MORGAN & CLAYPOOL PUBLISHERS

Microcontroller Programming and Interfacing with Texas Instruments MSP430FR2433 and MSP430FR5994 - Part I

Second Edition

Steven F. Barrett
Daniel J. Pack

SYNTHESIS LECTURES ON DIGITAL CIRCUITS AND SYSTEMS

Mitchell A. Thornton, *Series Editor*

Microcontroller Programming and Interfacing with Texas Instruments MSP430FR2433 and MSP430FR5994 – Part I

Second 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.

Microcontroller Programming and Interfacing with Texas Instruments MSP430FR2433 and MSP430FR5994 – Part I, Second Edition
Steven F. Barrett and Daniel J. Pack
2019

Synthesis of Quantum Circuits vs. Synthesis of Classical Reversible Circuits
Alexis De Vos, Stijn De Baerdemacker, and Yvan Van Rentergen
2018

Boolean Differential Calculus
Bernd Steinbach and Christian Posthoff
2017

Embedded Systems Design with Texas Instruments MSP432 32-bit Processor
Dung Dang, Daniel J. Pack, and Steven F. Barrett
2016

Fundamentals of Electronics: Book 4 Oscillators and Advanced Electronics Topics
Thomas F. Schubert and Ernest M. Kim
2016

Fundamentals of Electronics: Book 3 Active Filters and Amplifier Frequency
Thomas F. Schubert and Ernest M. Kim
2016

Bad to the Bone: Crafting Electronic Systems with BeagleBone and BeagleBone Black, Second Edition
Steven F. Barrett and Jason Kridner
2015

Fundamentals of Electronics: Book 2 Amplifiers: Analysis and Design
Thomas F. Schubert and Ernest M. Kim
2015

Fundamentals of Electronics: Book 1 Electronic Devices and Circuit Applications
Thomas F. Schubert and Ernest M. Kim
2015

Applications of Zero-Suppressed Decision Diagrams
Tsutomu Sasao and Jon T. Butler
2014

Modeling Digital Switching Circuits with Linear Algebra
Mitchell A. Thornton
2014

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. Shmerko
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

Microcontroller Programming and Interfacing with Texas Instruments MSP430FR2433 and MSP430FR5994, Second Edition

Steven F. Barrett and Daniel J. Pack

www.morganclaypool.com

ISBN: 9781681736242 paperback
ISBN: 9781681736259 ebook
ISBN: 9781681736273 hardcover
DOI 10.2200/S00936ED2V01Y201907DCS055
A Publication in the Morgan & Claypool Publishers series
SYNTHESIS LECTURES ON DIGITAL CIRCUITS AND SYSTEMS

Lecture #55
Series Editor: Mitchell A. Thornton, *Southern Methodist University*
Series ISSN
Print 1932-3166 Electronic 1932-3174

Microcontroller Programming and Interfacing with Texas Instruments MSP430FR2433 and MSP430FR5994 – Part I

Second Edition

Steven F. Barrett
University of Wyoming, Laramie, WY

Daniel J. Pack
University of Tennessee Chattanooga, TN

SYNTHESIS LECTURES ON DIGITAL CIRCUITS AND SYSTEMS #55

MORGAN & CLAYPOOL PUBLISHERS

ABSTRACT

This book provides a thorough introduction to the Texas Instruments MSP430™ microcontroller. The MSP430 is a 16-bit reduced instruction set (RISC) processor that features ultra-low power consumption and integrated digital and analog hardware. Variants of the MSP430 microcontroller have been in production since 1993. This provides for a host of MSP430 products including evaluation boards, compilers, software examples, and documentation. A thorough introduction to the MSP430 line of microcontrollers, programming techniques, and interface concepts are provided along with considerable tutorial information with many illustrated examples. Each chapter provides laboratory exercises to apply what has been presented in the chapter. The book is intended for an upper level undergraduate course in microcontrollers or mechatronics but may also be used as a reference for capstone design projects. Also, practicing engineers already familiar with another microcontroller, who require a quick tutorial on the microcontroller, will find this book very useful. This second edition introduces the MSP–EXP430FR5994 and the MSP430–EXP430FR2433 LaunchPads. Both LaunchPads are equipped with a variety of peripherals and Ferroelectric Random Access Memory (FRAM). FRAM is a nonvolatile, low-power memory with functionality similar to flash memory.

