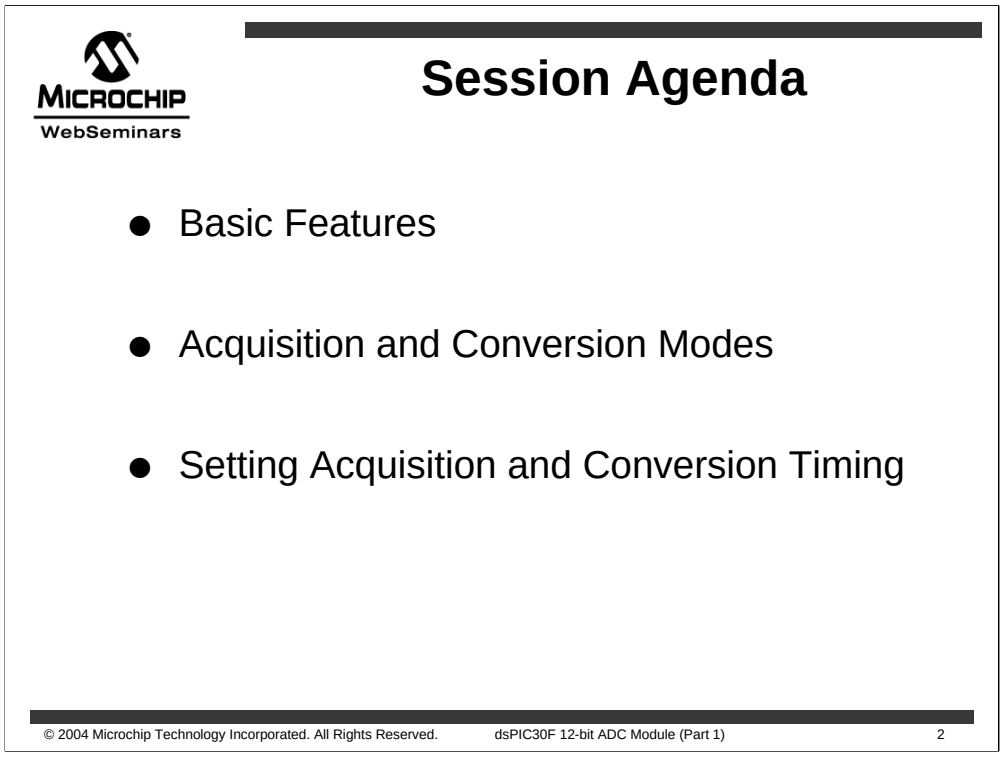
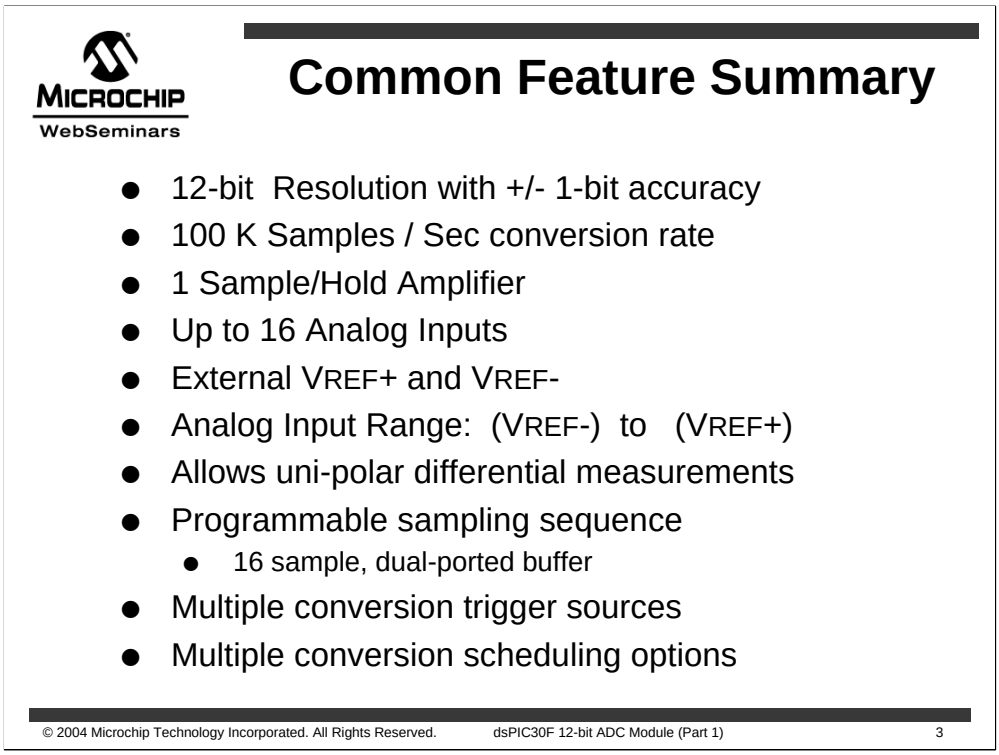
Welcome to Part 1 of the dsPIC30F 12-bit Analog to Digital Converter Web seminar.

