

Programmieren von Mikrocontrollersystemen

Dr. Stefan Enderle

1. Einleitung

Was ist „Programmieren“ ?

- Das Erstellen von Computerprogrammen!
- Aber was ist ein Programm?
- → Viele Programmiersprachen

Ablauf der Programmierung

- **Überlegen eines Algorithmus:**
Problem, Kreativität → Ablaufplan
- **Editieren des Quelltextes:**
Ablaufplan, Tastatur → Quelltext `test.cc`
- **Übersetzen (Compilieren):**
Quelltext → Binärcode `test.bin (.exe)`

Begriffe

- **Algorithmus:**
Abstrakter Programmablauf
Dargestellt z.B. durch Programmablaufplan
- **Quelltext (Source code, Programmtext):**
Eingetipptes Programm
Für (manche) Menschen verständlich
- **Binärcode/Maschinencode:**
Ausführbares Programm
Für Computer "verständlich"

Begriffe

- **Editor:**
Programm, das zum Eingeben des Quelltextes dient
- **Compiler:**
Programm, das Quelltext analysiert und daraus
Maschinencode erzeugt

Beispiel

- **Problem / Aufgabe:**

Ein Roboter soll andauernd ein Dreieck abfahren

Beispiel

- **"Algorithmus":**
 - Fahre ein Stück weit geradeaus
 - drehe dich um einen gewissen Winkel
 - Fahre wieder ein Stück weit geradeaus
 - drehe dich erneut
 - Fahre noch einmal ein Stück
 - drehe dich nochmal um den Winkel
 - Fange wieder von vorne an

Beispiel

- **Verbesserte Aufgabenstellung:**

Ein Roboter soll andauernd ein gleichseitiges Dreieck mit einer Kantenlänge von 50cm im Gegenuhrzeigersinn abfahren.

