

PROJECT

Time and Day of Week from Arduino RTC



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	11
Downloadable Resources	11

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.

This project showcases a TIMI-96 module controlled by an Arduino Uno to display time of day and day of week information from an RTC module.

Requirements

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)
- DS3231 RTC Module
- 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

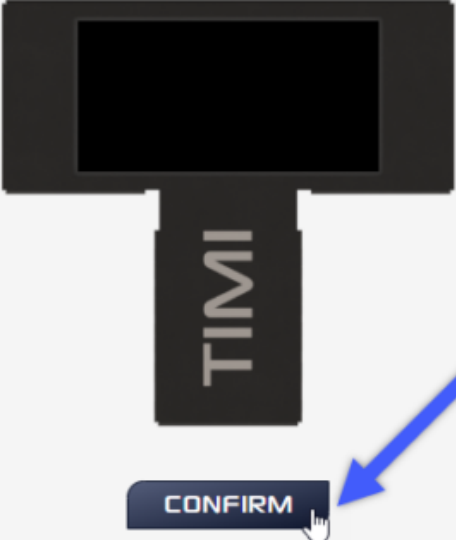
TED

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




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 predefined 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.



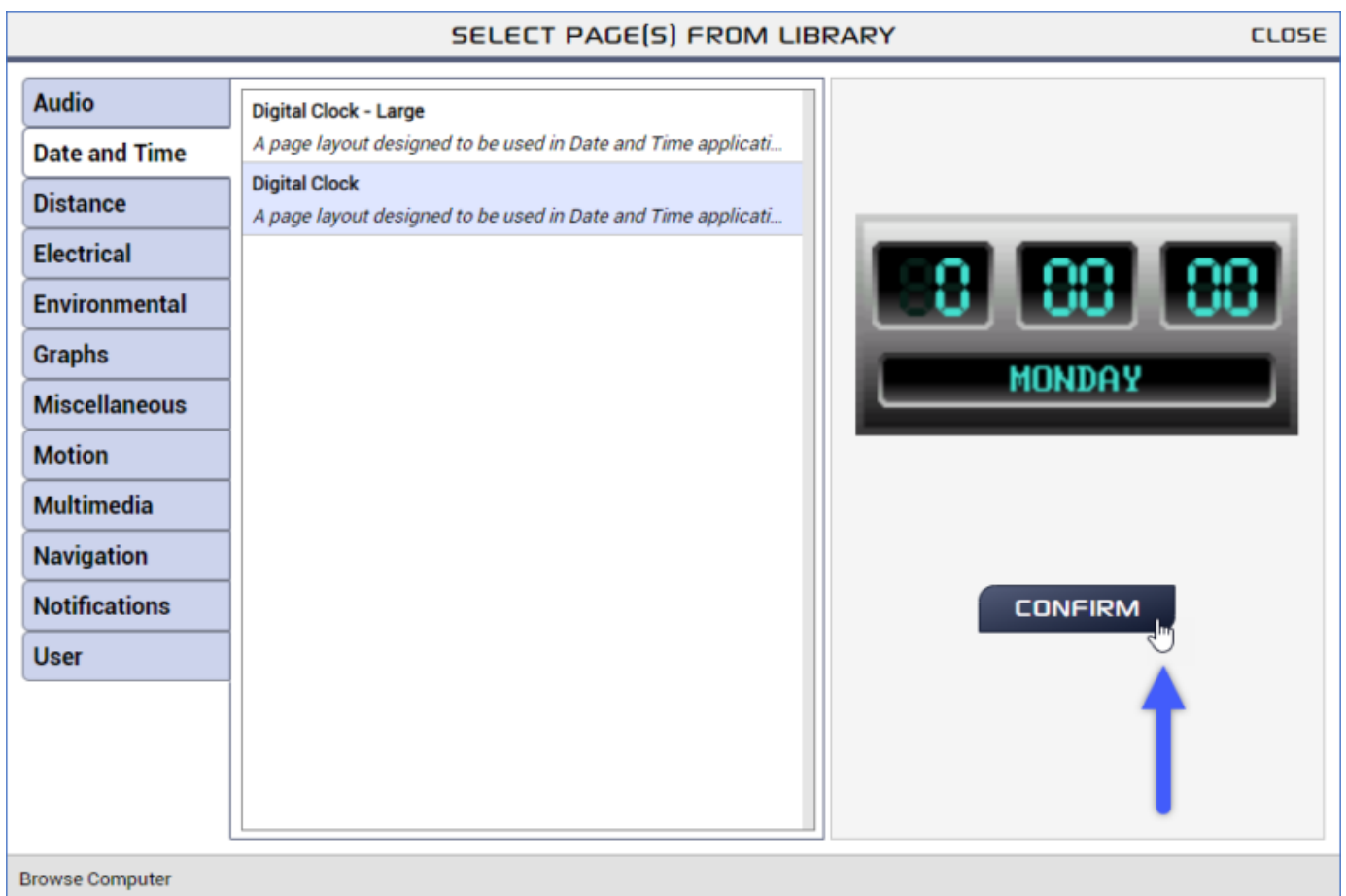
The Genius environment enables the user to design projects with custom pages and widgets and write code. This removes the need for an external host to control with the display.