I am Richard Fischer, DSC Applications Engineering Manager, and we will look at some basic features of the 12-bit ADC Module



In this session, we will start by taking a brief look at the important features of the 12-bit A/D Converter, or ADC module. Then, with the help of some examples, we will see how to configure various operational parameters of the module.

These parameters will include configuring port pins for accepting analog inputs, configuring the desired sampling rate, and interrupts.



General Purpose and Sensor dsPIC® family devices contain a 12-bit A/D Converter. This A/D Converter can convert one sample-and-hold channel at a maximum rate of 100 kilo samples per second, or KSPS.

Depending upon the specific dsPIC device being used, the A/D Converter may have up to 16 analog inputs which are all multiplexed on Port B. The A/D may use an external voltage reference for making conversions. The analog inputs may have a voltage range from the lower reference voltage, called Vref minus, to the higher voltage reference, called Vref plus.

Usually, the A to D will convert an input voltage relative to Vref minus. However, the module has operational modes that allow one input voltage to be converted relative to another input, thereby producing a differential measurement. The voltage on one input must exceed the other input, that is, the module allows uni-polar differential measurements.

The module contains a 16 word result buffer. The application may select from as few as 1 to as many as 16 sample buffer locations to be filled between interrupts.

The module has many programmable options specifying how analog inputs are selected while executing a set of sample cycles between interrupts. Programmable options also select the scheduling of acquisition and conversion functions.

Voltage References

VCFG<2:0>	VREFH	VREFL
000	AVDD	AVSS
001	VREF+	AVSS
010	AVDD	VREF -
011	VREF +	VREF -
1XX	AVDD	AVSS

- VCFG bits select AD reference high and low sources
- Either A/D power rails or Analog Inputs

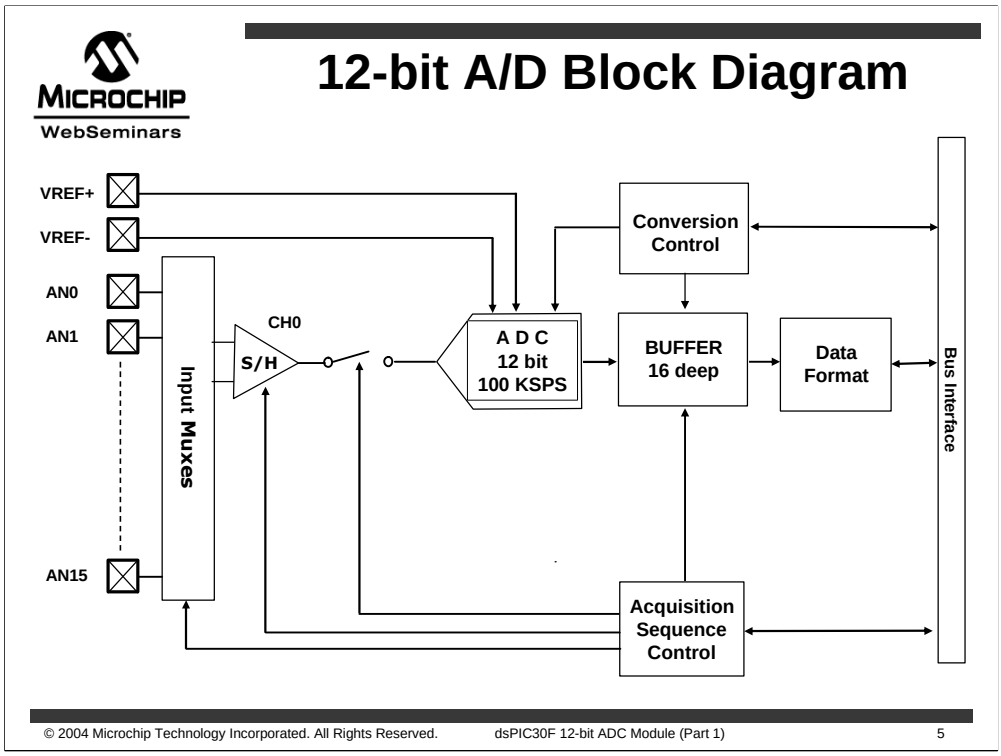
ADCON2 Register

VCFG2	VCFG1	VCGF0	-	-	CSCNA	-	-
bit15	14	13	12	11	10	9	bit8

BUFS	-	SMPI3	SMPI2	SMPI1	SMPI0	BUFM	ALTS
bit7	6	5	4	3	2	1	bit0

One such programmable option is the voltage reference input.

The VCFG bits in the ADCON2 register select the high and low reference sources. The references can be obtained from the A/D power rails. Alternatively, if the system requires an external reference source, the source can be provided on the V ref plus and V ref minus pins.



The A/D module includes logic that controls the sampling and conversion sequences. This includes application programmable registers to specify the sequence.

