Arduino Microcontroller Processing for Everyone!

Third Edition

Synthesis Lectures on Digital Circuits and Systems

Editor
Mitchell A. Thornton, ***Southern Methodist University***

The Synthesis Lectures on Digital Circuits and Systems series is comprised of 50- to 100-page books targeted for audience members with a wide-ranging background. The Lectures include topics that are of interest to students, professionals, and researchers in the area of design and analysis of digital circuits and systems. Each Lecture is self-contained and focuses on the background information required to understand the subject matter and practical case studies that illustrate applications. The format of a Lecture is structured such that each will be devoted to a specific topic in digital circuits and systems rather than a larger overview of several topics such as that found in a comprehensive handbook. The Lectures cover both well-established areas as well as newly developed or emerging material in digital circuits and systems design and analysis.

Arduino Microcontroller Processing for Everyone! Third Edition
Steven F. Barrett
2013

Boolean Differential Equations
Bernd Steinbach and Christian Posthoff
2013

Bad to the Bone: Crafting Electronic Systems with BeagleBone and BeagleBone Black
Steven F. Barrett and Jason Kridner
2013

Introduction to Noise-Resilient Computing
S.N. Yanushkevich, S. Kasai, G. Tangim, A.H. Tran, T. Mohamed, and V.P. Smerko
2013

Atmel AVR Microcontroller Primer: Programming and Interfacing, Second Edition
Steven F. Barrett and Daniel J. Pack
2012

Representation of Multiple-Valued Logic Functions
Radomir S. Stankovic, Jaakko T. Astola, and Claudio Moraga
2012

Arduino Microcontroller: Processing for Everyone! Second Edition
Steven F. Barrett
2012

Advanced Circuit Simulation Using Multisim Workbench
David Báez-López, Félix E. Guerrero-Castro, and Ofelia Delfina Cervantes-Villagómez
2012

Circuit Analysis with Multisim
David Báez-López and Félix E. Guerrero-Castro
2011

Microcontroller Programming and Interfacing Texas Instruments MSP430, Part I
Steven F. Barrett and Daniel J. Pack
2011

Microcontroller Programming and Interfacing Texas Instruments MSP430, Part II
Steven F. Barrett and Daniel J. Pack
2011

Pragmatic Electrical Engineering: Systems and Instruments
William Eccles
2011

Pragmatic Electrical Engineering: Fundamentals
William Eccles
2011

Introduction to Embedded Systems: Using ANSI C and the Arduino Development Environment
David J. Russell
2010

Arduino Microcontroller: Processing for Everyone! Part II
Steven F. Barrett
2010

Arduino Microcontroller Processing for Everyone! Part I
Steven F. Barrett
2010

Digital System Verification: A Combined Formal Methods and Simulation Framework
Lun Li and Mitchell A. Thornton
2010

Progress in Applications of Boolean Functions
Tsutomu Sasao and Jon T. Butler
2009

Embedded Systems Design with the Atmel AVR Microcontroller: Part II
Steven F. Barrett
2009

Embedded Systems Design with the Atmel AVR Microcontroller: Part I
Steven F. Barrett
2009

Embedded Systems Interfacing for Engineers using the Freescale HCS08 Microcontroller II: Digital and Analog Hardware Interfacing
Douglas H. Summerville
2009

Designing Asynchronous Circuits using NULL Convention Logic (NCL)
Scott C. Smith and JiaDi
2009

Embedded Systems Interfacing for Engineers using the Freescale HCS08 Microcontroller I: Assembly Language Programming
Douglas H.Summerville
2009

Developing Embedded Software using DaVinci & OMAP Technology
B.I. (Raj) Pawate
2009

Mismatch and Noise in Modern IC Processes
Andrew Marshall
2009

Asynchronous Sequential Machine Design and Analysis: A Comprehensive Development of the Design and Analysis of Clock-Independent State Machines and Systems
Richard F. Tinder
2009

An Introduction to Logic Circuit Testing
Parag K. Lala
2008

Pragmatic Power
William J. Eccles
2008

Multiple Valued Logic: Concepts and Representations
D. Michael Miller and Mitchell A. Thornton
2007

Finite State Machine Datapath Design, Optimization, and Implementation
Justin Davis and Robert Reese
2007

Atmel AVR Microcontroller Primer: Programming and Interfacing
Steven F. Barrett and Daniel J. Pack
2007

