Example files associated with the three books

Volume 1 Embedded Systems: Introduction to ARM Cortex M Microcontrollers Fourth Edition (new 10/2013) Available from Amazon e-book, Fourth edition includes material on TM4C123.

Volume 2 Embedded Systems: Real-Time Interfacing to ARM Cortex M Microcontrollers Errata Third Edition (new 11/2013) Available from Amazon e-book Third edition includes material on TM4C123.

Volume 3 Embedded Systems: Real-Time Operating Systems for ARM Cortex M Microcontrollers Errata Second edition (new 1/2014) Available on Amazon Second edition includes material on TM4C123.

These project files run on the LM3S811, LM3S1968, LM3S8962, LM4F120, or TM4C123 microcontrollers and will compile using the Keil uVision4 C compiler. Most of the examples also are configured to run on Texas Instrument's Code Composer Studio (for instructions on using CCS, refer to Appendix 3 of Volumes 1 or 2) These files are Copyright by Jonathan W. Valvano. You may use, edit, run or distribute these files as long as the copyright notices within the files remain. No specific warrantee exists concerning the accuracy or reliability of these examples. I think they work, but history has shown, sometimes I can be wrong. If you want to get more information on the example, click on the C file. The zip files are complete uVision4 projects. How to convert a project based on one Stellaris microcontroller into a project for a different microcontroller

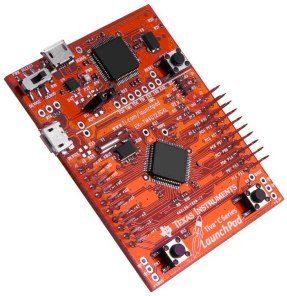
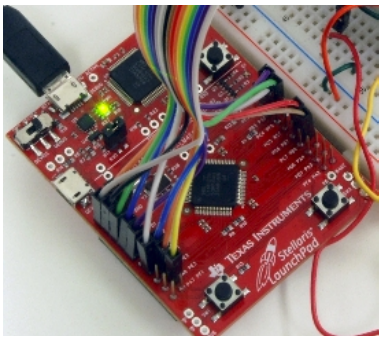
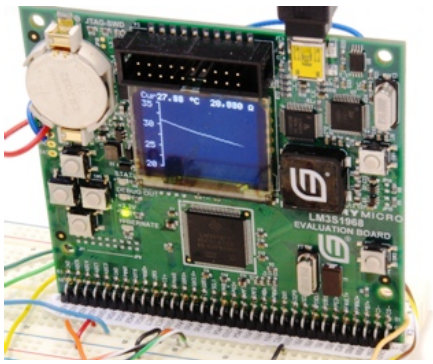
Go to Home Page

Keil uVision Reference Manual and instructions for download and setup
Installing windows drivers for the Launchpad
How to install drivers on a Windows 8 machine

To download all LM3S1968 software ValvanoWare_1968.zip
To download many TM4C123/LM4F123 software LaunchPadware.zip (include introductory software needed for EE319K Fall 2013)

To build the directory for the LM3S811, LM3S1968, or LM3S8962 is
ValvanoWare
_driverlib
_inc
LCD_1968 (any LM3S example on this page)

To build a directory for the LM4F120 or TM4C123
LaunchPadware
_driverlib (with driverlib-cm4f.lib in it)
_inc
GPIO_4F120 (any LM4F120 example on this page)

All of these LM4F120 examples will run on the new Tiva TM4C123 microcontroller.

