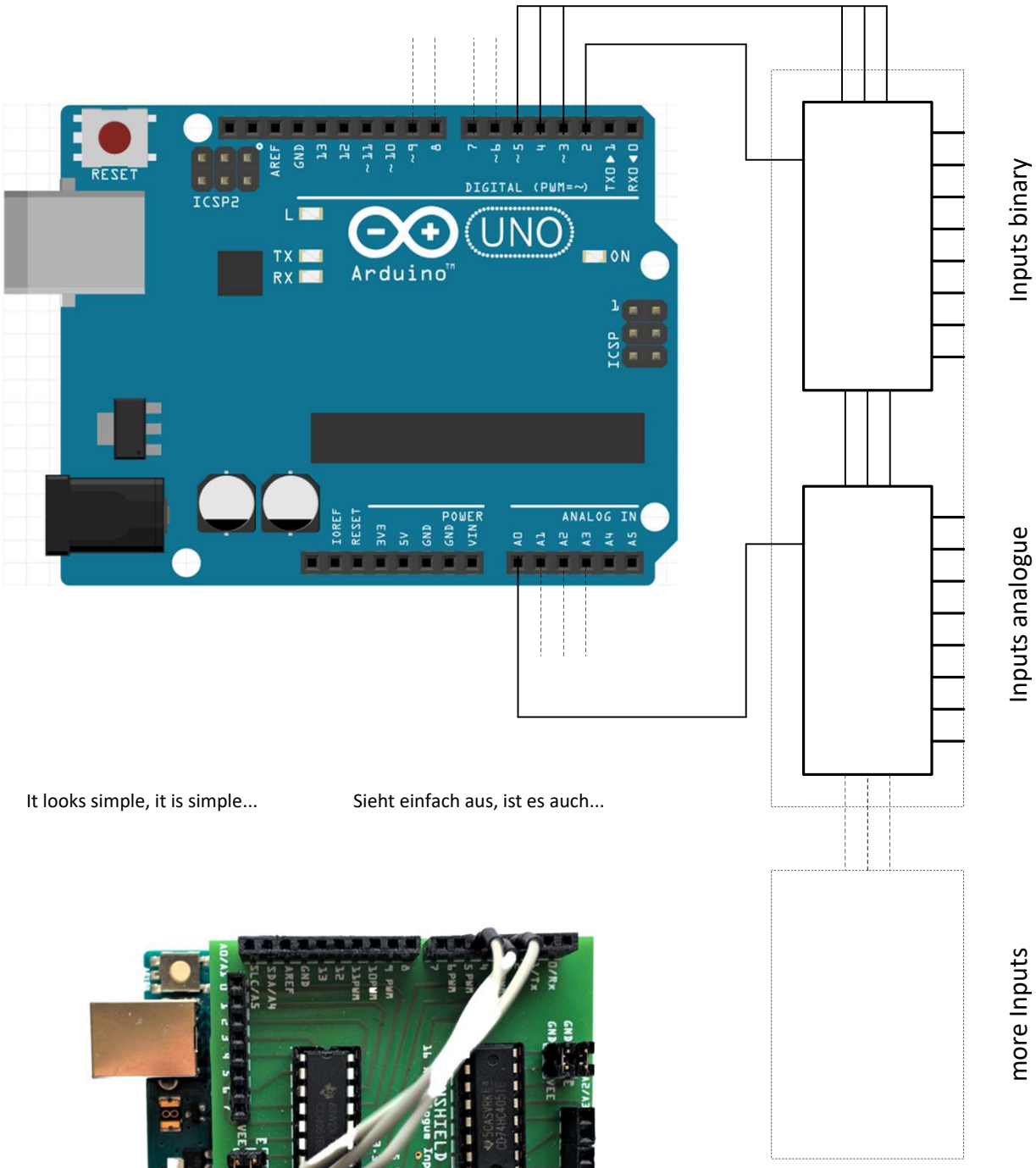
Making an intelligent I/O Interface with

A I N S H I E L D

Intelligentes I/O Interface mit

A I N S H I E L D



It looks simple, it is simple...

Sieht einfach aus, ist es auch...

