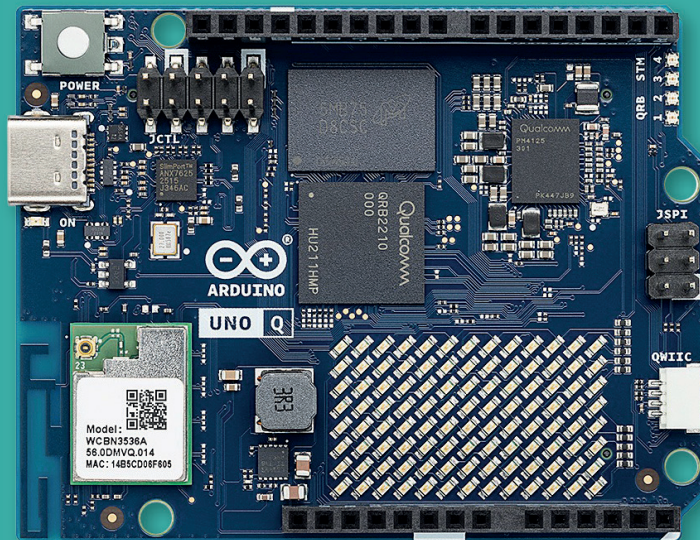


# Arduino UNO Q and AI

Learn to Build  
Intelligent Embedded Systems



Dogan Ibrahim

---

● This is an Elektor Publication. Elektor is the media brand of  
Elektor International Media B.V.  
PO Box 11, NL-6114-ZG Susteren, The Netherlands  
Phone: +31 46 4389444

● All rights reserved. No part of this book may be reproduced in any material form, including photocopying, or storing in any medium by electronic means and whether or not transiently or incidentally to some other use of this publication, without the written permission of the copyright holder except in accordance with the provisions of the Copyright Designs and Patents Act 1988 or under the terms of a licence issued by the Copyright Licensing Agency Ltd., 90 Tottenham Court Road, London, England W1P 9HE. Applications for the copyright holder's permission to reproduce any part of the publication should be addressed to the publishers.

● **Declaration**

The authors and publisher have used their best efforts in ensuring the correctness of the information contained in this book. They do not assume, or hereby disclaim, any liability to any party for any loss or damage caused by errors or omissions in this book, whether such errors or omissions result from negligence, accident or any other cause.

● British Library Cataloguing in Publication Data

A catalogue record for this book is available from the British Library

● **ISBN 978-3-89576-726-5** Print

**ISBN 978-3-89576-727-2** eBook

● © Copyright 2026 Elektor International Media  
[www.elektor.com](http://www.elektor.com)

Editor: Clemens Valens

Prepress Production: D-Vision, Julian van den Berg

Printers: Ipskamp, Enschede, The Netherlands

Cover photo courtesy Arduino SA

Elektor is the world's leading source of essential technical information and electronics products for pro engineers, electronics designers, and the companies seeking to engage them. Each day, our international team develops and delivers high-quality content - via a variety of media channels (including magazines, video, digital media, and social media) in several languages - relating to electronics design and DIY electronics. [www.elektormagazine.com](http://www.elektormagazine.com)

## Contents

|                                                                                         |           |
|-----------------------------------------------------------------------------------------|-----------|
| <b>Preface</b> .....                                                                    | <b>9</b>  |
| <b>Chapter 1 • The Hardware</b> .....                                                   | <b>11</b> |
| 1.1 Overview .....                                                                      | 11        |
| 1.2 Basic features .....                                                                | 11        |
| 1.3 Supplying power to the board .....                                                  | 13        |
| 1.4 Indicators on the board .....                                                       | 14        |
| 1.5 GPIO .....                                                                          | 14        |
| <b>Chapter 2 • Using the STM32U585 MCU in Projects.</b> .....                           | <b>18</b> |
| 2.1 Overview .....                                                                      | 18        |
| 2.2 STM32U585 MCU specifications .....                                                  | 18        |
| 2.3 Installing the Arduino UNO Q board .....                                            | 19        |
| 2.4 MCU user LED Arduino configuration .....                                            | 21        |
| 2.5 Project 1 – Flashing the red RGB LED3 connected to the MCU .....                    | 21        |
| 2.6 Project 2 – Flashing the red, green, and blue RGB LED3 of the MCU alternately ..... | 22        |
| 2.7 Project 3 – Warning signal .....                                                    | 23        |
| 2.8 Project 4 – LED flashing SOS .....                                                  | 24        |
| 2.9 Project 5 – Using an external LED .....                                             | 26        |
| 2.10 Project 6 – LED chaser .....                                                       | 29        |
| 2.11 Project 7 – Binary counting LEDs .....                                             | 31        |
| 2.12 Project 8 – Random flashing LEDs .....                                             | 34        |
| 2.13 Project 9 – Button controlled LED .....                                            | 35        |
| 2.14 Project 10 – Controlling the LED flashing rate – external interrupts .....         | 39        |
| 2.15 Project 11 – Traffic lights .....                                                  | 43        |
| 2.16 Using the print command .....                                                      | 48        |
| 2.17 Project 12 – Reaction timer .....                                                  | 57        |
| 2.18 LCD Displays .....                                                                 | 59        |
| 2.18.1 Project 13 – Seconds counter using 4-wire parallel LCD .....                     | 59        |
| 2.18.2 Project 14 – Seconds counter using I <sup>2</sup> C LCD .....                    | 66        |
| 2.19 Project 15 – LCD dice .....                                                        | 72        |
| 2.20 Project 16 – Analog ambient temperature sensor .....                               | 74        |

