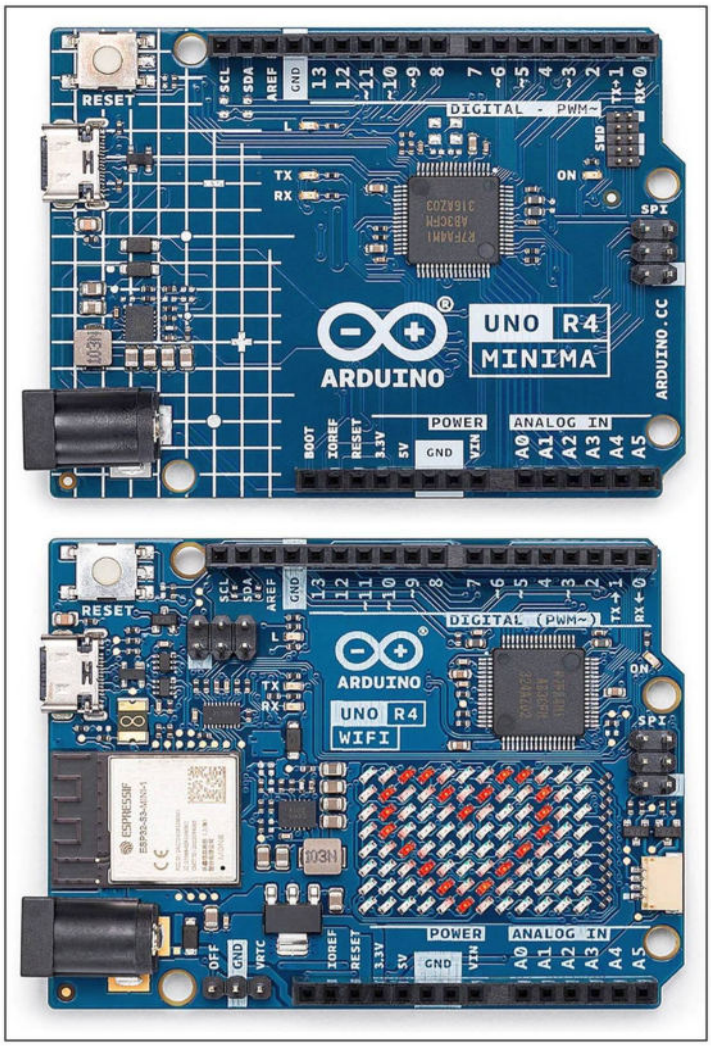
RADIO MAGiC
Robert C. Mazur, VA3ROM | www.va3rom.com | va3rom@gmail.com


*First published in the May-June 2024 issue of the Canadian Amateur*

# First Look: The Arduino Uno R4

**Figure 1: Arduino Uno R4 Minima and WiFi**
Credit: Arduino.




## Introduction

A while back, I wrote two articles about the Arduino Uno R3 in “Part Uno” (TCA Nov-Dec 2014) and “Part Duo” (TCA Jan-Feb 2015) so you can refer to them for the history of and introduction to the Arduino Uno microcontroller. As to why it took the company nearly a decade (a lifetime in the tech world) to make major changes is a mystery to me, but perhaps it was simply a case of “If it is not broken then do not ‘mess’ with it.” Last summer, and at long last, the Italian company introduced not one but two new Arduino Unos: the R4 “Minima” and R4 “WiFi” (Figure 1).

To some, they are a belated but somewhat crippled upgrade while others feel they are a leap forward, at least for the Uno brand. Both R4 versions are 100% backwards compatible with code written for the R3 and have the same form factor so they accept the same plug-in “shields”, but that is where the similarities end.
The programmer’s integrated development environment (IDE) was redesigned to utilize the new capabilities of the R4 in combination with the Arduino cloud IDE (“editor”) and internet-of-things (IoT) platforms. Additionally, the WiFi version supports Bluetooth low-energy (BLE) and Bluetooth 5, has an onboard programmable 12x8 surface mount device (SMD) light emitting diode (LED) matrix, plus the WiFi microprocessor (EXPRESSIF ESP32 microprocessor) shares some of the programming load with the Renesas RA4M1 microcontroller.