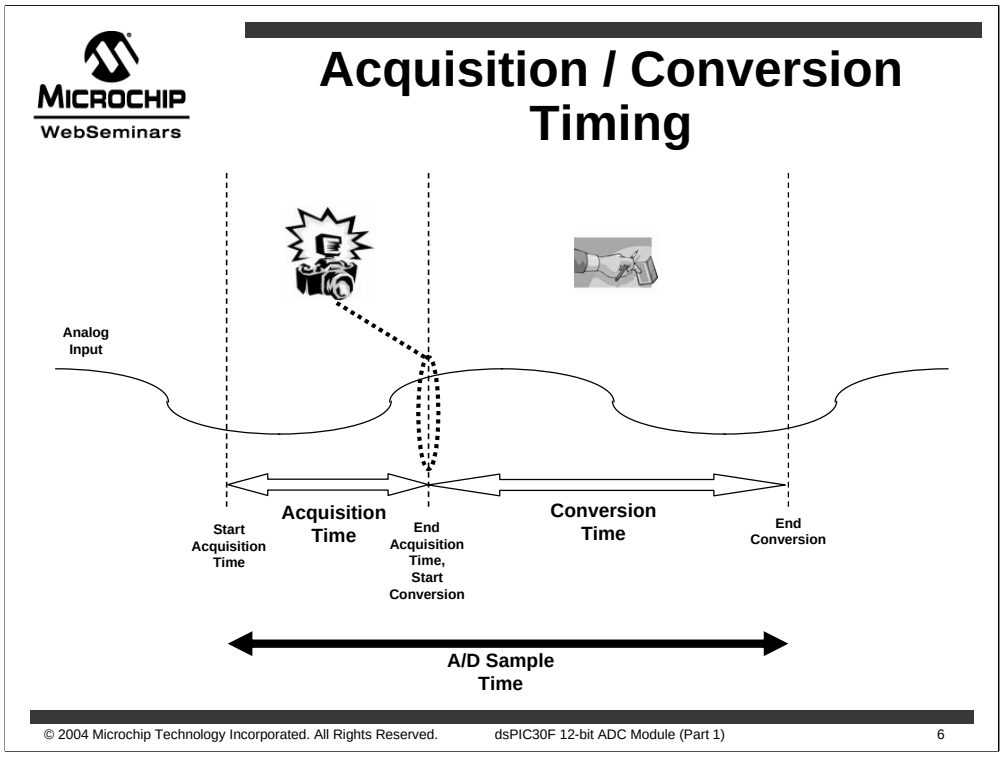
The sample sequence control determines which analog input is sampled and how it is sampled.

The A/D can sample up to 16 analog inputs on 16 input pins, labeled AN0 through AN15. A set of analog input multiplexers route the selected analog input to the sample/hold amplifier.

When ready, the sampled voltage held in the sample-and-hold amplifier is connected to the converter. The conversion sequence control logic schedules the converter operation and controls placing of the digital data into the result buffer.

After the ADC converts the analog input into a digital number, the digital result is stored in a buffer so that the application software can access the data. The buffer contains 16 words of data representing 16 converted samples. The ADC module writes to the buffer. The application software has only read access of this data.

As the application software reads the data, the module can format the data into one of four integer or fractional number formats.



As we discuss the A/D operation, let us use an analogy to define some critical terms.

The A/D module performs 2 major operations.

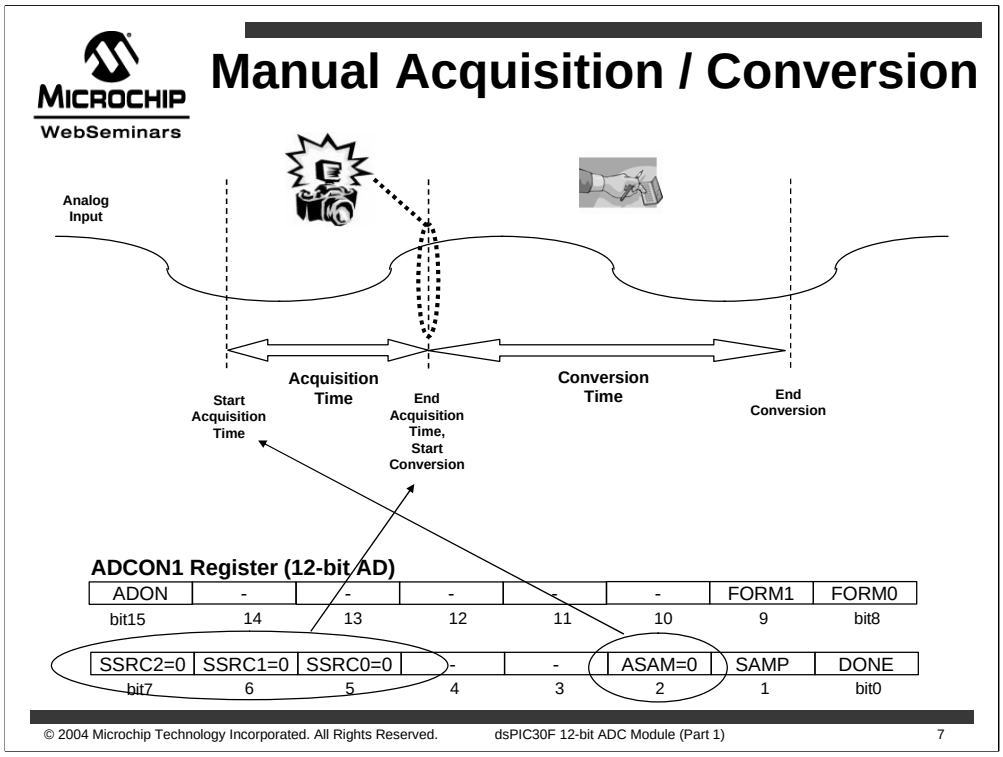
The first major operation is to take a ***sample*** of an analog input and hold that sample for analysis. This is analogous to taking a photograph of the analog waveform. The acquisition time is the time it takes to expose the photograph. The resulting photograph ***holds*** that sample for analysis.

