

ezDisplay LCM Command List

V.2 171113

Default baud rate of LCM is 115200

Except for codes for 0xd1~0xd9, 0xa0~ab and 0xf0~0xf6 all other codes only change the display memory, thus you have to excute the "0xd1" code (Function of refresh the display) to display the changes in the display memory after you write a character, string , pattern or draw a line.

Wrong Example:

```
Write_5X7_String(7, 17 , positive, "RPM");
Write_8X16_Pattern(1, 45, positive, 0);
Draw_Rectangle( 0, 0, 127, 127, positive );
```

*/*Without excute the Show_Display_Momery() function the change only in the memory, it won't display */*

Correct Example:

```
Write_5X7_String(7, 17 , positive, "RPM");
Write_8X16_Pattern(1, 45, positive, 0);
Draw_Rectangle( 0, 0, 127, 127, positive );
Show_Display_Momery();
```

*/*With the execution of Show_Display_Memory() fuction, the change of display memory will be displayed*/*

LCM default for Hex Command					
Code	Function	Sequence of HEX mode	API for Arduino (Hex mode)	Instruction of AT mode	API for Arduino (AT mode)
N/A	Send Page (128X64 bitmap) to OLED & LCM (An array consist of 1024 bytes bitmap)	<ol style="list-style-type: none"> 1. A "for" loop to send 1024 bytes user define display information 2. Wait until receive a module available byte ('E') from ezDisplay 	<pre>for (i = 0 ; i < 1024; i++) { Serial.write(User_define_array[i]); } while (Serial.read() !='E') {}</pre>	<ol style="list-style-type: none"> 1. A "for" loop to send 1024 bytes user define display information 2. Wait until receive a module available byte ('E') from ezDisplay 	<pre>for (i = 0 ; i < 1024; i++) { Serial.write(User_define_array[i]); } while (Serial.read() !='E') {}</pre>
N/A	Text input without AT command	Text input 5x7 string Total 8 lines, and 21 characters per line could be input on OLED & LCM screen			
0x80	Write a 5X7 Character	<ol style="list-style-type: none"> 1. Send 0x80 2. Send which line to put this character 3. Send which cloumn to put this character 4. Send character's ASCII code 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_5X7_Character(int line, int column, char Char) { Serial.write(0x80); Serial.write(line); Serial.write(column); Serial.write(Char); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT80=(line,column, Character) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT80=(0,0,A)</p>	<pre>Write_AT_Command("AT80=(0,0,A)"); void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x81	Write a 5X7 String	<ol style="list-style-type: none"> 1. Send 0x81 2. Send which line to start the string 3. Send which cloumn to start the string 4. Send string 5. Wait until receive a module available byte('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_5X7_String(int line, int column, char * string) { Serial.write(0x81); Serial.write(line); Serial.write(column); Serial.print(string); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT81=(line,column,String) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT81=(0,0,ABCD1234)</p>	<pre>Write_AT_Command("AT81=(0,0,ABCD1234)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x82	<p>Write a 8X16 Character</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x82 2. Send which line to put this character 3. Send which cloumn to put this character 4. Send character's ASCII code 5.Wait until receive a module available byte('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_8X16_Character(int line, int column, char Char) { Serial.write(0x82); Serial.write(line); Serial.write(column); Serial.write(Char); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT82=(line,column,Character) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT82=(0,0,A)</p>	<pre>Write_AT_Command("AT82=(0,0,A)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x83	<p>Write a 8X16 String</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x83 2. Send which line to stary the string 3. Send which cloumn to start the string 4. Send string 5. Wait until receive a module available byte('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_8X16_String(int line, int column, char * string) { Serial.write(0x83); Serial.write(line); Serial.write(column); Serial.print(string); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT83=(line,column,String) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT83=(0,0,ABCD1234)</p>	<pre>Write_AT_Command("AT83=(0,0,ABCD1234)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x84	<p>Dsisplay a 8X8 pattern</p>	<ol style="list-style-type: none"> 1. Send 0x84 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_8X8_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int Pattern_ID) { Serial.write(0x84); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT84=(X position, Y position, pattern ID) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT84=(16,32,1)</p>	<pre>Write_AT_Command("AT84=(16,32,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x85	<p>Dsisplay a 8X16 pattern</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x85 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_8X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int Pattern_ID) { Serial.write(0x85); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT85=(X position,Y position,pattern ID) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT85=(16,32,1)</p>	<pre>Write_AT_Command("AT85=(16,32,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x86	<p>Dsisplay a 16X16 pattern</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x86 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_16X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int Pattern_ID) { Serial.write(0x86); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT86=(X