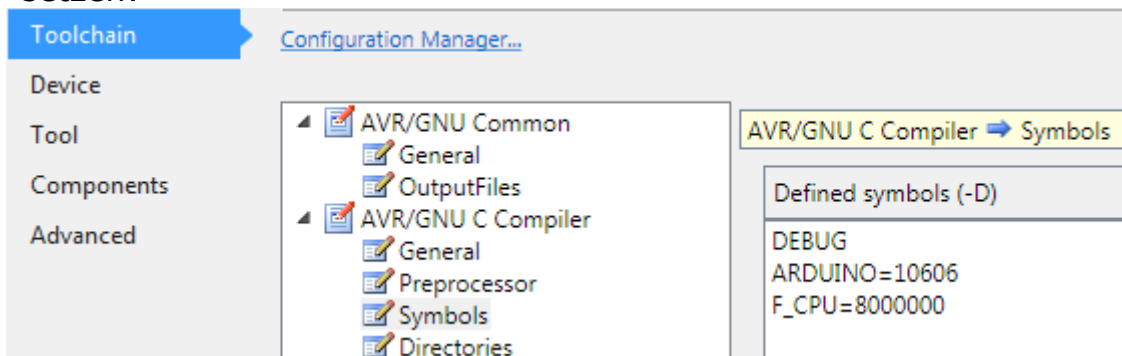


Einrichten des ATMEL-Studios

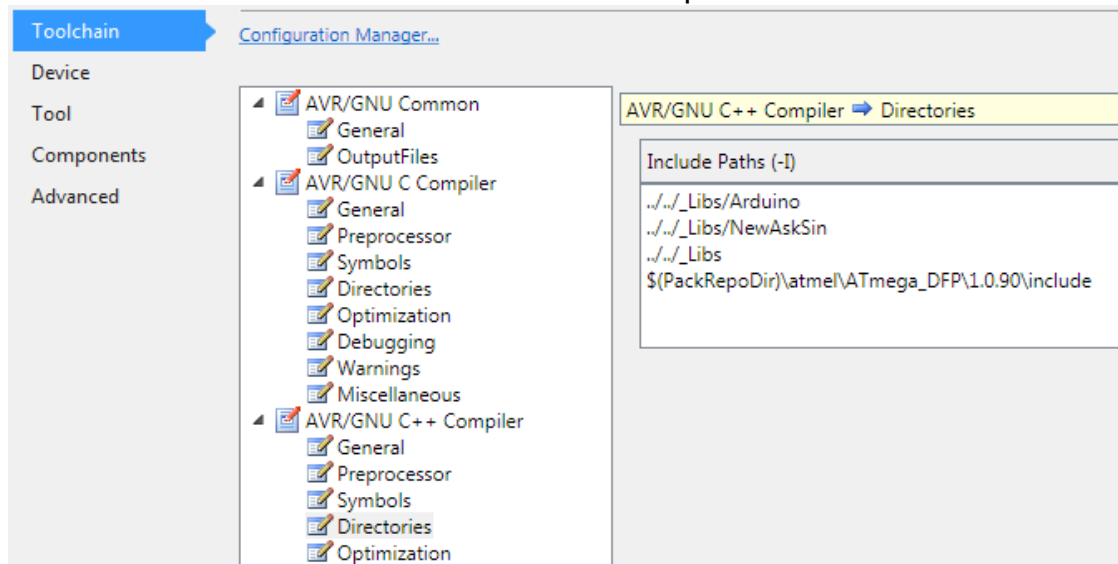
Da ich mit der Arduino-IDE auf „Kriegsfuß“ stehe, habe ich den bestehenden Universalsensor bzw. die Examples aus der AskSin/NewAskSin unter dem ATMEL-Studio zum Laufen gebracht.

