

Using an external OLED display

Learning Objectives



These slides should help you to

- Connect the external 128x64 OLED display to your Circuit Playground Express
- Install the Adafruit Graphics Libraries necessary to use the 128x64 OLED
- Display static text and dynamic numerical values on the 128x64 OLED
- Use functions to organize the setup and updating of the display



Hardware connections

Final configuration of hardware





3D-printed bracket





Other hardware components







Aligning the Qwiic cable to the CPX pads





Begin assembly



Use 3 screws to attach the bracket to the CPX



Place nut on the bracket side





Insert a 4-40 screw through the hole of the 3.3V pad next to the USB socket.

Place the washer on the screw and the nut on the screw – the split washer should be located under the nut

Place nut on the bracket side





Use a Phillips head screwdriver to tighten the screw.

Tighten just enough to keep the nut from coming loose.

Initially you want to allow the bracket to move so that other screws can be inserted

Three of the four screws should be installed





The hole for the GND pad next to the USB socket should be empty

Attach the black cable





Isolate the terminal of the black wire on the Qwiic cable.

The flat end of that terminal will be inserted between the bracket and the CPX

You may need to slightly loosen the three screws that you have already installed

Attach the black cable





Slide the flat part of the terminal between the CPX and the bracket

Attach the black cable





Insert a 4-40 screw through the hole aligned with the pad, the round terminal and the bracket.

Add a split washer and nut. Tighten the screw.

Now tighten all four screws that hold the bracket to the CPX

Attach the yellow, blue and red cables





Qwiic wiring					
Black	Ground				
Yellow	SCL (clock)				
Blue	SDA (data)				
Red	Power, 3.3V				

Note that the screw for the yellow wire will need to be oriented in the opposite direction to the screws holding the other terminals



Completed attachment of Qwiic cable





Locate the OLED and 4 #2 sheet metal screws



Locate the OLED and the four #2 sheet metal screws in your kit.

Attach the OLED to the bracket with the screws.

Be careful not to over-tighten the screws. The screws should be snug, and the OLED should now wobble relative to the bracket.

Completed attachment of the OLED





Next, we'll attach the Qwiic cable to the OLED.

Please be careful with the next step.

You may need to rotate the terminals to allow the free end of the Qwiic cable to reach the socket on the OLED.



Be careful when inserting the Qwiic cable into the socket on the OLED

Pins and sockets of Qwiic connectors have an offset alignment





Pins and sockets of Qwiic connectors have an offset alignment





Final configuration of hardware







Install graphics libraries

Install the Adafruit SSD1306 Library From the IDE menus ...

Select "Sketch" → "Include Library" → "Manage Libraries ..."

Arduino File Edit	Sketch Tools Help			
	Verify/Compile	ЖR	Manage Libraries	企 第1
	Upload	жU	Add ZIP Library	
	Upload Using Programmer	企業U		
	Export compiled Binary	₹₩S	Arduino libraries	
	Show Sketch Folder	ЖК	Arduino Cloud Provider Examples	
	Include Library	>	Arduino Ono WIFI Dev Ed Library	
	Add File		Arduino_ArD39900	
			Bridge	
			Esplora	



Install the Adafruit SSD1306 Library From the IDE menus ...

- Select "Sketch" → "Include Library" → "Manage Libraries ..."
- Enter "ssd" in the search box
- Choose Adafruit SSD1306
- Click "Install"





Install the Adafruit SSD1306 Library From the IDE menus ...

- Select "Sketch" → "Include Library" → "Manage Libraries ..."
- Enter "ssd" in the search box
- Choose Adafruit SSD1306
- Click "Install"
- Click "Install all" to add GFX

	Library Manager	
Type All	Topic All ssd	
by Adafruit SSD1305 library for Mo <u>More info</u>	onochrome OLEDs based on SSD1305 drivers SSD1305 library for Monochrome OLEDs based or	SSD1305 drivers
	Dependencies for library Adaman 350 1306-2.4.6	
- Adafruit SSD1306	The library Adafruit SSD1306:2.4.6 needs some other library dependencies currently not installed:	
SSD1306 oled driver l 128x32 displays More info	- Adafruit GFX Library	ne 128x64 and
	Would you like to install also all the missing dependencies?	Install
Adafruit SSD1306 by Stefan Bethke SSD1306 oled driver lii	Install all Install 'Adafruit SSD1306' only Cancel	added to support
the 64x48 display by mc <u>More info</u>	auser.	





