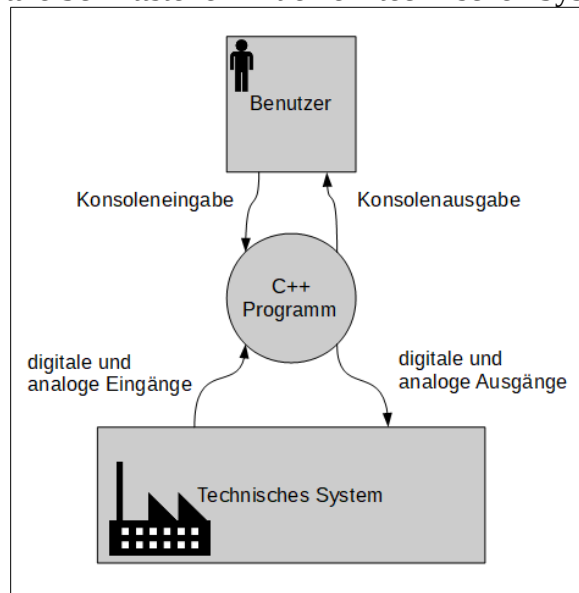


Simulationsumgebung SimSTB

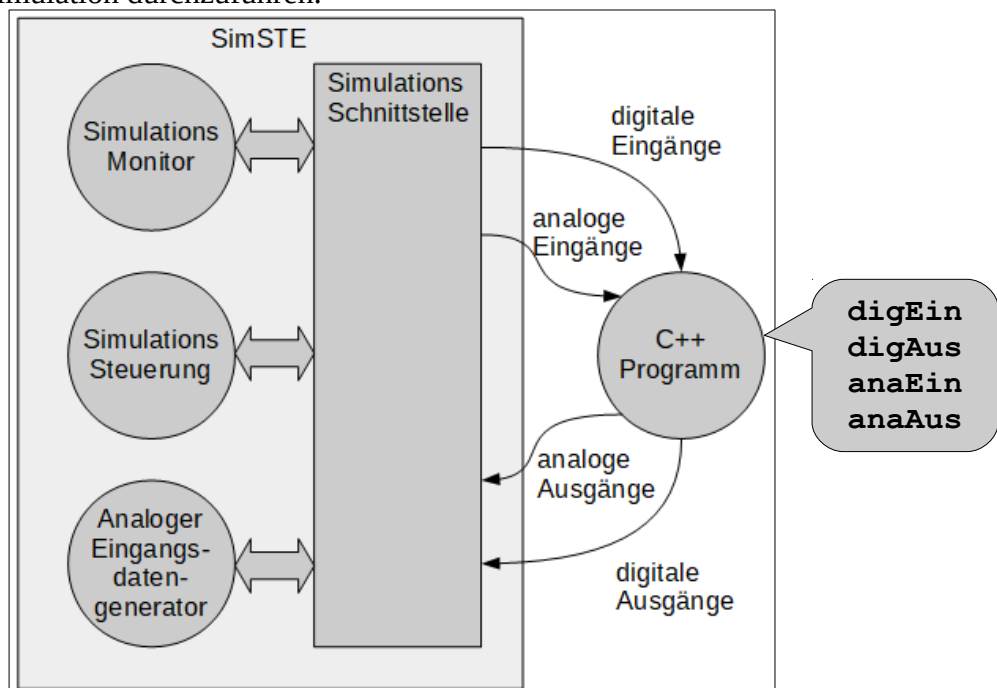
Version	0.2
Datum	24.07.2019

1 Allgemeines

Oft muss ein Programm nicht nur über die Konsole mit dem Benutzer kommunizieren, sondern auch über analoge und digitale Schnittstellen mit einem technischen System.



Die Simulationsumgebung **SimSTB** erlaubt es, dies für Schulungszwecke auch ohne zusätzliche Hardware mittels Simulation durchzuführen.



