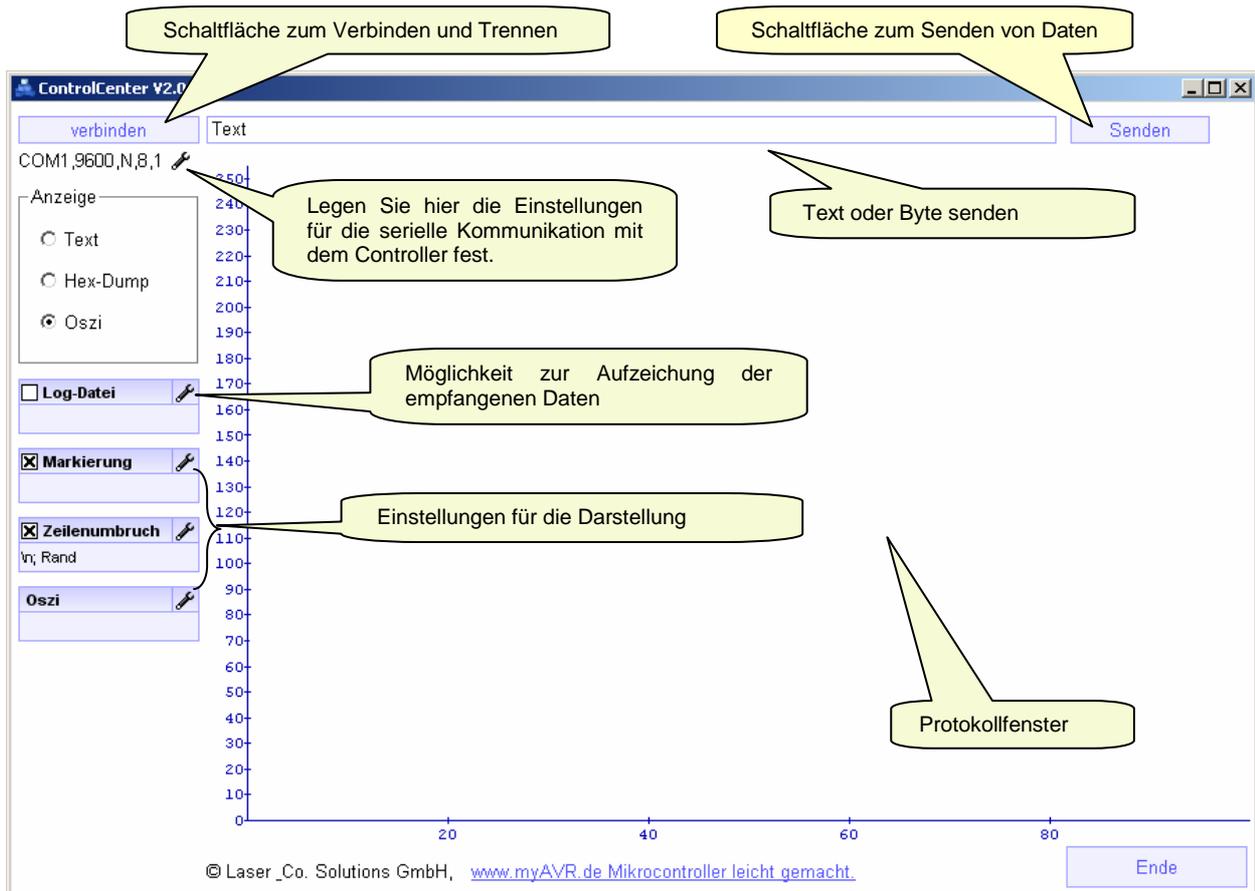


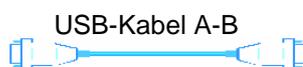
10.2 Das SiSy ControlCenter

Das SiSy ControlCenter ist ein universelles Terminalprogramm zur Kommunikation mit Mikrocontrollerapplikationen, die über eine serielle Schnittstelle (UART) oder USB Anbindung mit virtuellem COM-Port zum PC verfügen. Es kann für Test- und Debug-Meldungen sowie Visualisierung und Protokollierung von Messdaten genutzt werden. Dazu bietet das SiSy ControlCenter umfangreiche Konfigurationsmöglichkeiten.

