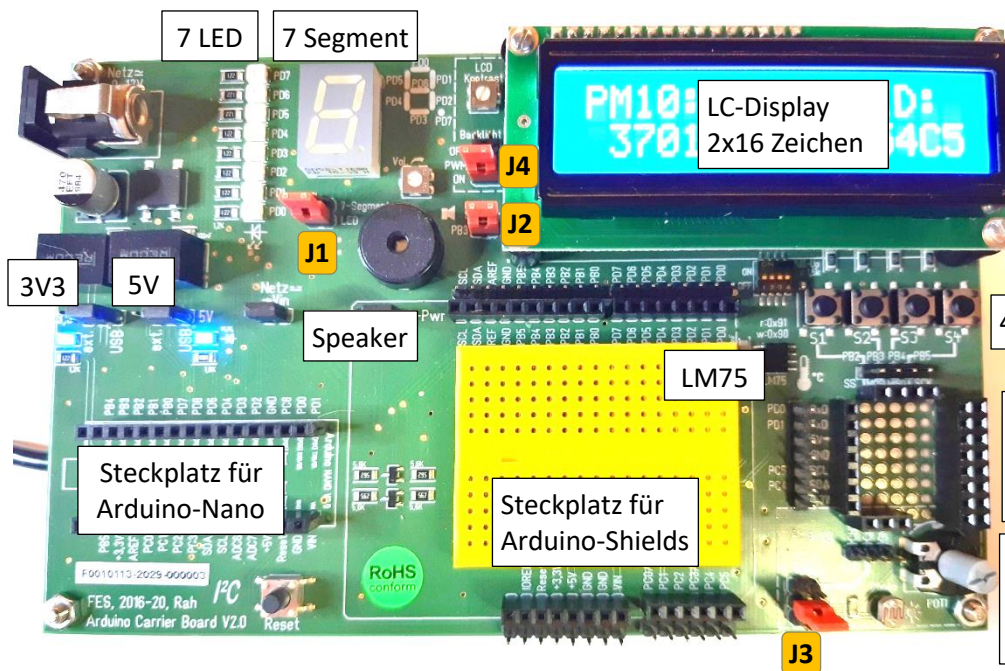
	Mikrocontroller-Labor	Name: Rahm Datum: 21.10.2017 1_1_Programmieren mit AtmelStudio.docx
	Erstellen eines Projektes für die C-Programmierung	1.1.1

Projekt: Programmierung des Mikrocontrollers mit AtmelStudio 7

Im Labor der FES wird eine zum Arduino compatible Mikrocontrollerplatine auf einem Trägerboard verwendet. Der Mikrocontroller ist ein weit verbreiteter AVR-Controller (ATmega328P).

Baugruppen des Trägerboards



Legende Steckbrücken

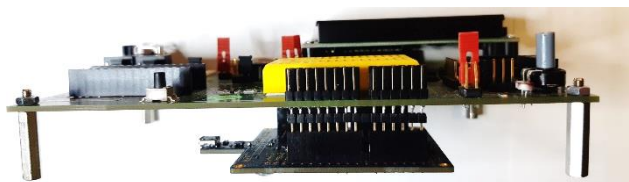
J1 Port D:	7-Segment LED
J2 Speaker:	PB3 (gezogen) OFF
J3 Analog: (PC3)	CH1 (Exp.) Poti LDR
J4 Backlight: (LCD)	OFF PWM (PB3) ON