But it is not perfect. For some reason there is no onboard micro-SD card support. It has a horrid software controlled real-time clock (RTC), albeit the WiFi version with internet access can use the network time protocol (NTP) to keep the RTC in sync. For some reason, Arduino added an onboard SMD digital thermometer but no humidity or pressure sensors to go with. I am disappointed that after 10 years this is the best that Arduino could come up with as compared to its clone competitors (Diligent, Adafruit, SparkFun, Seeed Studio et al). There is so much room on the Minima board because of SMD tech that it could have been an awesome multiple sensor, telemetry gathering and data logging device, in my opinion.

The R4 now limits current sourcing to eight milliamps (ma) and sinking to four ma per pin (Figure 2, left image, explains the difference) – a huge reduction from the R3s 20 ma per pin maximum. Arduino feels that modern microcontrollers are not designed to be current power supplies so if you need more, you will have to use a transistor or some other solid-state device to provide it off-board. Otherwise use a 1000-ohm current limiting resistor in series with any standard LEDs to keep the source/ sink current within their maximums. An inexpensive and handy to have LED tester let us you see just how bright any colour of LED will look to your eyes using various drive currents (Figure 2, next page, right image).

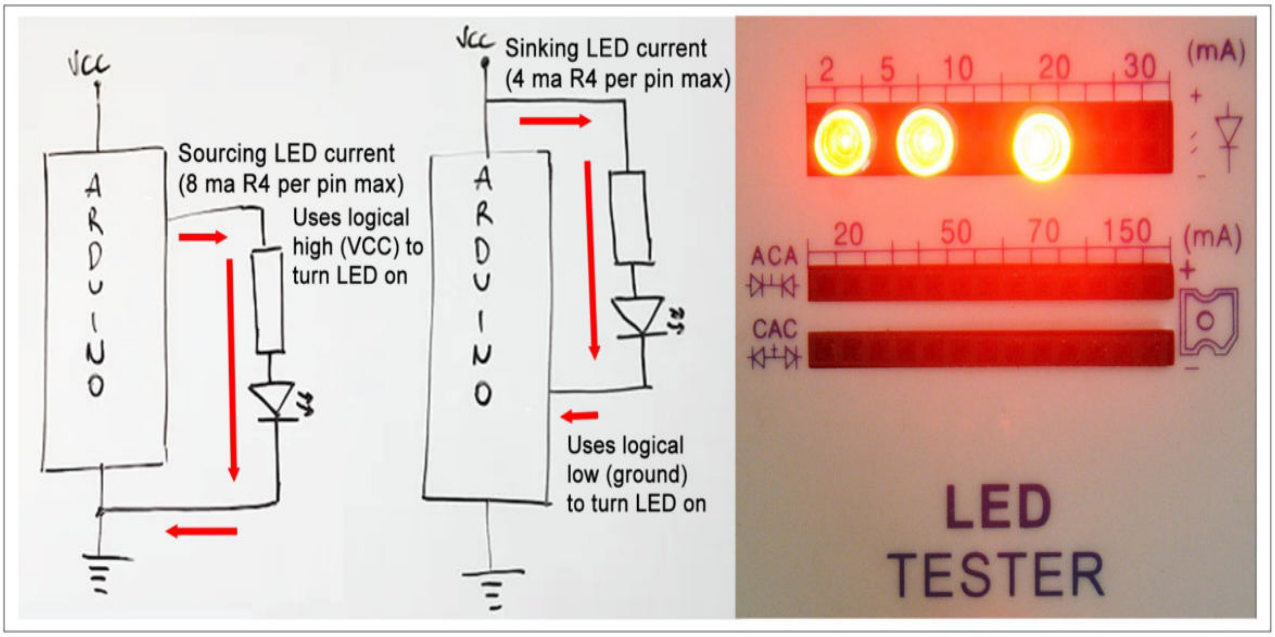



**Figure 2: Current Sourcing versus Sinking plus Comparing LED Brightness**
Microcontroller current sourcing versus sinking (left) explanation plus how clear top LEDs (same manufacturer and batch) appear to the camera/eye using three different drive currents: 4, 8 and 20 ma. Credit: Instructables.com for the sinking versus sourcing diagram.