Videos
How to change the name of an assembly project
How to build a not gate

Stellaris LaunchPad Tester booster pack and software application designed by Daniel Valvano

Assembly Language Examples for Volume 1

Link to download	Book reference	External Hardware	Ports used	Description
lm3s1968.s lm4f120.s			All	These assembly files contain all the port addresses for the microcontroller. To use this file, you open this file copy the lines you need and paste the lines into your program. You cannot include this file, like you can include the lm3S1968.h file in C

SimpleProject_811asm.zip SimpleProject_1968asm.zip SimpleProject_4F120asm.zip	Program 3.2	none	none	Random number generator using
Switch_811asm.zip Switch_1968asm.zip Switch_4F120asm.zip	Program 4.2	One switch	GPIO	Input from switch
GPIO_811asm.zip GPIO_1968asm.zip GPIO_4F120asm.zip	Program 4.5, Example 4.2	Four LEDs	GPIO	Output pattern 5 6 10 9 to Port D
InputOutput_811asm.zip InputOutput_1968asm.zip InputOutput_4F120asm.zip	Section 4.2	On board switches and LED	GPIO	Functional abstraction of the switches and LED. Switches include internal pull-up resistors.
NotGate_1968asm.zip GPIO_PORTG_1968asm.zip	Chapter 4 stuff	Switch, LED	Port D GPIO	Input on PD0, not gate, output on PD1, see http://www.youtube.com/watch?v=cg2EuTgJF7Y PORTG version uses SELECT and LED on LM3S1968 board
Lab2starter.zip			GPIO	EE319K lab 2 starter file with configuration for logic analyzer
SSR_1968asm.zip SSR_4F120asm.zip	Program 4.3	Switch and LED	GPIO	Switch input and LED (or SSR) output
Squarewaves_1968asm.zip SquarewavePG2_1968asm.zip Squarewaves_4F120asm.zip	Program 4.4		GPIO	Continuous output of two pins creating two squarewaves; PG2 version toggles LED every 1 second
PLL_811asm.zip PLL_1968asm.zip PLL_4F120asm.zip	Program 4.6	PG2 LED	PLL, GPIO	PLL used to change clock speed, LED flashes
SysTick_811asm.zip SysTick_1968asm.zip SysTick_4F120asm.zip	Program 4.6, Program 4.7	PG2 LED	PLL, SysTick, GPIO	LED flashes at constant rate, SysTick used to implement a time delay
Performance_1968asm.zip	Program 4.10, Section 4.7	PG2 LED	PLL, SysTick, GPIO	Performance measurements on a square root function
PointerTrafficLight_1968asm.zip	Example 6.6, Program 6.8	LEDs and switches	PLL, SysTick, GPIO	Finite state machine, linked structure
OLED_1968asm.zip	Example 6.5, Program 6.11-6.13	OLED	SSI	Graphics driver for OLED, bit matrix, graphics buffer
LLFifo_1968asm.zip	Section 6.6, Program 6.11-6.18	none	none	Linked list FIFO, dynamic memory manager
PeriodicSysTickInts_811asm.zip PeriodicSysTickInts_1968asm.zip PeriodicSysTickInts_4F120asm.zip	Program 9.7	LED	PLL, SysTick, GPIO	Interrupts are used to create a periodic task, need scope on LED output to debug (oscillates at 1000 Hz)
LinearInterpolation_1968asm.zip	Program 6.22			Linear interpolation, sine function
MealyEngineControl_1968asm.zip	Program 6.9			Mealy finite state machine
PeriodicTimer0AInts_1968asm.zip PeriodicTimer0AInts_4F120asm.zip	Program 9.8			Timer interrupt
FIFO_4F120asm.zip FIFO.s		none	none	Pointer implementation of FIFO queue, size must be a power of 2. Implemented on the LM4F120, but will run on any Cortex M.
UART2_4F120asm.zip UART2.s	Program 5.11	connection to PC	UART0	Interrupt serial port, transmit receive interrupts, both hardware and software FIFO queues

