Übung Echtzeitsysteme WS 2013 / 2014
Atmel AVR
Philipp Heise, Christian Buckl

Exercise 0 Analog digital conversion

In the following exercise we want to use ADC of the AVR to convert analog value into digital values.

• Read chapter 24 in the ATMega168 manual.
• What does it mean that the AVR has a 10bit ADC?
• Why is the conversion speed important? How can you set it?
• Try to find some widely available analog sensors on the internet and discuss other possible use cases of the ADC.

Exercise 1 ADC AVR

In this exercise we will connect a potentiometer to the AVR and want to measure the current voltage. For displaying the result we use our SPI display from the previous sheet.

• Connect the display as in the previous sheet and make sure that everything works as expected.
• Connect the potentiometer to the AVR. As shown in the image below the pin 1 of the poti should be connected to VTG and pin 3 to GND. Connect pin 2 of the poti to any ADCX of your choice (e.g. ADC0 on pin PC0).

• In the following we want to implement to functions. One function is \texttt{adc\_init} to initialize the ADC. The second function is \texttt{adc\_read( uint8\_t channel )} to read a value from \texttt{adc\_input\_channel}.
  
  \begin{itemize}
  \item \texttt{adc\_init}: We want to use \texttt{AVCC} as voltage reference. Further we want to use a prescaler division factor of 2 and perform single conversion without any interrupts. Without setting any specific \texttt{adc\_input} in the \texttt{ADMUX} register we need to perform and initial conversion to set everything up.
  \item \texttt{adc\_read( channel)}: shall perform a single conversion and return the resulting value. The function shall block until the conversion is finished.
  \end{itemize}
• Visualize the conversion result on the display.