## A Tale of Two Unos

All-in-all, the R4 is powerful 32-bit processor with four times the speed with far more programming and data storage memory, meaning that it can more easily do things in the digital signal processing (DSP) radio world that the R3 simply cannot (Table 1). I built and programmed my telemetry-over-WSPR transmitter (TCA Sep-Oct and Nov-Dec 2022) using the venerable R3, but had to leave out some features that can be implemented using the R4. The R4 has more efficient 5-volt and 3.3-volt power supplies generating less heat and less noise, plus the board also handles a wider range of external power supply voltages. A USB-C cable (replacing USB-B) is used for board power, programming and serial input/output with a host computer and external serial devices. USB-C has become the de facto standard for most new electronic devices using USB because it physically plugs in any which way and is less bulky.

| Feature | Arduino Uno R3 | Arduino Uno R4 |
|---|---|---|
| Processor | ATmega328P | Renesas RA4M1 |
| Word length (bits) | 8 | 32 |
| Clock speed | 16 MHz | 48 MHz |
| SRAM | 2 kB | 32 kB |
| Flash memory | 32 kB | 256 kB |
| EEPROM | 1 kB | 8 kB |
| Operating voltage | 5 volts | 5 volts |
| Timers | 3 | 10 |
| Capacitive touch sensing | None | 27 channels |
| Temperature sensor | None | 1 |
| USB connector | USB-B | USB-C |
| ADC | 10-bit | 14-bit |
| DAC | None | 12-bit |
| SPI | 1 | 1 |
| I2C | 1 | 2 |
| Qwiic I2C (3.3 volts) | None | 1 (WiFi version) |
| WiFi | None | WiFi version |
| RTC | None | 1 |
| HID | None | Yes |
| SWD debug | None | 1 |
| Bluetooth | None | WiFi version |
| CAN bus support | None | 1 |
| Op Amp (4.7 volts) | None | 1 |
| 12x8 LED matrix | None | WiFi version |
| External power supply | 7 to 12 volts | 6 to 24 volts |

**Table 1: Arduino Uno R3 versus R4**

Credit: Arduino.

Arduino finally added digital-to-analog (DAC) capability to the Uno and the R4 easily generates (12-bit resolution) analog audio frequency waveforms (sine, sawtooth and square) with only a few lines of code. It also supports the musical instrument digital interface (MIDI) so you can create a full-blown 88-key piano MIDI synthesizer, emulate human voice and/or create special sound effects, etc. Renesas has done an excellent job of developing software libraries to make what would normally be difficult coding for most of us so much easier.