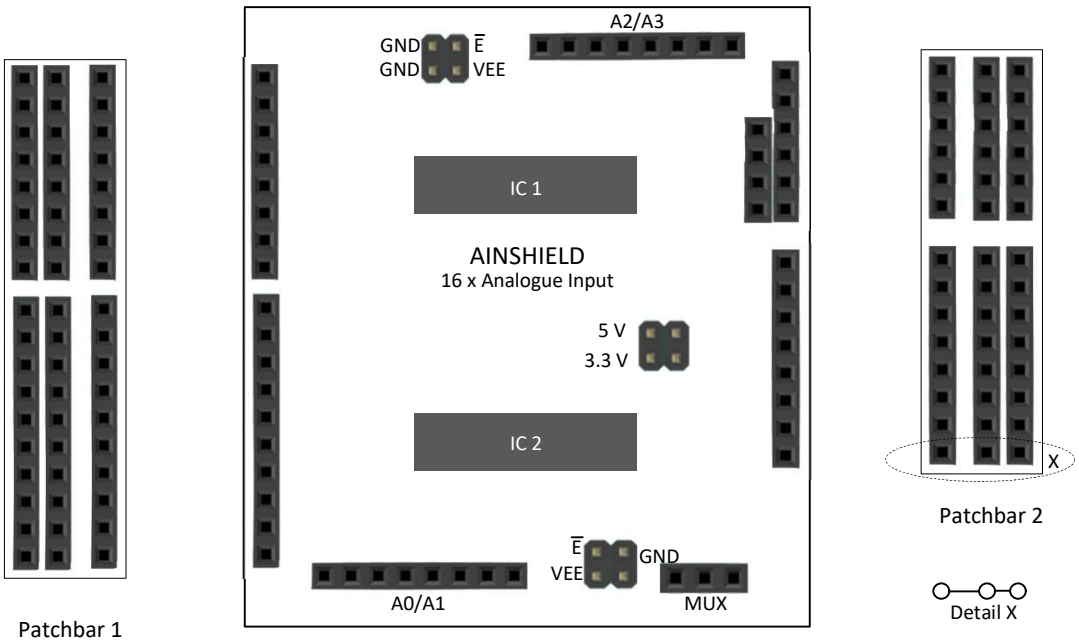


Fig.7 Example
Stacking with Patchbar

Fig.7 Beispiel
Stapeln mit Patchbar

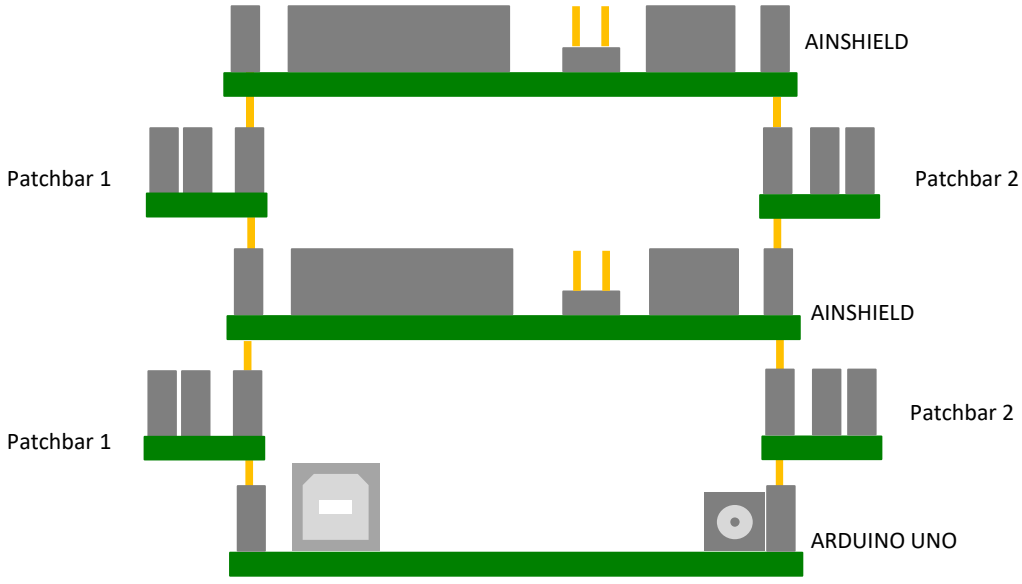
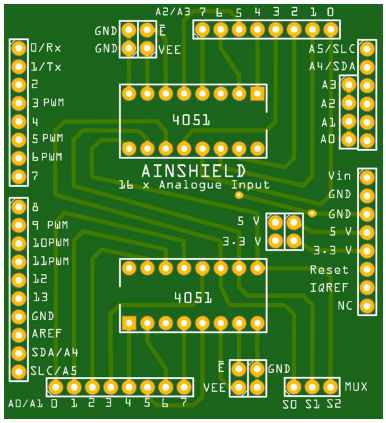
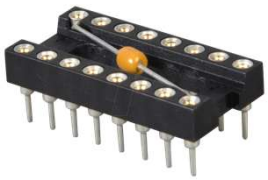
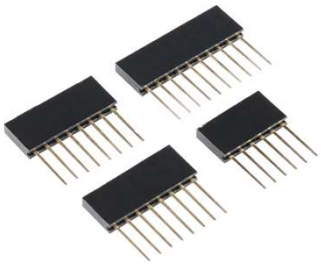