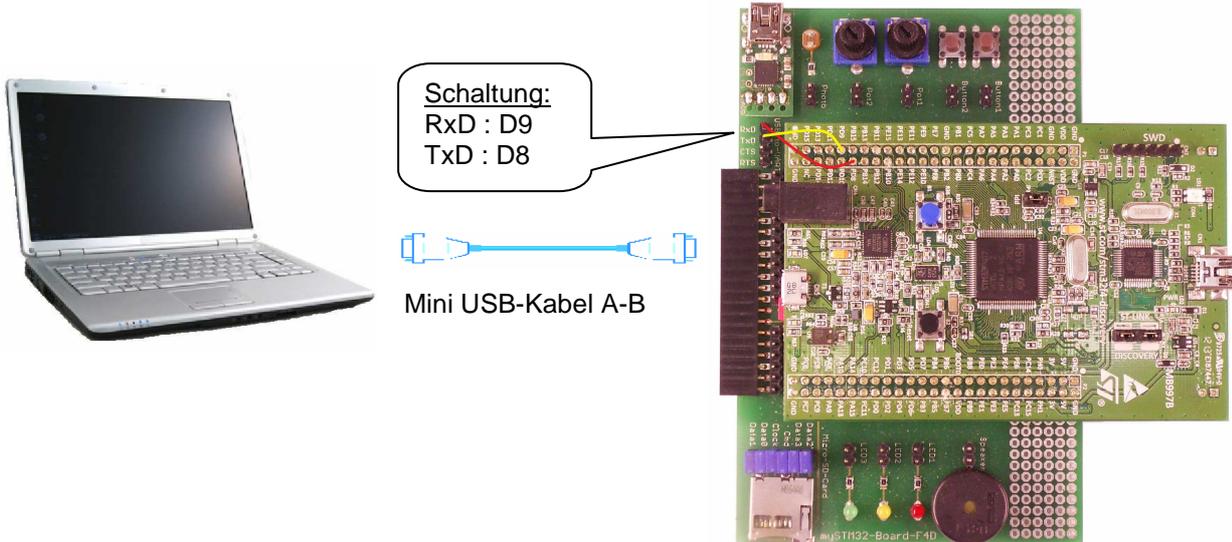


10.2.1 Kommunikation mit dem Controller

Das myAVR Board MK2 verfügt über den USB Programmer mySmartUSB MK2, das myAVR Board MK3 über den USB-Programmer mySmartUSB MK3. Diese Programmer stellen gleichzeitig einen virtuellen COM-Port für die Kommunikation zur Verfügung.