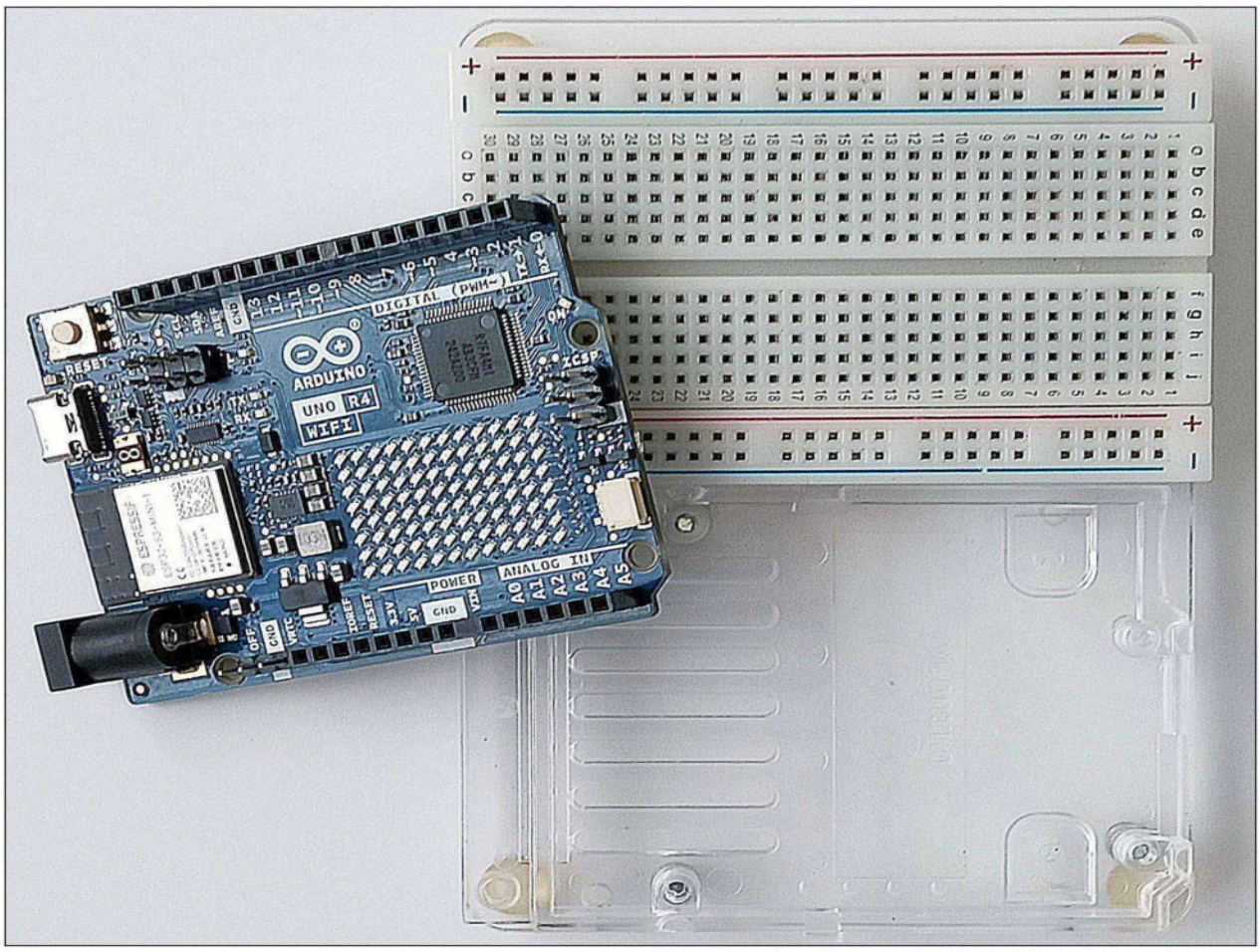
*Note: Because the price difference between the R4 Minima and WiFi boards is only about $10 Canadian, I did not bother buying the Minima. The rest of this article will focus mainly on the WiFi version unless noted otherwise.*

**Uno Mount and Solderless Breadboard**

The R4 comes with a useful snap-in/out acrylic cradle you can permanently attach to a standard Uno acrylic mounting plate with a half-sized solderless (stick-on) breadboard (Figure 3). It also fits the R3 so you can swap any Uno board in/out easily. There are many ways to create an Arduino microcomputer prototyping board but I prefer this method for most projects and ancillary circuits. Robot Shop Canada (https://ca.robotshop.com) and Universal Solder (https://universal-solder.ca) are my two main supply sources.

**Figure 3: My Uno Mount and Solderless Breadboard Option**

The clear acrylic cradle that comes with the R4 is a great accessory for quickly swapping boards.



**Machine Learning**

Both R4s support embedded machine learning (ML) in combination with, for example, Edge Impulse (https://edgeimpulse.com) which is free for non-commercial use. One avid Argentine football (soccer) fan uses his R4 Minima to analyze the FIFA football team statistics database to create a "fun sports machine learning prediction project" (http://tiny.cc/f4ecxz). Or we could use a variation to monitor and learn our driving habits by building a custom database as we drive, and then use machine learning to study that data and so on. The onus was always on the human coder to figure out all possible conditions and then the proper actions required.

**CAN-BUS, CAN-DO**

Another interesting addition is support for the controller area network bus (CAN, CAN-BUS or CAN BUS). It is an internationally standardized, bidirectional, serial data networking protocol that uses small gauge, end-terminated (120-ohms) twisted wire pairs that can create a long daisy-chain of nodes called electronic control units (ECUs). It uses packet radio with telemetry frames consisting of message identification, data and cyclic redundancy check (CRC) and requires an acknowledgement (ACK) from its host microcontroller to ensure the data is received understood and acted upon (Figure 4, next page).