position,Y position,pattern ID) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT86=(16,32,1)</p>	<pre>Write_AT_Command("AT86=(16,32,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x87	<p>Dsplay a 32X32 pattern</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x87 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_32X32_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int Pattern_ID) { Serial.write(0x87); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT87=(X position,Y position,pattern ID) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT87=(16,32,1)</p>	<pre>Write_AT_Command("AT87=(16,32,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x88	<p>Dsplay a 16X32 pattern</p> <p>(not available for 96x8 RGB LED)</p>	<ol style="list-style-type: none"> 1. Send 0x88 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED 	<pre>void Write_16X32_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int Pattern_ID) { Serial.write(0x88); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT88=(X position,Y position,pattern ID) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT88=(16,32,1)</p>	<pre>Write_AT_Command("AT88=(16,32,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x90	<p>Draw a line</p>	<ol style="list-style-type: none"> 1. Send 0x90 2. Send the X coordinate of first point 3. Send the Y coordinate of first point 4. Send the X coordinate of second point 5. Send the Y coordinate of second point 6. Send 0 or 1 <ol style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Line(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int positive) { Serial.write(0x90); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT90=(X0 position,Y0 position,X1 position,Y1 position,0~255) <ol style="list-style-type: none"> *0~255: Send 0 or 1 <ol style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT90=(0,0,127,63,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT90=(0,0,127,63,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x91	<p>Draw a Rectangle</p>	<ol style="list-style-type: none"> 1. Send 0x91 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send 0 or 1 <ol style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Rectangle(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int positive) { Serial.write(0x91); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT91=(X0 position,Y0 position,X1 position,Y1 position,0~255) <ol style="list-style-type: none"> *0~255: Send 0 or 1 <ol style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT91=(10,10,100,49,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT91=(10,10,100,49,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x92	Draw a filled Rectangle	<ol style="list-style-type: none"> 1. Send 0x92 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Rectangle(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int positive) { Serial.write(0x92); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT92=(X0 position,Y0 position,X1 position,Y1 position,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT92=(10,10,100,49,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT92=(10,10,100,49,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x93	Draw a Square	<ol style="list-style-type: none"> 1. Send 0x93 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the width of this square 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Square(int X0_Pos, int Y0_Pos, int width, int positive) { Serial.write(0x93); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(width); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT93=(X position,Y position,Width,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT93=(8,10,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT93=(8,10,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x94	Draw a Circle	<ol style="list-style-type: none"> 1. Send 0x94 2. Send the X coordinate of the center 3. Send the Y coordinate of the center 4. Send the radius of this circle 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Circle(int X0_Pos, int Y0_Pos, int radius, int positive) { Serial.write(0x94); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(radius); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT94=(X position,Y position,Radius,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT94(64,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT94(64,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x95	Draw a filled Circle	<ol style="list-style-type: none"> 1. Send 0x95 2. Send the X coordinate of the center 3. Send the Y coordinate of the center 4. Send the radius of this circle 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Circle(int X0_Pos, int Y0_Pos, int radius, int positive) { Serial.write(0x95); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(radius); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT95=(X position,Y position,Radius,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT95=(64,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT95=(64,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x96	Draw a tip upward Triangle	<ol style="list-style-type: none"> 1. Send 0x96 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the height of the tip to the bottom 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Triangle_Up_Ward(int X0_Pos, int Y0_Pos, int height, int positive) { Serial.write(0x96); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(height); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT96=(X position,Y position,Height,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT96=(64,10,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT96=(64,10,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x97	