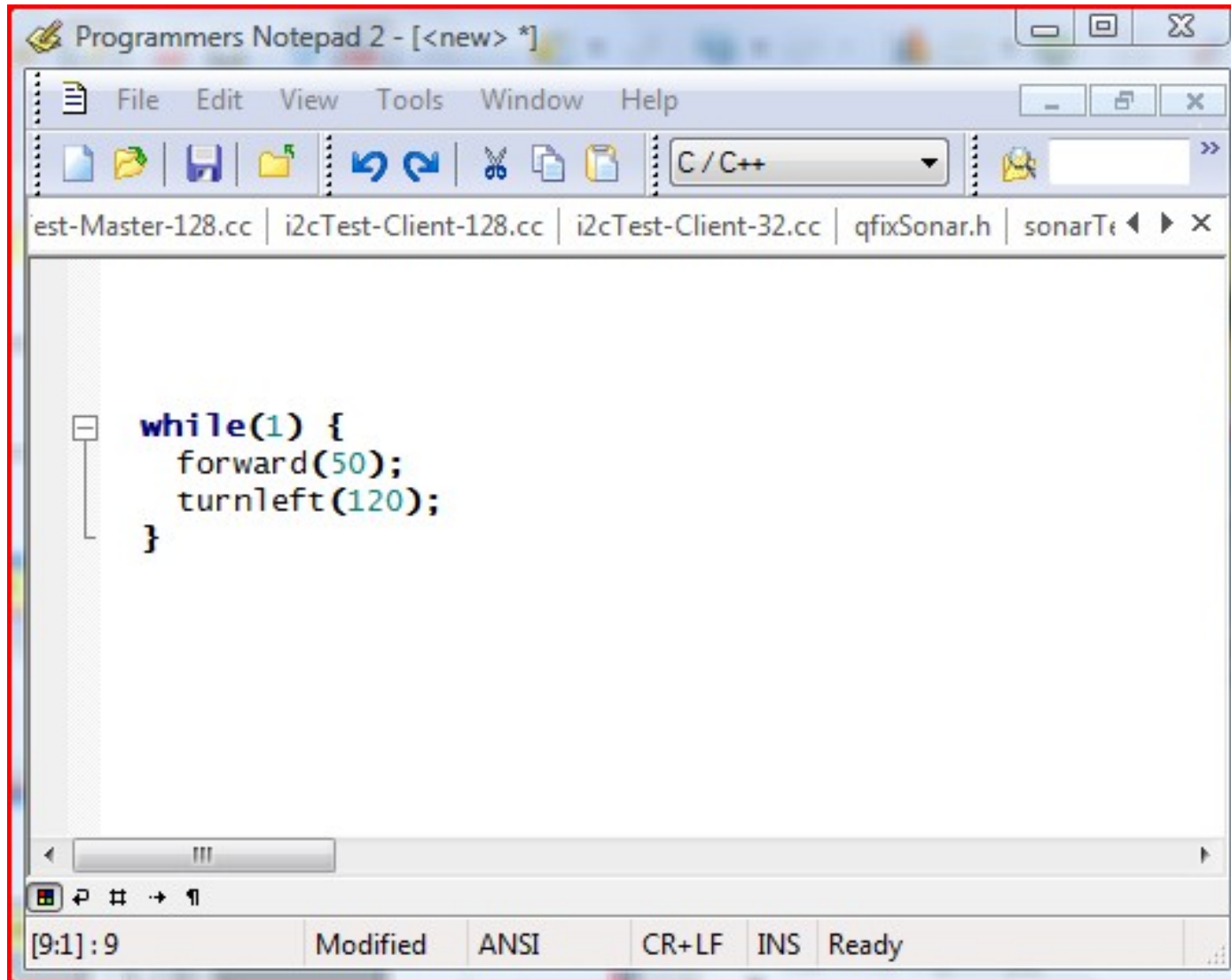
Beispiel

- **Besserer Algorithmus:**
 - Wiederhole für immer:
 - Fahre 50cm geradeaus
 - Drehe um 120 Grad nach links
 - Fahre 50cm geradeaus
 - Drehe um 120 Grad nach links
 - Fahre 50cm geradeaus
 - Drehe um 120 Grad nach links

Beispiel

- **Noch besserer Algorithmus:**
 - Wiederhole für immer:
 - Fahre 50cm geradeaus
 - Drehe um 120 Grad nach links

Editieren



The image shows a screenshot of a code editor window titled "Programmers Notepad 2 - [<new> *]". The window has a menu bar with "File", "Edit", "View", "Tools", "Window", and "Help". Below the menu bar is a toolbar with icons for file operations and editing. A dropdown menu shows "C/C++". The editor displays several open files: "est-Master-128.cc", "i2cTest-Client-128.cc", "i2cTest-Client-32.cc", "qfixSonar.h", and "sonarT...". The main editing area contains the following C++ code:

```
while(1) {  
    forward(50);  
    turnleft(120);  
}
```

The status bar at the bottom shows the cursor position "[9:1]: 9", the file encoding "Modified", the character set "ANSI", the line ending "CR+LF", the keyboard layout "INS", and the state "Ready".

