Table 14.1 shows the ADC0 register bits required to perform sampling on a single channel. There are two ADCs; you will use ADC0 and the grader uses ADC1. For more complex configurations refer to the specific data sheet. Bits 8 and 9 of the **SYSCTL_RCGC0_R** specify the maximum sampling rate, see Table 14.2. The TM4C123 can sample up to 1 million samples per second. Bits 8 and 9 of the **SYSCTL_RCGC0_R** specify how fast it COULD sample; the actual sampling rate is determined by the rate at which we trigger the ADC. In this chapter we will use software trigger mode, so the actual sampling rate is determined by the SysTick periodic interrupt rate; the SysTick ISR will take one ADC sample. On the TM4C123, we will need to set bits in the **AMSEL** register to activate the analog interface.

Address	31-17	16	15-10	9	8	7-0			Name
0x400F.E100		ADC		MAXADCSPD					SYSCTL_RCGC0_R
	31-14	13-12	11-10	9-8	7-6	5-4	3-2	1-0	
0x4003.8020		SS3		SS2		SS1		SS0	ADC0_SSPRI_R
	31-16				15-12	11-8	7-4	3-0	
0x4003.8014					EM3	EM2	EM1	EM0	ADC0_EMUX_R
	31-4				3	2	1	0	
0x4003.8000					ASEN3	ASEN2	ASEN1	ASEN0	ADC0_ACTSS_R
0x4003.80A0					MUX0				ADC0_SSMUX3_R
0x4003.80A4					TS0	IE0	END0	D0	ADC0_SSCTL3_R
0x4003.8028					SS3	SS2	SS1	SS0	ADC0_PSSI_R
0x4003.8004					INR3	INR2	INR1	INR0	ADC0_RIS_R
0x4003.800C					IN3	IN2	IN1	IN0	ADC0_ISC_R
	31-12				11-0				
0x4003.80A8					DATA				ADC0_SSFIFO3

Table 14.1. The TM4C ADC0 registers. Each register is 32 bits wide. You will use ADC0 and we will use ADC1 for the grader and to implement the oscilloscope feature.

Table 14.2 shows the configuration to set the maximum sampling speed. The actual sampling rate is determined by how many times per second the software starts the ADC.

Value	*Maximum sampling frequency*
0x3	1M samples/second
0x2	500K samples/second
0x1	250K samples/second
0x0	125K samples/second

Table 14.2. The ADC MAXADCSPD bits in the SYSCTL_RCGC0_R register specify the maximum sampling speed.

Table 14.3 shows which I/O pins on the TM4C123 can be used for ADC analog input channels. You will use PE2 for Labs 14 and 15, and TExaS uses PD3 for the voltmeter and oscilloscope.

IO	Ain	0	1	2	3	4	5	6	7	8	9	14
PB4	Ain10	Port		SSI2Clk		M0PWM2			T1CCP0	CAN0Rx		
PB5	Ain11	Port		SSI2Fss		M0PWM3			T1CCP1	CAN0Tx		
PD0	Ain7	Port	SSI3Clk	SSI1Clk	I_2C3SCL	M0PWM6	M1PWM0		WT2CCP0			
PD1	Ain6	Port	SSI3Fss	SSI1Fss	I_2C3SDA	M0PWM7	M1PWM1		WT2CCP1			
PD2	Ain5	Port	SSI3Rx	SSI1Rx		M0Fault0			WT3CCP0	USB0epen		
PD3	Ain4	Port	SSI3Tx	SSI1Tx				IDX0	WT3CCP1	USB0pflt		
PE0	Ain3	Port	U7Rx									
PE1	Ain2	Port	U7Tx									
PE2	Ain1	Port										
PE3	Ain0	Port										
PE4	Ain9	Port	U5Rx		I_2C2SCL	M0PWM4	M1PWM2			CAN0Rx		
PE5	Ain8	Port	U5Tx		I_2C2SDA	M0PWM5	M1PWM3			CAN0Tx		

Table 14.3. Twelve different pins on the LM4F/TM4C can be used to sample analog inputs. You will use ADC0 and PE2 to sample analog input. If your PE2 pin is broken, you will have the option to perform Lab 14 with PE3 or PE5. We use ADC1 and PD3 to implement the oscilloscope feature.

The ADC has four sequencers, but you will use only sequencer 3 in Labs 14 and 15. We set the **ADC0_SSPRI_R** register to 0x0123 to make sequencer 3 the highest priority. Because we are using just one sequencer, we just need to make sure each sequencer has a unique priority. We set bits 15–12 (**EM3**) in the **ADC0_EMUX_R** register to specify how the ADC will be triggered. Table 14.4 shows the various ways to trigger an ADC conversion. More advanced ADC triggering techniques are presented in the book Embedded Systems: Real-Time Interfacing to ARM® Cortex™-M Microcontrollers. However in this course, we use software start (**EM3**=0x0). The software writes an 8 (**SS3**) to the **ADC0_PSSI_R** to initiate a conversion on sequencer 3. We can enable and disable the sequencers using the **ADC0_ACTSS_R** register. There are twelve ADC channels on the LM4F120/TM4C123. Which channel we sample is configured by writing to the **ADC0_SSMUX3_R** register. The mapping between channel number and the port pin is shown in Table 14.3. For example channel 9 is connected to the pin PE4. The **ADC0_SSCTL3_R** register specifies the mode of the ADC sample. We set **TS0** to measure temperature and clear it to measure the analog voltage on the ADC input pin. We set **IE0** so that the **INR3** bit is set when the ADC conversion is complete, and clear it when no flags are needed. When using sequencer 3, there is only one sample, so **END0** will always be set, signifying this sample is the end of the sequence. In this class, the *sequence* will be just one ADC conversion. We set the **D0** bit to activate differential sampling, such as measuring the analog difference between two ADC pins. In our example, we clear **D0** to sample a single-ended analog input. Because we set the **IE0** bit, the **INR3** flag in the **ADC0_RIS_R** register will be set when the ADC conversion is complete, We clear the **INR3** bit by writing an 8 to the 8 to the **ADC0_ISC_R** register.

Value	*Event*
0x0	Software start
0x1	Analog Comparator 0
0x2	Analog Comparator 1
0x3	Analog Comparator 2
0x4	External (GPIO PB4)
0x5	Timer
0x6	PWM0
0x7	PWM1
0x8	PWM2
0x9	PWM3
0xF	Always (continuously sample)

Table 14.4. The ADC EM3, EM2, EM1, and EM0 bits in the ADC_EMUX_R register.