C Examples from Volumes 1, 2, and 3

Link to download	Book reference	External Hardware	Ports used	Description

SSR_811.zip SSR_1968.zip SSR_4F120.zip SSR.c	Example 2.1	Solid state relay	GPIO	Provide functions that initialize a GPIO pin and turn it on and off. Use bit-banded I/O.
GPIO_811.zip GPIO_1968.zip GPIO_4F120.zip GPIO.c	Example 2.2	Four LEDs	GPIO	Initialize four GPIO pins as outputs. Continually generate output to drive simulated stepper motor.
Switch_811.zip Switch_1968.zip Switch_4F120.zip Switch.c	Example 2.3	One switch	GPIO	Provide functions that initialize a GPIO as input, and allows software to read the status of a switch.
PLL_811.zip PLL_1968.zip PLL_4F120.zip PLL.c	Example 2.4		PLL	A software function to change the bus speed using the PLL.
SysTick_811.zip SysTick_1968.zip SysTick_4F120.zip SysTick_wait_4F120.zip SysTick.c	Program 2.11		SysTick	Provide functions that initialize the SysTick module, wait at least a designated number of clock cycles, and wait approximately a multiple of 10 milliseconds using busy wait.
OLED_811.zip OLED_1968.zip OLED_8962.zip Output.c	Section 3.4.5	oLED on the board	On the 811, uses I2C, on the 1968 and 8962 used SSI	Abstraction of the OLED as a general purpose output device allowing the use of printf to stream to the OLED
Logo_1968.zip Logo_8962.zip logo.c		OLED on board	SSI	Start with a 4-bit (16 color) BMP file less than 128 by 80, convert it to a ROM buffer in the LM3S, then display it on the OLED. Includes a DOS executable, BmpConvert.exe, used to convert BMP to C source code for image data.
TableTrafficLight_811.zip TableTrafficLight.c	Program 3.1, Example 3.1	Red, Yellow, Green LEDs, resistors, drivers, switches	GPIO, SysTick	Use a table implementation of a Moore finite state machine to operate a traffic light.
PointerTrafficLight_811.zip PointerTrafficLight_1968.zip PointerTrafficLight.c	Program 3.2, Example 3.1	Red, Yellow, Green LEDs, resistors, drivers, switches	GPIO, SysTick, PLL	Use a pointer implementation of a Moore finite state machine to operate a traffic light.
PortableTrafficLight_811.zip PortableTrafficLight.c	Program 3.3	Red, Yellow, Green LEDs, resistors, drivers, switches	GPIO, SysTick	Use a table implementation of a Moore finite state machine to operate a traffic light. This time, only the "#define" section is processor specific, and the rest of the code can easily be adapted to another system.
PointerRobot_811.zip PointerRobot.c	Example 3.2, Program 3.4	LEDs and switches simulate robot functions	GPIO	Use a pointer implementation of a Mealy finite state machine to control a robot.
FunctionRobot_811.zip FunctionRobot.c	Example 3.2, Program 3.5	LEDs and switches simulate robot functions	GPIO	Use a function pointer implementation of a Mealy finite state machine to control a robot. Use bit-banded I/O.
MealyEngineControl_1968.zip	Volume 1, Program 6.9			Mealy finite state machine
FIFO_811.zip FIFO_1968.zip FIFO.c FIFO.h	Programs 3.7, 3.8, 3.9 and 3.10		GPIO, Timer0A periodic interrupts	First in first out queue, pointer method and index method Provide functions that initialize a FIFO, put data in, get data out, and return the current size. The file includes an index and a pointer implementation and macros to create more FIFOs. Periodic interrupts are used to verify the FIFO has no critical sections.
		-		
HeapFixedBlock_8962.zip Heap.c Heap.h	Program 3.11, Volume 3, Program 3.1, Section 3.2.2	-	-	Fixed size memory manager. Allocate memory block, and deallocated block.
Clock_811.zip Clock.c			SysTick	Provide functions that initialize the SysTick module, wait at least a designated number of clock cycles, and wait approximately a multiple of 1 millisecond.
Performance_811.zip Performance_1968.zip Performance.c	Program 3.17		SysTick	Use the SysTick timer to measure approximately how long it takes to calculate a square root.
LCD_1968.zip lcd.c	Section 4.7.1, Program 4.2, Program 4.3	HD44780 LCD	GPIO, SysTick	LCD interface using 8-bit parallel port mode, blind-cycle synchronization using SysTick timer. Will work with both 3.3V and 5V devices
Nokia5110_4F120.zip Nokia5110.c		Nokia 5110	SSI0	48x84 LCD graphics; example can output characters and draw images on the screen, https://www.sparkfun.com/products/10168
ST7735.c ST7735_4F120.zip		ST7735	SSI0, PPL, Systick	128x160 pixels, 1.8" 18-bit color TFT LCD display; example can output characters and draw images on the screen, http://www.adafruit.com/products/358

