

PROJECT

Simulated Integer and Float Values from Arduino

Content may change at any time. Please refer to the resource centre for latest documentation.

© 2024 4D Systems. All rights reserved.

Contents

Introduction	3
Requirements	3
Hardware	3
Software	3
Graphics Design	4
Programming the Arduino	7
Running the Project	9
Downloadable Resources	9

Introduction

Arduino's are very common microcontroller boards used to study and design programmable electronics. It is often used with multiple peripherals such as buttons, sliders, sensors and motors.

Together with a TIMI acting as a small fancy display, Arduino boards become a lot more powerful and interesting to use in prototyping.

Mates Studio widgets are mostly capable of displaying 16-bit integer values. In addition to that, the LedDigits widget can also display 32-bit integers and 32-bit float values.

This project showcases a TIMI-96 module controlled by an Arduino Uno to display simulated integer and float values and display these to LedDigits widgets.

Requirements

To proceed with the project, the following are required.

Hardware

- [TIMI-96](#)
- [Mates Programmer](#)
- USB Type A to microUSB cable (for the Mates Programmer)
- USB Type A to Type B cable (for the Arduino, replace as necessary)
- Connecting Wires
- Arduino Uno
- Breadboard

Software

- [Mates Studio](#)
- [Arduino IDE](#)

Graphics Design

Step 1: Open *Mates Studio* and create a *Commander* project for *TIMI-96* with *Reversed Landscape* orientation

SELECT PRODUCT
CLOSE

ALL

TIMI

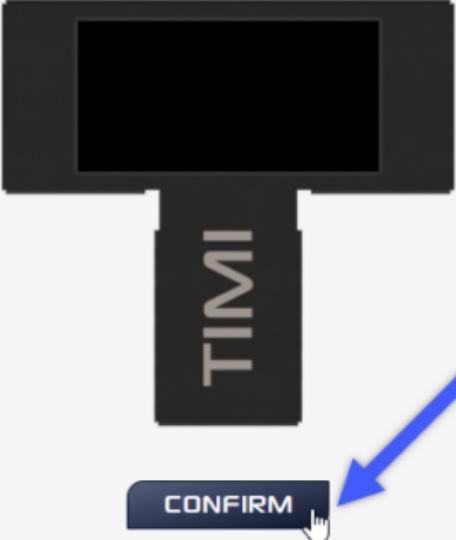
TED

MIHA

REPTOR

TIMI-96	160x80
<i>A 0.96-inch TIMI powered by 4D Labs' Pixxi28 graphics proce...</i>	
TIMI-Click	80x160
<i>A 0.96-inch TIMI for Click interface powered by 4D Labs' Pixxi...</i>	
TIMI-130	240x240
<i>A 1.30-inch TIMI powered by 4D Labs' Pixxi28 graphics proce...</i>	
TED-96	160x80
<i>A 0.96-inch TED powered by 4D Labs' Pixxi28 graphics proces...</i>	

Click Image to Rotate




CONFIRM


Browse Recent Projects
Browse Computer

SELECT ENVIRONMENT
BACK


Commander




The Commander environment enables the user to create projects by selecting page layouts from a selection of predesigned user interfaces from Breadboard Mates team and community.



The Architect environment enables the user to design projects with custom pages and widgets. This gives more designing capabilities than the Commander environment.