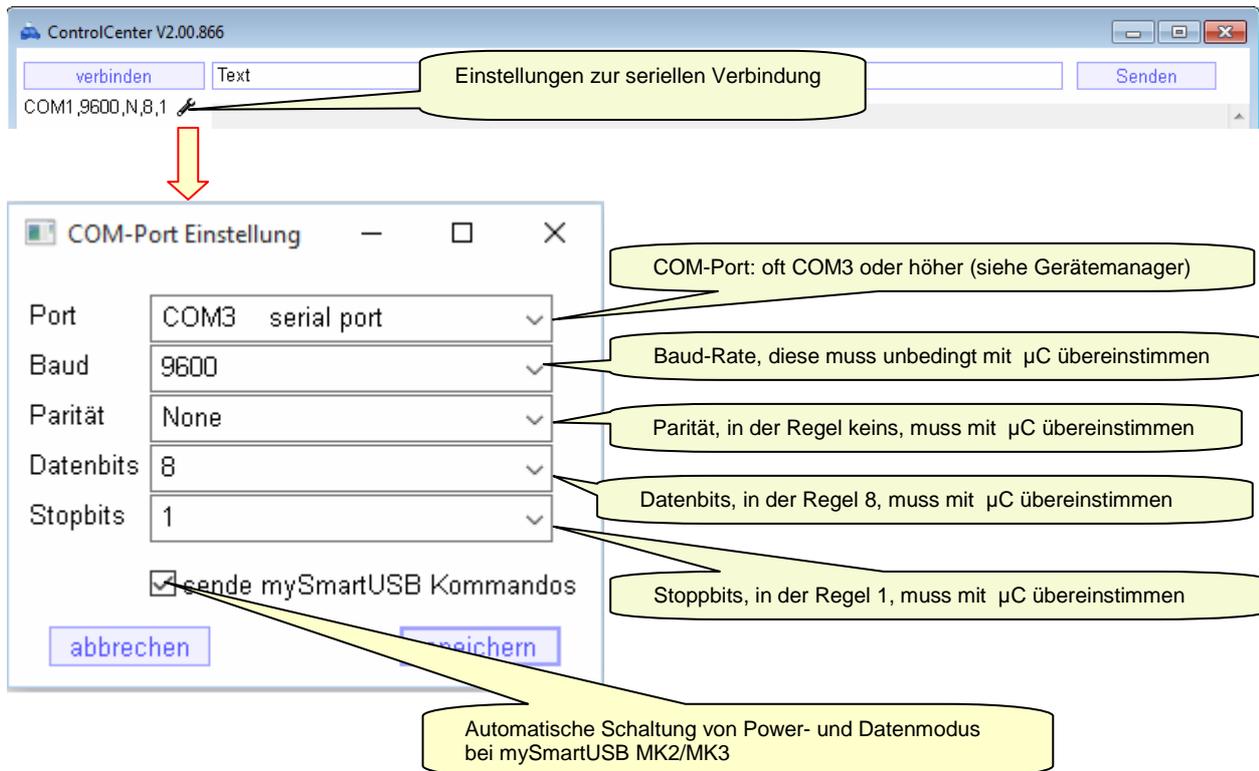
Beispiel mit myAVR Board MK2



Beispiel mit einem mySTM32-Board-F4D

Einstellungen für die seriellen Verbindung

Für eine erfolgreiche Kommunikation mit dem Mikrocontroller ist es wichtig, dass Sender und Empfänger von seriellen Daten die gleichen Parameter für die Datenübertragung konfiguriert haben. Auf der PC Seite erfolgt die Konfiguration der Kommunikationsparameter im SiSy ControlCenter über das Symbol für die Einstellungen der Verbindung. Der entsprechende COM-Port ist z.B. aus dem Gerätemanager zu ermitteln.



Besonderheiten für myAVR Systemboards und mySmartUSB MK2/MK3

Bei einem myAVR Systemboard sowie bei den Programmern mySmartUSB MK2/MK3 können Sie die Spannungsversorgung des Boards und die Umschaltung in den Datenmodus aus dem SiSy ControlCenter heraus steuern. Dazu wählen Sie in den Einstellungen „sende mySmartUSB Kommandos“. Eine manuelle Umstellung ist damit nicht nötig.