SimSTB besteht aus zwei Teilen:

1. Der eigentlichen **Simulationsumgebung** und **Steuerprogrammen** für diese. Die Steuerprogramme sind in Abschnitt 3 beschrieben. Die Installation in Abschnitt 2.
2. Einer **Programmier-Schnittstelle**, um aus eigenen Programmen die Simulationsumgebung zu nutzen. In Abschnitt 4 ist beschrieben, wie Sie eigene Programme erstellen können.

Um die Simulationsumgebung SimSTB aus eigenen Programmen zu nutzen, stehen 4 einfache C++-Funktionen zur Verfügung.

Schnittstelle	Funktion	Kanäle	Typ	Richtung
Digitaler Eingang	digEin	0 .. 15	digital	Eingang
Digitaler Ausgang	digAus	0 .. 15	digital	Ausgang
Analoger Eingang	anaEin	0 .. 7	analog	Eingang
Analoger Ausgang	anaAus	0 .. 7	analog	Ausgang

Die Kanäle bestimmen die Anzahl der jeweiligen Schnittstellen.

2 Lokale Installation der Simulationsumgebung

1. Kopieren Sie das bereitgestellte Simulationsverzeichnis samt Unterverzeichnissen nach „C:\“
2. Kontrollieren Sie, ob folgende Verzeichnis-Struktur und Dateien vorhanden sind.

```

C:
├── Sim
│   ├── anaaus.txt
│   ├── anaein.txt
│   ├── digaus.txt
│   ├── digein.txt
│   ├── beispiele
│   │   └── beispiel_ls4_01.cpp
│   ├── bin
│   │   ├── SimAnaEinGenerator.exe
│   │   ├── SimMonitor.exe
│   │   └── SimSTBuerung.exe
│   └── source
│       ├── simintern.h
│       ├── simulation.cpp
│       └── simulation.h

```

3 Steuerung der Simulationsumgebung SimSTB

Im Unterverzeichnis `bin` finden Sie drei Programme zur Bedienung der Simulationsumgebung:

1. Simulations Monitor

Mit Hilfe des Programms `SimMonitor.exe` können Sie digitalen und analogen Ein- und Ausgänge überwachen. Die Werte werden im Sekundentakt aktualisiert. Starten können Sie den Monitor über einen einfachen Doppelklick auf die Exe-Datei.

The screenshot shows the 'Simulator-Monitor' window. It displays two main sections: 'Digitale Eingänge' and 'Analoge Eingänge'. Each section has a corresponding 'Ausgänge' (Outputs) section. The digital inputs are listed from 0 to 15, and the analog inputs from 0 to 7. The outputs are also listed from 0 to 15 for digital and 0 to 7 for analog. The values are updated in real-time.

2. Simulations Steuerung

Mit Hilfe des Programms `SimSTBuerung.exe` können Sie digitalen und analogen Ein- und Ausgänge manuell setzen.

Die digitalen Eingänge können mit Hilfe der Befehle (1) – (4) sowohl pauschal als auch individuell gesetzt werden. Die analogen Eingänge können mit Hilfe der Befehle (5) – (6) sowohl pauschal als auch individuell gesetzt werden. Die Ausgänge können mit Hilfe der Befehle (a) und (b) zurückgesetzt werden.

The screenshot shows the 'SimSTBuerung.exe' window. It lists several control options: (1) - Alle digitalen Eingänge 0, (2) - Alle digitalen Eingänge 1, (3) - Digitaler Einzeleingang auf 0, (4) - Digitaler Einzeleingang auf 1, (5) - Alle analogen Eingänge 0, (6) - Analogen Eingänge setzen, (a) - Alle digitalen Ausgänge 0, (b) - Alle analogen Ausgänge 0, and (e) - Programm beenden. The user is prompted to select an option.

3. Analoger Eingangsdatengenerator

Mit Hilfe des Programms `SimAnaEinGenerator.exe` können Sie simulierte Messwerte für die analogen Eingänge erzeugen. Zur Zeit können Zufallswerte und sinusförmige Werte erzeugt werden.