Pragmatic Logic
William J. Eccles
2007

PSpice for Filters and Transmission Lines
Paul Tobin
2007

PSpice for Digital Signal Processing
Paul Tobin
2007

PSpice for Analog Communications Engineering
Paul Tobin
2007

PSpice for Digital Communications Engineering
Paul Tobin
2007

PSpice for Circuit Theory and Electronic Devices
Paul Tobin
2007

Pragmatic Circuits: DC and Time Domain
William J. Eccles
2006

Pragmatic Circuits: Frequency Domain
William J. Eccles
2006

Pragmatic Circuits: Signals and Filters
William J. Eccles
2006

High-Speed Digital System Design
Justin Davis
2006

Introduction to Logic Synthesis using Verilog HDL
Robert B.Reese and Mitchell A.Thornton
2006

Microcontrollers Fundamentals for Engineers and Scientists
Steven F. Barrett and Daniel J. Pack
2006

© Springer Nature Switzerland AG 2022

Reprint of original edition © Morgan & Claypool 2013

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means—electronic, mechanical, photocopy, recording, or any other except for brief quotations in printed reviews, without the prior permission of the publisher.

Arduino Microcontroller Processing for Everyone! Third Edition

Steven F. Barrett

ISBN: 978-3-031-79863-4 paperback
ISBN: 978-3-031-79864-1 ebook

DOI 10.1007/978-3-031-79864-1

A Publication in the Springer series
SYNTHESIS LECTURES ON DIGITAL CIRCUITS AND SYSTEMS

Lecture #43
Series Editor: Mitchell A. Thornton, *Southern Methodist University*
Series ISSN
Synthesis Lectures on Digital Circuits and Systems
Print 1932-3166 Electronic 1932-3174

Arduino Microcontroller Processing for Everyone!

Third Edition

Steven F. Barrett
University of Wyoming, Laramie, WY

SYNTHESIS LECTURES ON DIGITAL CIRCUITS AND SYSTEMS #43

ABSTRACT

This book is about the Arduino microcontroller and the Arduino concept. The visionary Arduino team of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis launched a new innovation in microcontroller hardware in 2005, the concept of open source hardware. Their approach was to openly share details of microcontroller-based hardware design platforms to stimulate the sharing of ideas and promote innovation. This concept has been popular in the software world for many years. This book is intended for a wide variety of audiences including students of the fine arts, middle and senior high school students, engineering design students, and practicing scientists and engineers. To meet this wide audience, the book has been divided into sections to satisfy the need of each reader. The book contains many software and hardware examples to assist the reader in developing a wide variety of systems. The book covers two different Arduino products: the Arduino UNO R3 equipped with the Atmel ATmega328 and the Arduino Mega 2560 equipped with the Atmel ATmega2560. The third edition has been updated with the latest on these two processing boards, changes to the Arduino Development Environment and multiple extended examples.

KEYWORDS

Arduino microcontroller, Arduino UNO R3, Atmel microcontroller, Atmel AVR, ATmega328, Arduino Mega 2560, microcontroller interfacing, ATmega2560, embedded systems design

Contents

Preface . . . . . **xix**

1 Getting Started **1**

1.1 Overview 1
1.2 Getting Started 2
1.3 Arduino UNO R3 processing board 3
1.3.1 Arduino UNO R3 host processor–the ATmega328 4
1.3.2 Arduino UNO R3/ATmega328 hardware features 6
1.3.3 ATmega328 Memory 6
1.3.4 ATmega328 Port System 8
1.3.5 ATmega328 Internal Systems 9
1.3.6 Arduino UNO R3 open source schematic 12
1.4 Arduino Mega 2560 processing board 12
1.4.1 Arduino Mega 2560 host processor–the ATmega2560 14
1.4.2 Arduino Mega 2560 /ATmega2560 hardware features 14
1.4.3 ATmega2560 Memory 16
1.4.4 ATmega2560 Port System 18
1.4.5 ATmega2560 Internal Systems 20
1.5 Arduino Mega 2560 open source schematic 22
1.6 Example: Autonomous Maze Navigating Robot 23
1.6.1 Structure chart 23
1.6.2 UML activity diagrams 23
1.6.3 Arduino UNO R3 Systems 23
1.7 Other Arduino–based platforms 24
1.8 Extending the hardware features of the Arduino platforms 25
1.9 Application: Arduino Hardware Studio 25
1.10 Summary 26
1.11 References 26
1.12 Chapter Problems 29