SSD2119.c we are working on this example		Kentec EB-LM4F120-L35	All PortB, PA7-4	SSD2119 interface on a 320x240 pixels, 16-bit color, 3.5 in, 15 pin
Stepper_811.zip Stepper_4F120.zip stepper.c	Example 4.1, Programs 4.4, 4.5 and 4..6	Stepper motor	GPIO, SysTick	Provide functions that step the motor once clockwise, step once counterclockwise, initialize the stepper motor interface, and turn the motor to the valid desired position.
ParallelKeypad_811.zip ParallelKeypad.c	Example 4.2, Program 4.7	Parallel keypad	GPIO	Use a busy-wait loop to wait for a rising edge and then return the parallel data from the keypad.
ParallelADC_811.zip ParallelPrinter_811.zip ParallelPrinter.c		Parallel ADC	GPIO	Use a handshaked interface to request an ADC conversion, wait for a rising edge signaling completion, and return the parallel result.
Handshake_1968.zip	Example 4.3, Program 4.8	Parallel printer	GPIO	Use a handshaked interface to request a character print while outputting a parallel character and wait for a rising edge signaling completion.
Receiver.c Transmitter.c	Example 4.4, Programs 4.9, 4.10, and 4.11	Two microcontrollers	GPIO	Two microcontrollers interfaced via parallel ports, synchronization with busy-wait, handshaked protocol
UART_811.zip UART_1968.zip UART_8962.zip UART_4F120.zip UART.c	Program 4.12	Virtual COM port through debugger USB	UART	Provide functions that initialize the UART, wait for and return a character, and print a character.
MatrixKeypad_1968.zip MatrixKeypad.c	Example 4.5, Programs 4.13, 4.14	4 by 4 matrix keyboard	GPIO	Busy-wait synchronization of a keyboard. Row by row scanning of the matrix keyboard. Not debounced.
EdgeInterrupt_811.zip EdgeInterrupt_4F120.zip EdgeInterrupt.c	Program 5.6	External switch	GPIO	Request an interrupt on the falling edge of PC4 (when the user button is pressed) and increment a counter in the interrupt. Note that button bouncing is not addressed.
TwoButtonVector_811.zip TwoButtonVector.c	Example 5.1, Program 5.7	One button and resistor	GPIO	Use vectored interrupts to respond to two button presses. Note that button bouncing is not addressed.
TwoButtonPoll_811.zip	Example 5.1, Program 5.8	Two buttons and resistors	GPIO	Use polled interrupts to respond to two button presses. Note that button bouncing is not addressed.
ParallelKeypadInt_811.zip	Example 5.2, Program 5.9	Parallel keypad	GPIO	Use an interrupt on the rising edge and then return the parallel data from the keypad.
ParallelPrinterInt_811.zip	Example 5.3, Program 5.10	Parallel printer	GPIO	Use a handshaked interface synchronized with interrupts to request a character print on a parallel printer. Interrupt when the printing is complete.
TwoButtonFunct_811.zip	old version	Two buttons and resistors	GPIO	Use vectored interrupts to run functions when buttons are pressed. One button and function pair is higher priority.