The CAN-BUS is used in modern robotic machinery, automobiles, other industrial vehicles like trucks and tractors, commercial and military aircraft, etc. It can prioritize critical ECU data in life-or-death emergency situations and take autonomous action – when required if we do not. It is a throwback to the past and is simple, serial, inexpensive, easy to implement and maintain plus extremely reliable. Most importantly of all, it is virtually immune to electromagnetic interference (EMI) meaning that our smartphones or much higher-powered Amateur Radio transmitters will not jam/ scramble CAN-BUS signals of say a vehicle's autopilot, lane drift and collision avoidance systems, or suddenly slam on brakes or deploy airbags just because we are chatting up the local repeater!

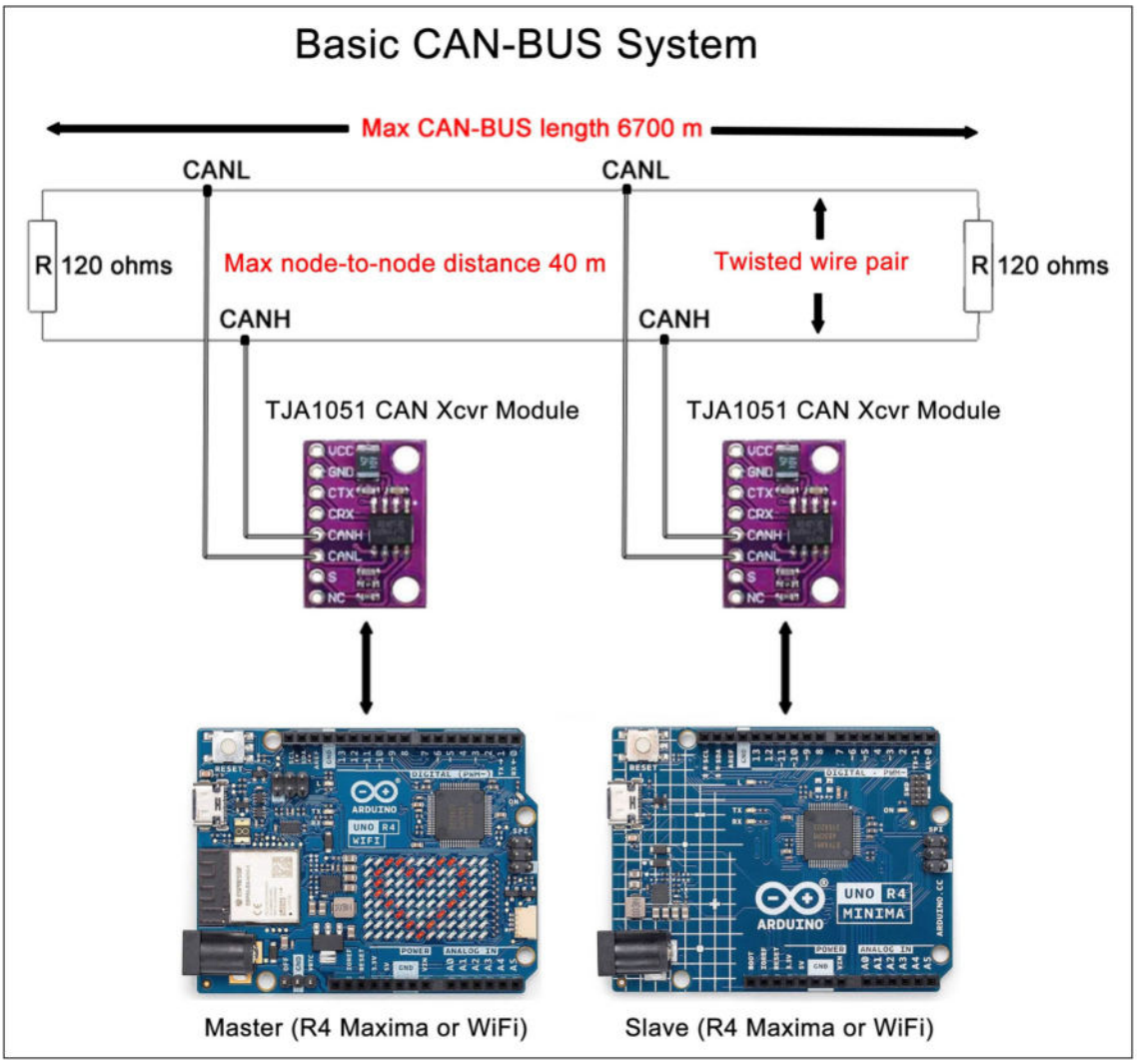



**Figure 4: The Arduino CAN-BUS Variant**

Credit: Arduino.