The A/D will then analyze the output of the sample-and-hold amplifier and ***convert*** the information into a digital number. The conversion time is the time it takes to perform the analysis and report the result.

The A/D performs a sequence of acquisition operations and conversion operations. The application selects how this sequence occurs.

The time it takes to perform the acquisition is specified by the application. Since acquisition is like exposing the photograph, we must ensure that the acquisition time is long enough to capture an accurate image of the input waveform. Later, we will see how to ensure that the acquisition time is adequate.

The A/D module uses a state machine to control the conversion time. The clock for the state machine is usually derived from the device instruction clock. The application can specify how the A/D bit clock is derived, but the resulting clock speed must not exceed the maximum conversion rate of the A/D module.



The simplest method to start an acquisition conversion sequence is to have the application software initiate all the actions.

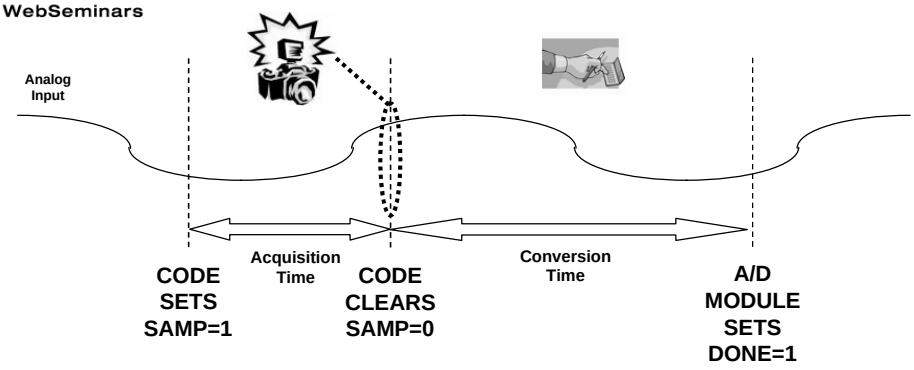
The SSRC bits, along with the ASAM bit in the ADCON1 register, control the acquisition conversion sequence type.

The SSRC bits control how acquisition is stopped. Note that when acquisition stops, conversion automatically starts, so these bits may also be considered as conversion start bits.

The ASAM bit controls how acquisition starts, either manually or automatically.

For example, when SSRC=000 and ASAM=0, the acquisition conversion sequences are manually controlled by application software.

Manual Acquisition / Conversion



ADCON1 Register (12-bit AD)

ADON	-	-	-	-	-	FORM1	FORM0
bit15	14	13	12	11	10	9	bit8

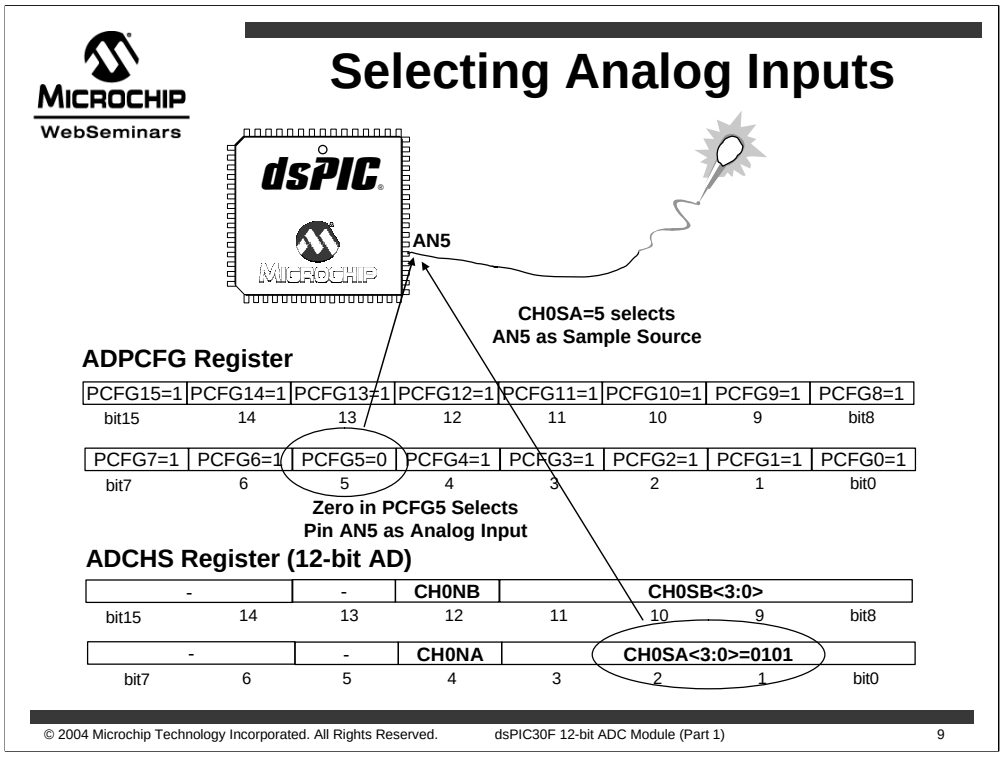
SSRC2	SSRC1	SSRC0	-	-	ASAM	SAMP	DONE
bit7	6	5	4	3	2	1	bit0

