

# myAVR MK2-Treiber für WinFACT

## Leistungsumfang

Der vorliegende Treiber ermöglicht die Kommunikation zwischen BORIS und dem Microcontrollerboard MK2 der Fa. myAVR bzw. anderen Boards mit einem ATmega8-Prozessor. Da die Quelltexte des Treibers mitgeliefert werden, kann er Bedarf auch an andere Boards/Prozessoren angepasst werden. Der Treiber besteht aus zwei Komponenten:

- Der User-DLL *myAVRMK2.dll* für BORIS
- Dem HEX-File *WFmyAVRMK2.hex* für das MK2-Board

Der Treiber unterstützt folgende Funktionen:

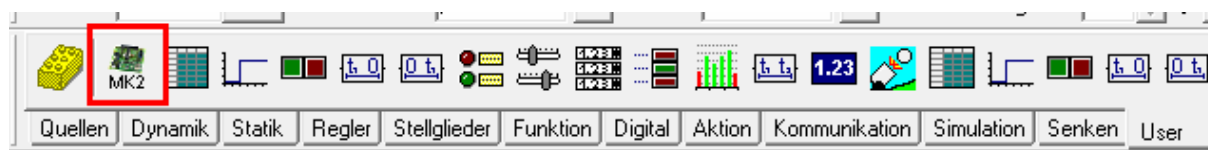
- Einstellbarer COM-Port
- Digitale und Analoge Eingänge (10 Bit)
- Digitale Ausgänge
- PWM-Ausgänge (8 Bit)
- Alle I/O-Ports beliebig konfigurierbar
- Lauffähig unter WinFACT 6/7/8

## Hinweise zur Installation

Zur Installation des Treibers gehen Sie wie folgt vor:

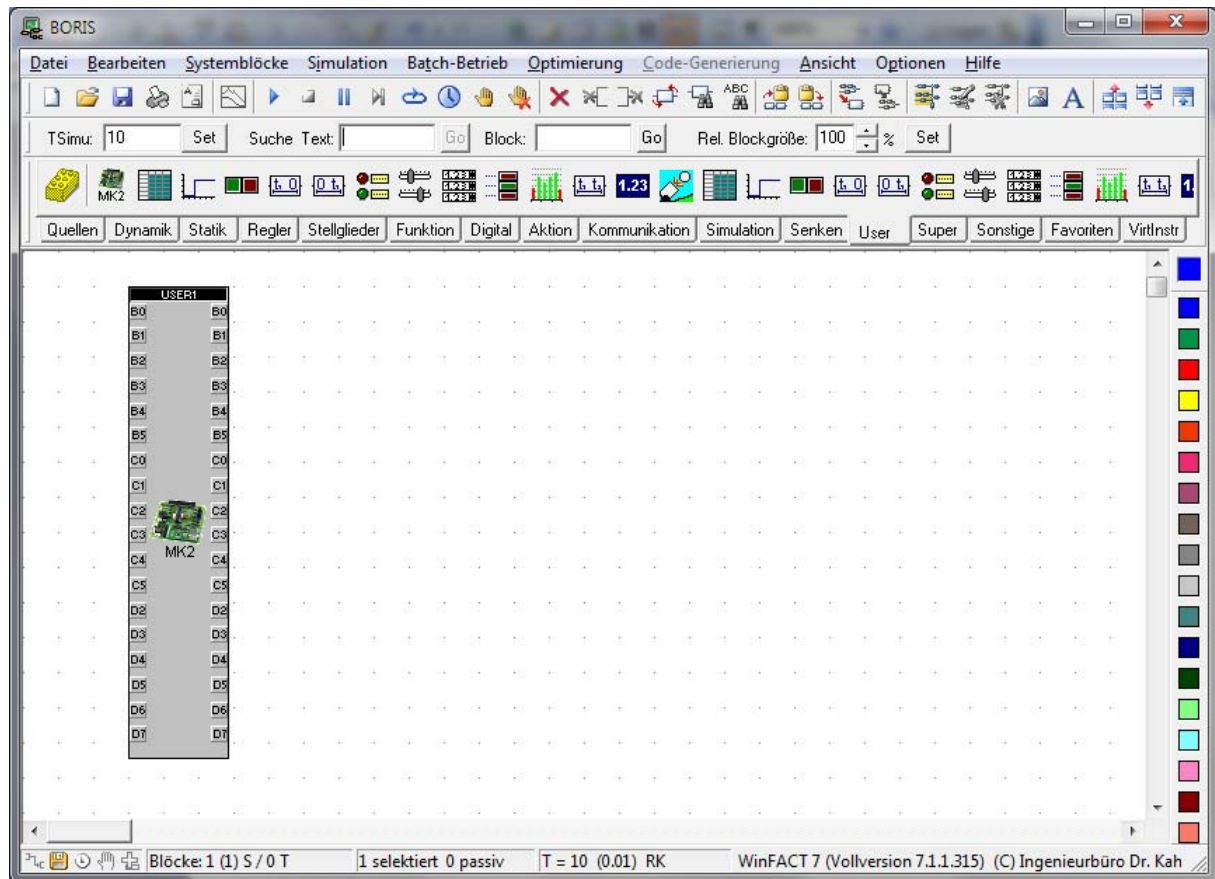
- Kopieren Sie die Dateien *myAVRMK2.dll*, *myAVRMK2.bmp*, *myAVRMK2\_t.bmp* und *myAVRMK2\_p.bmp* in das Unterverzeichnis \UserDLLs Ihrer WinFACT-Installation.
- Brennen Sie das HEX-File *WFmyAVRMK2.hex* auf das MK2-Board.

