AN1798
APPLICATION NOTE

STR71X ADC CONVERSION SPEED-UP

INTRODUCTION

This application note gives an example of how to reduce the conversion time of the A/D Converter of the STR71x microcontroller, when only one input needs to be converted. To do this we use Round Robin mode instead of Single Channel mode to convert an analog input signal which must be connected to all four analog input pins of the converter.

Rev. 2

1 STR71X ADC

The STR71x analog to digital converter (ADC) is a 12-bit Sigma-Delta converter with an input range between 0 and 2.5V.

Sigma-Delta converters, known also as oversampling converters, sample the input signal many times for each output sample with a frequency rate much greater than the Nyquist frequency (twice the input bandwith). A Sigma-Delta converter consists of two blocks, the first is the Sigma-Delta modulator which produces the bit stream to the second part which consists of a digital filter and decimator.

The STR71x ADC is composed of a second-order Sigma-Delta modulator followed by a $sinc^3$ digital filter and decimator.

It offers two conversion modes:

Round-robin: Repeats the conversion process for each of the four channels continuously in turn. In this mode, the converter samples each input channel for 511 cycles of the oversampling clock.

Single channel: Converts the selected analog input only. In this mode, the converter samples the selected input channel 4 times for 511 cycles of the oversampling clock to maintain the same output frequency as Round Robin mode. Consequently, a valid sample for the channel is produced only every 2048 clock cycles.

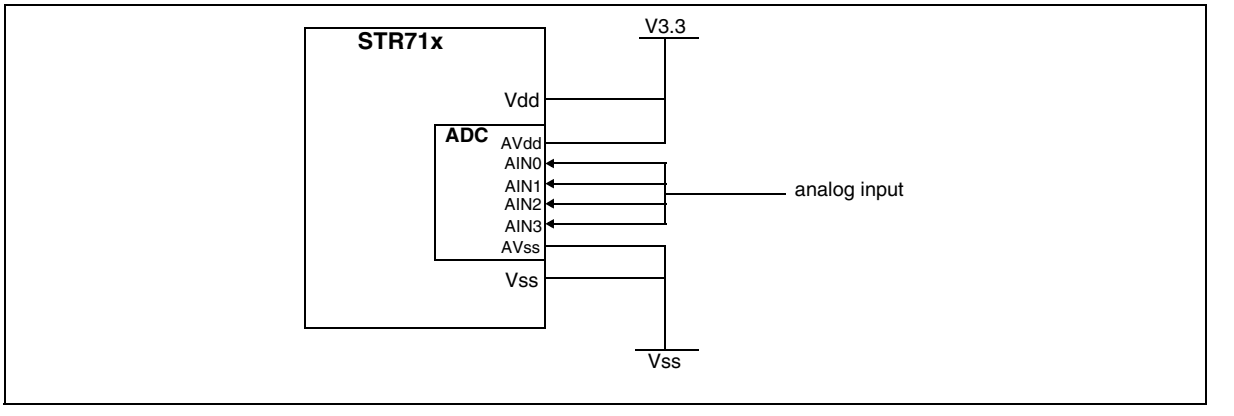
End of Conversion is indicated by four flags or by interrupt generation. The four flags are the Data Available flags (DA[n]) in the Control Status register. They allow the application software to determine which channel data register has a new sample ready to be read. Each DA[n] flag corresponds to ADC channel n. They are set by hardware as soon as a new sample on the corresponding channel is available and they are automatically cleared when the corresponding data register is read.

The STR71x ADC is able to generate an interrupt at the end of conversion. This interrupt depends on the conversion mode. In Single Channel mode an interrupt is generated if, in the CSR register, the interrupt bit and Data Available flag for the selected channel are set. In Round Robin mode, an interrupt is generated if all interrupt bits and all Data Available flags in the CSR are set.

2 OVERVIEW

In Single Channel mode, a valid sample for the selected channel is produced every 2048 clock cycles only, with the same output frequency as Round Robin mode which samples the four channels (511 clock cycles for each channel). Consequently, the conversion time for one analog voltage can be reduced by using Round Robin mode instead of Single Channel mode. The four ADC inputs have to be connected to the same analog voltage in order to use Round Robin conversion mode.

Figure 1. Connection to the STR71x ADC



3 SOFTWARE CONFIGURATION

To use the procedure described above, the ADC has to be driven in polling mode because in Round Robin mode it generates an interrupt only when it has finished converting the four input channels and all the Data Available flags in the Control Status Register (CSR) are set. In polling mode, we can get the conversion result of the channel corresponding to the DAn flag that has been set without waiting for the end of conversion on all the channels.

In this Application note, the Extended Function Timer provided in the STR71x is used to compare the conversion time in Single Channel mode to the conversion time in Round Robin mode when converting one analog voltage.

We start the software configuration by configuring all the STR71x analog pins as high impedance analog input. Next, the STR71x ADC has to be selected in Round Robin conversion mode. Then the prescaler has to be configured. After that, the ADC has to be enabled by setting the ADC_En bit in the PCU_BOOTCR register. Then we wait for data to become available in one of the four DATA register by looping on all Data Available flags (DAn) in the Control Status Register. This loop is stopped when a DAn flag is set. Then we get the conversion result from the Data Register corresponding to the Data Available flag that has been set.

The following flowchart shows how to implement this procedure.

Figure 2. Main Program Flowchart