 dsPIC30F 12-bit ADC Module (Part 1) 8

Two bits in the ADCON1 register then control the A/D operation.

When the SAMP bit is set, the acquisition begins. When the SAMP bit is cleared, acquisition will stop and conversion of that sample will begin. After the required conversion time, the module stores the digital value into the result buffer, sets the DONE bit and generates an interrupt.

Let us look at an example. Suppose we want to record an analog audio input connected to an external microphone.

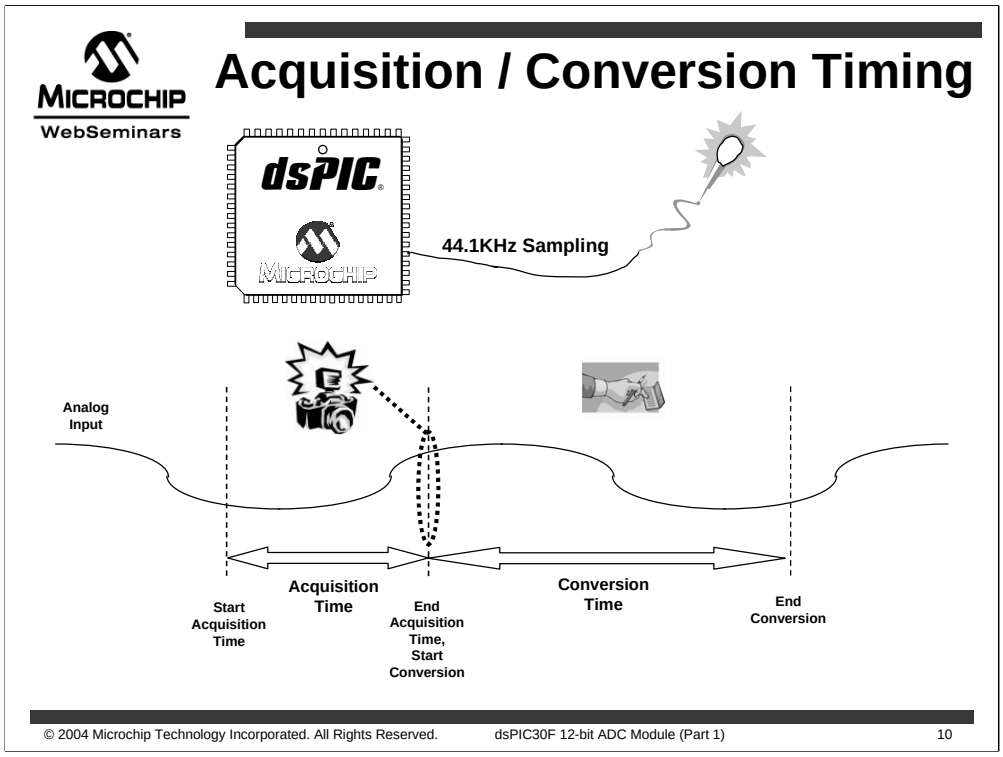


Let us assume that the audio input is on the AN5 input pin of a dsPIC® device. First, we need to set up that pin for analog input operation.

The ADPCFG register specifies which input port pins are dedicated to analog input functions. In this case, we will write zero to bit 5 of this register to indicate that port AN5 is an analog input. The other bits will be set, indicating the remaining port pins are for digital Input/Output functions.

Note that at device reset, all bits of this register are cleared, indicating that all pins will be available for analog input. At initialization, the software should set those bits that correspond to pins that require digital Input/Output.

We also need to tell the A/D which of the available analog pins to sample. The ADCHS register specifies which inputs are to be sampled. In this case, the CH0SA bits are programmed to 5, specifying that analog input AN5 is to be sampled. The value of the CH0SA bits corresponds to the number of the sampled analog input pin.



Since we are going to operate the A/D manually, we need to clear the SSRC and ASAM bits.