## The Arduino IDE, Cloud Tools and IoT Support

The latest standalone Arduino IDE is version 2.3.x and it is still my primary programming platform because you can use it standalone without needing internet access. It supports the R4 plus all Arduino compatible boards and clones. It even looks similar to and functions the same as the older IDE version 1.8.x but has several nice additions. Updating boards and libraries are now easier with just one click of a mouse button (or manually and selective, if you prefer) plus the serial monitor and plotter (with optional interpolation) can both run at the same time.

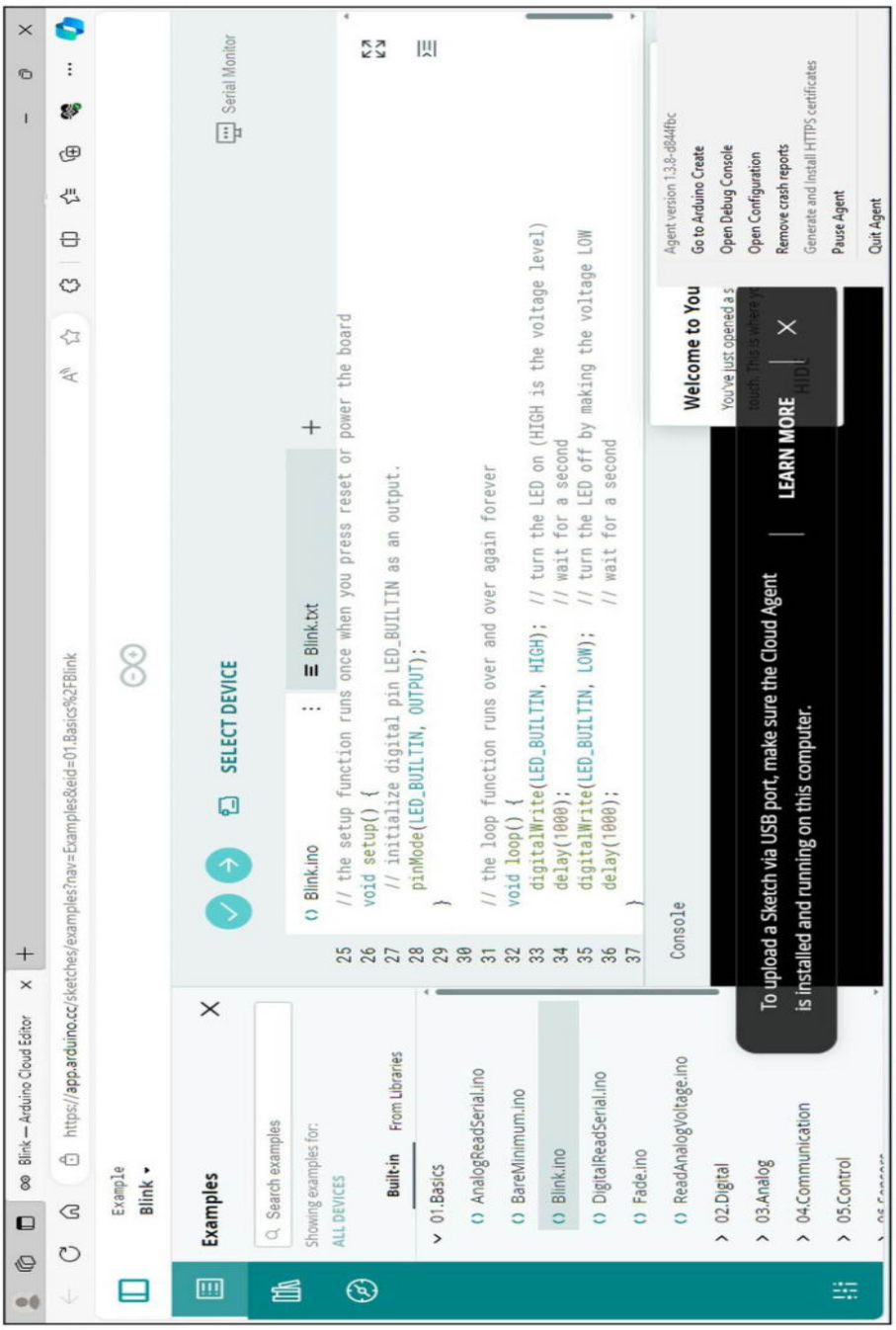


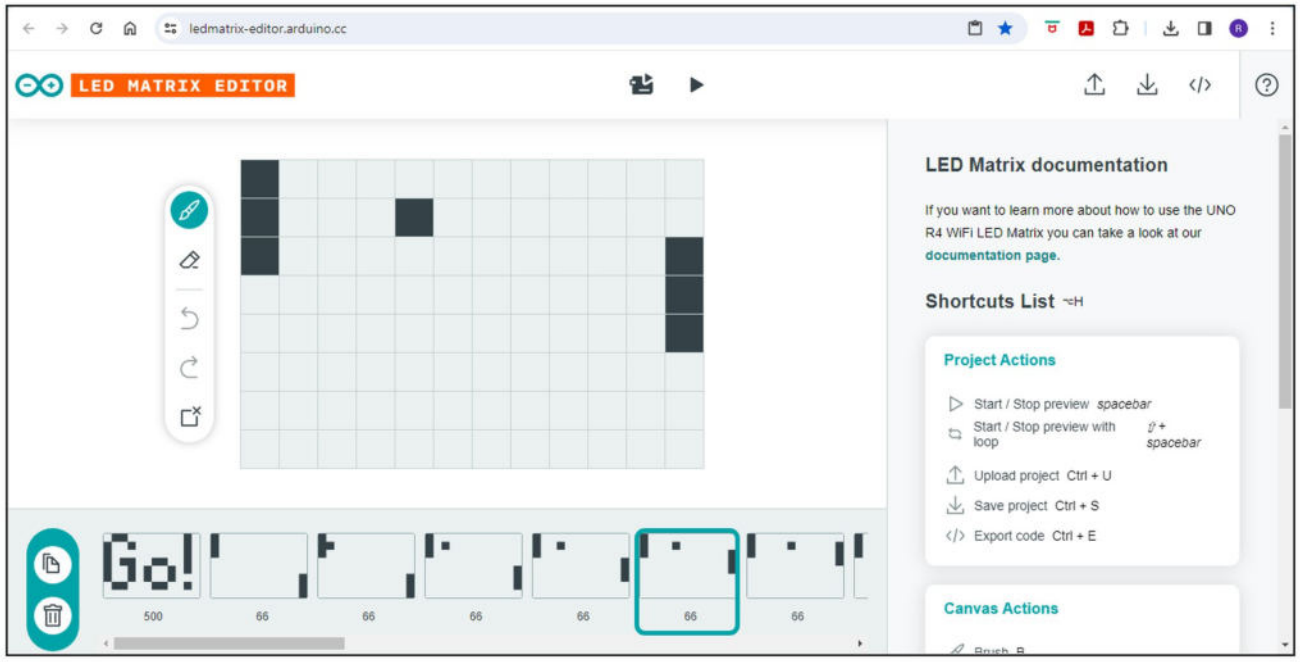

**Figure 5: Arduino Cloud Code Editor**
Here is literally everything, everywhere, all at once at your fingertips.
Credit: Arduino.

Arduino cloud-based programming projects (solo/group) as well as internet of things (IoT) setup is done using a web Google Chrome (recommended) browser (Figure 5). There are many online tutorials and other tools that work hand-in-hand with the Arduino web editor. To use it, you need to set up a free Arduino cloud/IoT account https://cloud.arduino.cc or for groups/schools https://cloud.arduino.cc/schools then download/install the free Arduino "Agent".

It interfaces the Arduino cloud with the Uno connected to your computer so you can write code using the cloud editor then download it the Uno via the Agent. There is a free basic account and three other paid plans that allow for more connected IoT "things" (devices), more cloud storage, longer data retention periods plus dashboard control sharing with others.

The Arduino cloud has excellent tutorials to teach you how to use the new features of the R4, cloud editor and IoT programming. There is also a cloud LED matrix editor for designing your own 12x8 custom graphics, sprites, glyphs and icons (Figure 6).