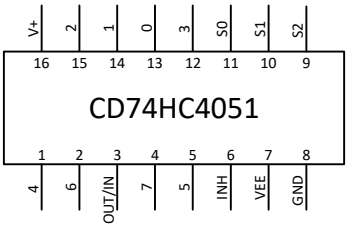
Fig.7

1 x PCB
2 x IC DIP SOCKET 16 LEGS (100pF inside)
6 x HEADER MALE 2 LEGS
2 x STACKING HEADER SET ARDUINO SHIELD
2 x CD74HC4051

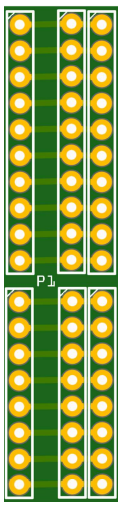


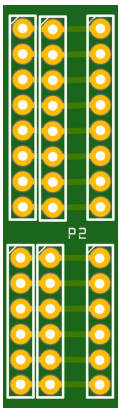


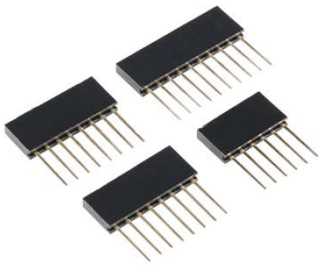




2 x PCB
3 x STACKING HEADER SET ARDUINO SHIELD







IC1 and IC2 (HC4051) are 8 channel bidirectional multiplexers in high speed technology (Typ. t_{Pd} = 15 ns). The analogue operating range is V_{ss} – VEE. For operation as a digital multiplexer also, VEE is connected to ground (GND). High Signal on $\overline{E}$ disable the switches.

IC1 und IC2 (HC4051) sind schnelle, achtkanalige bidirektionale Multiplexer (Typ. t_{Pd} = 15 ns). Der Messbereich ist V_{ss} – VEE. Zur Nutzung von binären Signalen wird VEE auf Masse verdrahtet (GND). Mit High Signal auf $\overline{E}$ kann die Messung verhindert werden.

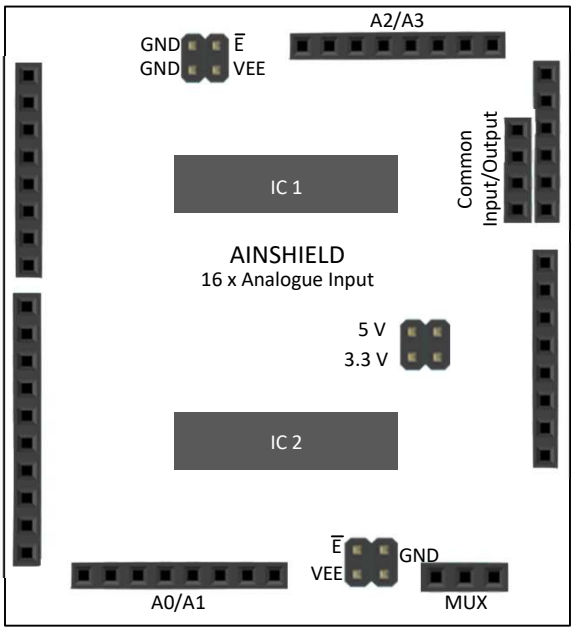


Fig.1

Fig.2 shows simplified how it works.

Die Arbeitsweise ist vereinfacht in Fig.2 dargestellt.

MUX (Selector)			
S2	S1	S0	Input Output
0	0	0	0
0	0	1	1
0	1	0	2
0	1	1	3
1	0	0	4
1	0	1	5
1	1	0	6
1	1	1	7

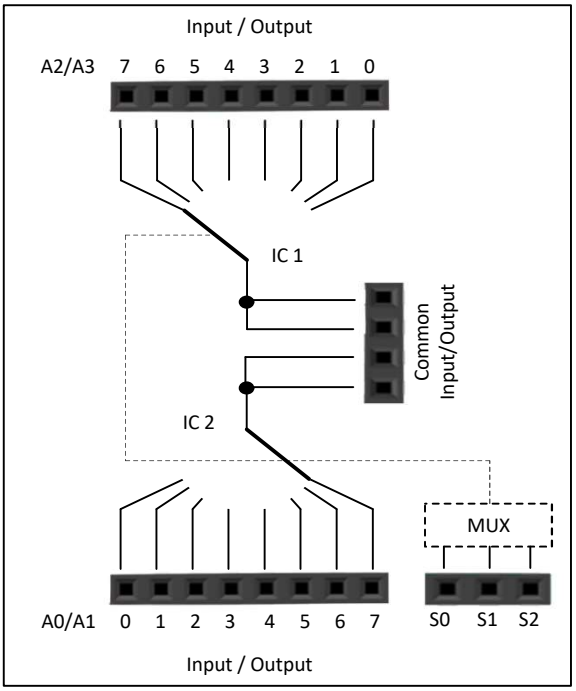


Fig.2

Fig.3 shows the input network.