2 Programming **33**
2.1 Overview 33
2.2 The Big Picture 34
2.3 Arduino Development Environment 34
2.3.1 Background 36
2.3.2 Quick start Guide 36
2.3.3 Arduino Development Environment overview 37
2.3.4 Sketchbook concept 38
2.3.5 Arduino software, libraries, and language references 38
2.3.6 Writing an Arduino Sketch 38
2.4 Anatomy of a Program 46
2.4.1 Comments 47
2.4.2 Include files 47
2.4.3 Functions 48
2.4.4 Program constants 51
2.4.5 Interrupt handler definitions 51
2.4.6 Variables 51
2.4.7 Main program 52
2.5 Fundamental programming concepts 52
2.5.1 Operators 52
2.5.2 Programming constructs 56
2.5.3 Decision processing 58
2.6 Application 1: Robot IR sensor 61
2.7 Application 2: Art piece illumination system 65
2.8 Application 3: Friend or Foe Signal 65
2.9 Summary 67
2.10 References 67
2.11 Chapter Problems 68

3 Embedded Systems Design **69**
3.1 What is an embedded system? 69
3.2 Embedded system design process 70
3.2.1 Project Description 70
3.2.2 Background Research 70
3.2.3 Pre–Design 72
3.2.4 Design 72

3.2.5 Implement Prototype 74
3.2.6 Preliminary Testing 74
3.2.7 Complete and Accurate Documentation 75
3.3 Example: Blinky 602A autonomous maze navigating robot system design 75
3.4 Application: Control algorithm for the Blinky 602A Robot 78
3.5 Summary 89
3.6 References 89
3.7 Chapter Problems 90

4 Atmel AVR Operating Parameters and Interfacing 91

4.1 Overview 91
4.2 Operating Parameters 92
4.3 Battery Operation 95
4.3.1 Embedded system voltage and current drain specifications 95
4.3.2 Battery characteristics 95
4.4 Input Devices 95
4.4.1 Switches 96
4.4.2 Pullup resistors in switch interface circuitry 97
4.4.3 Switch Debouncing 97
4.4.4 Keypads 99
4.4.5 Sensors 104
4.5 Output Devices 109
4.5.1 Light Emitting Diodes (LEDs) 109
4.5.2 Seven Segment LED Displays 109
4.5.3 Code Example 110
4.5.4 Tri–state LED Indicator 113
4.5.5 Dot Matrix Display 113
4.5.6 Liquid Crystal Character Display (LCD) in C 116
4.5.7 Programming a serial configured LCD 124
4.5.8 Liquid Crystal Character Display (LCD) using the Arduino Development Environment 125
4.5.9 High Power DC Devices 125
4.6 DC Solenoid Control 126
4.7 DC Motor Speed and Direction Control 126
4.7.1 DC Motor Operating Parameters 127
4.7.2 H–bridge direction control 128
4.7.3 Servo motor interface 130

4.7.4 Stepper motor control 130
4.7.5 AC Devices 139
4.8 Interfacing to Miscellaneous Devices 139
4.8.1 Sonalerts, Beepers, Buzzers 139
4.8.2 Vibrating Motor 141
4.9 Application: Special Effects LED Cube 141
4.9.1 Construction Hints 142
4.9.2 LED Cube Arduino Sketch Code 144
4.10 Summary 159
4.11 References 159
4.12 Chapter Problems 160

5 Analog to Digital Conversion (ADC) 163

5.1 Overview 163
5.2 Sampling, Quantization and Encoding 164
5.2.1 Resolution and Data Rate 166
5.3 Analog–to–Digital Conversion (ADC) Process 167
5.3.1 Transducer Interface Design (TID) Circuit 168
5.3.2 Operational Amplifiers 169
5.4 ADC Conversion Technologies 172
5.4.1 Successive–Approximation 172
5.5 The Atmel ATmega328 and ATmega2560 ADC System 173
5.5.1 Block Diagram 175
5.5.2 ATmega328 ADC Registers 175
5.5.3 ATmega2560 ADC Registers 178
5.6 Programming the ADC using the Arduino Development Environment 180
5.7 ATmega328: Programming the ADC in C 181
5.8 ATmega2560: Programming the ADC in C 182
5.9 Example: ADC Rain Gage Indicator with the Arduino UNO R3 184
5.9.1 ADC Rain Gage Indicator using the Arduino Development Environment 185
5.9.2 ADC Rain Gage Indicator in C 189
5.9.3 ADC Rain Gage using the Arduino Development Environment–Revisited 196
5.10 One–bit ADC – Threshold Detector 198
5.11 Digital–to–Analog Conversion (DAC) 198

5.11.1 DAC with the Arduino Development Environment 199
5.11.2 DAC with external converters 200
5.11.3 Octal Channel, 8–bit DAC via the SPI 201
5.12 Application: Art piece illumination system – Revisited 201
5.13 Arduino Mega 2560 example: Kinesiology and Health Laboratory Instrumentation 205
5.14 Summary 212
5.15 References 213
5.16 Chapter Problems 214