Im Mikrocontrollerprogramm sind die gleichen Parameter bei der Initialisierung der UART zu wählen.

```

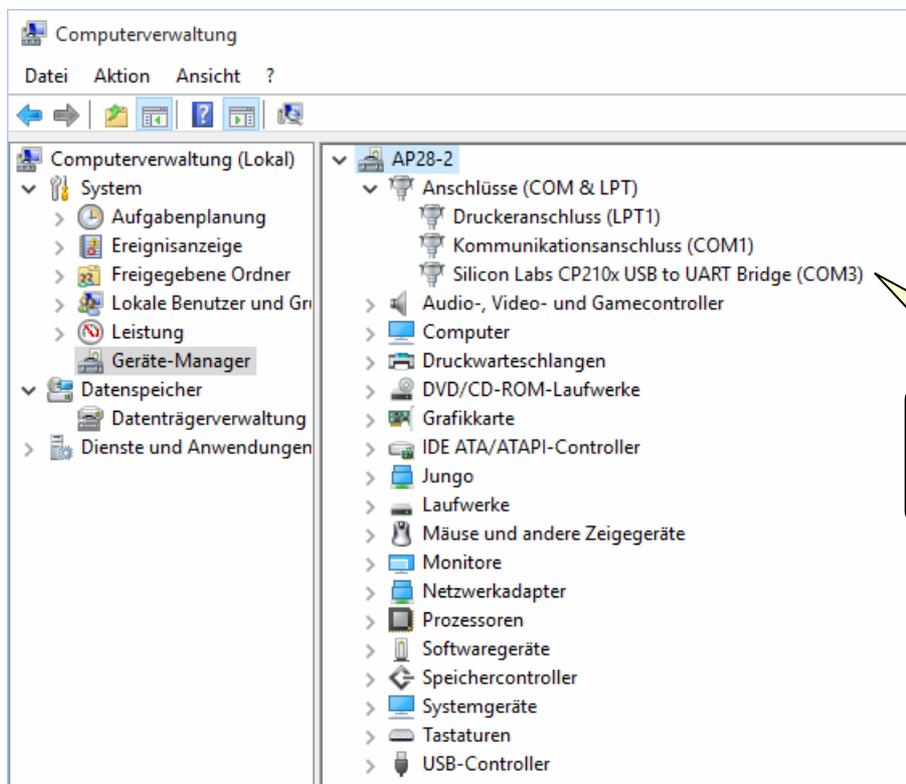
...
public: void onStart()
{
    // hier Initialisierungen durchführen
    wert=0;
    uart.config(9600,UDR);
}
public: void onWork()
...

```

Beispiel für die Konfiguration der UART des Mikrocontrollers mit 9600 Baud.

Geräte-Manager

Die erfolgreiche Verbindung mit dem Controllerboard kann aus dem Gerätemanager entnommen werden, ebenso der entsprechende COM-Port (Nummerierung).



Ermittlung des virtuellen COM-Ports für die Einstellung der seriellen Verbindung

10.2.2 Daten empfangen vom Controller

Das SiSy ControlCenter empfängt Daten über den gewählten COM-Port und stellt diese im Protokollfenster dar. Damit können Statusmeldungen, Fehlermeldungen oder auch Messwerte erfasst werden. Die Voraussetzung ist, dass die serielle Verbindung hergestellt (USB Kabel), korrekt konfiguriert und aktiviert wurde. Die Kommunikation beginnt mit dem Betätigen der Schaltfläche „verbinden“, diese wird ersetzt mit der Schaltfläche „trennen“, womit die Kommunikation beendet werden kann.

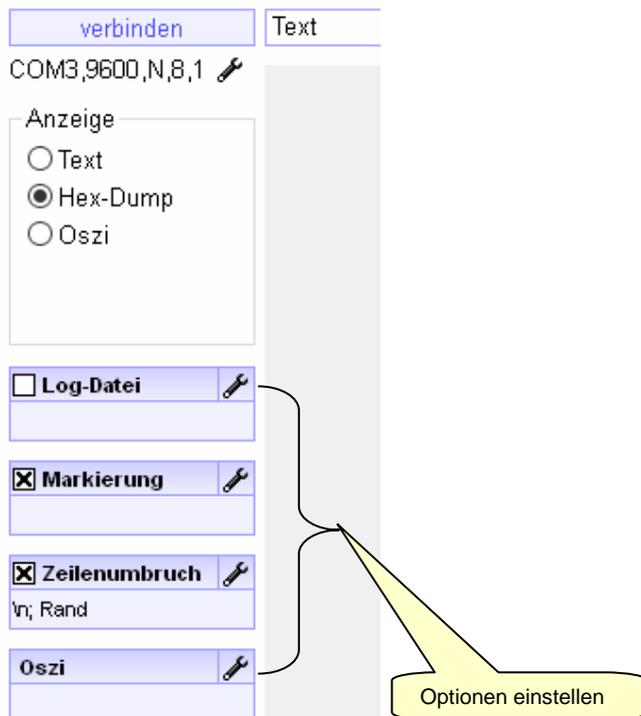
Darstellung der empfangenen Daten

Die empfangenen Daten werden fortlaufend im Protokollfenster dargestellt. Der Darstellungsmodus kann während der Kommunikation umgeschaltet werden.