KEYWORDS

MSP430 microcontroller, microcontroller interfacing, embedded systems design, Texas Instruments

To our families

Contents

Preface

Texas Instruments is well known for its analog and digital devices, in particular, Digital Signal Processing (DSP) chips. Unknown to many, the company quietly developed its microcontroller division in the early 1990s and started producing a family of controllers aimed mainly for embedded meter applications, which require an extended operating time without intervention for power companies. It was not until the mid 2000s that the company began serious effort to present the MSP430 microcontroller family, its flagship microcontroller, to the academic community and future engineers. Their efforts have been quietly attracting many educators and students due to the MSP430's cost and the suitability of the controller for capstone design projects requiring microcontrollers. In addition, Texas Instruments offers many compatible analog and digital devices that can expand the range of the possible embedded applications of the microcontroller. Texas Instruments has continually added new innovation to the MSP430 microcontroller line. The second edition introduces the MSP–EXP430FR5994 and the MSP–EXP430FR2433 LaunchPads. Both LaunchPads are equipped with a variety of peripherals and Ferroelectric Random Access Memory (FRAM). FRAM is a nonvolatile, low-power memory with functionality similar to flash memory.

This book is about the MSP430 microcontroller family. We have three goals in writing this book. The first is to introduce readers to microcontroller programming. The MSP430 microcontrollers can be programmed either using assembly language or a high–level programming language such as C. The second goal of the book is to teach students how computers work. After all, a microcontroller is a computer within a single integrated circuit (chip). Finally, we present the microcontroller's input/output interface capabilities, one of the main reasons for developing embedded systems with microcontrollers.

Background

This book provides a thorough introduction to the Texas Instruments MSP430 microcontroller. The MSP430 is a 16-bit reduced instruction set (RISC) processor that features ultra-low power consumption and integrated digital and analog hardware. Variants of the MSP430 microcontroller have been in production since 1993 with a host of MSP430-related products including evaluation boards, compilers, software examples, and documentation.

This book is intentionally tutorial in nature with many worked examples, illustrations, and laboratory exercises. An emphasis is placed on real-world applications such as smart home concepts, mobile robots, an unmanned underwater vehicle, and a DC motor controller to name a few.

Intended Readers

The book is intended for an upper level undergraduate course in microcontrollers or mechatronics but may also be used as a reference for capstone design projects. Also, practicing engineers who are already familiar with another line of microcontrollers, but require a quick tutorial on the MSP430 microcontroller, will find this book beneficial.

Approach and Organization

This book provides a thorough introduction to the MSP430 line of microcontrollers, programming techniques, and interface concepts. Each chapter contains a list of objectives, background tutorial information, and detailed information on the operation of the MSP430 system under study. Furthermore, each chapter provides laboratory exercises to apply what has been presented in the chapter and how the concepts are employed in real applications. Each chapter concludes with a series of homework exercises divided into Fundamental, Advanced, and Challenging categories. The reader will get the most out of the book by also having the following references readily available:

- MSP430FR2433 Mixed–Signal Microcontroller, SLASE59B;
- MSP430FR4xx and MSP430FR2xx Family User's Guide, SLAU445G;
- MSP430FR599x, MSP430FR596x Mixed–Signal Microcontrollers, SLASE54B; and
- MSP430FR58xx, MSP430FR59xx, and MSP430FR6xx Family User's Guide, SLAU367O.

This documentation is available for download from the Texas Instruments website [www.ti.com].

Chapter 1 provides a brief review of microcontroller terminology and a short history followed by an overview of the MSP430 microcontroller. The overview surveys systems onboard the microcontroller and also various MSP430 families. The chapter concludes with an introduction to the hardware and software development tools that will be used for the remainder of the book. Our examples employ the MSP–EXP430FR5994 and the MSP430FR2433 LaunchPads, the Energia rapid prototyping platform, and the Texas Instruments' Code Composer Studio Integrated Development Environment (IDE). The information provided can be readily adapted to other MSP430 based experimenter boards.