Test wiring and library with the demo code from the Adafruit library

Load ssd1306_128x64_i2c from examples

Select "File" \rightarrow "Examples" \rightarrow "Adafruit SSD1306" \rightarrow "ssd1306_128x64_i2c"

Upload the sketch

ad the sketch	Arduino	File	Edit	Sketch	Tools Help			
		New Oper Oper Sketo Exam Close Save Save Page Print	Lun n Recent chbook nples e As Setup	※日に1 ※ N ※ O ※ O ※ O ※ S ① 第 S ① 第 S ① 第 P ※ P	Built-in Examples 01.Basics 02.Digital 03.Analog 04.Communication 05.Control 06.Sensors 07.Display 08.Strings 09.USB 10.StarterKit_BasicKit 11.ArduinoISP	> > > > > > > > > > >		
Adafruit SSD1306 – appears after you add the library				\rightarrow	Adafruit SPIFlash Adafruit SSD1306 Adafruit ST7735 and ST7789 Library Adafruit STMPE610 Adafruit TouchScreen Adafruit Unified Sensor Adafruit Zero FFT Library DallasTemperature	> > > > > > > > > > > > > > > > > > > >	OLED_featherwing ssd1306_128x32_i2c ssd1306_128x32_spi ssd1306_128x64_i2c ssd1306_128x64_spi	1 i2 p

128x64 is resolution

i2c is communication protocol



Demonstrate static text and dynamic values with the OLEDdisplayFunctions.ino sketch

- Download <u>OLEDdisplayFunctions.ino</u>
- Upload the sketch
- Inspect the code with code-folding



- Turn on Code Folding in Preferences for Arduino IDE
- Open ODEdisplay functions

	Preferences	
	Settings Network	
Sketchbook location:		
/Users/gerry/Documents/A	rduino	Browse
Editor language:	System Default	ᅌ (requires restart of Arduin
Editor font size:	12	
Interface scale:	✓ Automatic 100 0% (requires restart of Arduino))
Theme:	Default theme 📀 (requires restart of Arduino)	
Show verbose output during	🗌 compilation 🛛 upload	
Compiler warnings:	None 📀	
Display line numbers	Enable Obde Folding	
Verify code after upload	Use external editor	
🗹 Check for updates on st	artup 🛛 🔽 save when verifying or	uploading
Use accessibility feature	\$	
Additional Boards Manager L	RLs: https://www.adafruit.com/package_adafruit_index.j	ison,http://arduino.esp82 🔲
More preferences can be edi	ed directly in the file	
/Users/gerry/Library/Arduin	o15/preferences.txt	
(edit only when Arduino is n	t running)	
		OK Cancel
		ent currect

- Turn on Code Folding in Preferences for Arduino IDE
- Open ODEdisplay functions
- Notice + and signs near function declarations



