

Digital Electronics

Table of Contents

1. [Digital Electronics](#)
 1. [Course Landing Page](#)
 1. [General Student Resources](#)
 2. [Copyright Notice](#)
 1. [It is illegal to make copies of this course without the permission of Project Lead The Way, Inc.](#)
2. [Unit 1: Foundations in Electronics](#)
 1. [Unit Landing Page](#)
3. [Lesson 1.1: Introduction to Electronics](#)
 1. [Introduction to Digital Electronics](#)
 2. [Quiz 1.1.1: General Safety in the Electronics Classroom](#)
 3. [Activity 1.1.2: Investigating Basic Circuits \(DMS\)](#)
 1. [Procedure](#)
 1. [Part A: Creating a Circuit and Measuring a Circuit's Properties](#)
 2. [Part B: Series and Parallel Circuits](#)
 3. [Going Further](#)
 2. [Startup Guide: The myDAQ and myDigital Protoboard](#)
 1. [NI Digital MiniSystem \(DMS\)](#)
 2. [NI myDAQ™](#)
 3. [NI myDigitalProtoboard™](#)
 4. [Required Hardware and Software:](#)
 5. [Powering the myDigital Protoboard](#)
 3. [Investigating Basic Circuits](#)
 4. [Activity 1.1.3: Scientific and Engineering Notation](#)
 1. [Procedure](#)
 2. [Scientific and Engineering Notation](#)
 5. [Activity 1.1.4: Component Identification: Analog](#)
 1. [Procedure](#)
 2. [Introduction to Resistors and Capacitors](#)
 6. [Activity 1.1.5.a: Circuit Theory: Hand Calculations](#)
 1. [Procedure](#)
 1. [Going Further – Optional](#)
 2. [Circuit Theory Laws](#)
 7. [Activity 1.1.5.b: Circuit Theory: Simulations](#)
 1. [Procedure](#)
 8. [Activity 1.1.5.c: Circuit Theory: Breadboarding](#)
 1. [Procedure](#)
 2. [The Breadboard](#)
 9. [Activity 1.1.6: Component Identification: Digital](#)
 1. [Procedure](#)
 1. [Combinational Logic](#)
 2. [Sequential Logic](#)
 3. [Clock Signals \(The 555 Timer\)](#)
 2. [Introduction to Logic Gates and Integrated Circuits](#)
 10. [Activity 1.1.7: Introduction to Datasheets](#)
 1. [Procedure](#)
 11. [Optional Project 1.1.8: Soldering Practice: Fun Light Project](#)
 1. [Procedure](#)

1. [Parts List – Fun Light](#)
 2. [Assembly Procedure](#)
 3. [Functional Test Procedure](#)
 4. [Project Completion](#)
2. [Soldering and Desoldering](#)
12. [Project 1.1.9: Soldering: Random Number Generator](#)
 1. [Procedure](#)
 2. [Introduction to the Board Game Counter](#)
4. [Lesson 1.2: Introduction to Circuit Design](#)
 1. [Circuit Design](#)
 2. [Activity 1.2.1: Introduction to Combinational Logic Design: Seat Belt Circuit](#)
 1. [Procedure](#)
 1. [Going Further – Optional](#)
 3. [Activity 1.2.2 Analog and Digital Signals](#)
 1. [Procedure](#)
 2. [Analog Digital Signals](#)
 4. [Activity 1.2.3 Binary Number System](#)
 1. [Procedure](#)
 1. [Going Further – Optional](#)
 2. [The Binary Number System](#)
 5. [Activity 1.2.4 Introduction to Sequential Logic Design: Counters \(DMS\)](#)
 1. [Procedure](#)
 - 1.
 6. [Activity 1.2.5 Clock Signals: The 555 Timer](#)
 1. [Procedure](#)
 2. [Clock Signals: 555 Timer](#)
 7. [Project 1.2.6 Understanding Analog Design: The Random Number Generator](#)
 1. [Procedure](#)
 2. [Understanding Analog Design \(RNG\)](#)
 8. [Project 1.2.7 Understanding Digital Design: Random Number Generator](#)
 1. [Procedure](#)
 2. [Understanding Digital Design \(RNG\)](#)
5. [Unit 2: Combinational Logic](#)
 1. [Unit Landing Page](#)
6. [Lesson 2.1: AOI Combinational Logic Circuit Design](#)
 1. [Combinational Logic Design Process](#)
 2. [Activity 2.1.1 AOI Design: Truth Tables to Logic Expressions](#)
 1. [Procedure](#)
 1. [Truth Tables to Logic Expressions](#)
 2. [Logic Expressions to Truth Tables](#)
 3. [Seat Belt Alarm Circuit](#)
 4. [Humidity Sensor Circuit](#)
 2. [AOI Design: Truth Tables to Logic Expressions](#)
 3. [Activity 2.1.2 AOI Logic Analysis: Circuit to Truth Table to Logic Expression](#)
 1. [Procedure](#)
 2. [AOI Design: Logic Analysis](#)
 4. [Activity 2.1.3 AOI Logic Implementation](#)
 1. [Procedure](#)
 2. [AOI Logic Implementation](#)
 5. [Activity 2.1.4 Circuit Simplification: Boolean Algebra](#)
 1. [Procedure](#)
 2. [Circuit Simplification: Boolean Algebra](#)