Chapter 2 provides a brief introduction to programming in C. The chapter contains multiple examples for a new programmer. It also serves as a good review for seasoned programmers. Also, software programming tools including Energia, Code Composer Studio IDE, and debugging tools are explored. This chapter was adapted from material originally written for the Texas Instruments MSP432, a 32-bit processor that has close ties to the 16-bit MSP430.[1] Embed-

[1]This chapter was adapted with permission from *Arduino Microcontroller Processing for Everyone*, S. Barrett, 3rd ed., Morgan & Claypool Publishers, San Rafael, CA, 2013.

ded system developers will find a seamless transition between the MSP430 and MSP432 line of processors.

Chapter 3 introduces the MSP430 hardware architecture, software organization, and programming model. The chapter also presents an introduction to the MSP430 orthogonal instruction set, including its 27 instructions and 9 emulated instructions.

Chapter 4 describes a wide variety of input and output devices and how to properly interface them to the MSP430 microcontroller. We believe it is essential for the embedded system designer to understand the electrical characteristics of the processor so a proper interface to peripheral components may be designed. We have included a chapter on these concepts for the books we have written for the Synthesis Lecture Series. We continue to add material as new microcontroller peripherals are developed. The chapter begins with a review of the MSP430 electrical operating parameters followed by a discussion of the port system. The chapter includes a description of a wide variety of input device concepts including switches, interfacing, debouncing, and sensors. Output device concepts are then discussed including light-emitting diodes (LEDs), tri–state LED indicators, liquid crystal displays (LCDs), high-power DC and AC devices, motors, and annunciator devices.

Chapter 5 provides an in–depth discussion of the MSP430 power management system. The power management system provides for ultra-low power operation and practices.

Chapter 6 is dedicated to the different memory components onboard the MSP430 including the new FRAM nonvolatile memory, RAM, EEPROM and the associated memory controllers. The Direct Memory Access (DMA) controller is also discussed.

Chapter 6.11 discusses the clock and timer systems aboard the MSP430. The chapter begins with a detailed discussion of the flexible clock system, followed by a discussion of the timer system architecture. The timer architecture discussion includes the Watchdog timers, timers A and B, real-time clocks, and pulse width modulation (PWM).

Chapter 6.11 provides an introduction to the concepts of resets and interrupts. The various interrupt systems associated with the MSP430 are discussed, followed by detailed instructions on how to properly configure and program them.

Chapter 6.11 discusses the analog systems aboard the MSP430. The chapter discusses the analog–to–digital converters (ADCs), the digital–to–analog converters (DACs), and the comparators.

Chapter 6.11 is designed for a detailed review of the complement of serial communication systems resident onboard the MSP430, including the universal asynchronous receiver transmitter (UART), the serial peripheral interface (SPI), the I2C system, the radio frequency (RF) link, USB, and the IrDA infrared link. The systems are contained within the MSP430 universal serial communication interfaces eUSCI_A and eUSCI_B subsystems.

Chapter 6.11 provides a detailed introduction to the data integrity features aboard the MSP430 including a discussion of noise and its sources and suppression, an Advanced Encryption Standard (AES) 256 accelerator module, and a 16- or 32-bit cyclic redundancy check

(CRC) engine. This chapter was adapted from material originally written for the Texas Instruments MSP432, a 32-bit processor that has close ties to the 16-bit MSP430.[2] Embedded system developers will find a seamless transition between the MSP430 and MSP432 line of processors.

Chapter 6.11 discusses the system design process followed by system level examples. We view the microcontroller as the key component within the larger host system. It is essential the embedded system designer has development, design, and project management skills to successfully complete a project. This chapter provides an introduction some of the skills used for project development. We have included a chapter on these concepts for the books we have written for the Synthesis Lecture Series. The examples have been carefully chosen to employ a wide variety of MSP430 systems discussed throughout the book.

Table 1 provides a summary of chapter contents and related MSP430 subsystems.

Steven F. Barrett and Daniel J. Pack
July 2019

[2] *Embedded Systems Design with the Texas Instruments MSP432 32-bit Processor*, Dung Dang, Daniel J. Pack, and Steven F. Barrett, Morgan & Claypool Publishers, San Rafael, CA, 2017.

Table 1: MSP-EXP430FR5994 and the MSP-EXP430FR2433 LaunchPad subsystems.