OEDCIDENT V/ File: 0LEDdisplayFunctions.ino /// File: 0LEDdisplayFunctions.ino /// dotsing and isolate the OLED setup and OLED display years and isolate the OLED setup and OLED display years and isolate the OLED setup and OLED display years and isolate the OLED setup and OLED display years and isolate the OLED display // Librories needed for the OLED display #include dire.he #include dire.he // Rest pin 4 (or -1 if shafing Analino reset pin) #define OLEDARSET // Rest pin 4 (or -1 if shafing Analino reset pin) #define SCREEN.HEIGHT // VLED display height pinets #define SCREEN.HEIGHT, SCREEN.HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIGHT, SCREEN.HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIGHT, SCREEN.HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIGHT, SCREEN_HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIGHT, SCREEN_HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIGHT, SCREEN_HEIGHT, &Wire, OLED_RESET); // #define SCREEN.HEIG	OLDDisplayfunctions // File: OLEDdisplayfunctions.ino /// Demonstrate a minimumal use case for an Adafruit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // - Libraries needed for the OLED display // - Libraries needed for the OLED display #include differit_GRX.b // Wire.h provides IZC support #include differit_GRX.b // Use on 1 if shafing Aduiton reset pin) #define GREENTIGH // DEED display width in pixels ////////////////////////////////////	•••	OLEDdisplayFunctions Arduino 1.8.15	
OLEDdsplayfunctions // File: OLEDdsplayfunctions.ino /// Demonstrate a minimumal use case for an Addrruit micro OLED display // Demonstrate a minimumal use case for an Addrruit micro OLED display // main solute the OLED display and OLED display update operations in functions. // Show the current time value from millis() // Libraries meeded for the OLED display #include -Addrruit_GSN.m. // Wire.h provides I2C support #include -Addrruit_SD306.bipset called OLED that is connected by I2C #define OLED_RESET // Nire.h growides is connected by I2C #define GEDENNETH 12K Meest pin 4 (Or - 1 if shading Arduino reset pin) #define SCREENNETH 12K Meest pin 4 (Or - 1 if shading Arduino reset pin) #define SCREENNETH 12K OLED display width in pixels #define SCREENNETH 12K OLED display bright in pixels #define SCREENNETH 12K OLED display the splat in pixels #define SCREENNETH 12K OLED display the splat in pixels #define SCREENNETHEGH 6 4 (VolED display) 12K address is used in setupOLED() //	OLEDdsplayfunctions // File: OLEDdsplayfunctions.ino // Demostrate a minimumal use case for an Adafruit micro OLED display // Demostrate a minimumal use case for an Adafruit micro OLED display // Show the current time value from millis() // - Librarise needed for the OLED display #include during: Needed for the CLED display #include during: Needed for the constant of the micro OLED display // Create an SD0386 object called OLED that is connected by IZC #define OLED, DESET // Neest pin # (Or - 1 if shafing Anduino reset pin) #define SCREEN_MIDH 128 // OLED display width in pixels #define IZADDR 0x30 // IZC address is used in setupOLED() Addruit_SSD186 OLED(SCREEN_MIDH, SCREEN_MEIGHT, & Mire, OLED_RESET); //	🤊 📀 🗈 🖬		
<pre>// File: OLEDdisplayFunctions.ino // Demonstrate a minimumal use case for an Addrruit micro OLED display // and isolate the OLED stup and OLED display update operations in functions. // Show the current time value from millis() // Libraries needed for the OLED display update operations in functions. // Show the current time value from millis() // Libraries needed for the OLED display update operations in functions. // Show the current time value from millis() // Create an SSD1306 object called OLED that is connected by I2C #define OLED_RESET 4 // Reset pin # (or -1 if shorting Arduino reset pin) #define SCREW_HEIDMT 128 // OLED display width in pixels #define SCREW_HEIDMT 64 // OLED display height in pixels #define SCREW_HEIDMT 127 UIC address is used in setupOLED() Addruit_SSD1306 OLED(SCREEN_WEIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); //</pre>	<pre>// File: OLEDdisplayFunctions.ino // Demonstrate a minimumal use case for an Addrnuit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // Show the current time value from millis() // Libraries meeded for the OLED display // Libraries meeded for the OLED display // Libraries meeded for the OLED display // Create an SSD1366 object called OLED that is connected by IZC #define OLED SEST 4 // Krest pin # (OLED display for the micro OLED display // Create an SSD1366 object called OLED that is connected by IZC #define OLED SEST 4 // Krest pin # (OLED display midth in pixels #define SCREEN_WIDTH 128 // OLED display width in pixels #define IZCADOR 0x30 // IZC address is used in setupOLED() Addrnuit_SSD1366 OLEDESCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); //</pre>	OLEDdisplayFunction	ins	
<pre>/// Demonstrate a minimumal use cose for an Adafruit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // ***********************************</pre>	<pre>/// Demonstrate a minimumal use case for an Adafruit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // Choste and SD1306 object colled display // Choste an SD1306 object colled OLED that is connected by IZC // Choste an SD1306 object colled OLED that is connected by IZC // Choste an SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED that is connected by IZC // Choste and SD1306 object colled OLED Choste and another message. // These steps are and internal voltage supply for the OLED display. // If configuration is successful, display the splash screen and another message. // These steps are and internal voltage supply for the OLED display. // Existence of a global Addrivit_SD1306 object colled OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of a global Addrivit_SD1080 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and se</pre>	// File: OLEDo	isplayFunctions.ino	