|                                                                                                    |            |
|----------------------------------------------------------------------------------------------------|------------|
| 2.21 Project 17 – ON-OFF temperature controller . . . . .                                          | 77         |
| 2.22 Project 18 – Ultrasonic distance measurement . . . . .                                        | 80         |
| 2.23 Project 19 – Geographical coordinates . . . . .                                               | 85         |
| 2.24 The DAC . . . . .                                                                             | 91         |
| 2.24.1 Project 20 – Generating a triangle waveform . . . . .                                       | 92         |
| 2.24.2 Project 21 - Generate a sinewave . . . . .                                                  | 94         |
| 2.25 Pulse-Width Modulation (PWM) . . . . .                                                        | 96         |
| 2.25.1 PWM channels of the Arduino UNO Q. . . . .                                                  | 97         |
| 2.25.2 Project 22 – Generating a PWM waveform . . . . .                                            | 98         |
| 2.26 OLED displays . . . . .                                                                       | 98         |
| 2.26.1 Using OLED displays . . . . .                                                               | 99         |
| 2.26.2 Project 23 - Using the SSD1306 OLED display module to display<br>text and numbers . . . . . | 101        |
| 2.26.3 Project 24 – Displaying shapes . . . . .                                                    | 103        |
| 2.26.4 Project 25 – Accurate digital thermometer with TMP102 . . . . .                             | 105        |
| <b>Chapter 3 • The LED Matrix . . . . .</b>                                                        | <b>110</b> |
| 3.1 Overview . . . . .                                                                             | 110        |
| 3.2 The LED matrix . . . . .                                                                       | 110        |
| 3.3 Example 1 – Draw a plus ('+') symbol . . . . .                                                 | 112        |
| 3.4 Example 2 – Drawing a rectangle . . . . .                                                      | 115        |
| 3.5 Using the other options of the template program. . . . .                                       | 116        |
| 3.6 Example 3 - Simple animation. . . . .                                                          | 117        |
| 3.7 Example 4 – Displaying and scrolling numbers 0 to 9. . . . .                                   | 119        |
| 3.8 Project 1 – Dice with LED matrix display. . . . .                                              | 120        |
| 3.9 Project 2 – Thermometer . . . . .                                                              | 122        |
| 3.10 Packing the LED matrix bit pattern. . . . .                                                   | 123        |
| 3.10.1 Program to generate the packed bit pattern from 13 × 8 bit array . . . . .                  | 125        |
| <b>Chapter 4 • Arduino UNO Q Programming Modes . . . . .</b>                                       | <b>128</b> |
| 4.1 Overview . . . . .                                                                             | 128        |
| 4.2 The Arduino App Lab . . . . .                                                                  | 128        |
| 4.2.1 Programming modes . . . . .                                                                  | 128        |
| 4.3 Required hardware and software . . . . .                                                       | 129        |

---

|                                                                              |            |
|------------------------------------------------------------------------------|------------|
| 4.4 SBC features and assembly . . . . .                                      | 130        |
| 4.5 Starting to use in SBC mode . . . . .                                    | 131        |
| 4.6 Using the Applications menu . . . . .                                    | 135        |
| 4.7 Accessing the Arduino UNO Q remotely . . . . .                           | 140        |
| 4.7.1 Accessing the Arduino UNO Q when in SBC mode . . . . .                 | 140        |
| 4.7.2 Accessing the Arduino UNO Q in wireless (network) mode . . . . .       | 142        |
| 4.7.3 Accessing in PC mode . . . . .                                         | 142        |
| 4.8 Accessing Arduino App Lab . . . . .                                      | 143        |
| <b>Chapter 5 • Arduino App Lab . . . . .</b>                                 | <b>144</b> |
| 5.1 Overview . . . . .                                                       | 144        |
| 5.2 Running an existing Arduino App Lab project . . . . .                    | 146        |
| 5.3 Project 1 – Flashing an on-board LED – First project . . . . .           | 147        |
| 5.4 Project 2 – Control the LED flashing rate with a potentiometer . . . . . | 149        |
| 5.5 Project 3 – Seconds counter with LCD . . . . .                           | 152        |
| 5.6 Project 4 – Room temperature controller . . . . .                        | 155        |
| 5.7 Arduino App Lab AI projects . . . . .                                    | 162        |
| 5.7.1 Project 5 – Keyword spotting – Hey Arduino. . . . .                    | 162        |
| 5.8 Arduino App Lab AI examples . . . . .                                    | 166        |
| 5.8.1 Blink LED with UI . . . . .                                            | 167        |
| 5.8.2 UNO Q Pin Toggle . . . . .                                             | 168        |
| 5.8.3 Weather Forecast on LED Matrix . . . . .                               | 170        |
| 5.8.4 Air quality on LED Matrix . . . . .                                    | 171        |
| 5.8.5 Glass breaking sensor . . . . .                                        | 173        |
| 5.8.6 Concrete crack detector. . . . .                                       | 175        |
| 5.8.7 LED Matrix Painter . . . . .                                           | 177        |
| 5.8.8 Bedtime storyteller . . . . .                                          | 179        |
| 5.8.9 Classify images . . . . .                                              | 184        |
| 5.8.10 Detect objects on images. . . . .                                     | 186        |
| 5.8.11 System resources logger . . . . .                                     | 187        |
| 5.8.12 QR and barcode scanner . . . . .                                      | 188        |
| 5.8.13 Face detection . . . . .                                              | 190        |