Draw a filled tip upward Triangle	<ol style="list-style-type: none"> 1. Send 0x97 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the height of the tip to the bottom 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Triangle_Up_Ward(int X0_Pos, int Y0_Pos, int height, int positive) { Serial.write(0x97); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(height); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT97=(X position,Y position,Height,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT97=(64,10,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT97=(64,10,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x98	Draw a tip downward Triangle	<ol style="list-style-type: none"> 1. Send 0x98 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the height of the tip to the top 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Triangle_Down_Ward(int X0_Pos, int Y0_Pos, int height, int positive) { Serial.write(0x98); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(height); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT98=(X position,Y position,Height,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT98=(64,50,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT98=(64,50,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x99	Draw a filled tip downward Triangle	<ol style="list-style-type: none"> 1. Send 0x99 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the height of the tip to the top 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Triangle_Down_Ward(int X0_Pos, int Y0_Pos, int height, int positive) { Serial.write(0x99); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(height); Serial.write(positive); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT99=(X position,Y position,Height,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT99=(64,50,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT99=(64,50,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x9a	Draw a tip leftward Triangle	<ol style="list-style-type: none"> 1. Send 0x9a 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the width of the tip to the right 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Triangle_Left_Ward(int X0_Pos, int Y0_Pos, int width, int positive) { Serial.write(0x9a); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(width); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9a=(X position,Y position,Width,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9a=(16,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT9a=(16,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x9b	Draw a filled tip leftward Triangle	<ol style="list-style-type: none"> 1. Send 0x9b 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the width of the tip to the right 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Triangle_Left_Ward(int X0_Pos, int Y0_Pos, int width, int positive) { Serial.write(0x9b); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(width); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9b=(X position,Y position,Width,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9b=(16,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT9b=(16,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x9c	Draw a tip rightward Triangle	<ol style="list-style-type: none"> 1. Send 0x9c 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the width of the tip to the left 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Triangle_Right_Ward(int X0_Pos, int Y0_Pos, int width, int positive) { Serial.write(0x9c); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(width); Serial.write(positive); /* Send 0 or 1 */ while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9c=(X position,Y position,Width, 0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9c=(120,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT9c=(120,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x9d	Draw a filled tip rightward Triangle	<ol style="list-style-type: none"> 1. Send 0x9d 2. Send the X coordinate of the tip 3. Send the Y coordinate of the tip 4. Send the width of the tip to the left 5. Send 0 or 1 <ul style="list-style-type: none"> 1: Positive Mode 0: Negative Mode 6. Wait until receive a module available byte ('E') from ezDisplay 7. Wait 2ms, but NO need for OLED. 	<pre>void Draw_Filled_Triangle_Right_Ward(int X0_Pos, int Y0_Pos, int width, int positive) { Serial.write(0x9d); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(width); Serial.write(positive); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9d=(X position,Y position,Width,0~255) *0~255: Send 0 or 1 1: Positive Mode 0: Negative Mode 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9d=(120,32,30,1) : '1' positive mode</p>	<pre>Write_AT_Command("AT9d=(120,32,30,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0x9e	Set a pixel for positive display (show pixel)	<ol style="list-style-type: none"> 1. Send 0x9e 2. Send the X coordinate of the pixel 3. Send the Y coordinate of the pixel 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED. 	<pre>void Set_Pixel(int X0_Pos, int Y0_Pos) { Serial.write(0x9e); Serial.write(X0_Pos); Serial.write(Y0_Pos); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9e=(X position,Y position) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9e=(120,32)</p>	<pre>Write_AT_Command("AT9e=(120,32)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0x9f	Set a pixel for negative display (clear pixel)	<ol style="list-style-type: none"> 1. Send 0x9f 2. Send the X coordinate of the pixel 3. Send the Y coordinate of the pixel 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED. 	