<pre>// Libraries meeded for the OLED display #include</pre>	<pre>// Libraries needed for the OLED display #include dire.h. // Wire.h provides IZC support #include dadFruit_SSD1306.h. // Units.provides IZC support #include dadFruit_SSD1306.h. // Units.provides IZC support #include dadFruit_SSD1306.h. // Support #include dadFruit_SSD1306.h. // Units.provides IZC support #define GLED_RESET 4 // Reset pin # (or -1 if shan_ng Andino reset pin) #define GLED_RESET 4 // Reset pin # (or -1 if shan_ng Andino reset pin) #define GLED_RESET 4 // Reset pin # (or -1 if shan_ng Andino reset pin) #define GLED_RESET 4 // CLED display height in pixels #define GLED_RESET 64 // OLED display height in pixels #define GLED_RESET 64 // OLED display height in pixels #define GLED_RESET.provides GLED_GLED_RESET); //</pre>	<pre>// Demonstrate // and isolate // Show the cur</pre>	a minimumal use case for an Adafruit micro OLED display the OLED setup and OLED display update operations in functions. rent time value from millis()	
<pre>// Create an SSD1306 object called OLED that is connected by IZC #define OLED_KESET 4 // Reset pin # (or -1 if sharfing Arduino reset pin) #define SCREEN_MIDIH 128 // OLED display width in pixels #define SCREEN_HEIGHT 64 // OLED display height in pixels #define SCREEN_HEIGHT 64 // OLED display height in pixels #define SCREEN_HEIGHT 74 // OLED display height in pixels #define SCREEN_HEIGHT 74 // OLED display height in pixels #define SCREEN_HEIGHT 74 // OLED display. //</pre>	<pre>// Create an SSD1306 object called OLED that is connected by I2C #define OLED_RESET 4 // Reset pin # (or -1 if sharfing Arduino reset pin) #define CREMINDTH 128 // OLED display height in pixels #define SCREMENTH 12 // OLED display height in pixels #define SCREMENTH 12 // OLED display height in pixels #define SCREMENTH 12 // OLED display height in pixels #define SCREMENTH 12 // OLED display height in pixels #define SCREMENTH 12 // OLED display height in pixels #define SCREMENTH 24 // OLED display. // display height in pixels #define SCREMENTH 12 // OLED display. // foorfiguration is successful, display the splath screen and another message. // Trees steps are only needd once at the start of a sketch, and presume the // existence of a global Adafruit_SSD1306 object called OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and sec</pre>	// Libraries #include <wire.h #include <adafru #include <adafru< td=""><td>needed for the OLED display > /// Wine.h provides I2C support it.GFX.h> // Generic graphics library: fonts, lines, effects it.SSD1306.h> // Library for the micro OLED display</td><td></td></adafru<></adafru </wire.h 	needed for the OLED display > /// Wine.h provides I2C support it.GFX.h> // Generic graphics library: fonts, lines, effects it.SSD1306.h> // Library for the micro OLED display	
<pre>#define SCREEN_HEIGHT 64 // OLED display height in pixels #define SCREEN_HEIGHT 64 // OLED display height in pixels #define ISCAMOR 0x30 // ISC address is used in setupOLED() Adefruit_SSD1306 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); // world abop() { // Divoid loop() { // // Set the IZC address and internal voltage supply for the OLED display. // Fee the IZC address and internal voltage supply for the OLED display. // Fee the IZC address and internal voltage supply for the OLED display. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Addrnut_SSD1306 object called OLED. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD1306 object. // This function assumes that OLED is a global Addrnut_SSD130 object. // This function assumes that OLED is a global Addrnut_SSD130 object. // This function assumes that OLED is a global Addrnut_SSD130 object. // This function assumes that OLED is a global Addrnut_SSD130 object. // This function assumes that OLED is a global Addrnut Addrnut</pre>	<pre>#define SCREEN_HEIGHT 64 // OLED display height in pixels #define SCREEN_HEIGHT 64 // OLED display height in pixels #define SCREEN_HEIGHT, SCREEN_HEIGHT, &Wire, OLED_RESET); //</pre>	// Create an #define OLED_RES #define SCREEN_V	SSD1306 object called OLED that is connected by I2C ET 4 // Reset pin # (or -1 if sharking Arduino reset pin) IDTH 128 // OLED display width in pixels	
<pre>#define IZADUR 05:00 // IZL address is used in setupUEU() Addruit_SSD196 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); // // // // // // // // // // // // //</pre>	<pre>#define LXADUM 0530 // 12L address is used in setupOLEU() Addruit_SSD196 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Mire, OLED_RESET); // Byoid setup() { // Woid setup() { //</pre>	#define SCREEN_H	EIGHT 64 // OLED display height in pixels	
<pre>variation content of a state of a state</pre>	<pre>word setup() { // // // // // // // // // // // // //</pre>	Adafruit SSD1306	OLEDCSCREEN WIDTH SCREEN HEIGHT &Wire OLED RESET)	
<pre>//</pre>	<pre>//</pre>	Auti 1 01 C_3301300	occoccentration, sector internet, and e, occoccestry,	
<pre>brow broky () //</pre>	<pre>//</pre>	//		
<pre>// woid loop() { // // Set the IZC address and internal voltage supply for the OLED display. // If configuration is successful, display the splash screen and another message. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Adafruit_SSD1306 object called OLED. // Woid setupOLED() { // // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD130 object. // This function assumes that OLED is a global Adafruit_SSD130 object. // Woid updateOLED(unsigned long tms, unsigned long ts) { //</pre>	<pre>// wid loop({ // Set the I2C address and internal voltage supply for the OLED display. // Set the I2C address and internal voltage supply for the OLED display. // If configuration is successful, display the splash screen and another message. // These steps are only needed ance at the start of a sketch, and presume the // existence of a global AddrivtL_SSD1386 object called OLED. Bvoid setupOLEDC) { // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD183 object. // Woid updateOLED(unsigned long tms, unsigned long ts) { </pre>			