Nach der Installation des Treibers finden Sie den Treiber-Block in BORIS auf der Palette *User* der Systemblock-Toolbar (Bild 1).



**Bild 1** myAVR MK2-Block in der BORIS-Systemblock-Toolbar

Durch einen Klick auf das MK2-Icon der Toolbar wird ein MK2-Block in die BORIS-Systemstruktur eingefügt (Bild 2).



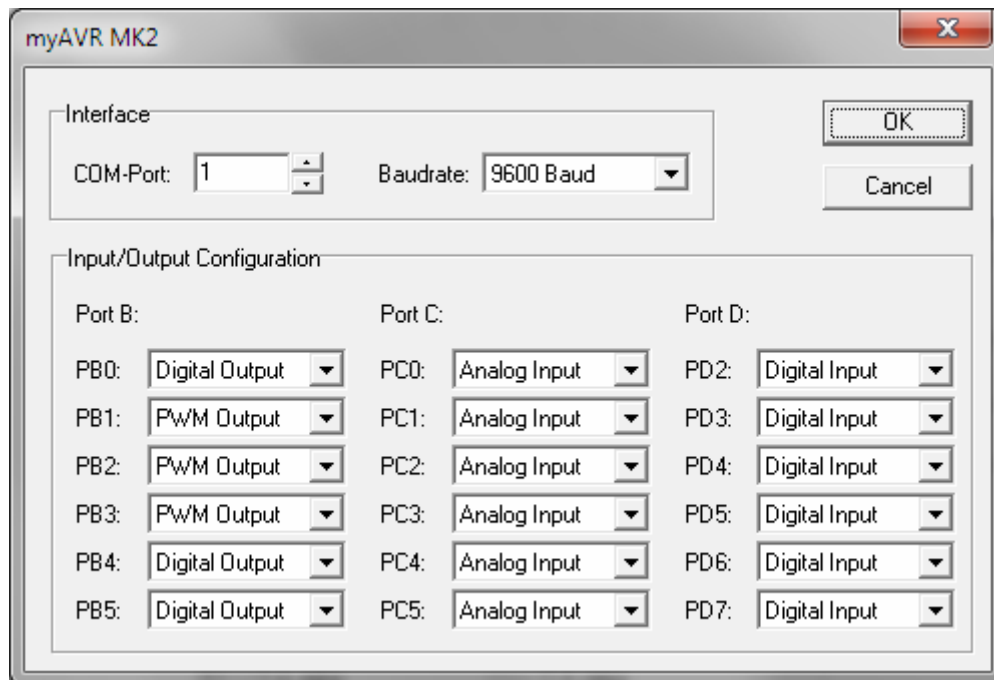
**Bild 2** BORIS-Hauptfenster mit eingefügtem MK2-Block