3. [Digital Electronics Equations and Theorems](#)
 1. [Boolean Theorems](#)
 2. [Commutative Laws](#)
 3. [Associative Laws](#)
 4. [Distributive Laws](#)
 5. [Consensus Theorems](#)
 6. [DeMorgan's Theorems](#)
6. [Activity 2.1.5 Circuit Simplification: DeMorgan's Theorems](#)
 1. [Procedure](#)
 2. [Circuit Simplification: DeMorgan's Theorems](#)
7. [Project 2.1.6 AOI Logic Design: Majority Vote](#)
 1. [Procedure](#)
7. [Lesson 2.2: Alternative Design: Universal Gates and K-Mapping](#)
 1. [Activity 2.2.1 Circuit Simplification: Karnaugh Mapping](#)
 1. [Procedure](#)
 1. [Going Further – Optional](#)
 2. [Karnaugh Mapping](#)
 2. [Activity 2.2.2 Universal Gates: NAND Only Logic Design](#)
 1. [Procedure](#)
 2. [Universal Gate - NAND](#)
 3. [Activity 2.2.3 Universal Gates: NOR Only Logic Design](#)
 1. [Procedure](#)
 2. [Universal Gate – NOR](#)
 4. [Activity 2.2.4 Design Tool: Logic Converter](#)
 1. [Procedure](#)
 5. [Project 2.2.5 Universal Gates and K-Mapping: Fireplace Control Circuit](#)
 1. [Procedure](#)
 1. [Design](#)
 2. [Simulation](#)
 3. [Prototyping](#)
 2. [Combinational Logic Design Process](#)
8. [Lesson 2.3: Specific Combinational Logic Designs](#)
 1. [Activity 2.3.1 Hexadecimal and Octal Number Systems](#)
 1. [Procedure](#)
 1. [Going Further – Optional](#)
 2. [Octal and Hexadecimal Number Systems](#)
 2. [Activity 2.3.2 Seven Segment Displays](#)
 1. [Procedure](#)
 1. [Seven-Segment Display Drivers](#)
 2. [Seven Segment Displays](#)
 3. [Seven Segment Display Driver](#)
 3. [Activity 2.3.3 Multiplexers \(MUX\) and Demultiplexers \(DEMUX\)](#)
 1. [Procedure](#)
 2. [Multiplexed Signals and Demultiplexed Signals](#)
 4. [Activity 2.3.4 Two's Complement Arithmetic](#)
 1. [Procedure](#)
 2. [2's Complement Arithmetic](#)
 5. [Activity 2.3.5 Binary Adders: XOR and XNOR](#)
 1. [Procedure](#)
 1. [Going Further](#)
 2. [XOR, XNOR, and Binary Adders](#)
9. [Lesson 2.4: Introduction to Programmable Logical Devices \(PLDs\)](#)