<pre>// Joint toop() { // // Set the IZC address and internal voltage supply for the OLED display. // If configuration is successful, display the splash screen and another message. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Addruit_SSD1306 object called OLED. // display.new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addruit_SSD130 object. // This function assumes that OLED is a global Addruit_SSD130 object. // Use the start of the st</pre>	<pre>// Set the IZC address and internal voltage supply for the OLED display. // Set the IZC address and internal voltage supply for the OLED display. // If configuration is successful, display the splash screen and another message. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Addriut_SSD1306 object called OLED. /// // Display new volues of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a global Addriut_SSD108 object. // This function assumes that OLED is a globa</pre>	//		
<pre>//</pre>	<pre>//</pre>			
<pre>// If configuration is successful, display the splash screen and another message. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Adafruit_SSD1306 object called OLED. // Display new values of system Clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // This function assumes that OLED is a global Adafruit_SSD103 object. // This function assumes that OLED is a global Adafruit_SSD103 object.</pre>	<pre>// If configuration is successful, display the splash screen and another message. // These steps are only needed once at the start of a sketch, and presume the // existence of a global Adafruit_SSD1306 object called OLED. // Total setupOLED() { // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // Woid updateOLED(unsigned long tms, unsigned long ts) { </pre>	//	address and internal voltage supply for the OLED display	
<pre>// These steps are only needed once at the start of a sketch, and presume the // existence of a global Adafruit_SSD1306 object called 0LED. // Woid setupOLED() { // // // // // // // This function assumes that 0LED is a global Adafruit_SSD103 object. // // Woid updateOLED(unsigned long tms, unsigned long ts) { </pre>	<pre>// These steps are only needed once at the start of a sketch, and presume the // existence of a global Addriut_SS01306 object called OLED. /// Vioid setupOLEDC { // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addriut_SS0180 object. // This function assumes that OLED is a global Addriut_SS0180 object. // This function assumes that OLED is a global Addriut_SS0180 object. // This function assumes that OLED is a global Addriut_SS0180 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addriut_SS0180 object. // This function assumes that OLED is a global Addriut_SS0180 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addriut_SS0180 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addriut_SS0180 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in micro OLED is a global Addriut_SS0180 object. // Display new values of system clock in micro OLED is a global Addriut_SS0180 object. // Display new values of system clock in micro OLED is a global Addriut_SS0180 object. // Display new values of system clock</pre>	// If configure	tion is successful, display the splash screen and another message.	
<pre>// Consider the a ground inclusion of gro</pre>	<pre>// Characteristic of a grout multilisebook object carted octo. void setupOLED() { // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addrnit_SSD103 object. // This function assumes that OLED is a global Addrnit_SSD103 object. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds and seconds on the micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in</pre>	// These steps	are only needed once at the start of a sketch, and presume the	
<pre>Woid setupOLED() { // ·/· // Display new values of system Clock in milliseconds and seconds on the micro OLED. /// This function assumes that OLED is a global Adafruit_SSD103 object. /// woid updateOLED(unsigned long tms, unsigned long ts) { /// ·/· /// ·/·/ ·/·// ·/·/ ·/</pre>	<pre>Woid setupOLED() { // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in milliseconds and seconds on the micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new values of system clock in micro OLED. // Display new va</pre>	//	a groot Additure_SSDING object carred ofen.	
<pre>// Display new values of system Clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adofruit_SD103 object. // Would updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>// Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adefruit_SSD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD183 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 object. // This function assumes that OLED is a global Adefruit_SD184 obj</pre>	∃ void setupOLED()	{	
<pre>// Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // alvoid updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>// Display new values of system clock in millisconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // Jooid updateOLED(unsigned long tms, unsigned long ts) {</pre>	//		
∃void updateOLED(unsigned long tms, unsigned long ts) {	<pre>Would updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>// Display new // This functio //</pre>	values of system clock in milliseconds and seconds on the micro OLED. n assumes that OLED is a global Adafruit_SSD103 object.	
	•	• void updateOLED(unsigned long tms, unsigned long ts) {	