Da die I/O-Ports B, C und D beliebig konfigurierbar sind, sind alle 18 verfügbaren Pins des MK2 (B0 ... B5, C0 ... C5, D2 ... D7) sowohl als Blockeingänge als auch als Blockausgänge verfügbar. Ist ein Pin als *Eingang* konfiguriert, erscheint der über den Pin eingelesene Wert am entsprechenden *Blockausgang* in BORIS und umgekehrt.

**Beispiel:** Wurde Pin PB2 als digitaler Eingang konfiguriert, erscheint am Blockausgang B2 eine 0, wenn an PB2 LOW-Signal anliegt, und eine 5 (entsprechend dem HIGH-Digitalpegel in BORIS), wenn an PB2 HIGH-Signal anliegt.

## Parametrierung des Blocks

Bild 3 zeigt den Parameterdialog des MK2-Blocks.



**Bild 3** Parameterdialog des MK2-Blocks

Im oberen Bereich werden COM-Port und Baudrate eingestellt (Hinweis: Die voreingestellte Baudrate von 9600 darf nur verändert werden, wenn das auf dem MK2 laufende Programm entsprechend angepasst wird!). Im unteren Bereich können alle verfügbaren Ein-/Ausgänge des Boards konfiguriert werden.

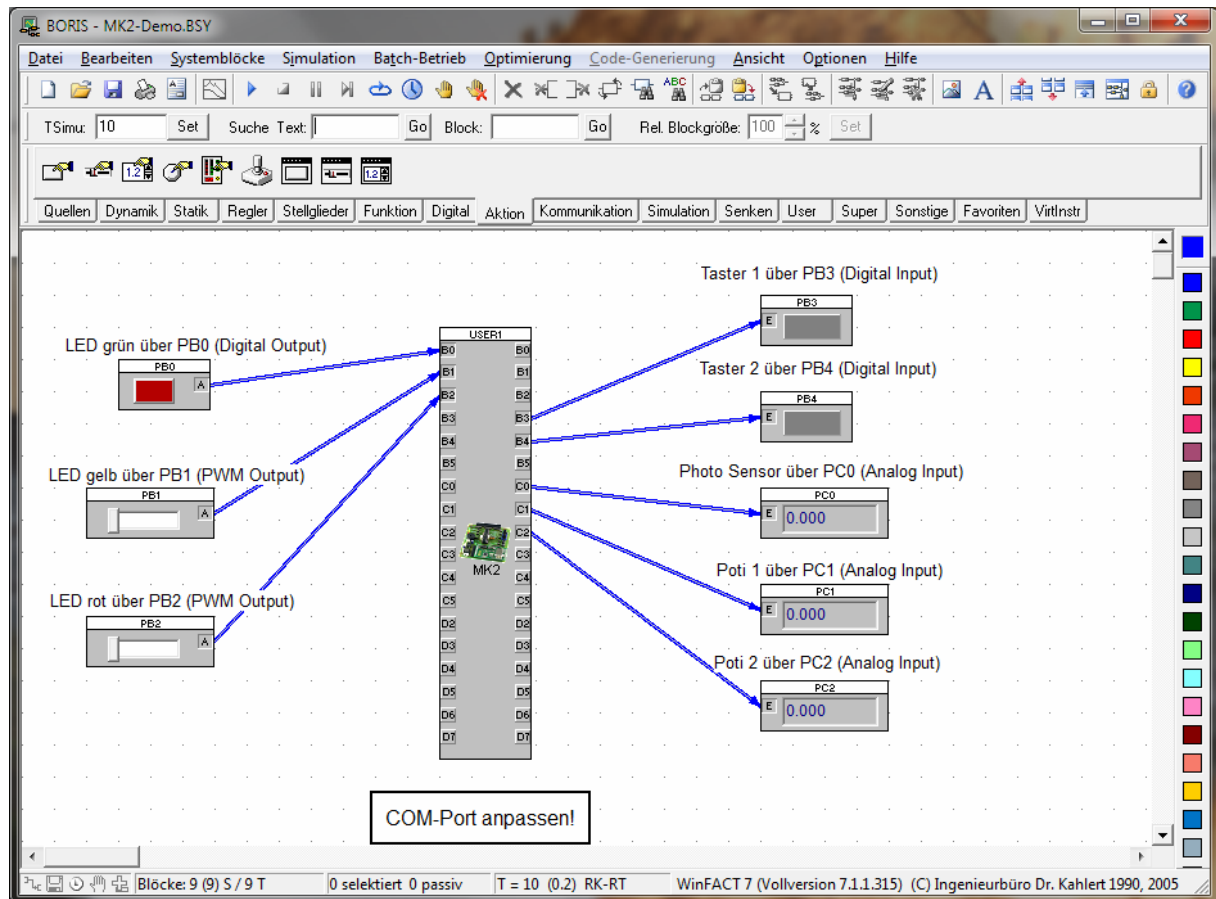
## Hinweise zum Betrieb

Beachten Sie bei der Nutzung des Treibers folgende Hinweise:

- Das MK2-Board muss sich im *Datenmodus* befinden (grüne LEDs am USB-Konnektor müssen leuchten!)
- Der ATmega8 muss mit dem externen 3,6864 MHz-Quartz laufen (ggf. Fuse-Bits anpassen!)
- BORIS sollte im *Echtzeitmodus* laufen (Simulationsschrittweite 0.1 sec oder darüber)
- Der Wertebereich für PWM-Ausgänge liegt zwischen 0 ... 255 (8 Bit), der Wertebereich für Analogeingänge zwischen 0 ... 1023 (10 Bit)