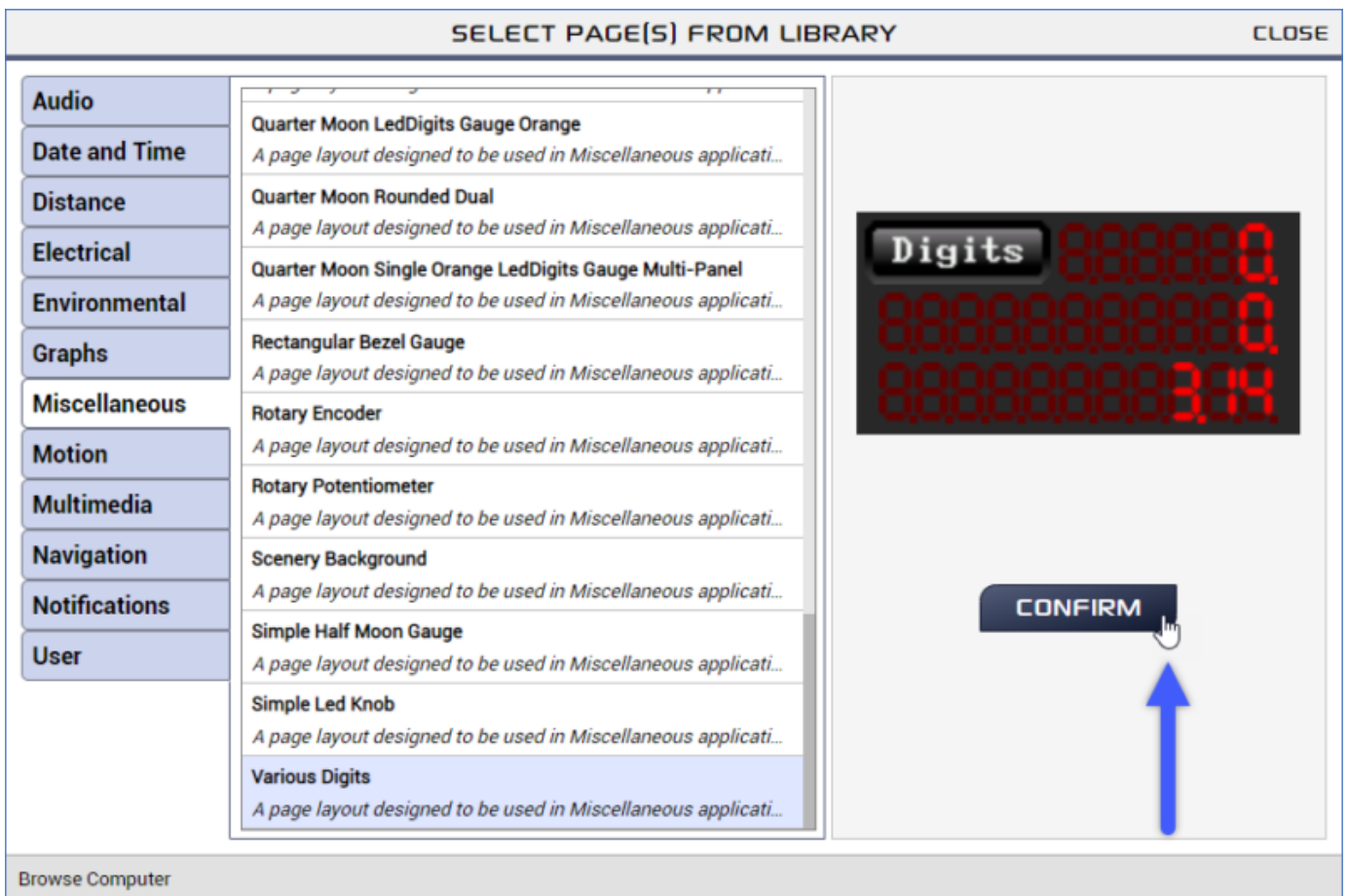
The Genius environment enables the user to design projects with custom pages and widgets and write code. This removes the need for an external host to control with the display.