4 Taster

Lochrasterfeld für Erweiterungen

Analogeingabe:
- Poti
- LDR

Trägerboard mit aufgesteckter Controllerplatine



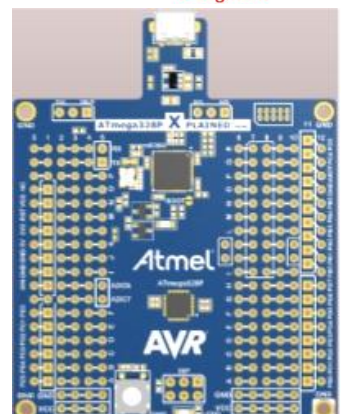
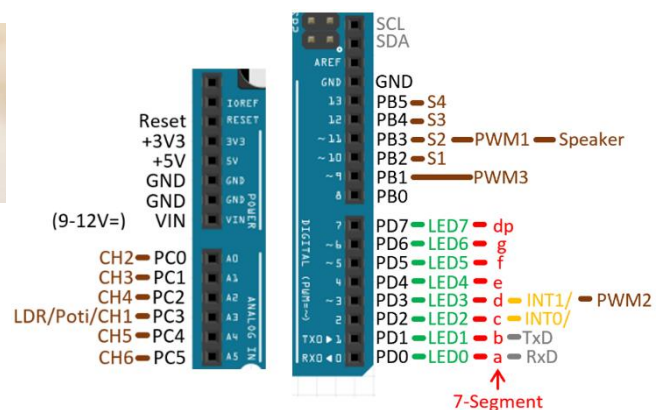
Das Arduino-Carrier-Board ist optimiert für die Verwendung mit unterschiedlichsten, zum Arduino-Pinheader kompatiblen Controllerboards. Ebenso befindet sich ein Arduino-Nano-Steckplatz auf dem Board.


Wichtig: Es darf immer nur ein Controllermodul gesteckt sein.

Im Laborunterricht verwenden wir ein ATmega328P Explained Mini-Controller-Board von Atmel/Microchip. Das Board verfügt über einen integrierten Debugger für die Programmierung in Atmel/Microchip-Studio.

Dies ist eine kostenlose integrierte Entwicklungsumgebung (IDE = Integrated Development Environment), die aber auch professionellen Ansprüchen genügt. Als Programmiersprache verwenden wir C und die für die Berufsschul-Abschlussprüfung der Elektroniker/-innen für Geräte und System verwendeten FA205-Funktionsbibliotheken.

Belegung der Arduino-Pinheader



 Friedrich-Ebert-Schule Esslingen FES	Mikrocontroller-Labor	Name: Rahm Datum: 21.10.2017 1_1_Programmieren mit AtmelStudio.docx
	Erstellen eines Projektes für die C-Programmierung	1.1.2

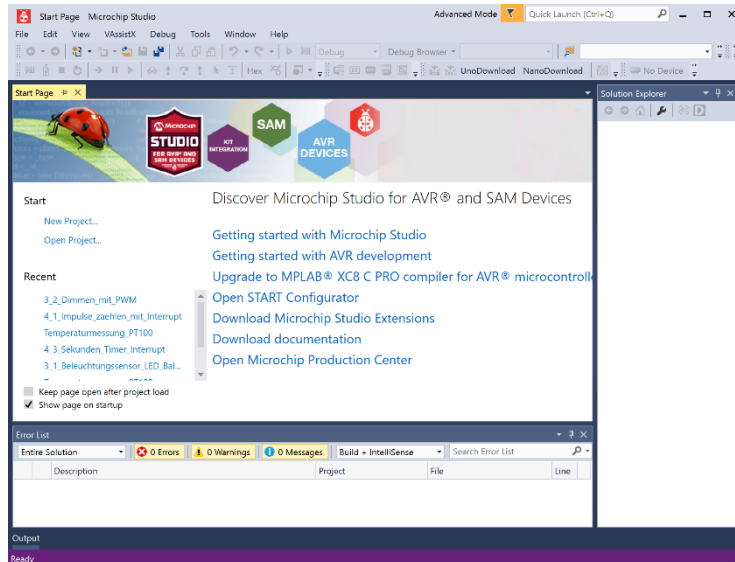
Arbeitsauftrag

◆ Vorbereitungen (bei Bedarf)

