	Default baud rate of LED Display is 115200			void Write_AT_Command(char *string)
				{
				Serial.print(string);
				while (Serial.read() != 'E') {}
				delay(2);
				}

Code	**Function**	**Instruction of AT Command mode**	**API for Arduino**	**Example of using Write_AT_Command() subroutine above**

N/A	Sent a image(192X64 bitmap) to LED Display
	(An array consist of 1536 bytes bitmap information)
		1. A "for" loop to send 1536 bytes user define display information
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			for (i = 0 ; i < 1536; i++)
			{
			Serial.write(User_define_array[i]);
			}
			while (Serial.read() !='E') {}
			delay(2);	

0x80	Write a 5X7 Character	1. AT80=(line,column,Character)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT80=(0,0,A)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT80=(0,0,A)")

0x81	Write a 8X8 String	1.AT81=(line,column,String)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT81=(0,0,ABCD1234)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT81=(0,0,ABCD1234)")

0x82	Write a 8X16 Character	1.AT82=(line,column,Character)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT82=(0,0,A)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT82=(0,0,A)")

0x83	Write a 8X16 String	1.AT83=(line,column,String)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT83=(0,0,ABCD1234)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT83=(0,0,ABCD1234)")

0x84	Dsiplay a 8X8 pattern	1. AT84=(X position,Y position,pattern ID)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT84=(16,32,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT84=(16,32,1)")

0x85	Dsiplay a 8X16 pattern	1.AT85=(X position,Y position,pattern ID)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT85=(16,32,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT85=(16,32,1)")

0x86	Dsiplay a 16X16 pattern	1. AT86=(X position,Y position,pattern ID)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT86=(16,32,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT86=(16,32,1)")

0x87	Dsiplay a 32X32 pattern	1. AT87=(X position,Y position,pattern ID)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT87=(16,32,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT87=(16,32,1)")

0x90	Draw a line	1. AT90=(X0 position,Y0 position,X1 position,Y1 position,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT90=(0,0,127,63,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT90=(0,0,127,63,1)")

0x91	Draw a Rectangle	1. AT91=(X0 position,Y0 position,X1 position,Y1 position,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT91=(10,10,100,49,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT91=(10,10,100,49,1)")

0x92	Draw a filled Rectangle	1. AT92=(X0 position,Y0 position,X1 position,Y1 position,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT92=(10,10,100,49,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT92=(10,10,100,49,1)")

0x93	Draw a Square	1. AT93=(X position,Y position,Width,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT93=(8,10,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT93=(8,10,30,1)")

0x94	Draw a Circle	1. AT94=(X position,Y position,Radius,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT94(64,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT94(64,32,30,1)")

0x95	Draw a filled Circle	1. AT95=(X position,Y position,Radius,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT95=(64,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT95=(64,32,30,1)")

0x96	Draw a tip upward Triangle	1. AT96=(X position,Y position,Height,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT96=(64,10,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT96=(64,10,30,1)")

0x97	Draw a filled tip upward Triangle	1. AT97=(X position,Y position,Height,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT97=(64,10,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT97=(64,10,30,1)")

0x98	Draw a tip downward Triangle	1. AT98=(X position,Y position,Height,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT98=(64,50,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT98=(64,50,30,1)")

0x99	Draw a filled tip downward Triangle	1. AT99=(X position,Y position,Height,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT99=(64,50,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT99=(64,50,30,1)")

0x9a	Draw a tip leftward Triangle	1. AT9a=(X position,Y position,Width,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9a=(16,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9a=(16,32,30,1)")

0x9b	Draw a filled tip leftward Triangle	1. AT9b=(X position,Y position,Width,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9b=(16,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9b=(16,32,30,1)")

0x9c	Draw a tip rightward Triangle	1. AT9c=(X position,Y position,Width,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9c=(120,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9c=(120,32,30,1)")

0x9d	Draw a filled tip rightward Triangle	1. AT9d=(X position,Y position,Width,0 or 1)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9d=(120,32,30,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9d=(120,32,30,1)")

0x9e	Set a pixel for positive display (show pixel)	1. AT9e=(X position,Y position)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9e=(120,32)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9e=(120,32)")

0x9f	Set a pixel for negative display (clear pixel)	1. AT9f=(X position,Y position)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("AT9f=(120,32)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("AT9f=(120,32)")

0xa0	Display image row by row Up Ward	1. ATa0=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa0=(20)")
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa0=(20)")

0xa1	Display image row by row Down Ward	1. ATa1=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa1=(20)")
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa1=(20)")

0xa2	Display image column by column Left Ward	1. ATa2=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa2=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa2=(20)")

0xa3	Display image column by column Right Ward	1. ATa3=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa3=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa3=(20)")

0xa4	Erase image row by row Up Ward	1. ATa4=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa4=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa4=(20)")

0xa5	Erase image row by row Down Ward	1. ATa5=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa5=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa5=(20)")

0xa6	Erase image column by column Left Ward	1. ATa6=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa6=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa6=(20)")

0xa7	Erase image column by column Right Ward	1. ATa7=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa7=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa7=(20)")

0xa8	Display image Inside Out	1. ATa8=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa8=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa8=(20)")

0xa9	Display image Outside In	1. ATa9=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATa9=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATa9=(20)")

0xaa	Erase image Inside Out	1. ATaa=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATaa=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATaa=(20)")

0xab	Erase image Outside In	1. ATab=(Speed in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATab=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATab=(20)")

0xd0	Clear display	1. ATd0=()
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd0=()");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd0=()")

0xd1	Show the data in the display memory	1. ATd1=()
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd1=()");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd1=()")

0xd2	Scroll the whole display upward	1. ATd2=(shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd2=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd2=(20)")

0xd3	Scroll the whole display downward	1. ATd3=(shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd3=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd3=(20)")

0xd4	Scroll the whole display leftward	1. ATd4=(shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd4=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd4=(20)")

0xd5	Scroll the whole display rightward	1. ATd5=(shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd5=(20)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd5=(20)")

0xd6	Scroll the section display upward	1. ATd6=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd6=(10,16,120,50,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd6=(10,16,120,50,1)")

0xd7	Scroll the section display downward	1. ATd7=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd7=(10,16,120,50,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd7=(10,16,120,50,1)")

0xd8	Scroll the section display leftward	1. ATd8=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd8=(10,16,120,50,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd8=(10,16,120,50,1)")

0xd9	Scroll the section display rightward	1. ATd9=(X0 position,Y0 position,X1 position,Y1 position, shif time in ms)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATd9=(10,16,120,50,1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATd9=(10,16,120,50,1)")

0xda	Display quarter of display memory
	(Available for Mode0, 1, and 2 only)
		1. ATda=(Quadrant 0~3)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATda=(1)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATda=(1)")

0xf0	Turn display Off	1. ATf0=()
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf0=()");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf0=()")

0xf1	Turn display On	1. ATf1=()
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf1=()");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf1=()")

0xf2	Set the brightness of the LED Display	1. ATf2=(levele of brightness 0~11)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf2=(5)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf2=(5)")

0xf3	Inverse image	1. ATf3=()
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf3=()");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf3=()")

0xf6	Change Instruction mode
	(0 for Hex Coammand, 1 for AT Command)
		1. ATf6=(0)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf6=(0)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf6=(0)")

0xf7	Change Display Mode	1. ATf7=(Display Mode)
		2. Wait until receive a module available byte ('E') from LED Display
		3. Wait 2ms
			Serial.print("ATf7=(0)");
			while (Serial.read() !='E') {}
			delay(2);
				Write_AT_Command("ATf7=(0)")