Chapter	**MSP- EXP430FR2433**	**MSP-EXP430FR5994**
Ch. 1: Introduction		
Ch. 2: Programming	MSP430 port system	MSP430 port system
Ch. 3: HW and SW	Joint Test Action Group (JTAG) serial debug port, Enhanced Emulation Module (EEM) onboard debug tool, serial Spy-Bi-Wire (SBY) JTAG	Joint Test Action Group (JTAG) serial debug port, Enhanced Emulation Module (EEM) on-board debug tool, serial Spy-Bi-Wire (SBY) JTAG
Ch. 4: Interfacing	MSP430 port system	MSP430 port system
Ch. 5: Power Mgt	Power Mgt Module	Power Mgt: LDO, SVS, Brownout
Ch. 6: Memory	FRAM: 15KB + 512B RAM: 4KB	FRAM: 256KB RAM: 4 KB + 4 KB DMA Controller Memory Protection Unit (MPU) IP Encapsulation Segment (IPE)
Ch. 7: Timer Systems - Clock - Timers	Clock system (CS), LFXT Timer_A3(2), Timer_A2(2) Watchdog, Real-Time Clock	Clock system (CS), TB0: Timer_B, TA0: Timer_A, TA1: Timer_A, TA4: Timer_A, Watchdog, Real-Time Clock
Ch. 8: Resets and Interrupts		
Ch. 9: Analog Peripherals	ADC: 8 ch, SE, 10-bit, 200 ksps	Comp_E: 16 ch, Ref_A ADC 12_B: 16 ch SE/8 DE, 12-bit
Ch. 10: Comm Sys	eUSCI_A(2) - UART, IrDA, SPI eUSCI_B0 - SPI, I2C	eUSCI_A(4) (A0 to A3) - UART, IrDA, SPI eUSCI_B(4) (B0 to B3) - SPI, I2C
Ch. 11: System Integrity	CRC16: 16-bit cyclic redundancy check	CRC16: CRC-16-CCITT CRC32: CRC-32-ISO-3309 AES 256: security encryption/decryption
Ch. 12: System Design		

Acknowledgments

There have been many people involved in the conception and production of this book. We especially want to thank Doug Phillips, Mark Easley, and Franklin Cooper of Texas Instruments. The future of Texas Instruments is bright with such helpful, dedicated engineering and staff members. In 2005, Joel Claypool of Morgan & Claypool Publishers, invited us to write a book on microcontrollers for his new series titled "Synthesis Lectures on Digital Circuits and Systems." The result was the book *Microcontrollers Fundamentals for Engineers and Scientists.* Since then we have been regular contributors to the series. Our goal has been to provide the fundamental concepts common to all microcontrollers and then apply the concepts to the specific microcontroller under discussion. We believe that once you have mastered these fundamental concepts, they are easily transportable to different processors. As with many other projects, he has provided his publishing expertise to convert our final draft into a finished product. We thank him for his support on this project and many others. He has provided many novice writers the opportunity to become published authors. His vision and expertise in the publishing world made this book possible. We thank Sara Kreisman of Rambling Rose Press, Inc. for her editorial expertise. We also thank Dr. C.L. Tondo of T&T TechWorks, Inc. and his staff for working their magic to convert our final draft into a beautiful book. Finally, we thank our families who have provided their ongoing support and understanding while we worked on books over the past fifteen plus years.

Steven F. Barrett and Daniel J. Pack
July 2019

CHAPTER 1

Introduction to Microcontroller Technology

Objectives: After reading this chapter, the reader should be able to:

- describe the key technological accomplishments leading to the development of the microcontroller;
- define microprocessor, microcontroller, and microcomputer;
- identify examples of microcontroller applications in daily life;
- list key attributes of the MSP430 microcontroller;
- describe different features that differentiate MSP430 microcontroller family members;
- list the subsystems onboard the MSP430FR2433 and the MSP430FR5994 microcontrollers;
- provide an example application for each subsystem onboard the MSP430 microcontrollers;
- describe the hardware, software, and emulation tools available for the MSP430 microcontrollers; and
- employ the development tools to load and execute simple programs on the MSP-EXP430FR2433 and the MSP-EXP430FR5994 evaluation boards.