Das SiSy ControlCenter bietet folgende Darstellungsmodi:

- Text
- Hexadezimal
- Grafik (Oszi)

Für die Darstellungsmodi können über die entsprechenden Schaltflächen Optionen eingestellt werden.



weitere Funktionen

Für die Darstellungen der Daten im Protokollfenster stehen weitere Funktionen zur Verfügung, z.B. das Löschen der angezeigten Daten. Zu erreichen sind diese Funktionen mit Klick der rechten Maustaste im Protokollfenster, womit ein Fenster zur Auswahl der zusätzlichen Funktionen öffnet. Die Funktionsauswahl ist während der Abarbeitung eines Programms möglich.



Der Textmodus

Der Textmodus dient zur Visualisierung alphanumerischer Werte im ASCII Format (Zeichenketten). Zahlen, die mit Zeichenketten gesendet werden, müssen vom Mikrocontrollerprogramm zuvor ins ASCII-Format gewandelt werden (siehe `itoa` und `sprintf`).

Über die Schaltfläche „Zeilenumbbruch“ lässt sich das Protokollfenster weiter konfigurieren. So kann der Zeilenumbbruch z.B. nach einer festgelegten Zeichenanzahl erzwungen werden.

Optionen

Zeilenumbbruch

- Newline-Zeichen ('\n' = 10 = 0x0A)
- ganze Zeichenkette:
- jedes dieser Zeichen:
- nach Zeichen
- am Fensterrand

Über die Schaltfläche „Markierung“ lassen sich im Text festgelegte Zeichen oder Zeichenketten hervorheben.

Optionen

Markierung

Abc	<input type="checkbox"/>	<input style="width: 95%;" type="text" value="myAVR"/>	<input type="text" value="ganze Zeichenkette"/>
Abc	<input type="checkbox"/>	<input style="width: 95%;" type="text" value="e"/>	<input type="text" value="jedes einzelne Zeichen"/>
Abc	<input type="checkbox"/>	<input style="width: 95%;" type="text" value="'A' 0x0d 10"/>	<input type="text" value="Bytefolge"/>
Abc	<input type="checkbox"/>	<input style="width: 95%;" type="text" value="\t"/>	<input type="text" value="jedes einzelne Byte"/>

trennen

Text

COM3,9600,N,8,1

✎

Hallo Welt und myAVR, Hallo Welt und myAVR,

Der Grafikmodus (Oszi)

Messwerte können auch grafisch visualisiert werden. Dabei werden die Werte fortlaufend und byteweise (Wertebereich 0 bis 255) als Punkte in einem Koordinatensystem dargestellt. Die X-Achse repräsentiert den zeitlichen Verlauf, die Y-Achse den Wertebereich der Daten. Die Anzeige kann als einzelne Punkte oder Linien erfolgen.

Die Einstellung der Optionen erfolgt über die Schaltfläche „Oszi“.

Optionen

Oszi

Anzeigeart:

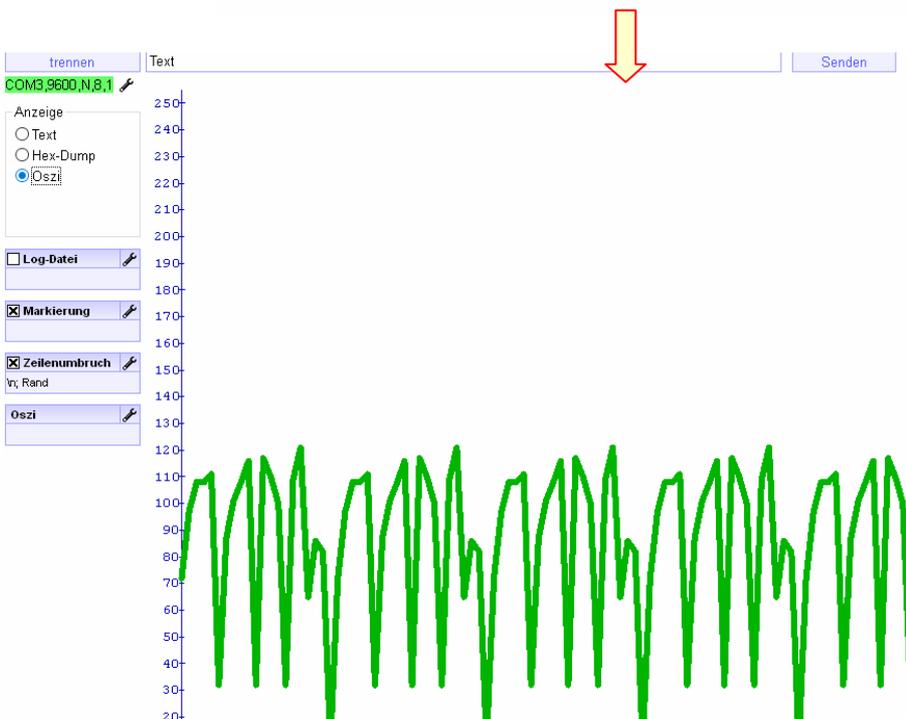
Linie

Punkte

Linienstärke und Punktgröße

mittel ▾

maximale Anzahl der dargestellten Werte: 200



10.2.3 Empfangene Daten speichern

Bei der Erfassung von Messdaten die vom Mikrocontroller an den PC gesendet werden ist es oft wichtig, diese in eine Datei zu speichern. Damit wird eine Weiterverarbeitung der Daten in entsprechenden Programmen möglich. Das SiSy ControlCenter ermöglicht es, Protokolldaten aufzuzeichnen (Log-Datei, Rekorderfunktion). Um diese Funktion zu aktivieren, wählen Sie die Schaltfläche „LogDatei“.

In den Optionen muss der Dateiname und Pfad der Log-Datei angegeben werden. Die Weiterverarbeitung der Daten erfolgt entsprechend der Möglichkeiten der Zielanwendung (z.B.: Öffnen, Importieren oder Zwischenablage).

Optionen

Log-Datei

Dateiname:

Anzahl der Log-Dateien:

- eine Datei = immer überschreiben/anfügen
- mehrere Dateien mit Nummerierung (z.B. data_001.log)
- mehrere Dateien mit Zeitstempel (z.B. data_2012-10-16_13-56-01.log)

Datei neu beginnen bei:

- verbinden
- Dateigröße KByte

Zusätzliche Texte bei:

- verbinden
- trennen

Hinweise:
Die zusätzlichen Texte können mit aktuellen Zeitangaben versehen werden (z.B.: %Y-%m-%d %H:%M:%S).



```
Test.txt - Editor
Datei Bearbeiten Format Ansicht ?
Hallo Welt und myAVRHallo Welt und myAVRHallo Welt und myAVR
Hallo Welt und myAVRHallo Welt und myAVRHallo Welt und myAVR
Hallo Welt und myAVRHallo Welt und myAVRHallo Welt und myAVR
Hallo Welt und myAVRHallo Welt und myAVRHallo Welt und myAVR
Hallo Welt und myAVRHallo Welt und myAVRHallo Welt und myAVR
Hallo Welt und mvAVRHallo Welt und mvAVRHallo Welt und mvAVR
```