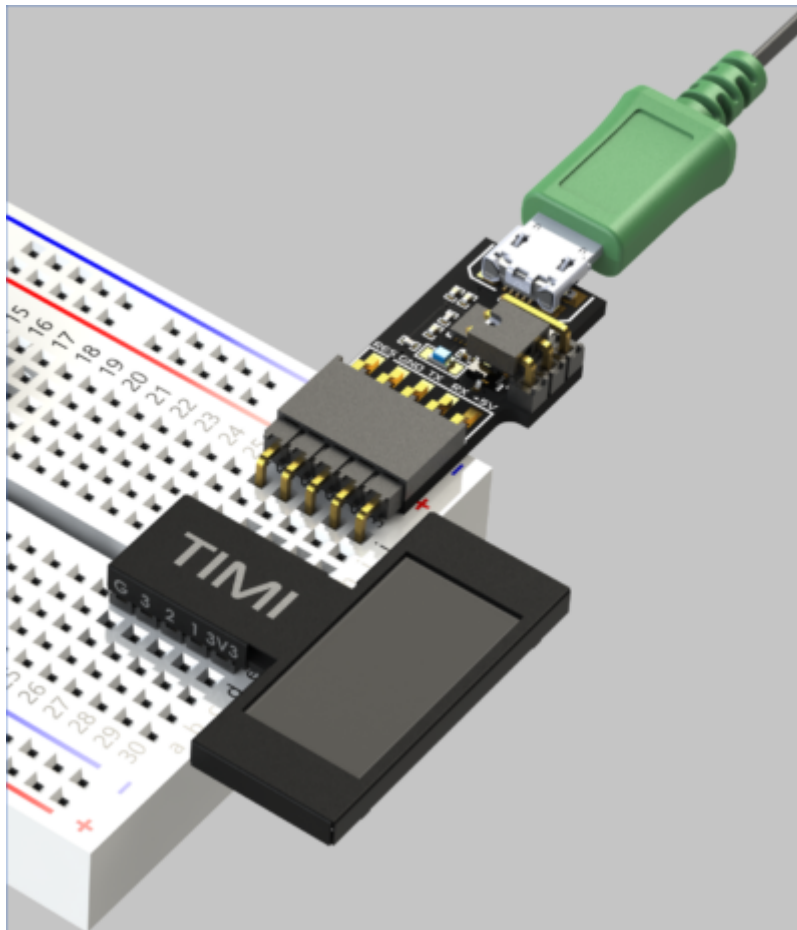
The Builder environment enables the user to design projects with custom pages and widgets and build the process flow using graphical/block programming. This removes the need for an external host to control with the display.

Browse Recent Projects
Browse Computer

Step 2: Browse the library for appropriate page designs. For this project, Digital Clock page under Date and Time category was used.