- Turn on Code Folding in Preferences for Arduino IDE
- Open ODEdisplay functions
- Notice + and signs near function declarations
- Hover over + sign (do not click) to reveal code



OLEDduphyfunctions // File: OLEDdisployfunctions.ino // File: OLEDdisployfunctions.ino // Jenostrote a minimumal use case for an Adafruit micro OLED display // and isolate the OLED satup and OLED display update operations in functions. // Show the current time value from millis() // (lipraries needed for the OLED display // (reate an SSD1306 object called OLED that is connected by IZC #define OLED.RESET 4 // Reset pin # (O-1 - 1 if sharring Arduino reset pin) #define SCREEN.HEIDHT 128 // OLED display buildth in pixels #define ICZADDR 0x30 // IZC address is used in setupOLED() Adafruit_SSD1306 OLEOCKEEN.HIDTH, SCREEN.HEIGHT, &Wire, OLED_RESET); //		OLEDdisplayFunctions Arduino 1.8.15	
<pre>OLEDdisplayfunctions // File: OLEDdisplayfunctions.ino // Jemonstrate a minimumol use case for an Adafruit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // Show the current time value from millis() // Libraries needed for the OLED display %include -Adafruit_SSD1306.bs // Library for the micro OLED display // Create an SSD1306.object called OLED that is connected by IZC #define OLED.RESET 4 // Reset pi # (or - 1 if sharing Arduino reset pin) #define SCREP.NEIDTH 128 // OLED display beight in pixels #define ICZADDR 0x50 // IZC address is used in setupOLED() Adafruit_SSD1306 OLEDCKEEN_WIDTH, SCREEN_WEIGHT, &Wire, OLED_RESET); //</pre>) 🕘 🗈 🖻		
<pre>// File: OLEDdisplayFunctions.ino // Denostrate a minimumal use case for an Adafruit micro OLED display // and isolate the OLED setup and OLED display update operations in functions. // Show the current time value from millis() // Libraries needed for the OLED display // Libraries needed for the OLED display // Libraries needed for the OLED display // Create an SSD1366 object called OLED that is connected by I2C #define OLED.RESET 4 // Reset pin f (or -1 if sharing Arduino reset pin) #define SCREEN_MIDTH 128 // OLED display width in pixels #define SCREEN_MIDTH 128 // OLED display width in pixels #define SCREEN_MIDTH 128 // OLED display width in setupOLED() Adafruit_SSD1366 OLEO(SCREEN_WIDTH, SCREEN_HEIGHT, &Mire, OLED_RESET); //</pre>	OLEDdisplayFunction	15	
<pre>// Demonstrate a minimumal use case for an Adafruit micro OLED display/ // and isolate the OLED setup and OLED display udate operations in functions. // Show the current time volue from millis() // Libraries needed for the OLED display // Libraries needed for the OLED display finclude dAdafruit_CSCN.b. // Generic graphics library: fonts, lines, effects finclude dAdafruit_CSCN.b. // Generic graphics library: fonts, lines, effects finclude dAdafruit_CSCN.b. // Library for the micro OLED display // Create an SOLI366 object called OLED that is connected by I2C ddefine OLED.RESET 4 // Reset pin # (or -1 if sharing Arduino reset pin) ddefine SCREEN.WIDTH 128 // OLED display height in pixels ddefine GCREEN.WIDTH 128 // OLED display methods in setupOLED() Adafruit_SSDI366 OLEDCSCREEN.WIDTH, SCREEN.HEIGHT, &Wire, OLED_RESET); //</pre>	// File: OLEDdi	.splayFunctions.ino	
<pre>// Libraries needed for the OLED display #Include defire.hs // Wire.h provides IZC support #Include defire.iE, the // Generic graphics library: fonts, lines, effects #Include defire.iE, the fire.h provides IZC support #Include defire.iE, the iteration of the iteration o</pre>	// Demonstrate a // and isolate t // Show the curr	n minimumal use case for an Adafruit micro OLED display che OLED setup and OLED display update operations in functions. rent time value from millis()	
<pre>// Create an SSD1306 object called OLED that is connected by I2C MadFine OLED_RESET 4 // Reset pin # (or -1 if sharing Arduino reset pin) MadFine SCREM_MIDT 122 (// OLED display height in pixels MadFine SCREM_MIDTM 122 // OLED display height in pixels MadFine IZCADMOR 0x50 // IZC address is used in setupOLED() MadFruit_SSD1306 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); //</pre>	// Libraries n #include < <mark>Wire.h</mark> > #include <adafrui #include <adafrui< td=""><td>.eeded for the OLED display . // Wire.h provides I2C support LGFX.h⊳ // Generic graphics library: fonts, lines, effects t_SSD1306.h⊳ // Library for the micro OLED display</td><td></td></adafrui<></adafrui 	.eeded for the OLED display . // Wire.h provides I2C support LGFX.h⊳ // Generic graphics library: fonts, lines, effects t_SSD1306.h⊳ // Library for the micro OLED display	