## Beispiel

Bild 4 zeigt die mit dem Treiber gelieferte Beispieldatei *MK2-Demo.bsy*.



**Bild 4** Beispieldatei *MK2-Demo.bsy*

Zum Testen des Beispiels müssen auf dem MK2-Board folgende Verbindungen gezogen werden:

Sensor/Aktor	Pin
LED grün	PB0
LED gelb	PB1
LED rot	PB2
Photo Sensor	PC0
Poti 1	PC1
Poti 2	PC2
Taster 1	PB3
Taster 2	PB4

## Übertragungsprotokoll

```
// myAVR MK2 Driver for WinFACT (all versions)
// DELHPI 2009
// AVR Studio 4
//
// ===== Commands:
//
// ----- Write Output Pin
```

```

//
// Step 1: BORIS sends 4-Byte-Command to MK2
//
// Command structure: Wxyz
//
// x: 'B', 'C', 'D' Port to write (char)
// y: 0 ... 7 Pin to write (byte)
// z: Pin Mode (char)
//     'O': Digital Output
//     'P': PWM Output
//
// Step 2: MK2 returns 'OK'
//
// Step 3: BORIS sends value to write (byte):
//         0 or 1 for Digital Output
//         0 ... 255 for PWM Output
//
// Step 4: MK2 returns 'OK'
//

// ----- Read Input Pin
//
// Step 1: BORIS sends 4-Byte-Command to MK2
//
// Command structure: Rxyz
//
// x: 'B', 'C', 'D' Port to read (char)
// y: 0 ... 7 Pin to read (byte)
// z: Pin Mode
//     'I': Digital Input
//     'A': Analog Input
//
// Step 2:
//
// MK2 returns byte value 0 or 1 for Digital Input
// MK2 returns 2-byte value 0 ... 1023 for Analog Input (Low Byte
first)
//

```

## Treiber-Quelltexte

Die Treiber wurden mit DELPHI 2009 (BORIS-User-DLL) bzw. dem AVR Studio 4 (MK2-Programm) erstellt. Anpassungen an andere Entwicklungsumgebungen sollten aber ohne allzu großen Aufwand möglich sein.