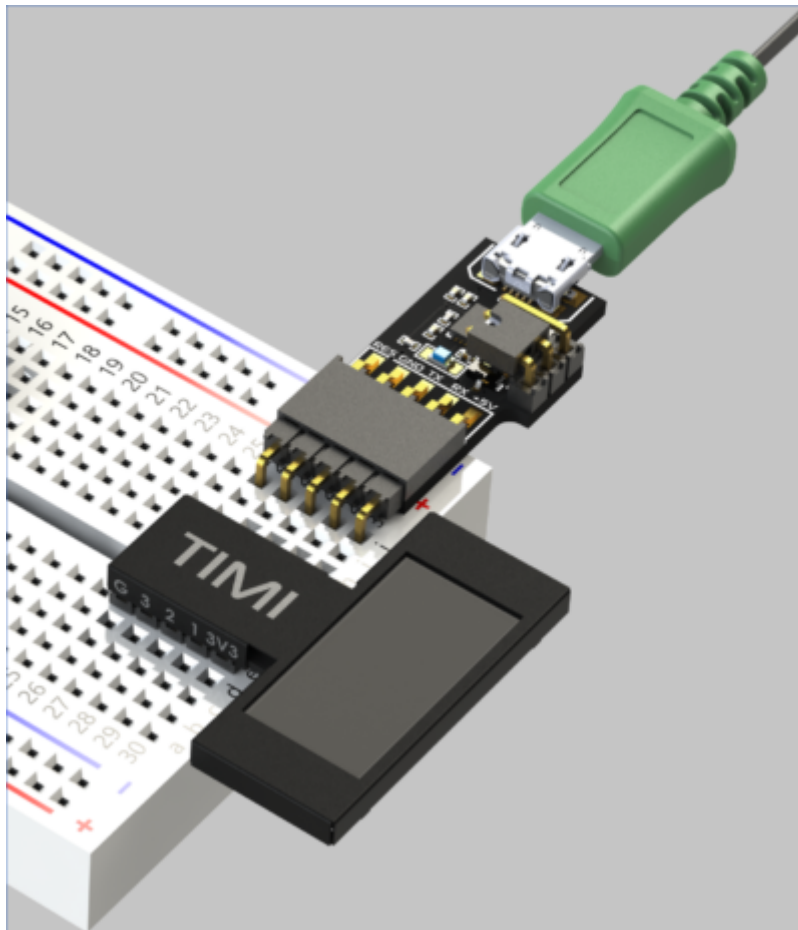
The Builder environment enables the user to design projects with custom pages and widgets and build the process flow using graphical/block programming. This removes the need for an external host to control with the display.

Browse Recent Projects
Browse Computer

Step 2: Browse the library for appropriate page designs. For this project, *Various Digits* page under *Miscellaneous* category was used.



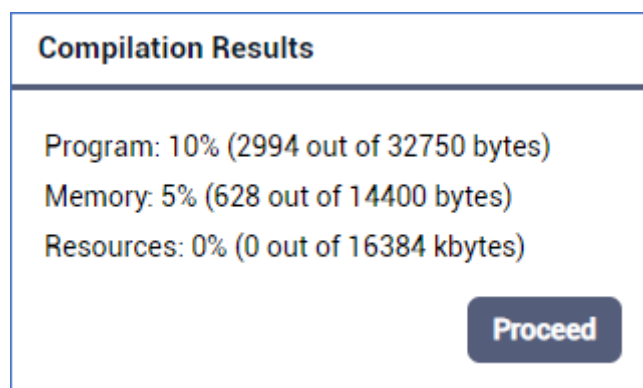
Step 3: After finalizing the design, connect TIMI-96 to your computer



Step 4: Upload the project to the appropriate COM port



Step 5: When prompted, click *Proceed* to continue with upload.