Addfruit_SSD1306 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET); //	// Create an S #define OLED_RESE #define SCREEN_WI #define SCREEN_HE #define I2CADDR	SD1306 object called OLED that is connected by I2C I 4 // Reset pin ≇ (or -1 if sharing Arduino reset pin) DTH 128 // OLED display width in pixels SIGMT 64 // OLED display height in pixels Øx30 // IZC address is used in setupOLED()	
<pre>// void loop() { void loop() { void loop() { unsigned long timeWillis, timeSeconds; timeWillis = millis(); // Read the system clock, value in milliseconds timeWillis = millis(); // Convert milliseconds to seconds updateOLED(timeWillis, timeSeconds); // Update display in function to keep loop() simple // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) { </pre>	// void setup() {	vicu(Joneen_miuin, Joneen_neiun), amire, Uleu_Neisel);	
<pre>void loop() { unsigned long timeWillis, timeSeconds; timeWillis = millis(); // Read the system clock, value in milliseconds timeSeconds = timeWillis/1000; // Convert milliseconds to seconds updateOLED(timeWillis, timeSeconds); // Update display in function to keep loop() simple // // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addrivit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) { </pre>			
<pre>unsigned long timeHillis, timeSeconds; timeHillis = millis(); // Read the system clock, value in milliseconds timeSeconds = timeHillis/1000; // Convert milliseconds to seconds updateOLEO(timeHillis, timeSeconds); // Update display in function to keep loop() simple // // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Addrnut_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) {</pre>	//		
<pre>timeMillis = millis(); // Read the system clock, value in milliseconds timeSeconds = timeMillis/1000; // Convert milliseconds to seconds updateOUED(timeMillis, timeSeconds); // Update display in function to keep loop() simple // // Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>// void loop() { void loop() {</pre>		
<pre>timeSeconds = timeHillis/1000; // Convert milliseconds to seconds updateOLED(timeHillis, timeSeconds); // Update display in function to keep loop() simple // // Display new values of system clock in milliseconds and seconds on the micro OLED. /// This function assumes that OLED is a global Adafruit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) { </pre>	<pre>//</pre>	meMillis, timeSeconds:	
<pre>updateOLED(timeWillis, timeSeconds); // Update display in function to keep loop() simple //</pre>	<pre>void loop() { void loop() { unsigned long ti timeWillis = mil</pre>	imeMillis, timeSeconds;	
<pre>// Display new volues of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>void loop() { void loop() { unsigned long ti timeMillis = mil timeSeconds = ti</pre>	<pre>imeMillis, timeSeconds; imeMillis, timeSeconds; imeMillis/1800; // Convert milliseconds to seconds</pre>	
<pre>// Display new values of system clock in milliseconds and seconds on the micro OLED. // This function assumes that OLED is a global Adafruit_SSD103 object. // void updateOLED(unsigned long tms, unsigned long ts) {</pre>	<pre>//</pre>	<pre>imeMillis, timeSeconds; llis(); // Read the system clock, value in milliseconds imeMillis/1000; // Convert milliseconds to seconds dillis, timeSeconds); // Update display in function to keep loop() simple</pre>	
/// void updateOLED(unsigned long tms, unsigned long ts) {	<pre>void loop() { void loop() { unsigned long ti timeMillis = mil timeSeconds = ti updateOLED(time } //</pre>	<pre>imeMillis, timeSeconds; llis(); // Read the system clock, value in milliseconds imeMillis/1000; // Convert milliseconds to seconds 4illis, timeSeconds); // Update display in function to keep loop() simple</pre>	
	<pre>void loop() { void loop() { unsigned long ti timeWillis = mil timeSeconds = ti updateOLED(time# } //</pre>	<pre>imeMillis, timeSeconds; imeMillis, timeSeconds; imeMillis/1000; // Convert milliseconds to seconds 4illis, timeSeconds); // Update display in function to keep loop() simple</pre>	
	<pre>// void loop() { void loop() { unsigned long t; timeSeconds = t; updateOLED(time) // Display new v // This function // This function // void updateOLED(u</pre>	<pre>imeMillis, timeSeconds; llis(); // Read the system clock, value in milliseconds imeMillis/1000; // Convert milliseconds to seconds willis, timeSeconds); // Update display in function to keep loop() simple values of system clock in milliseconds and seconds on the micro OLED. vasumes that OLED is a global Addrout_SSI083 object. nsigned long tms, unsigned long ts) {</pre>	
	//	<pre>imeMillis, timeSeconds; lis(); // Read the system clock, value in milliseconds imeMillis/1800; // Convert milliseconds to seconds dillis, timeSeconds); // Update display in function to keep loop() simple values of system clock in milliseconds and seconds on the micro OLED. a global Adafruit_SD108 object. msigned long tms, unsigned long ts) {</pre>	