UART2_811.zip UART2_1968.zip UART2_4F120.zip UART2.c	Program 5.11		UART	Use UART0 to implement bidirectional data transfer to and from a computer running HyperTerminal. This time, interrupts and FIFOs are used.
PeriodicSysTickInts_811.zip PeriodicSysTickInts_1968.zip PeriodicSysTickInts_2110.zip PeriodicSysTickInts_8962.zip PeriodicSysTickInts_4F120.zip PeriodicSysTickInts.c	Program 5.12		GPIO, SysTick	Periodic interrupts using SysTick. Software allows you to select the interrupt period and attach a user program (hook)
MatrixKeypadPeriodic_1968.zip Matrix.c	Example 5.4, Figure 5.18, Program 5.13	4 by 4 matrix keyboard	GPIO, SysTick	Periodic polling synchronization of a keyboard. Row by row scanning of the matrix keyboard occurs during a period SysTick ISR. Data passed via a FIFO. This solution debounces the keyboard.
InputCapture_811.zip InputCapture_1968.zip InputCapture_4F120.zip InputCapture.c	Program 6.1		GPIO, Timer0A	Use Timer0A in edge time mode to request interrupts on the rising edge of PD4 (CCP0), and count the pulses.
PeriodMeasure_811.zip PeriodMeasure_8962.zip PeriodMeasure_4F120.zip PeriodMeasure.c	Example 6.2, Program 6.2		GPIO, Timer0A	Use Timer0A in edge time mode to request interrupts on the rising edge of PD4 (CCP0), and measure period between pulses.
PeriodicTimer0AInts_811.zip PeriodicTimer0AInts_1968.zip PeriodicTimer0AInts_4F120.zip Periodic32bitT0Ints_4F120.zip Timer0A.c	Program 6.5, Example 6.6		GPIO, Timer0A, PLL	Use Timer0A in periodic mode to request interrupts at a particular period. Uses the timer prescale.
PulseCount_811.zip			GPIO, Timer0A	Use Timer0A in edge time mode to request interrupts on the rising edge of PD4 (CCP0), and count the pulses.
HighPulseMeasure_811.zip			GPIO, Timer0A	Use Timer0A in edge time mode to record time at rising and falling edges of PD4 (CCP0), and subtract them to get high pulse duration.
HighPulseMeasureInts_811.zip			GPIO, SysTick	Use Timer0A in edge time mode to request interrupts on both edges of PD4 (CCP0), determine which edge occurred, and subtract times to get period.
HighPulseMeasureHW_811.zip			Timer0	Use both subtimers of Timer0 to record time at rising and falling edges of the signal. An interrupt makes the data available, but measurement is in hardware.
Timer0APWM_811.zip	Program 6.6		Timer0	Use Timer0A in PWM mode to generate a square wave of a given period with 50% duty cycle.
FlexibleTimer0APWM_811.zip Timer0APWM_4F120.zip			Timer0	Use Timer0A in PWM mode to generate a square wave of a given high period and low period.
FreqMeasure_811.zip	Program 6.8		GPIO, Timer0	Use Timer0B in edge count mode to count positive edges within a period set by Timer0A in periodic mode. In Timer0A periodic interrupts, calculate frequency.