In every chapter, we start with a motivation and background followed by a section on theory. After the theory section, an example application is used to demonstrate the operational use of chapter concepts. Each chapter includes a hands-on laboratory exercise and a list of chapter references, which you can use to explore further areas of interest. Each chapter concludes with a series of practice exercises, divided into Fundamental, Advanced, and Challenging levels.

1.1 MOTIVATION

This book is about microcontrollers! A microcontroller is a self-contained processor system in a single integrated circuit (IC or chip) that contains essential functional units of a general-purpose computer such as a central processing unit (CPU), a memory, and input/output (I/O) units.

Microcontrollers provide local computational resources to many products, requiring a limited amount of processing power to perform their functions. They are everywhere! In the routine of daily life, we use multiple microcontrollers. Take a few minutes and jot down a list of microcontroller equipped products, sometimes called embedded systems, you have used today.

This chapter introduces the Texas Instruments MSP430 line of microcontrollers. We begin with a brief history of computer technology followed by an introduction to the MSP430FR2433 and the MSP430FR5994 microcontrollers. After a review of these MSP430 microcontrollers, we introduce you to the powerful and user-friendly development tools.

1.2 BACKGROUND THEORY: A BRIEF HISTORY AND TERMINOLOGY

The development of microcontrollers can be traced back to the time of early computing with the first generation of computers. The generations of computer development are marked by breakthroughs in hardware and architecture innovation. The first generation of computers employed vacuum tubes as the main switching element. Mauchly and Eckert developed the electronic numerical integrator and calculator (ENIAC) in the mid 1940s. This computer was large and consumed considerable power due to its use of 18,000 vacuum tubes. The computer, funded by the U.S. Army, was employed to calculate ordnance trajectories in World War II. The first commercially available computer of this era was the UNIVAC I [Bartee, 1972].

The second generation of computers employed transistors as the main switching element. The transistor was developed in 1947 by John Bardeen and Walter Brattain at Bell Telephone Laboratories. Bardeen, Brattain, and William Schockley were awarded the 1956 Nobel Prize in Physics for development of the transistor [Nobel.org]. The transistor reduced the cost, size, and power consumption of computers.

The third generation of processors started with the development of the integrated circuit. The integrated circuit was developed by Jack Kilby at Texas Instruments in 1958. The integrated circuit revolutionized the production of computers, greatly reducing their size and power consumption. Computers employing integrated circuits were first launched in 1965 [Bartee, 1972], and Kilby was awarded the Nobel Prize in Physics in 2002 for his work on the integrated circuit [Nobel.org]. The first commercially available minicomputer of this generation was the digital equipment corporation's (DEC) PDP-8 [Osborne, 1980].

The fourth generation of computers was marked by the advancement of levels of integration, leading to very large-scale integration (VLSI) and ultra-large scale integration (ULSI) production techniques. In 1969, the Data Point Corporation of San Antonio, Texas had designed an elementary CPU. The CPU provides the arithmetic and control for a computer. Data Point contracted with Intel and Texas Instruments to place the design on a single integrated circuit. Intel was able to complete the task, but Data Point rejected the processor as being too slow for their intended application [Osborne, 1980].

Intel used the project as the basis for their first general-purpose 8-bit microprocessor, the Intel 8008. The microprocessor chip housed the arithmetic and control unit for the computer. Other related components such as memory (ROM), random access memory (RAM), I/O components, and interface hardware were contained in external chips. From 1971–1977, Intel released the 8008, 8080, and 8085 microprocessors which significantly reduced the number of system components and improved upon the number of power supply voltages required for the chips. Some of the high visibility products of this generation were the Apple II personal computer, developed by Steve Jobs and Steve Wozniak and released in 1977, and the IBM personal computer, released in 1981 [MCS 85, 1977, Osborne, 1980].

The first single-chip microcontroller was developed by Gary Boone of Texas Instruments in the early 1970s. A microcontroller contains all key elements of a computer system within a single integrated circuit. Boone's first microcontroller, the TMS 1000, contained the CPU, ROM, RAM, and I/O and featured a 400 kHz clock [Boone, 1973, 1978]. From this early launch of microcontrollers, an entire industry was launched. There are now over 35 plus companies manufacturing microcontrollers worldwide offering over 250 different product lines [Wendt]. The MSP430 line of microcontrollers was first developed in 1992 and became available for worldwide release in 1997.