<pre>void Clear_Pixel(int X0_Pos, int Y0_Pos) { Serial.write(0x9f); Serial.write(X0_Pos); Serial.write(Y0_Pos); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. AT9f=(X position,Y position) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> AT9f=(120,32)</p>	<pre>Write_AT_Command("AT9f=(120,32)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa0	Display image row by row Up Ward	<ol style="list-style-type: none"> 1. Send 0xa0 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Row_By_Row_Up_Ward(int Speed) { Serial.write(0xa0); Serial.write(speed); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa0=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa0=(20)</p>	<pre>Write_AT_Command("ATa0=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa1	Display image row by row Down Ward	<ol style="list-style-type: none"> 1. Send 0xa1 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Row_By_Row_Down_Ward(int speed) { Serial.write(0xa1); Serial.write(speed); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa1=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa1=(20)</p>	<pre>Write_AT_Command("ATa1=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa2	Display image column by column Left Ward	<ol style="list-style-type: none"> 1. Send 0xa2 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Column_By_Column_Left_Ward(int speed) { Serial.write(0xa2); Serial.write(speed); while (Serial.read() != 'E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa2=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa2=(20)</p>	<pre>Write_AT_Command("ATa2=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0xa3	Display image column by column Right Ward	<ol style="list-style-type: none"> 1. Send 0xa3 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Column_By_Column_Right_Ward(int Speed) { Serial.write(0xa3); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa3=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa3=(20)</p>	<pre>Write_AT_Command("ATa3=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa4	Erase image row by row Up Ward	<ol style="list-style-type: none"> 1. Send 0xa4 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Row_By_Row_Up_Ward(int Speed) { Serial.write(0xa4); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa4=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa4=(20)</p>	<pre>Write_AT_Command("ATa4=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa5	Erase image row by row Down Ward	<ol style="list-style-type: none"> 1. Send 0xa5 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Row_By_Row_Down_Ward(int Speed) { Serial.write(0xa5); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa5=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa5=(20)</p>	<pre>Write_AT_Command("ATa5=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa6	Erase image column by column Left Ward	<ol style="list-style-type: none"> 1. Send 0xa6 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Column_By_Column_Left_Ward(int Speed) { Serial.write(0xa6); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa6=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa6=(20)</p>	<pre>Write_AT_Command("ATa6=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa7	Erase image column by column Right Ward	<ol style="list-style-type: none"> 1. Send 0xa7 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Column_By_Column_Right_Ward(int Speed) { Serial.write(0xa7); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa7=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa7=(20)</p>	<pre>Write_AT_Command("ATa7=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0xa8	Display image Inside Out	<ol style="list-style-type: none"> 1. Send 0xa8 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Inside_Out(int Speed) { Serial.write(0xa8); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa8=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa8=(20)</p>	<pre>Write_AT_Command("ATa8=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xa9	Display image Outside In	<ol style="list-style-type: none"> 1. Send 0xa9 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Display_Outside_In(int Speed) { Serial.write(0xa9); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATa9=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATa9=(20)</p>	<pre>Write_AT_Command("ATa9=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xaa	Erase image Inside Out	<ol style="list-style-type: none"> 1. Send 0xaa 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Inside_Out(int Speed) { Serial.write(0xaa); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATaa=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATaa=(20)</p>	<pre>Write_AT_Command("ATaa=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xab	Erase image Outside In	<ol style="list-style-type: none"> 1. Send 0xab 2. Send the speed (typical time is 20ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Erase_Outside_In(int Speed) { Serial.write(0xab); Serial.write(Speed); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATab=(Speed in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATab=(20)</p>	<pre>Write_AT_Command("ATab=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xc0	Build user define 8X8 pattern bitmap into OLED's display memory (Maximun number of user define 8X8 pattern is 10 (0~9))	<ol style="list-style-type: none"> 1. Send 0xc0 2. Send the pattern ID 3. Sent the bitmap of this pattern ID 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED. 	<pre>void Build_User_Define_8X8_Pattern() { Serial.write(0xc0); Serial.write(0); for (i = 0; i < 8; i++) { Serial.write(User_Define_8X8_pattern_ID[i]); } while (Serial.read() !='E') {} delay(2); }</pre>	N/A	N/A