1. [Problem 2.4.1 Combinational Logic Circuit Design: Date of Birth](#)
 1. [Procedure](#)
 1. [Design](#)
 2. [Simulation](#)
 3. [Prototyping](#)
 2. [Combinational Logic Student Design Problem: Date of Birth](#)
2. [Activity 2.4.2 Introduction to PLDs Programming Tutorial \(DMS/Xilinx 14.7\)](#)
 1. [Procedure](#)
 1. [Part 1: Setting Up the CDS \(Multisim\) in PLD Mode](#)
 2. [Part 2: Simulating a PLD Mode Design in the CDS \(Multisim\)](#)
 3. [Part 3: Exporting the Design to the PLD Module](#)
 4. [Using Both Hardware Platforms \(DMS and DLB\)](#)
3. [Project 2.4.3 PLD Design: Date of Birth](#)
 1. [Procedure](#)
 1. [Circuit Design in PLD Design Mode](#)
 2. [Digital MiniSystem \(DMS\) - Mapping PLD Pin Inputs/Outputs \(PIO\) – Skip if using DLB](#)
 3. [Digital Logic Board \(DLB\) - Mapping PLD Inputs/Outputs – Skip if using DMS](#)
10. [Unit 3: Sequential Logic](#)
 1. [Unit Landing Page](#)
11. [Lesson 3.1: Sequential Logic Circuit Design](#)
 1. [Activity 3.1.1 Sequential Logic: D Flip-Flops and J/K Flip-Flops](#)
 1. [Procedure](#)
 2. [Flip-Flops and Latches](#)
 2. [Activity 3.1.2 Flip-Flop Applications: Event Detection](#)
 1. [Procedure](#)
 2. [Flip-Flop Applications](#)
 3. [Activity 3.1.3 Flip-Flop Applications: Shift Registers](#)
 1. [Procedure](#)
 1. [Simulation \(Design Mode\)](#)
 2. [Simulation \(PLD Mode\)](#)
 3. [Export to PLD \(PLD Mode\)](#)
12. [Lesson 3.2: Asynchronous Counters](#)
 1. [Activity 3.2.1 Asynchronous Counters: Small Scale Integration \(SSI\) Up/Down Counters](#)
 1. [SSI \(D flip-flops\)](#)
 2. [Procedure](#)
 1. [Simulation \(Design Mode\)](#)
 2. [Simulation \(PLD Mode\)](#)
 3. [Export to PLD \(PLD Mode\)](#)
 3. [Asynchronous Counter](#)
 2. [Activity 3.2.2 Asynchronous Counters: Small Scale Integration \(SSI\) Modulus Counters](#)
 1. [Procedure](#)
 1. [Simulation \(Design Mode\)](#)
 2. [Simulation \(PLD Mode\)](#)
 3. [Digital MiniSystem \(DMS\) \(Disregard if using the DLB\)](#)
 4. [Clock Signal](#)
 5. [DLB External Clock Signal \(RotCLK\)](#)
 3. [Activity 3.2.3 Asynchronous Counters: Medium Scale Integration \(MSI\) Suspend/Reset Counts](#)
 1. [Procedure](#)
 2. [Asynchronous Counters with MSI Gates](#)
 4. [Problem 3.2.4 Asynchronous Counters: Now Serving Display \(DMS\)](#)