|                                                                                       |            |
|---------------------------------------------------------------------------------------|------------|
| 5.8.14 Blinking an LED from the Arduino Cloud. . . . .                                | 191        |
| <b>Chapter 6 • Edge Impulse Studio – Build, Train, and Deploy AI Models . . . . .</b> | <b>198</b> |
| 6.1 Overview . . . . .                                                                | 198        |
| 6.2 Edge Impulse Studio – advantages and disadvantages. . . . .                       | 200        |
| 6.3 Edge AI key terms . . . . .                                                       | 202        |
| 6.4 Installing Edge Impulse Studio . . . . .                                          | 203        |
| 6.5 Developing Keyword Spotting AI model . . . . .                                    | 205        |
| 6.6 Developing Object Detection AI model . . . . .                                    | 217        |
| 6.7 The future of AI. . . . .                                                         | 227        |
| 6.8 Future of Edge AI . . . . .                                                       | 229        |
| <b>Chapter 7 • Bibliography . . . . .</b>                                             | <b>231</b> |
| <b>Index . . . . .</b>                                                                | <b>234</b> |

## Preface

The rapid evolution of embedded systems has significantly lowered the barrier between concept and real-world deployment. Modern development boards now combine powerful heterogeneous processors, rich software ecosystems, and accessible tools that enable engineers, students, and hobbyists to prototype intelligent systems with remarkable efficiency. This book is written with that spirit in mind and focuses on exploring the capabilities of the Arduino UNO Q board as a versatile platform for embedded and edge-AI development.

The Arduino UNO Q board integrates the STM32U585 MCU microcontroller alongside a Qualcomm Dragonwing QRB2210 MPU processor, offering a unique combination of low-power real-time control and high-level application processing on a single platform. The STM32U585 MCU is fully programmable using the familiar Arduino IDE, allowing users to develop embedded applications using standard Arduino workflows. In parallel, the Qualcomm Dragon processor enables higher-level application development using Python through Arduino App Lab, making it possible to rapidly prototype, test, and deploy applications without deep system-level configuration.

This book begins with an overview of the board's architecture and key features, providing readers with a solid understanding of both the microcontroller (MCU) and microprocessor (MPU) environments. It then progresses to a series of hands-on example projects that demonstrate how to program the STM32U585 MCU using the Arduino IDE. These projects emphasize practical design patterns and illustrate how to efficiently utilize the hardware resources of the board.

In addition to custom MCU-based projects, the book explores the use of example applications provided through the Arduino App Lab. These examples highlight Python-based development on the Qualcomm Dragonwing processor and demonstrate how App Lab can be used to accelerate application development, testing, and system integration using both the MCU and the MPU for advanced AI based applications.

The later chapters introduce Edge Impulse Studio, covering its installation, core features, and its role in developing machine learning models for embedded and edge devices. The book culminates with a complete walkthrough of building a keyword spotting application using Edge Impulse Studio, illustrating the end-to-end workflow from data collection and model training to deployment and on-device inference.

This book is intended for students, educators, embedded developers, and enthusiasts who have a basic understanding of microcontrollers and are interested in expanding their skills into modern embedded, heterogeneous, and edge-AI system design. Prior practical knowledge of using the Arduino IDE and the Python programming language will be an advantage, but not essential. By combining clear explanations with practical examples across both Arduino and Python-based workflows, it aims to serve as both a learning guide

and a reference for developing AI based intelligent applications on the Arduino UNO Q platform.

The author hopes that readers will be inspired to use the Arduino UNO Q as a platform for developing their own future projects, including applications that explore and apply the important topic of artificial intelligence.

Dr Dogan Ibrahim  
London, February, 2026