0xc1	Build user define 8X16 pattern bitmap into OLED's display memory (Maximun number of user define 8X16 pattern is 10 (0~9)) (not available for 96x8 RGB LED)	1. Send 0xc1 2. Send the pattern ID 3. Sent the bitmap of this pattern ID 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED.	void Build_User_Define_8X16_Pattern() { Serial.write(0xc1); Serial.write(0); for (i = 0; i < 16; i++) { Serial.write(User_Define_8X16_pattern_ID[i]); } while (Serial.read() != 'E') {}; delay(2); }	N/A	N/A
0xc2	Build user define 16X16 pattern bitmap into OLED's display memory (Maximun number of user define 16X16 pattern is 10 (0~9)) (not available for 96x8 RGB LED)	1. Send 0xc2 2. Send the pattern ID 3. Sent the bitmap of this pattern ID 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED.	void Build_User_Define_16X16_Pattern() { Serial.write(0xc2); Serial.write(0); for (i = 0; i < 32; i++) { Serial.write(User_Define_16X16_pattern_ID[i]); } while (Serial.read() != 'E') {} delay(2); }	N/A	N/A
0xc3	Build user define 32X32 pattern bitmap into OLED's display memory (Maximun number of user define 32X32 pattern is 5 (0~4)) (not available for 96x8 RGB LED)	1. Send 0xc3 2. Send the pattern ID 3. Sent the bitmap of this pattern ID 4. Wait until receive a module available byte ('E') from ezDisplay 5. Wait 2ms, but NO need for OLED.	void Build_User_Define_32X32_Pattern() { Serial.write(0xc3); Serial.write(0); for (i = 0; i < 128; i++) { Serial.write(User_Define_32X32_pattern_ID[i]); } while (Serial.read() != 'E') {} delay(2); }	N/A	N/A
0xc4	Dsisplay a user define 8X8 pattern (Build user define 8X8 pattern function needs to run before this function)	1. Send 0xc4 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED.	void Write_User_Define_8X8_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) { Serial.write(0xc4); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() != 'E') {} delay(2); }	N/A	N/A
0xc5	Dsisplay a user define 8X16 pattern (Build user define 8X16 pattern function needs to run before this function) (not available for 96x8 RGB LED)	1. Send 0xc5 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED.	void Write_User_Define_8X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) { Serial.write(0xc5); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() != 'E') {} delay(2); }	N/A	N/A