LongPeriodMeasure_811.zip	Program 6.9		GPIO, Timer0	Use Timer0A in edge time mode to request interrupts on the rising edge of PD4 (CCP0). In Timer0B periodic interrupts, count amount of time between rising edges to determine period.
PWM_811.zip PWM_1968.zip PWMDual_4C123.zip PWM_4C123.zip	Program 6.7		PWM0	Use PWM0 to generate a 100 Hz square wave with 50% duty cycle.
SyncPrinter_811.zip	Program 7.1		GPIO, UART	Use UART0 to implement a printer interface with DTR synchronization. When the user button is pressed, DTR is low, and the printer is ready.
MAX5353_811.zip MAX5353_4F120.zip	Program 7.2	MAX5353 12-bit DAC	SSI	Provide a function that initializes the SSI0 module to interface with a MAX5353 12-bit DAC, and use SSI0 to send a 16-bit code to the MAX5353 and return the reply.
MAX1246_811.zip	Program 7.3	MAX1246 12-bit ADC	SSI	Provide a function that initializes the SSI0 module to interface with a MAX1246 12-bit ADC, and use SSI0 to send an 8-bit channel number code to the MAX1246 and return a 12-bit ADC value.
74HC595_811.zip	Program 7.4	74HC595 shift register	SSI	Use SSI0 to interface with a 74HC595 shift register to convert 3 output ports to 8 output ports.
I2C_811.zip	Programs 7.5, 7.6, 7.7	HMC6352 compass or TMP102 thermometer	I2C	Provide functions that initialize the I2C0 module to interface with an HMC6352 compass or TMP102 thermometer, send 1, 2, or 3 bytes to a particular slave address, and receive 1 or 2 bytes from a particular slave address.
InternalResistors_811.zip		Two buttons	GPIO	Provide a function that initializes GPIO PortD for a pull-up resistor on PD0 and a pull-down resistor on PD1.
DebounceSysTick_811.zip		One button	GPIO, SysTick	Use the SysTick timer to de-bounce a switch using 10 ms blind waits.
DebounceTimer_811.zip		One button	GPIO, Timer0A	Use Timer0A in periodic mode to debounce a switch using an interrupt. Basically, the switch is read every 10 ms.
DebounceCombo_811.zip		One button	GPIO, Timer0A, Timer0B	Use Timer0A in edge time mode to request interrupts on any edge of PD4 (CCP0) and start Timer0B. In Timer0B one-shot interrupts, record the state of the switch once it has stopped bouncing. This interface features minimum latency and allows for the user to attach functions to switch touch and switch release.
Piano_811.zip		16 buttons	GPIO, Timer0A	Use Timer0A in periodic mode to request interrupts to record changes on the 16 piano keys, directly connected to 16 GPIO pins.
MatrixKeypad_811.zip	old version with timer interrupts	Matrix keypad, resistors	GPIO, Timer0A	Provide functions that initialize GPIO ports and timers, arm the matrix keypad to respond to a button press, and scan the matrix keypad and return the ASCII code for the key pressed and number of keys pressed. Create a data structure based on the appearance of the keypad. Finally, use key wakeup interrupts on any change of GPIO Port A, then use Timer0A to request an interrupt in 10 ms to scan the matrix keypad.
7Segment_811.zip		7-segment LEDs, resistors, 2N2907, 2N2222	GPIO, Timer0A	Use Timer0A in periodic mode to request interrupts to interface a scanned LED display and refresh the image. Panel select driven by PNP transistor; segment select driven by NPN transistor.
PWMSine_811.zip PWMSine_1968.zip	Program 8.7	Resistor capacitor filter	PWM0	Use PWM0 to generate a sine wave of a given frequency. Timer0A periodic interrupts are used to cycle through each element of the output wave sequence.
ADCSWTrigger_811.zip ADCSWTrigger_1968.zip ADCSWTrigger_4F120.zip ADCSWTrigger.c ADC_TwoChanSWTrigger_1968.zip ADCSWTriggerTwoChan_4F120.zip			ADC	Provide functions that initialize ADC SS3 to be triggered by software and trigger a conversion, wait for it to finish, and return the result. The TwoChan examples use SS2 sampling two channels with software start and busy-wait synchronization.
ADCT0ATrigger_811.zip ADCT0ATrigger_4F120.zip ADCT0ATrigger.c			ADC, Timer0A	Provide a function that initializes Timer0A to trigger ADC SS3 conversions and request an interrupt when the conversion is complete.
ADCPrintResults_811.zip ADCPrintResults_1968.zip			ADC, PLL, Timer0A, UART	Use a setup similar to ADCT0ATrigger.c to gather ADC samples into a buffer. When the buffer is full, print them to the UART separated by TABs.
PI_811.zip		DC motor, TIP120, diode, resistor	PWM0, Timer0	Use a setup similar to PeriodMeasure.c to measure the tachometer period. Implement a PI controller to keep this period near a desired value.
Incremental_811.zip		DC motor, TIP120, diode, resistor	PWM0, Timer0A	Use a setup similar to PeriodMeasure.c to measure the tachometer period. Implement an incremental controller to keep this period near a desired value.
PICalibrate_811.zip		DC motor, TIP120, diode, resistor	PLL, PWM0, Timer0A, Timer1, UART	Use PWM0 to drive the motor at a steady 25% duty cycle for a short time. Then, abruptly increase to 75% duty cycle and measure the tachometer period to calculate constants for the PI controller.
Servo_811.zip		HiTec HS-322HD servo	PWM0, UART	Accept a string from the UART, parse it into a period, use PWM to output this period to the servo, which rotates to a particular angle.
Linecamera_811.zip		TSL1401R-LF camera	ADC0, ADC1, GPIO, I2C0, PLL, Timer0, UART	Interface a line camera, periodically take images, display them on the OLED display. Exposure length set by potentiometer position.
rit128x96x4.c rit128x96x4.h	OLED graphics	oLED on the LM3S1968 board	SSI on LM1968 or LM8962	Additional functionality to plot measured signals versus time. See the comments for RIT128x96x4PlotClear,