1.3 MICROCONTROLLER SYSTEMS

Although today's microcontrollers physically bear no resemblance to their earlier computer predecessors, they all have a similar architecture. All computers share the basic systems shown in Figure 1.1. The processor or CPU contains both datapath and control hardware. The datapath is often referred to as the arithmetic logic unit (ALU). As its name implies, the ALU provides hardware to perform the mathematical and logic operations for the computer. The control unit provides an interface between the computer's hardware and software. It generates control signals to the datapath and other system components such that operations occur in the correct order and within an appropriate time to execute the desired actions of a software program.

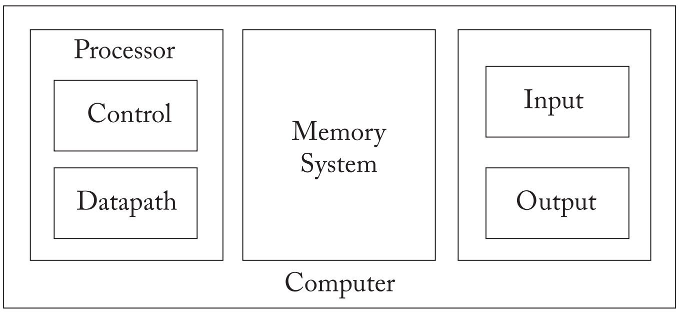

Figure 1.1: Basic computer architecture. (Adapted from Patterson and Hennessy [1994].)

The memory system contains a variety of memory components to support the operation of the computer. Typical memory systems aboard microcontrollers contain RAM, ROM, and electrically erasable programmable read only memory (EEPROM) components. RAM is volatile. When power is unavailable, the contents of RAM memory is lost. RAM is typically used in microcontroller operations for storing global variables, local variables, which are required during execution of a function, and to support heap operations during dynamic allocation activities. In contrast, ROM memory is nonvolatile. That is, it retains its contents even when power is not available. ROM memory is used to store system constants and parameters. If a microcontroller application is going to be mass produced, the resident application program may also be written into ROM memory at the manufacturer.

EEPROM is available in two variants: byte-addressable and flash programmable. Byte-addressable memory EEPROM, as its name implies, allows variables to be stored, read, and written during program execution. The access time for byte-addressable EEPROM is much slower than RAM memory; however, when power is lost, the EEPROM memory retains its contents. Byte-addressable EEPROM may be used to store system passwords and constants. For example, if a microcontroller-based algorithm has been developed to control the operation of a wide range of industrial doors, system constants for a specific door type can be programmed into the microcontroller onsite when the door is installed. Flash EEPROM can be erased or programmed in bulk. It is typically used to store an entire program.

Ferroelectric Random Access Memory (FRAM) is a nonvolatile, ultra-low power (ULP) with access speeds similar to RAM. It has been termed a universal memory because it can be used for storing program code, variables, constants, and for stack operations. Note these functions are typically performed by nonvolatile ROM and volatile RAM. FRAM also has a high level of write endurance on the order of 10^{15} cycles [SLAA526A, 2014, SLAA628, 2014].

The input and output system of a microcontroller usually consists of a complement of ports. Ports are fixed sized hardware registers that allow for the orderly transfer of data in and out of the microcontroller. In most microcontroller systems, ports are equipped for dual use. That is, they may be used for general-purpose digital I/O operations or may have alternate functions such as input access for the analog-to-digital (ADC) system.

Our discussion thus far has been about microcontrollers in general. For the remainder of this chapter and the rest of the book, we concentrate on the Texas Instruments MSP430 microcontroller, specifically the MSP430FR2433 and the MSP430FR5994.

1.4 WHY THE TEXAS INSTRUMENTS MSP430?

The MSP430 line of microcontrollers began development in 1992. Since this initial start, there have been multiple families of the microcontroller developed and produced with a wide range of features. This allows one to choose an appropriate microcontroller for a specific application. Texas Instruments invests considerable resources in providing support documentation, development tools, and instructional aids for this processor family.