**Figure 6: Arduino Cloud LED Matrix Editor**
My simple, multi-frame animation to which DAC sound effects could be added. Numbers at the bottom are each frame's display time in milliseconds (default 66 milliseconds). You can download matrix designs as Arduino #include files or download/upload them as ". mpj" (matrix project JSON) files.



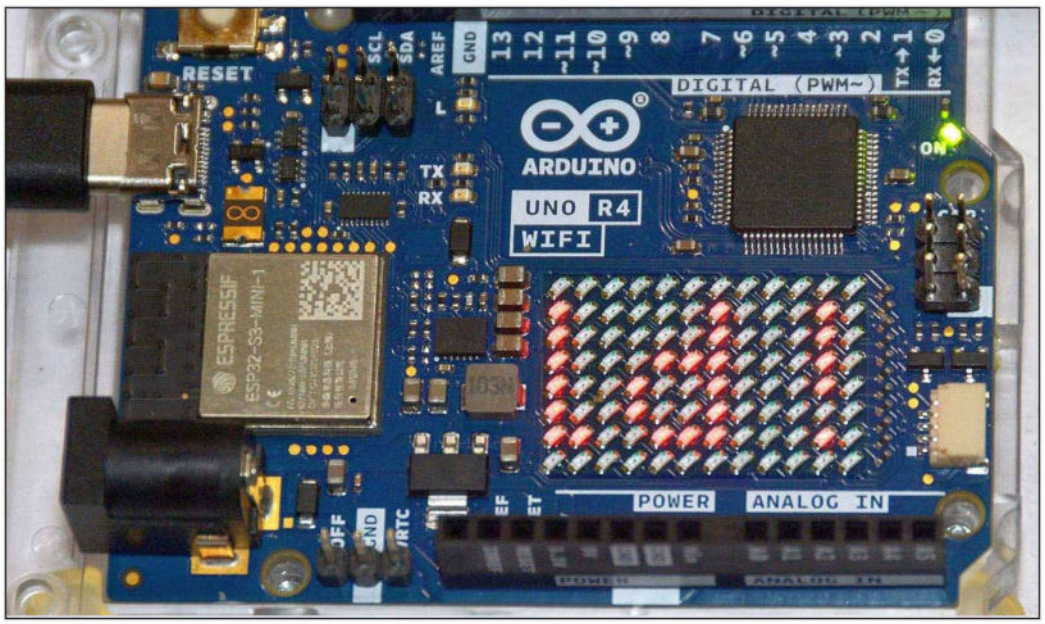

It did not take too long to learn to program the onboard 12x8 LED matrix used in combination with the "ArduinoGraphics" support library. I modified one of the LED matrix cloud sample programs (Figures 7A and 7B, next page) found at https://docs.arduino.cc/tutorials/uno-r4-wifi/led-matrix for use with the IDE 2.3.x (I prefer old-school versus cloud programming). There are several other coding examples in the standalone Arduino IDE 2.3.x under File > Examples > LED_Matrix>.

**Figure 7A: “Hello, World!” LED Matrix Code**

By tradition, the first program you code and load on any new computing device is “Hello, World!” Credit: Arduino with my comments with minor changes.


**Figure 7B: “Hello, World!” LED Matrix Live**

I could not capture the entire scrolling “Hello, World!” text in one photograph so you will need to load and run the code to see the LED matrix in action.



*Note: User installed Arduino libraries are located in an Arduino folder in the host computer's user documents directory while any default ones are in the libraries subdirectory where you installed the Arduino IDE software (the default is C:\Program Files). But for the new R4 boards, all required Renesas specific support libraries are hidden deep down inside invisible subdirectories. If you have the need to modify any, such as the default 5x7 character font, or just want to learn how their coding "magic" works then you will need to create a Windows shortcut (without quotes) pointing to: "C:\Users\VA3ROM\AppData\Local\ Arduino15\packages\arduino\hardware\ renesas_uno\1.0.5\libraries". My Windows 10/11 user name is "VA3ROM" so change it to whatever yours is.*

**My Final**

Taking everything into consideration, I rate the R4 (both versions) at a generous, but, 3.5 out of five stars, IMHO. And depending on your point of view, the R4 Minima is either a stripped down R4 WiFi or an slightly enhanced Uno R3 so I would recommend going with the R4 WiFi or just sticking with the R3 if you do not have the need for speed and more memory. —73