

C++ mit dem Arduino :

IDE mit Compiler und Loader aus dem Internet :

Suche nach „Code::Blocks für Arduino“ -> <http://sourceforge.net/projects/arduinoide/>

Laden, Installieren und den Arduino über USB an den PC anschließen.

In Systemsteuerung/System/Gerätmanager muß unter COM+LPT nun der Arduino erscheinen, üblicherweise als COM3:

Probleme ? Dann fehlt es am Treiber für die COM-Emulation an USB. Google

Nötige Dokumentationen :

Datenblatt für den ATMEGA328p laden -> <http://www.atmel.com/Images/doc8161.pdf>

Schaltplan laden -> http://arduino.cc/de/uploads/Main/Arduino_Uno_Rev3-schematic.pdf

Beispielprogramm (noch ohne Objektorientierung) :

Neues Projekt anlegen, in der main.c programmieren :

```
#include <avr/delay.h>
#include <avr/io.h>
int main(void)
{
  DDRB = 32;
  while(1)
  {
    PORTB = 32;
    _delay_ms(1000);
    PORTB = 0;
    _delay_ms(1000);
  };
  return 0;
}
```

Die Zeitverzögerung wird mit einer fertigen Bibliotheksroutine realisiert. Wenn man im Editor die Zeile `#include <avr` beginnt, kommt ein Pulldown-Menü mit den vorrätigen Bibliotheksroutinen.

Deren Funktionen findet man leicht in Google.

Die Bedienung des Ausgangsports findet man im Datenblatt. Kapitel 13 beschäftigt sich mit I/O, in 13.2.1 ist die Funktion der einzelnen Register erklärt. Hier wird Pin5 des Datenrichtungsregisters von PortB auf Ausgang geschaltet (Da hängt, wie im Schaltplan ersichtlich, eine LED dran).

Dann wird diese mit Zeitverzögerung immer umgeschaltet.

Übersetzen, testen :

In Code::Blocks auf den Menüpunkt BUILD, dann wieder auf BUILD. Das „Target“ sollte auf RELEASE stehen. Compiler läuft -> 0 Errors wäre prima !

In C::B auf TOOLS, dort den Arduino Builder, das Hexfile aus dem Project/Release suchen (LOAD SKETCH/HEX) und an COM3: (oder dem Port, an dem der Ardu eben hängt) ausgeben.

-> Na ?