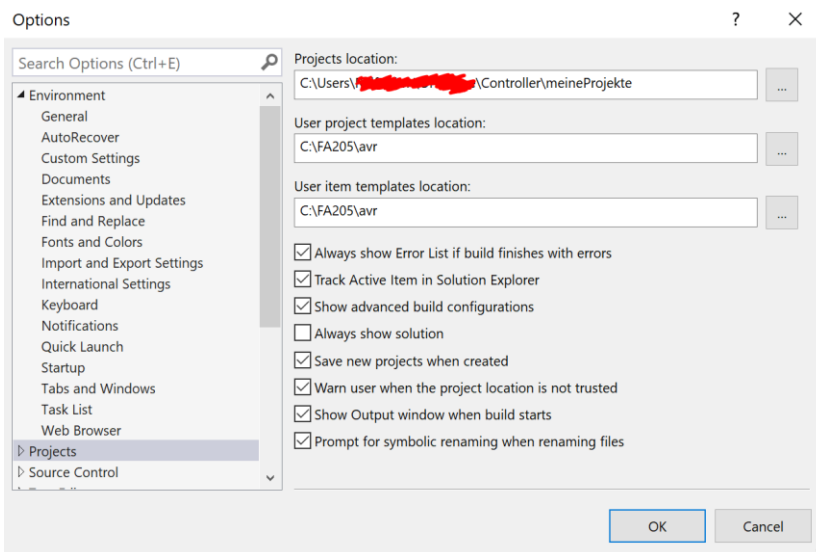
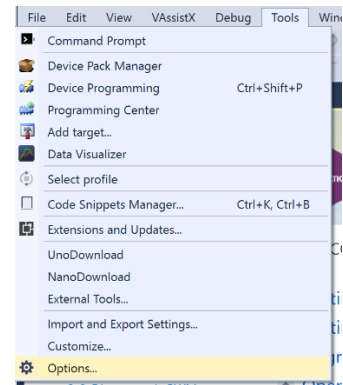
Öffnen Sie AtmelStudio,



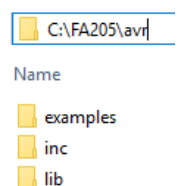
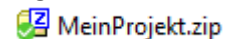
bzw. MicrochipStudio



Stellen Sie fest, ob der Verzeichnispfad zur Projekt-Vorlage (Template) richtig gesetzt ist (Tools > Options...).

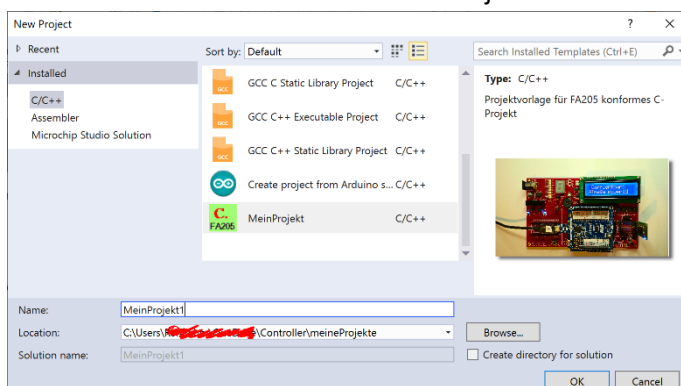


Stellen Sie die Pfade (Speicherorte) wie abgebildet ein. Kontrollieren Sie mit dem Explorer, dass im angegebenen Speicherort für die Benutzerprojektvorlagen die Datei **MeinProjekt.zip** liegt.



Kontrollieren Sie, ob die FA205-Bibliotheken im Verzeichnis **C:\FA205\avr** vorhanden sind.

