex
PRINT_STRING  2D hex
" Text South "
NULL          0 hex
TEXT_WEST     63 hex
PRINT_STRING  2D hex
" Text West  "
NULL          0 hex
```



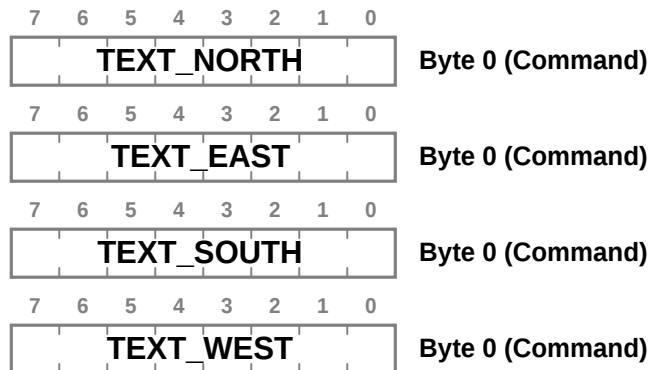
```
NULL          0 hex
TEXT_EAST     61 hex
PRINT_STRING  2D hex
" Text East  "
NULL          0 hex
TEXT_SOUTH    62 hex
PRINT_STRING  2D hex
" Text South "
NULL          0 hex
TEXT_WEST     63 hex
PRINT_STRING  2D hex
" Text West  "
NULL          0 hex
```

1.26 TEXT_SOUTH

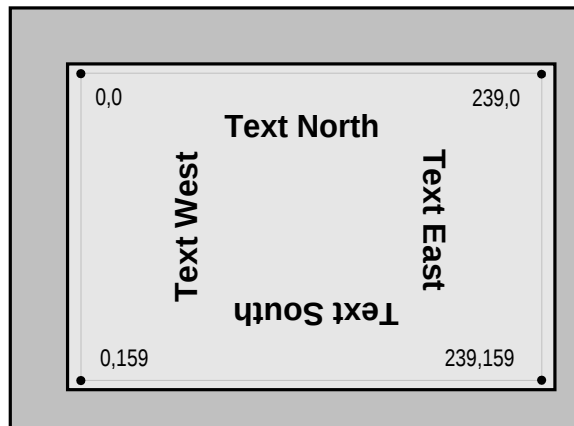
Description: Set the orientation of the text, as shown on the picture below

Class: Single Byte Commands

Code:
TEXT_NORTH: 60hex, 96dec, ' ASCII
TEXT_EAST : 61hex, 97dec, a ASCII
TEXT_SOUTH: 62hex, 98dec, b ASCII
TEXT_WEST : 2Fhex, 99dec, c ASCII



Note: TEXT_NORTH is the default text orientation



See Also: [PRINT_CHAR](#), [PRINT_STRING](#), [SELECT_FONT](#)

Example:

The following sequence will print the text pattern similar to the one on the picture above.

```
SET_XY      25 hex
60          60 dec
10          10 dec
SELECT_FONT 2B hex
0           0 dec
TEXT_NORTH  60 hex
PRINT_STRING 2D hex
"Text North "
```

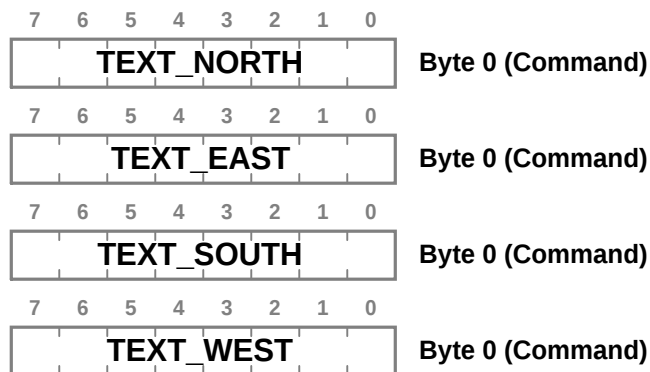
```
NULL          0 hex
TEXT_EAST     61 hex
PRINT_STRING  2D hex
" Text East  "
NULL          0 hex
TEXT_SOUTH    62 hex
PRINT_STRING  2D hex
" Text South "
NULL          0 hex
TEXT_WEST     63 hex
PRINT_STRING  2D hex
" Text West  "
NULL          0 hex
```

1.27 TEXT_WEST

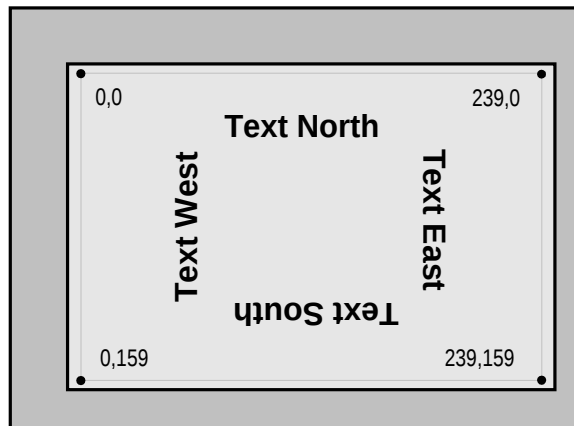
Description: Set the orientation of the text, as shown on the picture below

Class: Single Byte Commands

Code:
TEXT_NORTH: 60hex, 96dec, ' ASCII
TEXT_EAST : 61hex, 97dec, a ASCII
TEXT_SOUTH: 62hex, 98dec, b ASCII
TEXT_WEST : 2Fhex, 99dec, c ASCII



Note: TEXT_NORTH is the default text orientation



See Also: [PRINT_CHAR](#), [PRINT_STRING](#), [SELECT_FONT](#)

Example:

The following sequence will print the text pattern similar to the one on the picture above.