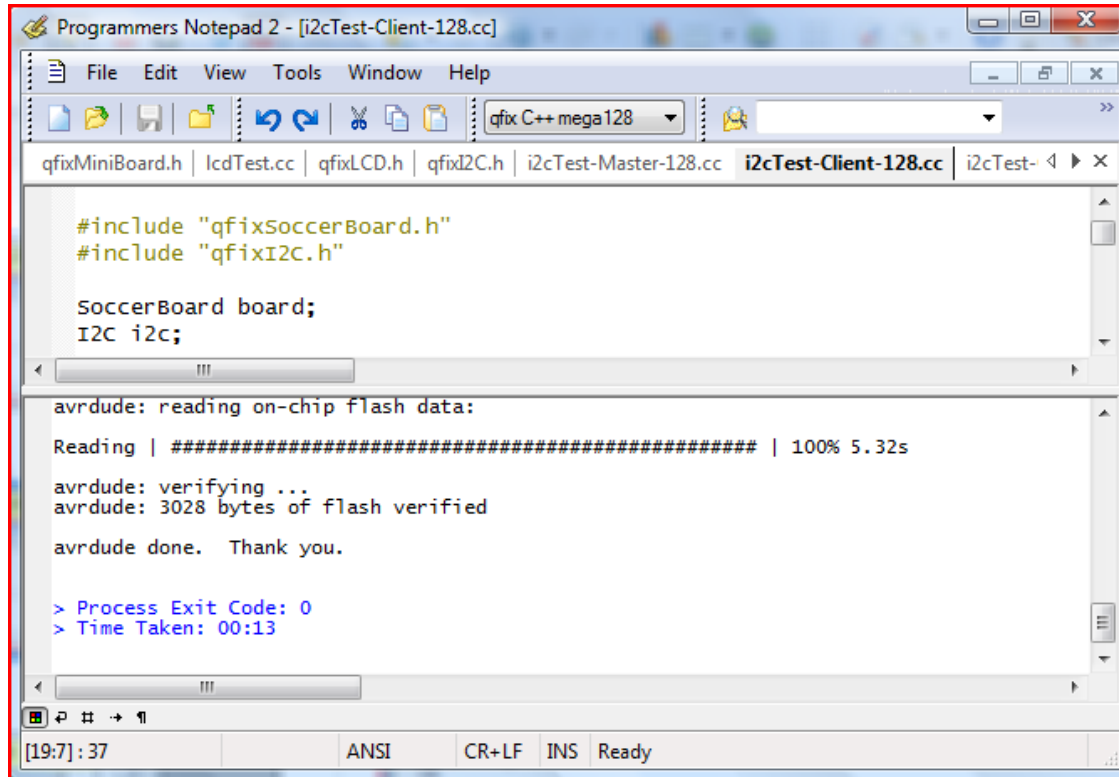
Compilieren (F5)

```
Programmers Notepad 2 - [lcdTest.cc]
File Edit View Tools Window Help
qfix C++ mega32
qfixMiniBoard.h | lcdTest.cc | qfixLCD.h | qfixI2C.h | i2cTest-Master-128.cc | i2cTest-Client-128.cc | i2cTest-C
#include "qfixMiniBoard.h"
#include "qfixLCD.h"

MiniBoard board;
LCD lcd;

int main()
{
> "c:\Programme\qfix\bin\compile-mega32.bat" lcdTest
compiling ...
OK
> Process Exit Code: 0
> Time Taken: 00:01
[1:1]: 32 ANSI CR+LF INS Ready
```

Übertragen (F6)



The screenshot shows a Notepad++ window titled "Programmers Notepad 2 - [i2cTest-Client-128.cc]". The code editor contains the following C++ code:

```
#include "qfixSoccerBoard.h"
#include "qfixI2C.h"

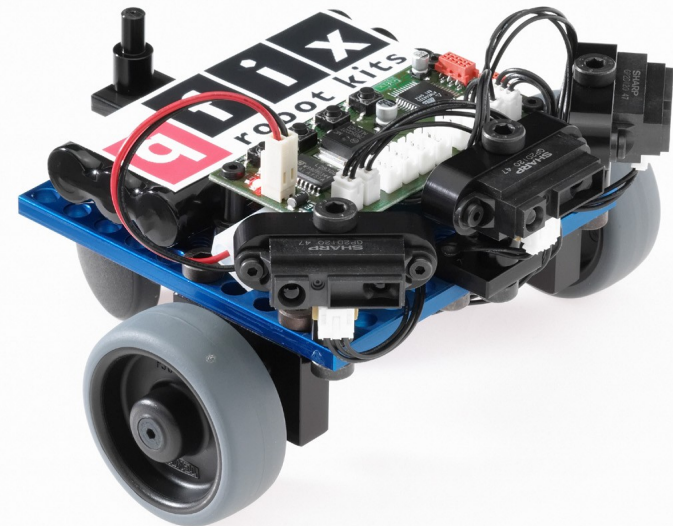
SoccerBoard board;
I2C i2c;
```

Below the code editor is a terminal window showing the output of an AVRDUDE command:

```
avrdude: reading on-chip flash data:
Reading | ##### | 100% 5.32s
avrdude: verifying ...
avrdude: 3028 bytes of flash verified
avrdude done. Thank you.

> Process Exit Code: 0
> Time Taken: 00:13
```

The status bar at the bottom of the terminal shows "[19:7]: 37" and "ANSI CR+LF INS Ready".



Ausführen