6 Interrupt Subsystem 217

6.1 Overview 217
6.1.1 ATmega328 Interrupt System 218
6.1.2 ATmega2560 Interrupt System 218
6.1.3 General interrupt response 221
6.2 Interrupt programming overview 221
6.3 Programming ATmega328 Interrupts in C and the Arduino Development Environment 222
6.3.1 External interrupt programming–Atmega328 222
6.3.2 ATmega328 Internal Interrupt Programming 226
6.4 Programming ATmega2560 Interrupts in C and the Arduino Development Environment 231
6.4.1 External interrupt programming–Atmega2560 231
6.4.2 ATmega2560 Internal Interrupt Programming 235
6.5 Foreground and Background Processing 240
6.6 Interrupt Examples 243
6.6.1 Application 1: Real Time Clock in C 243
6.6.2 Application 2: Real Time Clock using the Arduino Development Environment 246
6.6.3 Application 3: Interrupt Driven USART in C 248
6.7 Summary 259
6.8 References 259
6.9 Chapter Problems 260

7 Timing Subsystem 261

7.1 Overview 261
7.2 Timing related terminology 262

7.2.1 Frequency 262
7.2.2 Period 262
7.2.3 Duty Cycle 262
7.3 Timing System Overview 263
7.4 Timer System Applications 265
7.4.1 Input Capture – Measuring External Timing Event 265
7.4.2 Counting Events 267
7.4.3 Output Compare – Generating Timing Signals to Interface External Devices 267
7.4.4 Industrial Implementation Case Study (PWM) 268
7.5 Overview of the Atmel ATmega328 and ATmega2560 Timer Systems 269
7.6 Timer 0 System 270
7.6.1 Modes of Operation 272
7.6.2 Timer 0 Registers 274
7.7 Timer 1 275
7.7.1 Timer 1 Registers 275
7.8 Timer 2 281
7.9 Programming the Arduino UNO R3 and Mega 2560 using the built–in Arduino Development Environment Timing Features 281
7.10 Programming the Timer System in C 285
7.10.1 Precision Delay in C 285
7.10.2 Pulse Width Modulation in C 287
7.10.3 Input Capture Mode in C 288
7.11 Apprlication 1: Servo Motor Control with the PWM System in C 290
7.12 Application 2: Inexpensive Laser Light Show 295
7.13 Summary 297
7.14 References 297
7.15 Chapter Problems 298

8 Serial Communication Subsystem 301

8.1 Overview 301
8.2 Serial Communications 302
8.3 Serial Communication Terminology 302
8.4 Serial USART 303
8.4.1 System Overview 305
8.5 System Operation and Programming using Arduino Development Environment features 307

8.6 System Operation and Programming in C 313
8.6.1 Serial Peripheral Interface–SPI 315
8.7 SPI Programming in the Arduino Development Environment 317
8.8 SPI Programming in C 318
8.9 Two–wire Serial Interface–TWI 319
8.9.1 Arduino Development Environment 319
8.10 Application 1: USART communication with LCD 320
8.11 Application 2: SD/MMC card module extension via the USART 321
8.12 Application 3: Equipping an Arduino Processor with a Voice Chip 324
8.13 Application 4: Programming the Arduino UNO R3 ATmega328 via the ISP . . 326
8.13.1 Programming Procedure 326
8.14 Application 5: TMS1803 3–bit LED Drive Controller 328
8.15 Summary 330
8.16 References 330
8.17 Chapter Problems 330

9 Extended Examples 333

9.1 Overview 333
9.2 Extended Example 1: Automated Fan Cooling System 333
9.3 Extended Example 2: Fine Art Lighting System 342
9.4 Extended Example 3: Flight Simulator Panel 346
9.5 Extended Example 4: Submersible Robot 372
9.5.1 Requirements 373
9.5.2 Structure chart 374
9.5.3 Circuit Diagram 374
9.5.4 UML Activity Diagram 375
9.5.5 Microcontroller Code 375
9.5.6 Project Extensions 379
9.6 Extended example 5: Weather Station 380
9.6.1 Requirements 380
9.6.2 Structure chart 380
9.6.3 Circuit diagram 381
9.6.4 UML activity diagrams 381
9.6.5 Microcontroller code 381
9.7 Autonomous Maze Navigating Robots 393
9.8 Extended Example 6: Blinky 602A robot–revisited 394