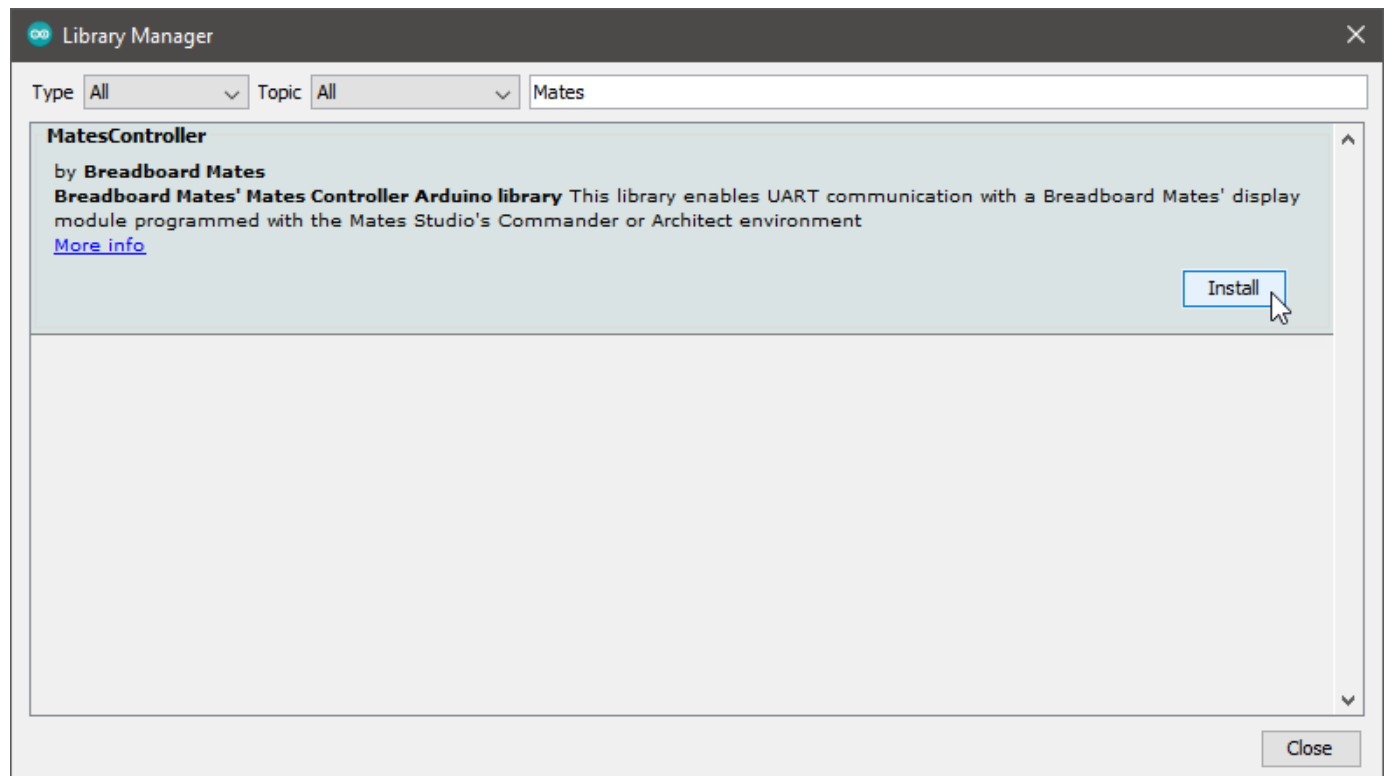


Note

It is recommended that the graphics design is finalized before moving to the next steps when working on a project

Programming the Arduino

Step 1: Install the *MatesController* library using Arduino's *Library Manager*.



Step 2: Include *MatesController.h* to your project.

```
#include "MatesController.h"
```

Step 3: Create a *MatesController* instance named *mates*.

```
MatesController mates = MatesController(Serial);
```

This will initialize the *MatesController* instance to the default reset pin 4 using a LOW pulse.

Step 4: (Optional) Create a function for toggling the built-in LED of the Arduino board. This can be used for debugging or showing errors if the Serial monitor can't be used.

```
int errLedStatus = LOW;
void ErrorLed_Toggle() {
    errLedStatus = ~errLedStatus;
    digitalWrite(LED_BUILTIN, errLedStatus);
}
```

Step 5: (Optional) At the beginning of the setup function, set the built-in LED pin to OUTPUT and set it to LOW.

```
pinMode(LED_BUILTIN, OUTPUT);
digitalWrite(LED_BUILTIN, errLedStatus);
```

Step 6: To start using the MatesController instance, use the `begin` function

```
mates.begin();
```

This will initialize the Serial UART at the default baudrate of 9600

Step 7: (Optional) The `begin` function can be enclosed in an if condition to handle initialization errors.

```
if (!mates.begin()) {  
  // Display didn't send ready signal in time  
  while (1) {  
    ErrorLed_Toggle();  
    delay(100);  
  }  
}
```

Step 8: Create variables for 16-bit and 32-bit integer, and 32-bit float.

```
int16_t value = 0;  
int32_t longValue = 100000;  
float floatValue = 3.1416;
```