Now, let us assume that we want to sample the audio input at a 44.1 kilohertz rate, which is the same rate a CD player operates at. This translates to one sample every 22.67 microseconds.

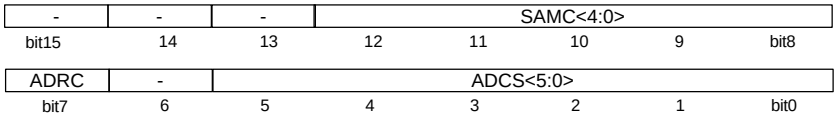
So, we set up a timer to give us an interrupt every 22.67 microseconds. On every interrupt, we start acquisition by setting the SAMP bit. When we are satisfied that the acquisition is complete, we clear the SAMP bit, thereby starting conversion. When the conversion is completed, we will get our digital value and then wait for the next interrupt.

But how do we know how long the conversion took? Did it actually complete in the time we desired? Let us see how to control conversion timing.

Setting A/D Conversion Time

- ADCS ⇒ sets conversion clock period
  - ADCS value limited by maximum SPS of A/D
  - Related to clock used by processor
- Tcy = 1 / MIPS
- Tconv = Desired conversion time
- Clocks per conversion = 14 for 12-bit A/D
- ADCS = ((2 * Tconv) / (14 * Tcy)) - 1

ADCON3 Register



The conversion time is specified by the ADCS bits in the ADCON3 register, which set the A/D conversion clock period (also known as A/D bit clock period)

To determine the proper value of ADCS, the user must first determine how fast the A/D is to perform conversion. The A/D has a minimum conversion time, which corresponds to a maximum conversion rate. This minimum conversion time is at least 10 microseconds, resulting in a maximum rate of 100 KSPS. The user must also know how fast the processor is running in terms of MIPS or instruction cycle rate. Note that 14 A/D clocks are required for completely converting the data held in the Sample-and-Hold amplifier.

Once this information is obtained, the formula shown here can be used to compute the ADCS value.

Setting A/D Conversion Time

- ADCS = ((2 * Tconv) / (CConv * Tcy)) - 1
- Example: 30 MIPS device, 12-bit A/D, max rate
  - Tcy = 1 / 30 MIPS = 33.3 nsec
  - Tconv = 10 µSec (min) by specification
- ADCS = ((2 * $10*10^{-6}$) / (14 * $33.3*10^{-9}$)) - 1
- ADCS = 42 = 0x24 = 0b101010

- Tad = (ADCS+1) * (Tcy/2) = 716 nsec
- Tconv = 14 * Tad= 10.03 µSec

So in our example:

A dsPIC® device, with a 12-bit A/D, is running at 30 MIPS, meaning that the instruction cycle time, Tcy, is 33.3 nano seconds.

It is always better to use the maximum available time for acquisition, so as to improve accuracy of the results. So, out of 22.67 microseconds, we wish to use as little of that time for conversions as possible and complete conversions as fast as the specification allows. We therefore set the conversion time, Tconv, to be as close as possible to 10 micro seconds. The calculated ADCS value, 41.8, rounded up, is 42.

We may also be able to round down to 41 but we must ensure the conversion bit clock specification is not exceeded.

When we compute the conversion time corresponding to this ADCS value, we have a conversion time of 10.03 microseconds out of our total time of 22.67 microseconds, leaving something less than 12.64 microseconds for acquisition.

So we can schedule 10 microseconds for acquisition, and the application software may be written such that it waits 10 microseconds between setting and clearing of the SAMP bit.

Setting A/D Conversion Time

- ADRC enables internal AD RC clock
- ADCS bits are ignored
- For 12-bit A/D
  - Nominal Rate = 47 KSPS
  - Nominal Tconv = 21 µSec
- Note that these are substantially less than the maximum rate

ADCON3 Register

-	-	-	SAMC<4:0>				
bit15	14	13	12	11	10	9	bit8

ADRC	-	ADCS<5:0>					
bit7	6	5	4	3	2	1	bit0

We could also use the A/D module's internal RC clock rather than a clock derived from the system clock.

This may be desirable when the processor clock is switching frequently or if we wish to complete a conversion while the processor is in SLEEP mode.

Selecting the ADRC clock renders the ADCS bits redundant.

Using the ADRC clock results in a substantially slower conversion rate than the maximum and has a nominal conversion rate of 47 kilo samples per second.