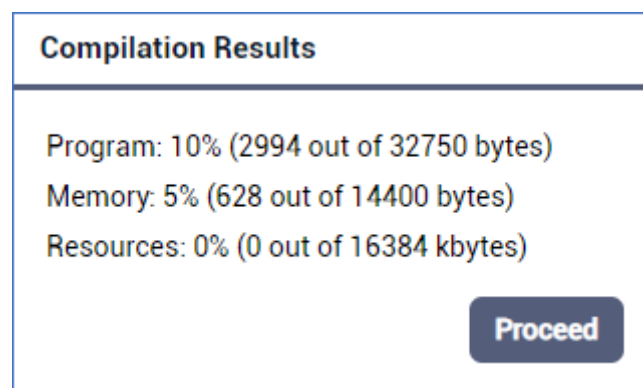
Step 3: After finalizing the design, connect TIMI-96 to your computer



Step 4: Upload the project to the appropriate COM port



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

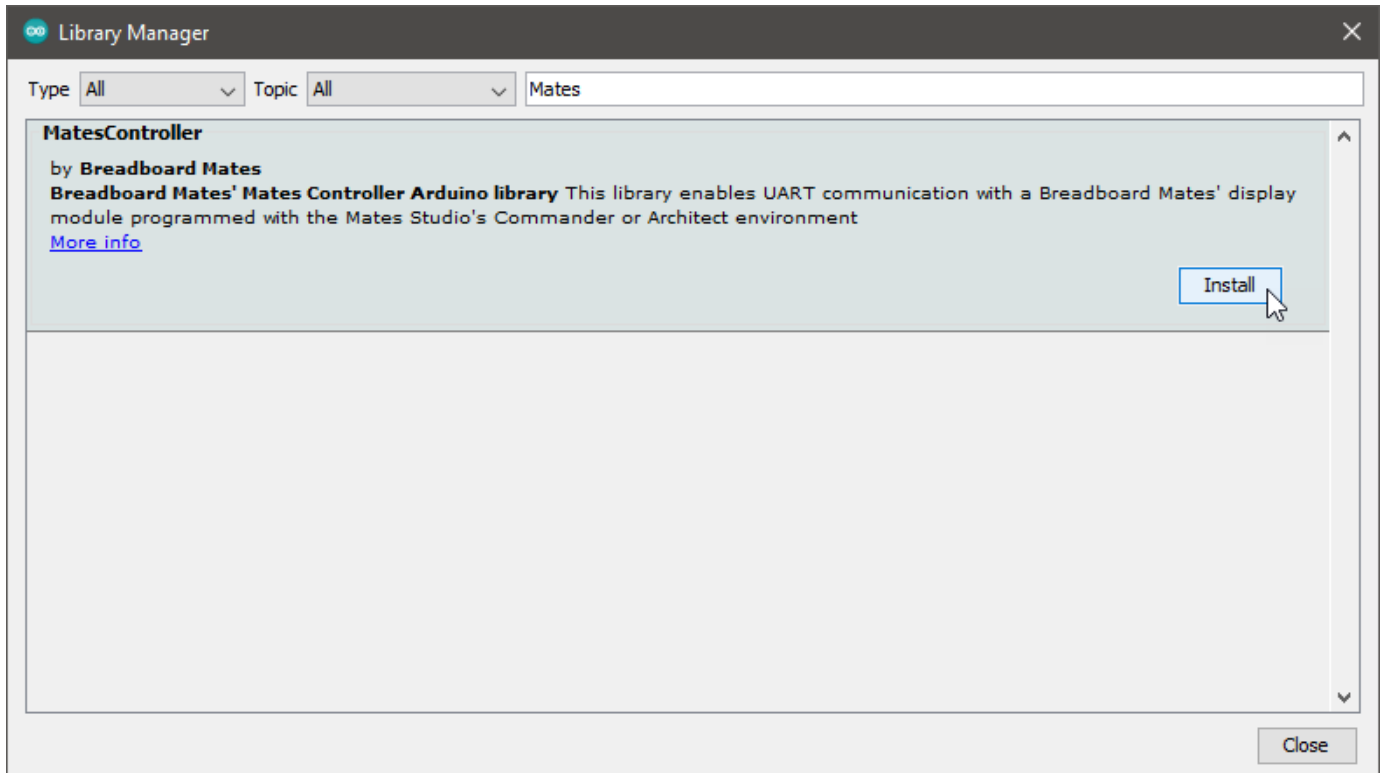


 **Note**

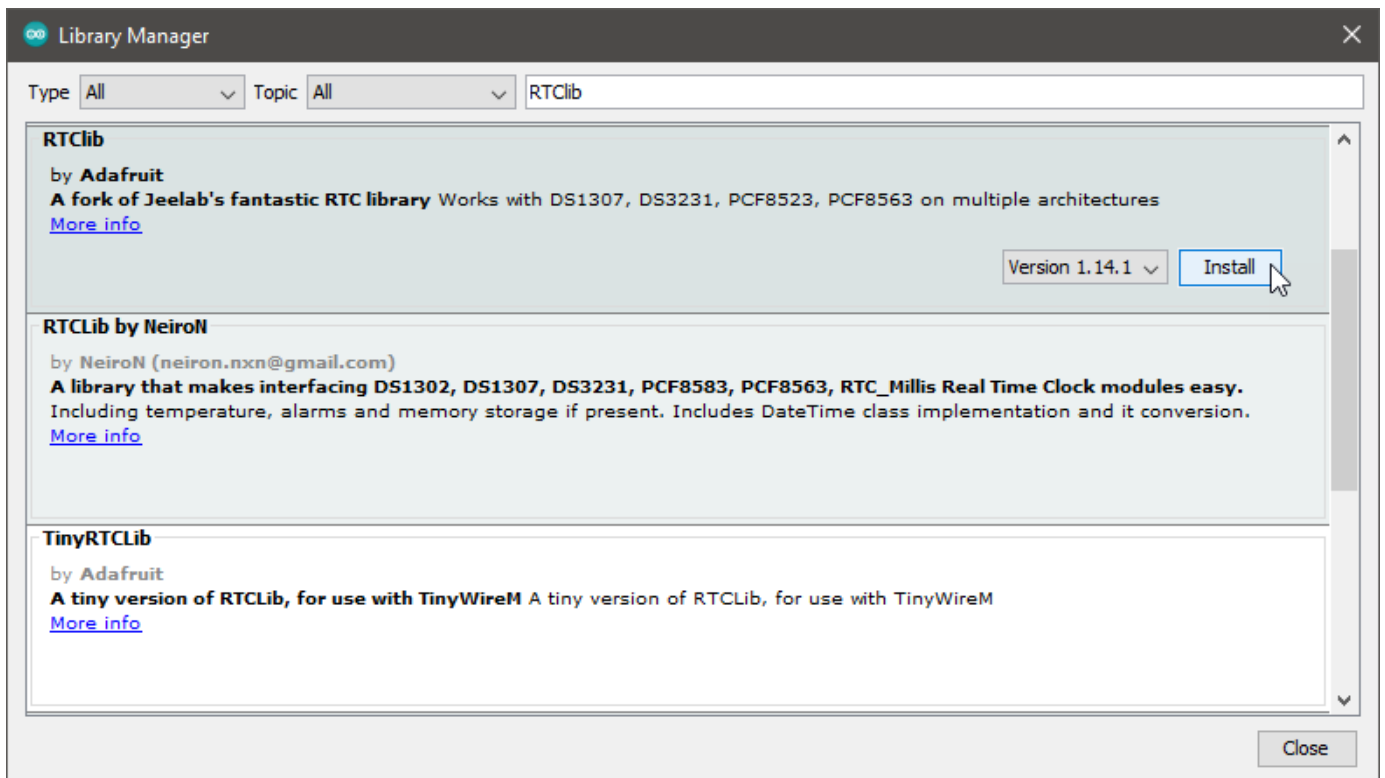
It is recommended that the graphics design is finalized before moving to the next steps when working on a project.

Programming the Arduino

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



Step 2: Install the RTCLib library using Arduino's Library Manager.



Step 3: Include `MatesController.h` and `RTCLib.h` to your project.

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

Step 4: Create a `MatesController` instance named `mates` and an `RTC_DS3231` instance named `rtc`.

```
MatesController mates = MatesController(Serial);
RTC_DS3231 rtc;
```

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

Step 5: (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 6: (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 7: 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 8: (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 9: To print the day of week, an array of strings needs to be prepared.

```
// Days of Week Strings  
const char * daysOfWeek[] = {  
  "SUNDAY",  
  "MONDAY",  
  "TUESDAY",  
  "WEDNESDAY",  
  "THURSDAY",  
  "FRIDAY",  
  "SATURDAY"  
};
```