0xc6	Display a user define 16X16 pattern (Build user define 16X16 pattern function needs to run before this function) (not available for 96x8 RGB LED)	<ol style="list-style-type: none"> 1. Send 0xc6 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_User_Define_16X16_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) { Serial.write(0xc6); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	N/A	N/A
0xc7	Display a user define 32X32 pattern (Build user define 32X32 pattern function needs to run before this function) (not available for 96x8 RGB LED)	<ol style="list-style-type: none"> 1. Send 0xc7 2. Send the Up Left X coordinate of pattern 3. Send the Up Left Y coordinate of pattern 4. Send the ID of pattern 5. Wait until receive a module available byte ('E') from ezDisplay 6. Wait 2ms, but NO need for OLED. 	<pre>void Write_User_Define_32X32_Pattern(int Up_Left_Xpos, int Up_Left_Ypos, int negative, int Pattern_ID) { Serial.write(0xc7); Serial.write(Up_Left_Xpos); Serial.write(Up_Left_Ypos); Serial.write(Pattern_ID); while (Serial.read() !='E') {} delay(2); }</pre>	N/A	N/A
0xd0	Clear display	<ol style="list-style-type: none"> 1. Send 0xd0 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>void Clear_Display_Momery(void) { Serial.write(0xd0); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd0=() 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd0=()</p>	<pre>Write_AT_Command("ATd0=()") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd1	Show the data in the display memory	<ol style="list-style-type: none"> 1. Send 0xd1 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>void Show_Display_Momery(void) { Serial.write(0xd1); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd1=() 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd1=()</p>	<pre>Write_AT_Command("ATd1=()") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd2	Scroll the whole display upward	<ol style="list-style-type: none"> 1. Send 0xd2 2. Send the shift time (typical time is 70ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Whole_Display_Memory_Up(int shift time) { Serial.write(0xd2); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd2=(shift time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd2=(20)</p>	<pre>Write_AT_Command("ATd2=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd3	Scroll the whole display downward	<ol style="list-style-type: none"> 1. Send 0xd3 2. Send the shift time (typical time is 70ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Whole_Display_Memory_Down(int shift time) { Serial.write(0xd3); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd3=(shift time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd3=(20)</p>	<pre>Write_AT_Command("ATd3=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0xd4	Scroll the whole display leftward	<ol style="list-style-type: none"> 1. Send 0xd4 2. Send the shift time (typical time is 70ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Whole_Display_Memory_Left(int shift time) { Serial.write(0xd4); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd4=(shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd4=(20)</p>	<pre>Write_AT_Command("ATd4=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd5	Scroll the whole display rightward	<ol style="list-style-type: none"> 1. Send 0xd5 2. Send the shift time (typical time is 70ms) 3. Wait until receive a module available byte ('E') from ezDisplay 4. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Whole_Display_Memory_Right(int shift time) { Serial.write(0xd5); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd5=(shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd5=(20)</p>	<pre>Write_AT_Command("ATd5=(20)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd6	Scroll the section display upward	<ol style="list-style-type: none"> 1. Send 0xd6 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send the shift time (typical time is 20ms) 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Section_Display_Memory_Up(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) { Serial.write(0xd6); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(shift time); while (Serial.read() !='E') {}; delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd6=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd6=(10,16,120,50,1)</p>	<pre>Write_AT_Command("ATd6=(10,16,120,50,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd7	Scroll the section display downward	<ol style="list-style-type: none"> 1. Send 0xd7 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send the shift time (typical time is 70ms) 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Section_Display_Memory_Down(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) { Serial.write(0xd7); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd7=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd7=(10,16,120,50,1)</p>	<pre>Write_AT_Command("ATd7=(10,16,120,50,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0xd8	Scroll the section display leftward	<ol style="list-style-type: none"> 1. Send 0xd8 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send the shift time (typical time is 20ms) 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Section_Display_Memory_Left(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) { Serial.write(0xd8); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd8=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd8=(10,16,120,50,1)</p>	<pre>Write_AT_Command("ATd8=(10,16,120,50,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xd9	Scroll the section display rightward	<ol style="list-style-type: none"> 1. Send 0xd9 2. Send the X coordinate of up left corner 3. Send the Y coordinate of up left corner 4. Send the X coordinate of bottom right corner 5. Send the Y coordinate of bottom right corner 6. Send the shift time (typical time is 70ms) 7. Wait until receive a module available byte ('E') from ezDisplay 8. Wait 2ms, but NO need for OLED. 	<pre>void Scroll_Section_Display_Memory_Right(int X0_Pos, int Y0_Pos, int X1_Pos, int Y1_Pos, int shift time) { Serial.write(0xd9); Serial.write(X0_Pos); Serial.write(Y0_Pos); Serial.write(X1_Pos); Serial.write(Y1_Pos); Serial.write(shift time); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATd9=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATd9=(10,16,120,50,1)</p>	<pre>Write_AT_Command("ATd9=(10,16,120,50,1)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xed	Set default Background	<ol style="list-style-type: none"> 1. Send 0xed 2. Send the backlight code 1 3. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>void Display_Off(void){ Serial.write(0xed); Serial.write(1); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATed=(Backlight 0~255) *Backlight 0~7: <for LCM with RGB backlight > '0': none '1': red color '2': green color '3': blue color '4': yellow color '5': purple color '6': cyan color '7': white color <for LCM with White backlight > '0': none '1~7': white color 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATed=(1) : '7' white color</p>	<pre>Write_AT_Command("ATed=(7)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xf0	Turn display Off	<ol style="list-style-type: none"> 1. Send 0xf0 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>void Display_Off(void){ Serial.write(0xf0); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATf0=() 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATf0=()</p>	<pre>Write_AT_Command("ATf0=()") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>

0xf1	Turn display On	<ol style="list-style-type: none"> 1. Send 0xf1 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>void Display_On(void) { Serial.write(0xf1); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATf1=() 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATf1=()</p>	<pre>Write_AT_Command("ATf1=()") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>
0xf6	<p>Change Instruction mode (</p> <p>0 for HEX command, 1 for AT command</p> <p>)</p>	<ol style="list-style-type: none"> 1. Send 0xf6 2. Send instruction mode 1 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms, but NO need for OLED. 	<pre>int Change_Instruction_Mode(int mode) { Serial.write(0xf6); Serial.write(1); while (Serial.read() !='E') {} delay(2); }</pre>	<ol style="list-style-type: none"> 1. ATf6=(instruction mode) 2. Wait until receive a module available byte ('E') from ezDisplay 3. Wait 2ms <p><example> ATf6=(0)</p>	<pre>Write_AT_Command("ATf6=(0)") void Write_AT_Command(char *string) { Serial.print(string); while (Serial.read() != 'E') {} delay(2); }</pre>