Setting A/D Acquisition Time

- When SSRC=111, the A/D will count clocks for acquisition time.
- Tacq = SAMC * Tad

ADCON3 Register

-	-	-			SAMC<4:0>		
bit15	14	13	12	11	10	9	bit8

ADRC	-	ADCS<5:0>					
bit7	6	5	4	3	2	1	bit0

ADCON1 Register (12-bit AD)

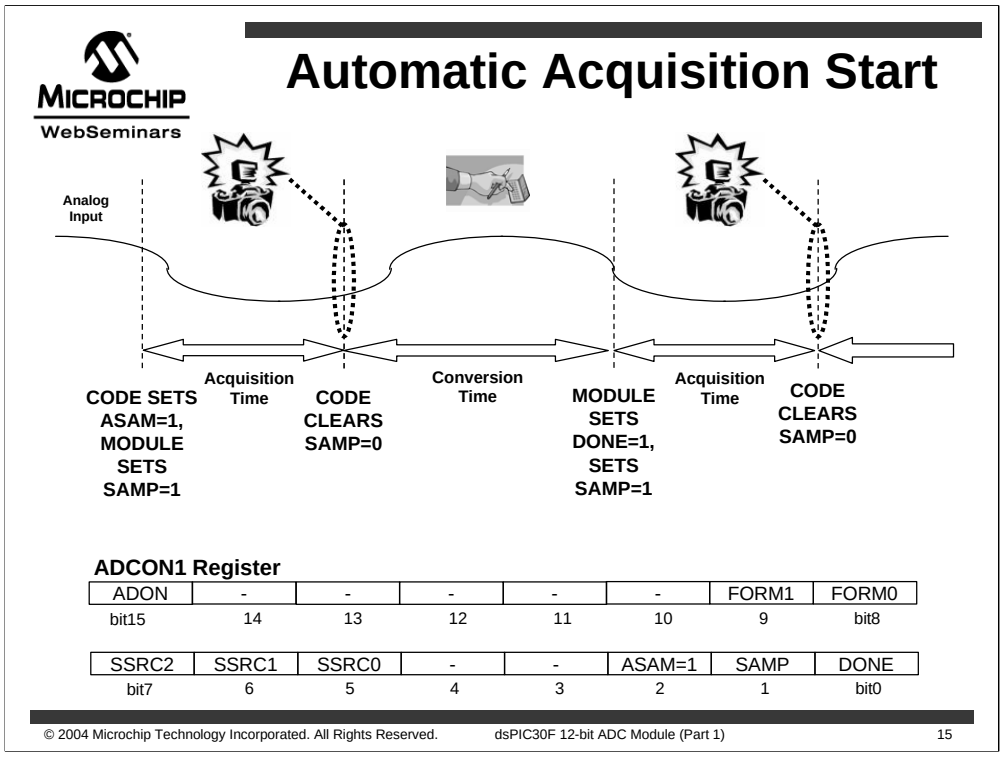
ADON	-	-	-	-	-	FORM1	FORM0
bit15	14	13	12	11	10	9	bit8

SSRC2=1	SSRC1=1	SSRC0=1	-	-	ASAM=0	SAMP	DONE
bit7	6	5	4	3	2	1	bit0

Let us now look at some other methods of configuring the sampling rate.

Rather than making the application control the acquisition time, the A/D module can control the acquisition time automatically.

Setting the SSRC bits to binary 111 puts the module into the auto convert mode. When the SAMP bit is set, the module will begin to count Tad bit clocks. As soon as the number of clocks specified by SAMC is complete, the module will automatically clear the SAMP bit and begin conversion.

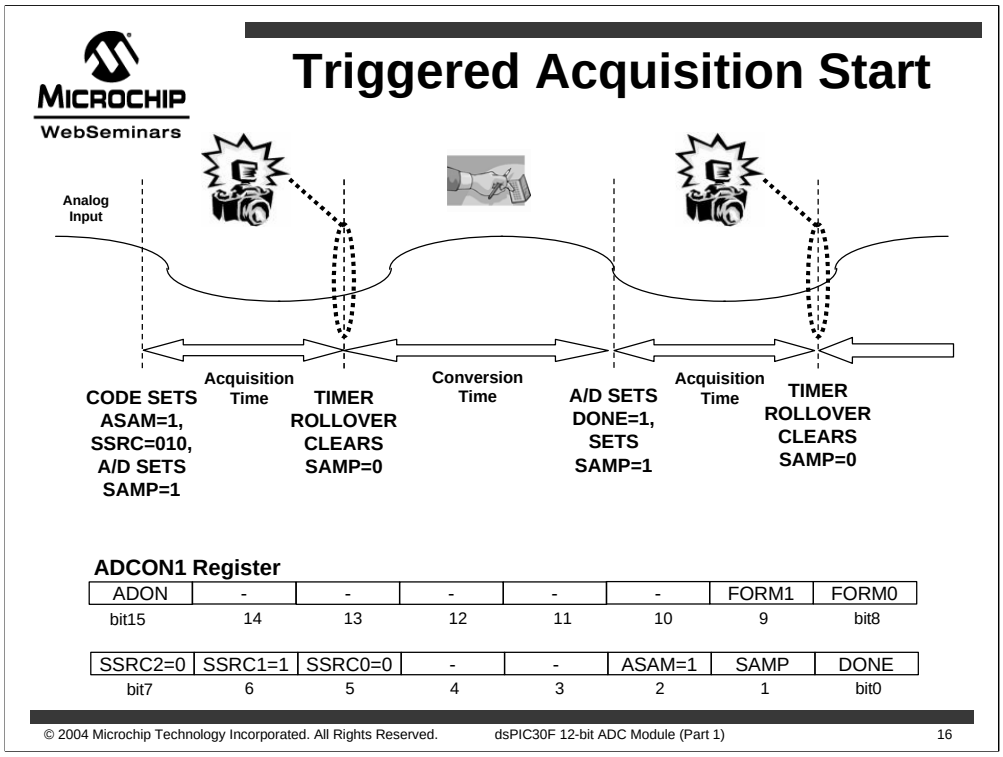


To maximize the acquisition time, acquisition could start as soon as the converter has completed acquiring the previous sample.