Step 10: In the loop function, the time and day values are read from the RTC and sent to TIMI as necessary.

```
void loop() {
  static int8_t lastDay = -1;

  DateTime now = rtc.now();

  int8_t dy = now.dayOfTheWeek();
  int16_t hr = now.hour();
  int16_t mn = now.minute();
  int16_t sc = now.second();

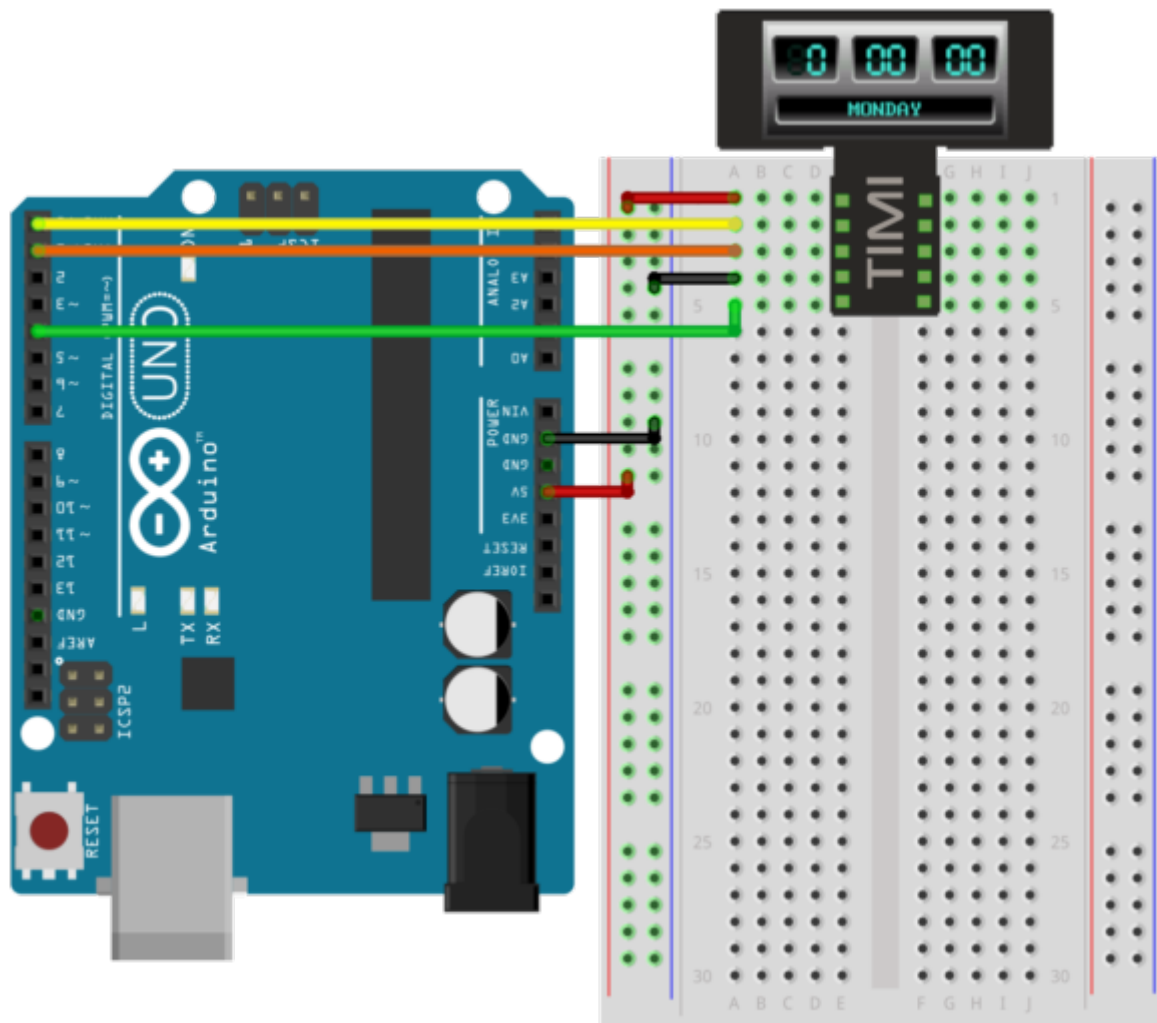
  mates.setLedDigitsValue(0, hr);
  mates.setLedDigitsValue(1, mn);
  mates.setLedDigitsValue(2, sc);

  if (lastDay != dy) {
    mates.updateTextArea(0, daysOfWeek[dy]);
    lastDay = dy; // prevents writing the same text to TextArea
  }
}
```

As shown, the RTC module is read every loop with the day of week and time values. The widgets are then updated accordingly.

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

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

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