Folgende Einstellungen sind hierzu notwendig (ATMEL Studio 7.0):

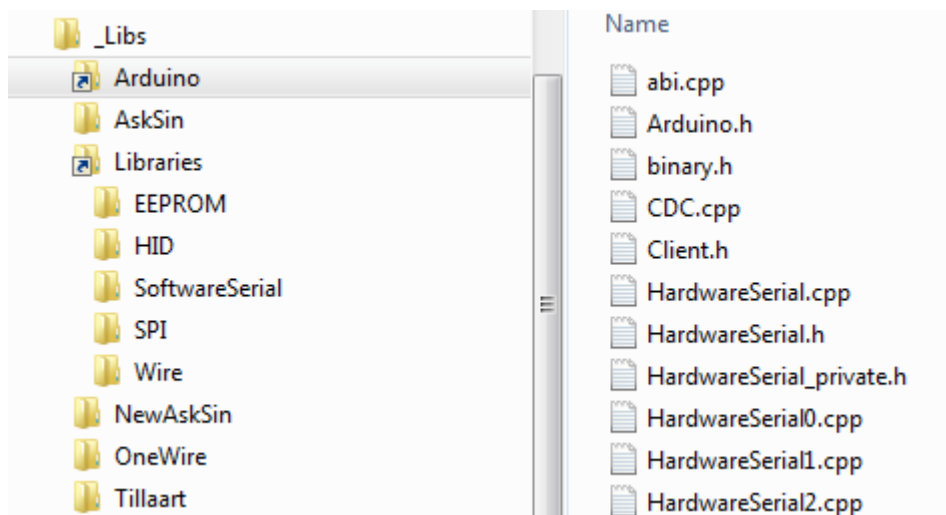
- Installieren des Atmel-Studios
- Installieren der Arduino-IDE (bei mir V1.6.8), damit die entsprechenden Libraries zur Verfügung stehen
- Anlegen eines C++ Projekts im Atmel-Studio
- Folgende Einstellungen:
In der ToolChain des C **und** des C++ - Compilers Arduino Version auf 10606 oder ggf. höher setzen und die CPU-Geschwindigkeit (F_CPU) setzen:



- Die Include-Pfade des C- und C++ - Compilers setzen:



- Unter dem Pfad `_Libs` binde ich dann meine Libraries ein, z.B. die `NewAskSin`, aber auch Links zu den Arduino Standard-Libs:

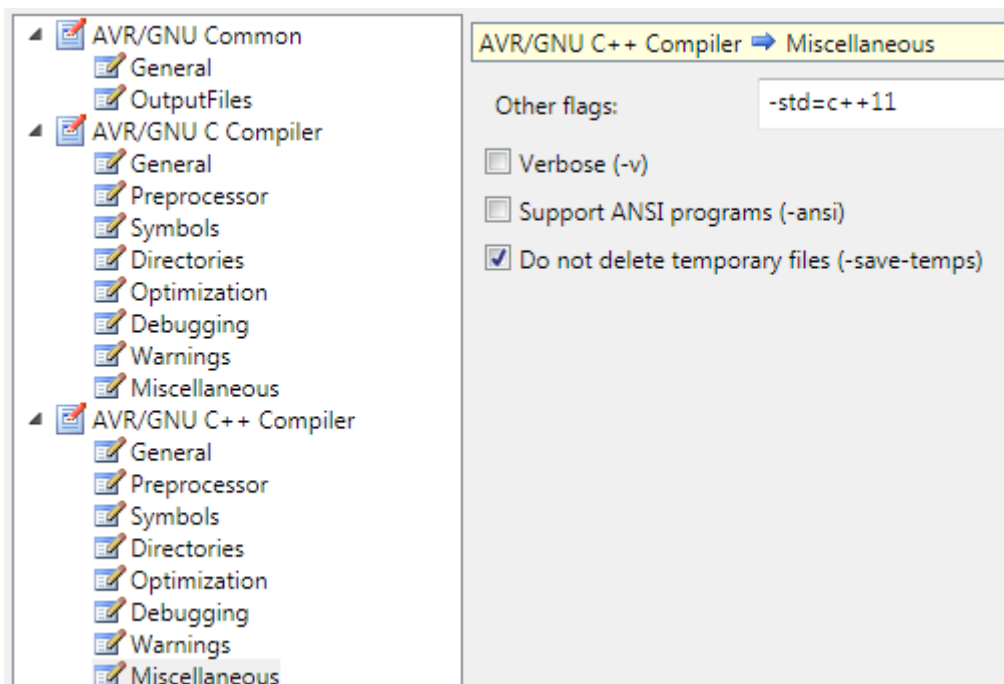


Die Symlinks sehen folgendermaßen aus:

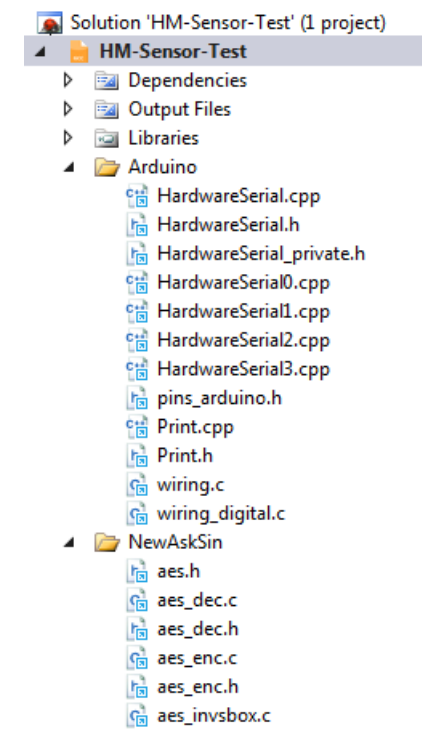
```
Verzeichnis von C:\Users\Martin\Documents\Atmel Studio\7.0\_Libs
16.03.2016 01:51 <DIR> .
16.03.2016 01:51 <DIR> ..
20.02.2016 22:12 <SYMLINKD> Arduino [C:\Program Files (x86)\Arduino\hardware\arduino\avr\cores\arduino]
10.04.2016 09:49 <DIR> AskSin
22.02.2016 00:20 <SYMLINKD> Libraries [C:\Program Files (x86)\Arduino\hardware\arduino\avr\libraries]
15.05.2016 12:39 <DIR> NewAskSin
01.05.2016 00:20 <DIR> OneWire
03.11.2015 17:19 5.994 pins_arduino.h
22.02.2016 00:39 <DIR> Tillaart
```

Sie können unter Windows mittels „mklink“ erzeugt werden.

- Ich habe dann noch beim C++ Compiler unter „Misc“ die Option „-std=c++11“ gesetzt – dann gibt’s weniger Warnings:



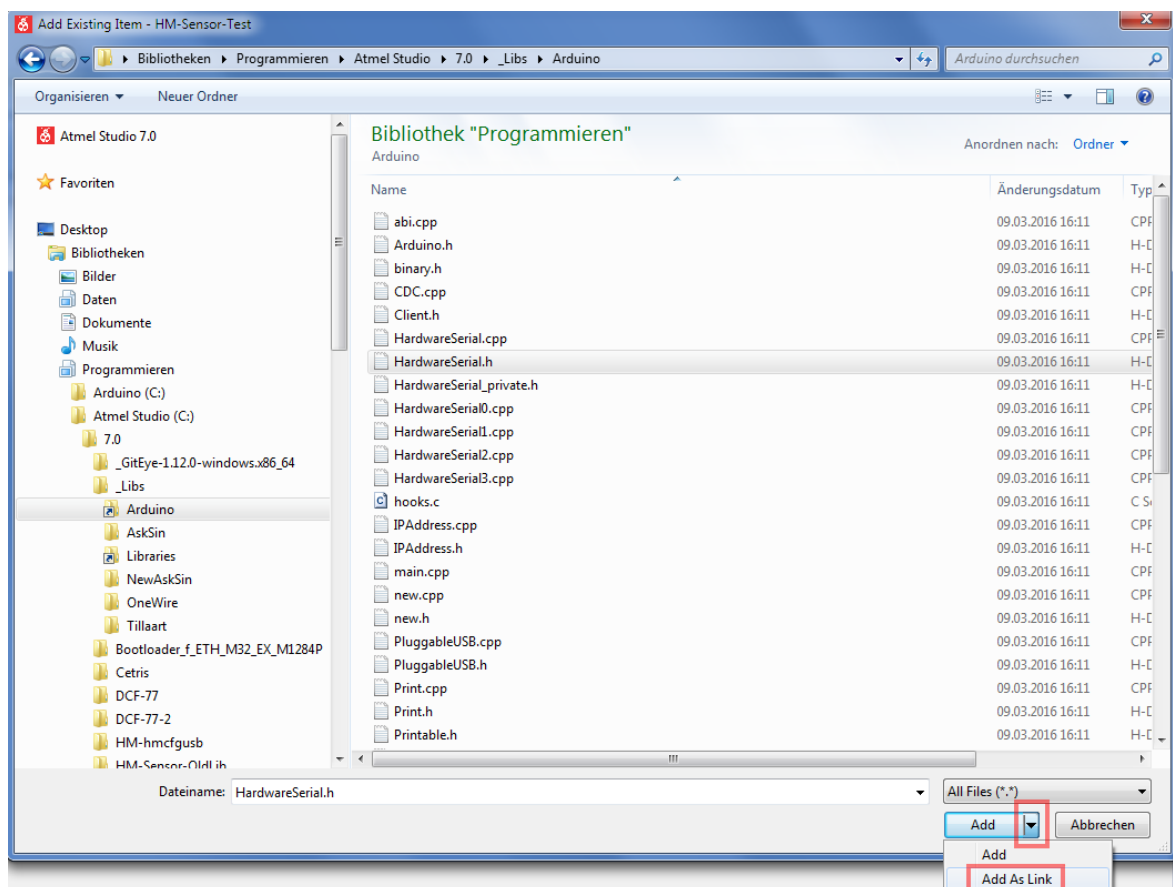
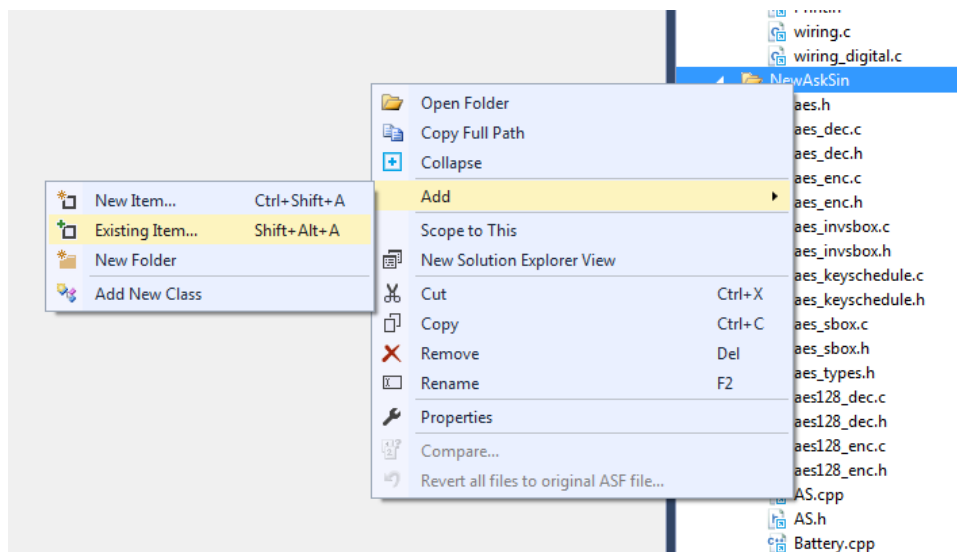
- Im Atmel-Project habe ich dann Unterordner für die NewAskSin und die Arduino-Lib angelegt:



...

- HAL.cpp
- HAL.h
- HAL_atmega328P.h
- HAL_atmega32U4.h
- HAL_extern.h
- Power.cpp
- Power.h
- Receive.cpp
- Receive.h
- Registrar.cpp
- Registrar.h
- Send.cpp
- Send.h
- StatusLed.cpp
- StatusLed.h
- THSensor.cpp
- THSensor.h
- Version.h
- dht.cpp
- dht.h
- hardware.cpp
- hardware.h
- hmkey.h
- main.cpp
- OneWire.cpp
- OneWire.h
- register.h

- Hierbei ist zu beachten, daß die Files der Libraries nur als „Links“ in das Projekt eingefügt werden:



So, das sollte es gewesen sein – ich hoffe ich habe nichts vergessen...

Jedenfalls habe ich damit den Compilervorgang wesentlich mehr unter Kontrolle und der Editor mit Highlighting und Springen zu Referenzen etc ist mir wesentlich sympathischer...

Martin