To accomplish this, the application can set the ASAM bit. When ASAM is set to 1, the module will revert to acquisition immediately after conversion completes.

In the example shown here, the SSRC bits are assumed to be 000, thus specifying the manual conversion start mode. In this case, the application is required to clear the SAMP bit in order to start conversion. Using this configuration, a timer interrupt can be used to start the conversion.

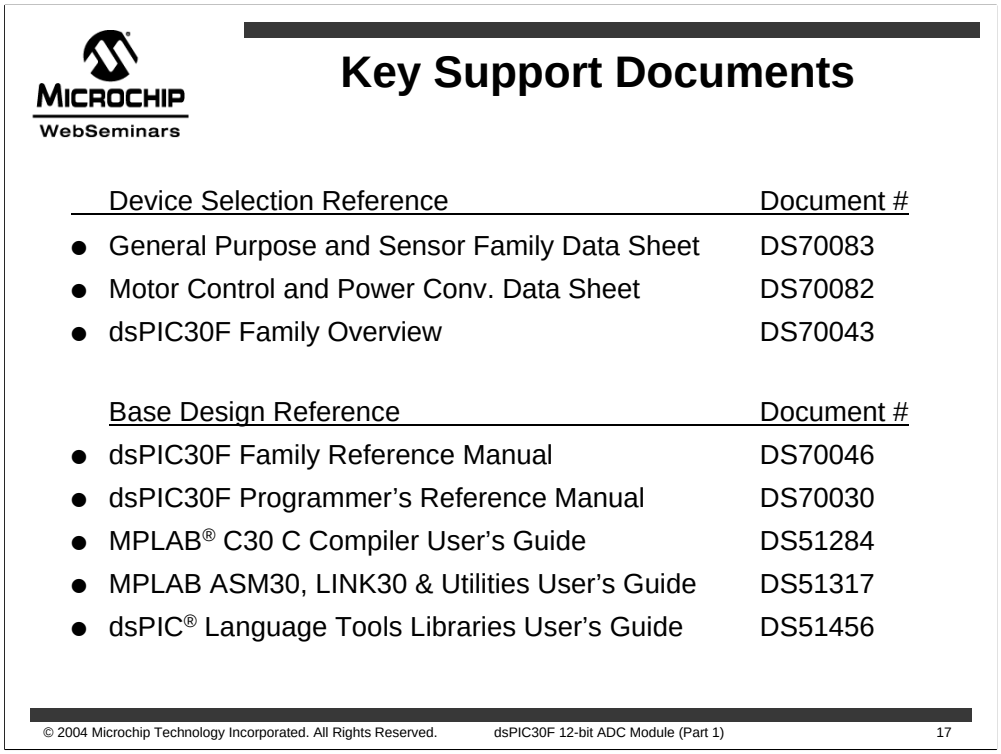
However, if we implemented the auto conversion trigger mode, by setting SSRC to binary 111, both the acquisition and conversion events would take place automatically.



The Triggered Conversion Start mode allows the module to automatically begin conversion based on a signal it receives from the timer 3 module.

Using this mode, in conjunction with the automatic acquisition mode, allows the A to D module to operate with no intervention from the application other than to read the conversion results.

There are 2 different conversion trigger sources that the application can choose from, and the SSRC bits will select the mode and the source. The trigger can come either from a timer event or an external interrupt event.



For more information, here are references to some important documents that contain a wealth of information about the dsPIC30F family of devices.

The Family Reference Manual contains detailed information about the architecture and peripherals, whereas the Programmer's Reference Manual contains a thorough description of the instruction set.

Key Support Documents

Device Specific Reference	Document #
dsPIC30F2010 Data Sheet	DS70118
dsPIC30F2011/2012/3012/3013 Data Sheet	DS70139
dsPIC30F3014/4013 Data Sheet	DS70138
dsPIC30F4011/4012 Data Sheet	DS70135
dsPIC30F5011/5013 Data Sheet	DS70116
dsPIC30F6010 Data Sheet	DS70119
dsPIC30F6011/12/13/14 Data Sheet	DS70117

Microchip Web Site: www.microchip.com

For device-specific information such as pinout diagrams, packaging and electrical characteristics, the device datasheets listed here are the best source of information.

All these documents can be obtained from the Microchip web site shown, by clicking on the “dsPIC® Digital Signal Controllers” or “Technical Documentation” link.

That concludes this presentation and I would like to thank you for attending. If you would like to find out more about this topic, please visit our website at www.microchip.com and look for Part 2 of the12-bit ADC module.