9.8.1 Requirements 394
9.8.2 Circuit diagram 394
9.8.3 Structure chart 397
9.8.4 UML activity diagrams 397
9.8.5 Microcontroller code 399
9.9 Extended Example 7: Mountain Maze Navigating Robot 407
9.9.1 Description 407
9.9.2 Requirements 407
9.9.3 Circuit diagram 407
9.9.4 Structure chart 407
9.9.5 UML activity diagrams 411
9.9.6 Microcontroller code 411
9.9.7 Mountain Maze 415
9.9.8 Project extensions 415
9.10 Extended Example 8: Robot Wheel Odometry 415
9.11 Summary 419
9.12 References 419
9.13 Chapter Problems 420

A ATmega328 Register Set 425

B ATmega328 Header File 431

C ATmega2560 Register Set 451

D ATmega2560 Header File 457

Author's Biography 493

Preface

This book is about the Arduino microcontroller and the Arduino concept. The visionary Arduino team of Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino, and David Mellis launched a new innovation in microcontroller hardware in 2005, the concept of open source hardware. There approach was to openly share details of microcontroller–based hardware design platforms to stimulate the sharing of ideas and innovation. This concept has been popular in the software world for many years.

This book is written for a number of audiences. First, in keeping with the Arduino concept, the book is written for practitioners of the arts (design students, artists, photographers, etc.) who may need processing power in a project but do not have an in depth engineering background. Second, the book is written for middle school and senior high school students who may need processing power for a school or science fair project. Third, we write for engineering students who require processing power for their senior design project but do not have the background in microcontroller–based applications commonly taught in electrical and computer engineering curricula. Finally, the book provides practicing scientists and engineers an advanced treatment of the Atmel AVR microcontroller. The third edition has been updated with the latest on the Arduino UNO R3 and the Arduino Mega 2560 processors, changes to the Arduino Development Environment and multiple extended examples.

APPROACH OF THE BOOK

To encompass such a wide range of readers, we have divided the book into several portions to address the different readership. Chapters 1 through 2 are intended for novice microcontroller users. Chapter 1 provides a review of the Arduino concept, a description of the Arduino UNO R3 and the Arduino Mega 2560 development boards, and a brief review of the features of the UNO R3's host processor, the Atmel ATmega 328 microcontroller. We also provide a similar treatment of the Arduino Mega 2560 processor board. Chapter 2 provides an introduction to programming for the novice programmer. Chapter 2 also introduces the Arduino Development Environment and how to program sketches. It also serves as a good review for the seasoned developer.

Chapter 3 provides an introduction to embedded system design processes. It provides a systematic, step–by–step approach on how to design complex systems in a stress free manner.

Chapter 4 introduces the extremely important concept of the operating envelope for a microcontroller. The voltage and current electrical parameters for the HC CMOS based Atmel AVR line of microcontrollers is reviewed and applied to properly interface input and output devices to the Arduino UNO R3, the Arduino Mega 2560, the ATmega328 microcontroller, and the ATmega 2560 microcontroller.

Chapters 5 through 8 provide detailed engineering information on the ATmega328 and the AT mega 2560 microcontroller systems including analog-to-digital conversion, interrupts, timing features, and serial communications. These chapters are intended for engineering students and practicing engineers. However, novice microcontroller users will find the information readable and well supported with numerous examples using the Arduino Development Environment and C.

The final chapter, Chapter 9, provides example applications for a wide variety of skill levels.

ACKNOWLEDGMENTS

A number of people have made this book possible. I would like to thank Massimo Banzi of the Arduino design team for his support and encouragement in writing the book. I would also like to thank Joel Claypool of Morgan & Claypool Publishers who has supported a number of writing projects over the last several years. He also provided permission to include portions of background information on the Atmel line of AVR microcontrollers in this book from several of our previous projects. I would also like to thank Sparkfun Electronics of Boulder, Colorado; Atmel Incorporated; the Arduino team; and ImageCraft of Palo Alto, California for use of pictures and figures used within the book.

I would like to dedicate this book to my close friend Dr. Daniel Pack, Ph.D., P.E. Much of the writing is his from earlier Morgan & Claypool projects. In 2000, Daniel suggested that we might write a book together on microcontrollers. I had always wanted to write a book but I thought that's what other people did. With Daniel's encouragement we wrote that first book (and several more since then). Daniel is a good father, good son, good husband, brilliant engineer, a work ethic second to none, and a good friend. To you good friend I dedicate this book. I know that we will do many more together. Finally, I would like to thank my wife and best friend of many years, Cindy.

Laramie, Wyoming, August 2013

Steve Barrett