◆ Erstellen Sie nun ein neues FA205 C-Projekt mit **File > New > Project...**

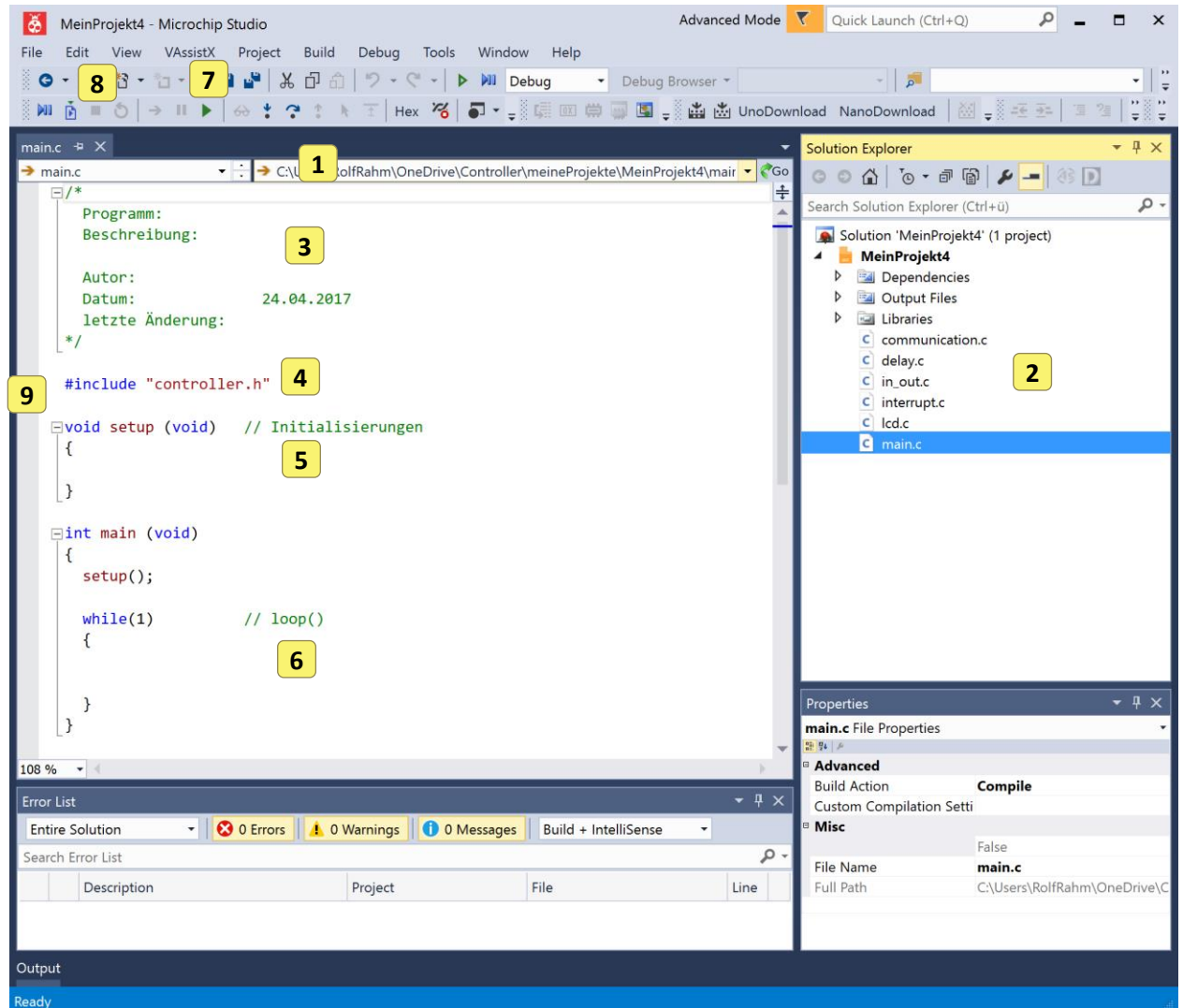




Klicken Sie auf die Vorlage **MeinProjekt** und geben Sie dem Projekt einen sinnvollen Namen. Speichern Sie das Projekt anschließend mit OK in Ihrem Projektordner.

Anschließend wird das Projekt erstellt und Sie können direkt mit der Programmierung in der Datei **main.c** beginnen.




Erläuterung der wichtigsten Elemente für die Programmentwicklung



- 1 Text-Editor: Hier wird der C-Quellcode eingetragen (**main.c**).
- 2 Projektmappen/Solution-Explorer: Hier stehen alle Quelldateien (*.c) die Teil des Projekt-Builds sind. Dazu gehören neben der main.c auch die fünf FA205-Bibliotheken.
- 3 Der Programmkopf sollte immer ausgefüllt werden und besteht mindestens aus einer Kurzbeschreibung des Programms, dem Namen des Programmiers und dem Datum.
- 4 Der Befehl **#include "controller.h"** sorgt dafür, dass die FA205-Funktionen in der Datei main.c bekannt sind.
- 5 In der Funktion setup() werden alle Befehle eingetragen, die nur am Anfang des Programms einmal ausgeführt werden müssen. Dies sind meistens spezielle Initialisierungsroutinen für die Hardware-Funktionen des Mikrocontrollers oder anderer Peripherie.
- 6 In der Endlosschleife while (1) { } stehen die Befehle, die „endlos“ immer wieder abgearbeitet werden, bis der Controller ausgeschaltet oder ein Reset erzeugt wird.
- 7  Programm übertragen und Debugging starten (F5).
- 8  Debugging stoppen (Strg+Umschalt+F5).
- 9 In der grauen Leiste können Breakpoints gesetzt werden.

```
temp = adc_in1();
pwm_duty_cycle((ui
```