```
SET_XY      25 hex
60          60 dec
10          10 dec
SELECT_FONT 2B hex
0           0 dec
TEXT_NORTH  60 hex
PRINT_STRING 2D hex
"Text North "
```

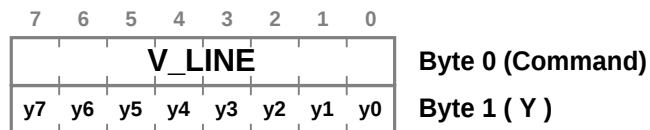
```
NULL          0 hex
TEXT_EAST     61 hex
PRINT_STRING  2D hex
" Text East  "
NULL          0 hex
TEXT_SOUTH    62 hex
PRINT_STRING  2D hex
" Text South "
NULL          0 hex
TEXT_WEST     63 hex
PRINT_STRING  2D hex
" Text West  "
NULL          0 hex
```

1.28 V_LINE

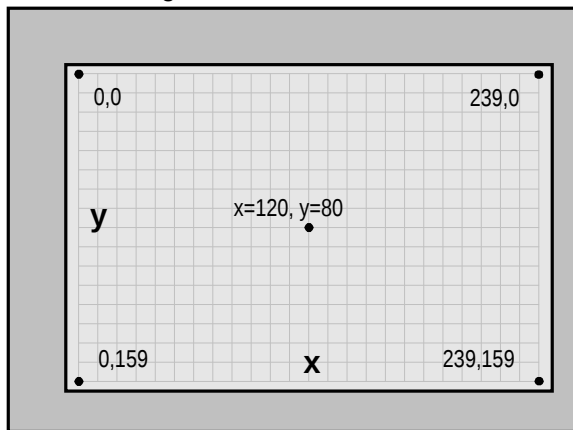
Description: Fast draws a vertical line from Current Position, to the row specified by the parameter.

Class: Double Byte Command

Code: 41hex, 65dec, A ASCII



Note: The screen size is 240x160. However, the valid Y range is 0 - 255



See Also: [H_LINE](#), [SET_XY](#)

Example:

The following sequence will draw the vertical blue line from (95, 10) to (95, 110)

```
SET_COLOR  24 hex
BLUE       11000000 bin
SET_XY     25 hex
95         95 dec
10         10 dec
V_LINE     41 hex
110       110 dec
```

Index

- A -

ARC 2

- B -

BOX 4

BOX_FILL 5

- C -

CIRCLE_R 6

CIRCLE_R_FILL 7

CLS 8

- H -

H_LINE 9

- L -

LIGHT_OFF 10

LIGHT_ON 11

LINE_TO_XY 12

- P -

PICTURE 13

PLOT 14

PLOT_XY 15

PRINT_CHAR 16

PRINT_CHAR_BG 17

PRINT_STRING 18

PRINT_STRING_BG 19

PUT_BITMAP 20

- S -

SELECT_FONT 23

SET_BG_COLOR 24

SET_COLOR 25

SET_XY 26

- T -

TEXT 27

TEXT_EAST 27

TEXT_NORTH 27

TEXT_SOUTH 27

TEXT_WEST 27

- V -

V_LINE 35