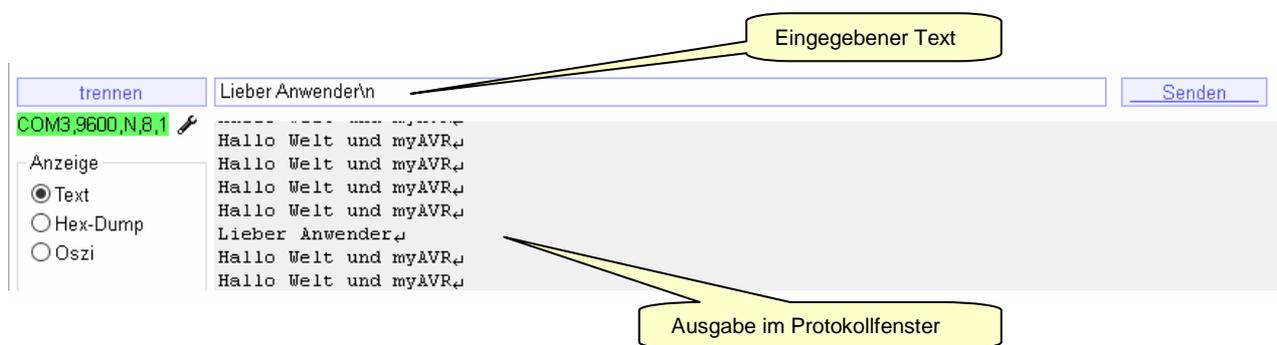
10.2.4 Daten an den Controller senden

Über das SiSy ControlCenter können Daten im Text- oder Zahlenformat an den Mikrocontroller gesendet werden. Es können einzelne Zeichen, aber auch Zeichenketten gesendet werden. Die Erfassung der Zeichen bzw. Zeichenkette erfolgt in der Zeile „Text“. Das Senden von Text und Zahlen funktioniert in C-Style:

- \n für Newline (=10=0x0A),
- \r für Return (=13=0x0D),
- \t für Tab (=8),
- \x.. für beliebiges Zeichen als HEX Code (z.B. \x0C für 12=0x0C)
-

Mit der zugehörigen Schaltfläche „Senden“ wird immer der gesamte Inhalt der Eingabezeile „Text“ gesendet.

In unserem Programmbeispiel wird die Ausgabe des im Programm hinterlegten Textes (hier „Hallo Welt und myAVR“) einmal mit dem Inhalt der Eingabezeile unterbrochen, sobald die Schaltfläche „Senden“ aktiviert wird.



Beispielprogramm

Die Daten des folgenden Programmbeispiels sollen als Testdaten dienen.

Es wird die Zeichenkette „Hallo Welt und myAVR“ an das SiSy ContolCenter gesendet. Desweiteren fragt das Programm ab, ob über die Textzeile des SiSy ContolCenters ein Text gesendet wird. Ist das der Fall wird dieser einmal im Protokollfenster ausgegeben.

```
//-----
// Titel      : Datenkommunikation per UART
//-----
// Funktion   : Senden und Empfangen von Daten über UART
//             Entwickelt mit SiSy und PEC (Portable Embedded Classes)
//-----
// Prozessor  : Atmega8 mit 3,6864 MHz
// Sprache    : AVR C++
//-----
class Application : public Controller
{
    // Bausteine und Attribute .....
    protected: uint8_t wert;
    Uart uart;

    // Funktionen .....
    public: void onPower()
    {
        // kleine aber nötige Vorabinitialisierungen
    }
    public: void onStart()
    {
        // hier Initialisierungen durchführen
        wert=0;
        uart.config(9600,UDR);
    }
    public: void onWork()
    {
        if (uart.hasData()) // wenn Daten da sind -> Echo
        {
            char c;
            c = uart.getBytes();
            uart.sendByte(c);
        }
    }
    public: void onTimer10ms() // alle 10 ms
    {
    }
    public: void onTimer100ms() // alle 100 ms
    {
    }
    public: void onTimer1s() // jede Sekunde
    {
        uart.sendString("Hallo Welt und myAVR\n");
    }
    public: void onEvent(const Object& sender, uint8 data)
    {
        // Ereignismeldungen von Appmodulen
        // Sender == Attribut, Data beschreibt das Ereignis
    }
} app; // Anwendungsinstanz
```