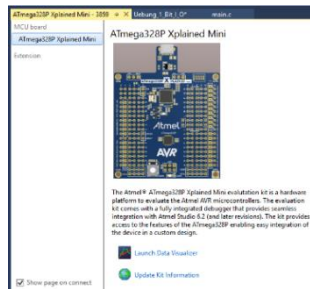
 Friedrich-Ebert-Schule Esslingen FES	Mikrocontroller-Labor	Name: Rahm Datum: 21.10.2017 1_1_Programmieren mit AtmelStudio.docx
	Erstellen eines Projektes für die C-Programmierung	1.1.4



ISP-Programmierung über die DebugWire-Schnittstelle

Für diese Funktion muss sich das Board im ISP-Modus¹ befinden. Wird nicht im Labor verwendet!


Um ein Programm (*.hex) auf den Mikrocontroller zu übertragen, schließen Sie das Board mit einem MicroUSB-Kabel am PC an.



Im Hauptfenster erscheint der Tab „ATmega328P Xplained Mini – xxxx“. Darüber haben Sie Zugriff auf Tools und Online-Informationen rund um das Xplained-Board. Der Tab kann bedenkenlos geschlossen werden.

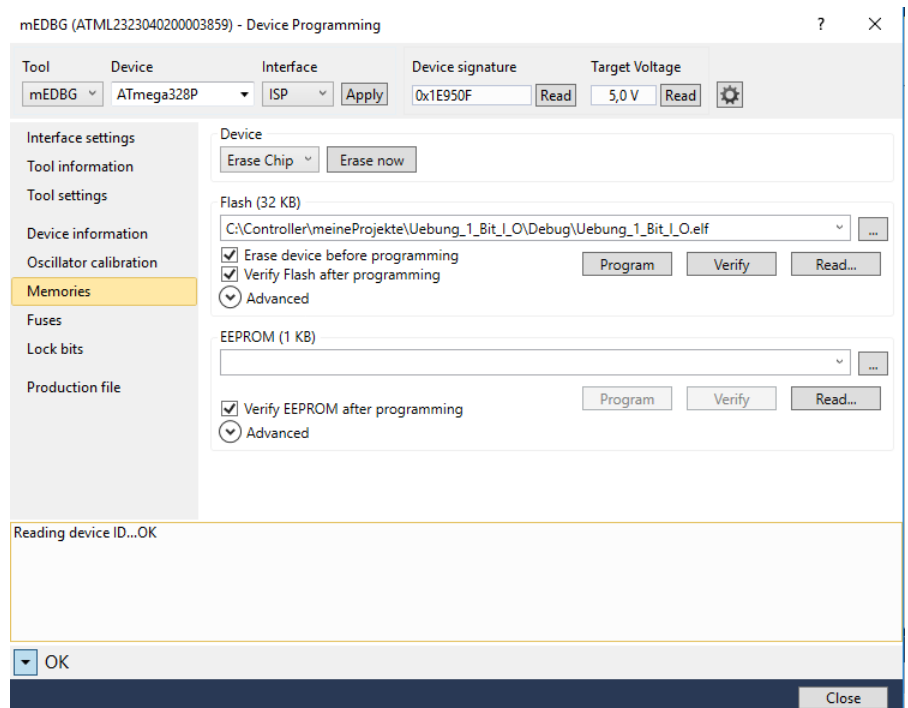
Schreiben Sie das Programm und erstellen Sie die Projektmappe  (F7).

Öffnen Sie das

Programmier-Tool 
(Strg+Umschalt+P).

Nehmen Sie die in der Abbildung gezeigten Einstellungen vor. Achten Sie darauf, dass die Target-Voltage nicht unter 4,5V angezeigt wird (Read betätigen), da sonst der Controller beim Programmieren Schaden nehmen kann.

Im Abschnitt "Memories" sollte die aktuelle Hex- (oder ELF-) Datei bereits ausgewählt sein. Betätigen Sie "Program".



Anschließend wird das Programm übertragen und ein Reset auf dem Controller ausgeführt. Schließen Sie das Programmier-Tool wieder und testen Sie die Funktion des Programms.

Viel Erfolg beim ersten Mikrocontroller-Projekt !

¹ Siehe Anleitung: Arduino Carrier Board mit AtmegaXplained und AtmelStudio 7 (S.9)