				RIT128x96x4PlotPoint(); RIT128x96x4PlotNext(); RIT128x96x4ShowPlot();
AGM1264_1968.zip AGM1264_4F120.zip		AGM1264 LCD graphics	GPIO, PLL on LM1968 or LM4F120	The LCD-00710 from www.sparkfun is a low-cost graphics LCD. It is 64 by 128 screen interfaced with 12 parallel output pins and is powered by 5V. It interfaces directly to the 3.3V LM3S without level shifters.
Flash_811.zip FlashProgram.c	Volume 3, Programs 6.2 and 6.3		Internal EEPROM	Provide functions that initialize the flash memory, write 32-bit data to flash, write an array of 32-bit data to flash, and erase a 1 KB block.
Heap_8962.zip Heap_4F120.zip heap.c heap.h	Volume 3, Section 3.2.3			Memory manager implementing malloc and free
RTOS_811.zip osasm.txt os.c	Volume 3, Programs 4.4 through 4.12, Section 4.2		GPIO, PLL, SysTick	Implement a real-time operating system with tasks
SDC_8962.zip SDC_2110.zip SDC_4F120.zip	Volume 3, Section 6.6, Program 6.4	Secure digital card	Systick, and SSI	Low-level device driver for secure digital card
Camera_811.zip camera.c	Volume 3, Program 6.5	TCM8230MD digital camera	I2C, PWM, and parallel ports	An interface of a camera with a resolution of 640 by 256 with RGB=5:6:5 color
CAN_8962_2110.zip CAN_4F120.zip	Volume 3	CAN cable	Controller Area Network, CAN	The LM3S8962 board is interfaced to the LM3S2110 board via CAN. The switch position on one board is displayed as LED status on the other. Communication is both directions.
PI_811.zip PI.c	Volume 3, Program 10.3	DC motor and tachometer	PWM, input capture	Proportion Integral Digital Controller, speed is measured with a tachometer and input capture. Power is controlled with PWM and a Darlington transistor, The system runs the motor at constant speed under variable load
ezLCD_811.zip Application Note ezLCD.h ezLCD.c		ezLCD301 and speaker	UART, PWM, ADC, Timer0	The ezLCD301 is a color graphics module with touch screen. See http://store.earthlcd.com/ezLCD-301. It is interfaced to a LM3S811. The zip file includes PCBartist files to create a PCB that fits into a PacTec XP-RB enclosure.
Ethernet_8962.zip	Volume 3, Program 9.1	Crossover Ethernet cable	Ethernet, SysTick	This example will appear in the second edition of Volume 3 due to be published Jan 2013. Information is communicated from one LM3S8962 to another via Ethernet.
LinearInterpolation_1968.zip LinearInterpolation_1968asm.zip LinearInterpolation.c	Volume 1, Program 6.22			Linear interpolation, sine function
DMASoftware_4F120.zip DMASoftware.c	Volume 3, Program 6.1	none	uDMA	Memory to memory transfer using uDMA
DMATimer_4F120.zip DMATimer.c	Volume 3, Chapter 6	PORTF input	Timer, uDMA	PORTF to memory transfer using uDMA, triggered by a periodic timer.
DMASPI_4F120.zip	Volume 3, Chapter 6	MAX5353 12-bit DAC	Timer, SSI, uDMA	Streaming data from memory to DAC/SSI using periodic timer, and ping-pong DMA.

Last updated March 26, 2014