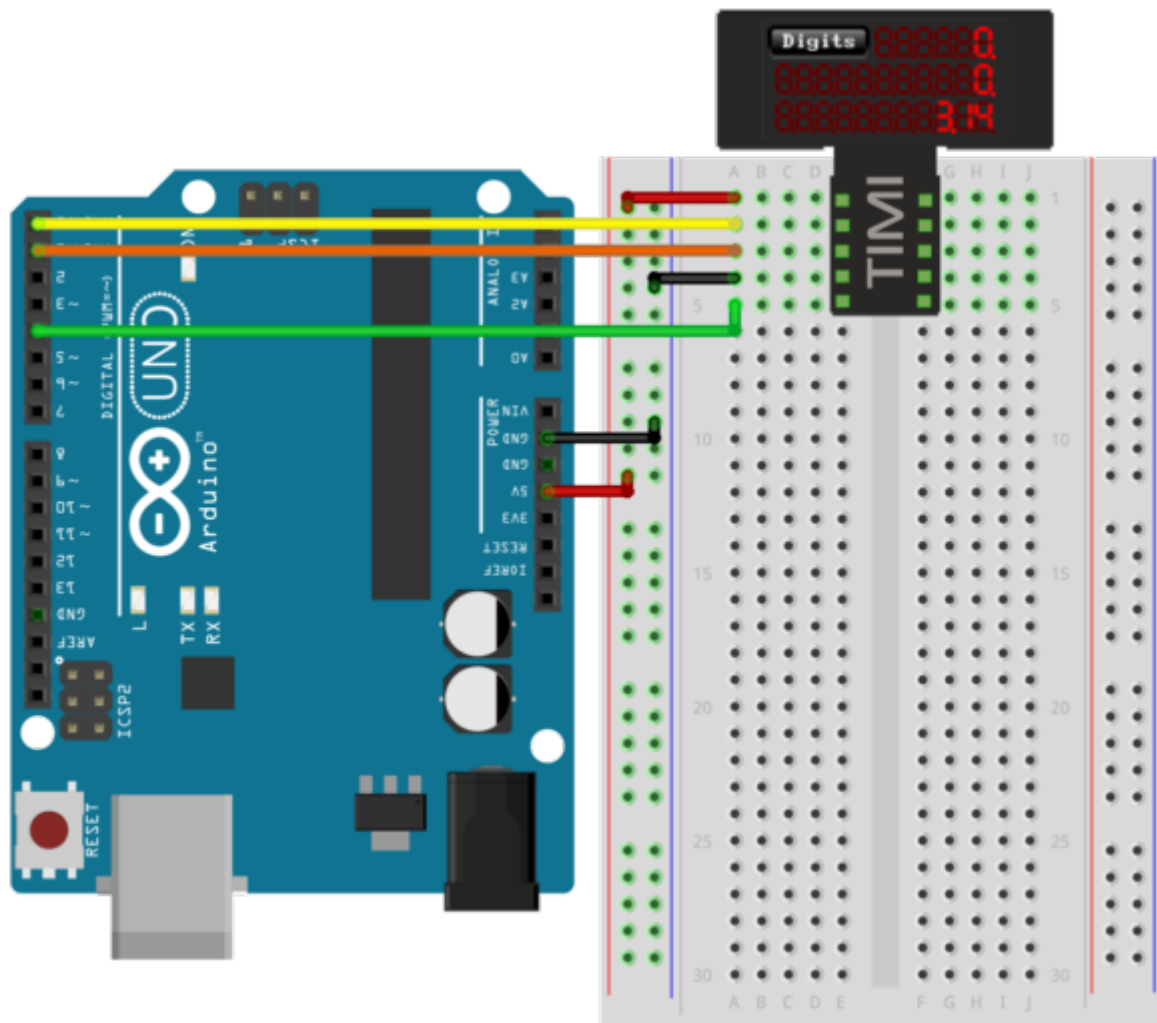
Step 9: In the loop function, the values are simulated and sent to TIMI as necessary.

```
void loop() {  
  // mates.setWidgetValue(MATES_LED_DIGITS, 0, value);  
  mates.setLedDigitsValue(0, value);  
  mates.setLedDigitsValue(1, longValue);  
  mates.setLedDigitsValue(2, floatValue);  
  
  value++;  
  longValue += 12345;  
  floatValue += 3.1416;  
}
```

As shown, the code simply increments each simulated value a certain amount every iteration and writes it to LedDigits widget.

Running the Project

After designing the user interface for TIMI and writing code for the Arduino and programming them, it is time to connect the devices together. Follow the diagram below for the connection between TIMI and Arduino.



Finally, supply power to the Arduino and observe the behavior of the project.

Downloadable Resources

The Mates Studio – Commander project and Arduino sketch is included in the MatesController library.

The Commander project is available under the extras folder of the library. You can find it in (if the library was installed using Arduino Library Manager):

C:\Users\%USERNAME%\Documents\Arduino\libraries\MatesController\extras\Various Digits.mates

Here are the links to the software applications, libraries and completed project files.

- [Mates Studio](#)
- [Arduino IDE](#)
- [Arduino Mates Controller Library](#)
- [Project Files](#)