The screenshot shows the 'Generator für analoge Eingangssignale' window. It displays parameters for generating analog input signals: Id: 0, Form: 1, Amplitude: 15, Periodendauer (s): 30. It also shows the start time, current time, and elapsed time. The current value is displayed as -12.9904.

4 Erstellung eigener Programme für die Simulationsumgebung SimSTB

1. Um die Simulationsumgebung SimSTB nutzen zu können, müssen die Header und CPP-Dateien aus dem `source`-Verzeichnis der Simulationsumgebung in das Verzeichnis kopiert werden, indem auch die CPP-Datei des nutzenden Programms ist.
2. Die Datei `simulation.cpp` ist als bereits existierende Datei in das Projekt einzubinden. Ansonsten kann sie, wie auch die Header-Datei `simintern.h` vollständig ignoriert werden.
3. Die Header-Datei `simulation.h` muss in der eigenen CPP-Datei inkludiert werden. Dabei sind Anführungszeichen und keine spitzen Klammern zu verwenden.
4. Danach kann normal weiter programmiert werden, wobei man die vier Funktionen zur Nutzung der Simulationsumgebung benutzen darf.

Funktionsprototypen:

```
const int DIGMAXLAENGE = 16;
const int ANAMAXLAENGE = 8;

bool digEin( int id);
void digAus( int id, bool wert);

double anaEin( int id);
void anaAus( int id, double wert);
```

Beispiel-Code (Auszug; vollständig in Unterordner beispiele):

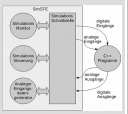
```
...
#include "simulation.h"
...
int main()
{
    bool ende = false;
    double wert;
    ...
    while( ende != true)
    {
        wert = anaEin( 0);
        cout << wert << endl;
        Sleep( 1000);
        ende = digEin( 0);
    }
    digAus( 15, 1);
    anaAus( 7, -123.456);
    ...
}
```

Analoger Eingang

Digitaler Eingang

Digitaler und
Analoger Eingang

5 SimSTB Ein- und Ausgangsbelegung

SinSTE Ein- und Ausgangsbelegung					
Digital Eingänge			Digitale Ausgänge		
	<input type="radio"/>	DE0	DA0	<input type="radio"/>	
	<input type="radio"/>	DE1	DA1	<input type="radio"/>	
	<input type="radio"/>	DE2	DA2	<input type="radio"/>	
	<input type="radio"/>	DE3	DA3	<input type="radio"/>	
	<input type="radio"/>	DE4	DA4	<input type="radio"/>	
	<input type="radio"/>	DE5	DA5	<input type="radio"/>	
	<input type="radio"/>	DE6	DA6	<input type="radio"/>	
	<input type="radio"/>	DE7	DA7	<input type="radio"/>	
	<input type="radio"/>	DE8	DA8	<input type="radio"/>	
	<input type="radio"/>	DE9	DA9	<input type="radio"/>	
	<input type="radio"/>	DE10	DA10	<input type="radio"/>	
	<input type="radio"/>	DE11	DA11	<input type="radio"/>	
	<input type="radio"/>	DE12	DA12	<input type="radio"/>	
	<input type="radio"/>	DE13	DA13	<input type="radio"/>	
	<input type="radio"/>	DE14	DA14	<input type="radio"/>	
	<input type="radio"/>	DE15	DA15	<input type="radio"/>	
Analoge Eingänge			Analoge Ausgänge		
	<input type="radio"/>	AE0	AA0	<input type="radio"/>	
	<input type="radio"/>	AE1	AA1	<input type="radio"/>	
	<input type="radio"/>	AE2	AA2	<input type="radio"/>	
	<input type="radio"/>	AE3	AA3	<input type="radio"/>	
	<input type="radio"/>	AE4	AA4	<input type="radio"/>	
	<input type="radio"/>	AE5	AA5	<input type="radio"/>	
	<input type="radio"/>	AE6	AA6	<input type="radio"/>	
	<input type="radio"/>	AE7	AA7	<input type="radio"/>	