1. [Procedure](#)
 1. [Design](#)
 2. [Design Specification](#)
 3. [Multiplexed Seven Segment Display Design](#)
 4. [Simulation: \(Design Mode or PLD Mode\)](#)
 5. [Prototyping](#)
13. [Lesson 3.3: Synchronous Counters](#)
 1. [Activity 3.3.1 Synchronous Counters: Small Scale Integration \(SSI\)](#)
 1. [Procedure](#)
 2. [Synchronous Counters](#)
 2. [Activity 3.3.2 Synchronous Counters: Medium Scale Integration \(MSI\) 74LS163 Up Counter](#)
 1. [Procedure](#)
 1. [Simulation \(Design Mode\)](#)
 2. [Simulation \(PLD Mode\)](#)
 2. [Synchronous Counters With MSI Gates](#)
 3. [Activity 3.3.3 Synchronous Counters: Medium Scale Integration \(MSI\) 74LS193 Up/Down Counter](#)
 1. [Procedure](#)
 1. [Design Mode](#)
 2. [PLD Mode](#)
 4. [Problem 3.3.4 Synchronous Counters: Sixty Second Timer \(DMS\)](#)
 1. [Procedure](#)
 1. [Design \(60 Second Timer\)](#)
 2. [Design Specifications](#)
 3. [Simulation \(Design Mode or PLD Mode\)](#)
 4. [Prototyping](#)
14. [Unit 4: Controlling Real World Systems](#)
 1. [Unit Landing Page](#)
15. [Lesson 4.1: Introduction to State Machines](#)
 1. [Project 4.1.1 Sensors and Motors: Copier Jam Detector \(VEX/DMS\)](#)
 1. [Procedure](#)
 1. [Design](#)
 2. [Design Specifications](#)
 3. [Simulation \(Design Mode or PLD Mode\)](#)
 4. [Prototyping](#)
 2. [Introduction to Sensors and Motors: Copier Jam Detector DMS-VEX or DLB-VEX](#)
 2. [Activity 4.1.2 State Machines: Phone Number \(DMS\)](#)
 1. [Procedure](#)
 1. [Going Further \(Optional\)](#)
 2. [State Machine Design](#)
 3. [Problem 4.1.3 State Machines: Tollbooth \(VEX/DMS\)](#)
 1. [Procedure](#)
 1. [Design](#)
 2. [State Graph Analysis](#)
 3. [State Transition Table Analysis](#)
 4. [Tollbooth State Transition Table Analysis](#)
 5. [Design Equations](#)
 6. [Simulation](#)
 7. [Prototyping](#)
 8. [Tollbooth Test Fixture – Build](#)
 9. [Tollbooth Test Fixture – Wiring Diagram](#)

16. [Lesson 4.2: Introduction to Microcontrollers](#)
 1. [Tutorial 4.2.1 Installing Arduino IDE Software](#)
 1. [Procedure](#)
 1. [Installation](#)
 2. [Test](#)
 3. [Parallax Sensor Kit with Arduino VEX with Arduino](#)
 4. [Microcontroller Test Bed](#)
 2. [Activity 4.2.2 Introduction to Microcontrollers](#)
 1. [Structure of a Sketch \(Program\)](#)
 2. [Procedure](#)
 1. [Sketch 1: "Blink"](#)
 2. [Sketch 2: "AltBlink"](#)
 3. [Sketch 3: "Pushbutton"](#)
 4. [Sketch 4: "DigitalReadSerial"](#)
 5. [Sketch 5: "DigitalReadSerialLED"](#)
 6. [Sketch 5: "PIRDigitalReadSerial"](#)
 7. [Sketch 6: "AnalogReadSerial"](#)
 8. [Sketch 7: "Joystick"](#)
 3. [Microcontrollers, Microcomputers, and Microprocessors](#)
 3. [Activity 4.2.3 PWM Signals](#)
 1. [Input Device: Dual-axis Accelerometer](#)
 2. [Features](#)
 3. [Communication Protocol](#)
 4. [Procedure](#)
 1. [Sketch 1: "Two Axis Accelerometer"](#)
 2. [Sketch 1: "Two Axis Accelerometer"](#)
 3. [Output Device: VEX 393 Motor with Motor Controller 29](#)
 4. [Sketch 2 "ServoPot"](#)
 5. [Sketch 3 "microServoPot"](#)
 5. [PWM: Pulse Width Modulation](#)
 4. [Problem 4.2.4 Microcontrollers: The Tollbooth Revisited](#)
 1. [Procedure](#)
 1. [Design](#)
 2. [Design Proposal](#)
17. [Appendix](#)
 1. [Remarks](#)
 2. [Glossary](#)