OLEDdisplayFunctions.ino (header)

// File: OLEDdisplayFunctions.ino

//

// Demonstrate a minimumal use case for an Adafruit micro OLED display

- // and isolate the OLED setup and OLED display update operations in functions.
- // Display the time value returned by millis() in both milliseconds and seconds.

<pre>// Libraries needed for the</pre>	OLED	display
<pre>#include <wire.h></wire.h></pre>	//	Wire.h provides I2C support
<pre>#include <adafruit_gfx.h></adafruit_gfx.h></pre>	//	Generic graphics library: fonts, lines, effects
<pre>#include <adafruit_ssd1306.h></adafruit_ssd1306.h></pre>	11	Library for the micro OLED display

```
// -- Create an SSD1306 object called OLED that is connected by I2C
#define OLED_RESET 4 // Reset pin # (or -1 if sharing Arduino reset pin)
#define SCREEN_WIDTH 128 // OLED display width in pixels
#define SCREEN_HEIGHT 64 // OLED display height in pixels
#define I2CADDR 0x3D // I2C address is used in setupOLED()
```

Adafruit_SSD1306 OLED(SCREEN_WIDTH, SCREEN_HEIGHT, &Wire, OLED_RESET);

OLEDdisplayFunctions.ino (setup)



OLEDdisplayFunctions.ino (loop)



Organize code into separate functions

Keep setup and loop compact

Using functions for discrete tasks is a big advantage when working with more complex codes

setupDisplay and updateDisplay can be reused or used as templates in other sketches







Displaying text and numbers on the OLED

- 1. Code in setupOLED initializes the OLED
 - Start the "oled" object
 - Display message that OLED is starting
- 2. Code in updateOLED
 - Display value in milliseconds, and append "ms"
 - Display "t = ", the value of time in seconds, and " s"

You will adapt updateOLED to other sketches

Steps to display fixed text

- 1. Set text size optional if current size is OK
- 2. Move cursor to a starting position
- 3. Add text to the display buffer with .print method
- 4. Update the display with .display method when butter is finished

```
OLED.setTextSize(1);
OLED.setCursor(0,0);
OLED.print(F("Message"));  NOTE: Fixed strings are
oLED.display();  enclosed in the F(...) macro
```

Steps to display dynamic numbers

- 1. Set text size optional if current size is OK
- 2. Move cursor to a starting position
- 3. Add numerical to the display buffer with .print method
- 4. Update the display with .display method when butter is finished

```
OLED.setTextSize(1);
OLED.setCursor(20,0);
OLED.print(t);
OLED.display();
```





OLED display is a 2D grid of pixels

Origin of the display coordinates is the upper left corner

- x is the horizontal position (long axis), increasing to the right
- y is the vertical position, increasing downward





See: <u>https://learn.adafruit.com/monochrome-oled-breakouts/arduino-library-and-examples</u>



Characters are drawn as bit-maps

Each character is a predefined bit-map, ie. the pixel pattern

- Locate text from upper left corner
- GFX library hands the details

```
OLED.setCursor(3,4);
OLED.print(F("A"));
OLED.display();
```



Image from https://learn.adafruit.com/adafruit-gfx-graphics-library



Practice



Modify OLEDdisplayFunctions.ino

- Add a new variable, timeMinutes to loop, convert seconds to minutes
- ✤ Add timeMinutes as a 3rd argument in the call to updateDisplay
- Modify updateDisplay
 - Add input argument for timeMinutes
 - Add steps to display time in minutes
- Add a display of output from the on-board light sensor
- Add a display of output from the on-board temperature sensor