Fig.3 zeigt die Eingangsbeschaltung.

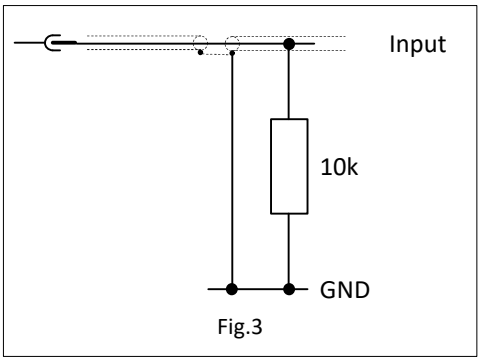


Fig.3

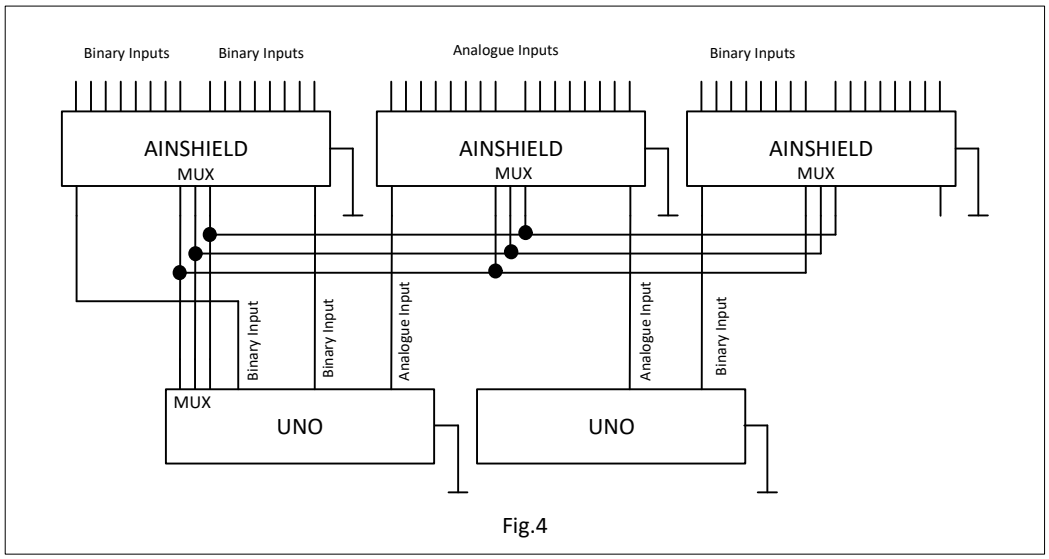


Fig.4

Fig.4 shows an example for interconection between Arduinos and AINSHIELDS.

Fig.4 zeigt ein Beispiel wie Arduinos und AINSHIELDS miteinander verbunden werden können.

Fig.5 Example
Input Program

Fig.5 Beispiel
Eingang Programm

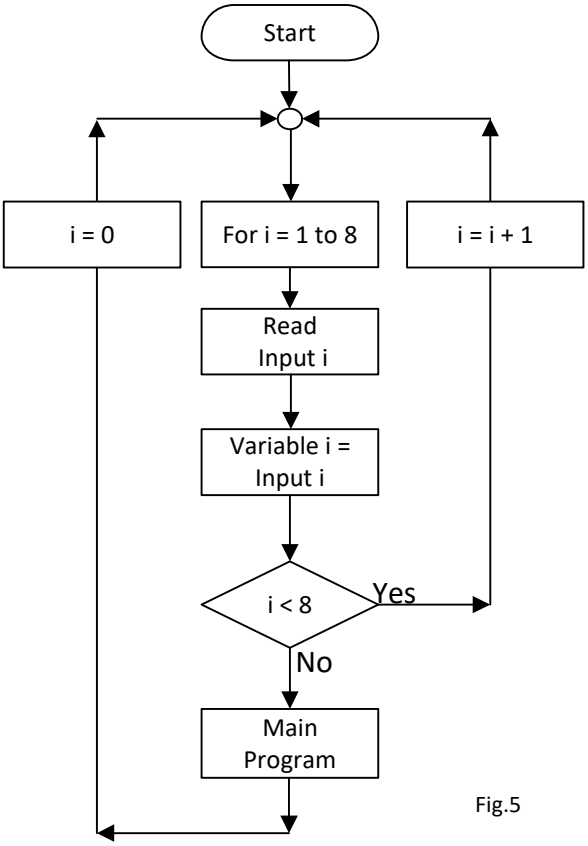


Fig.5

Fig.6 Example
Output Program

Fig.6 Beispiel
Ausgang Programm

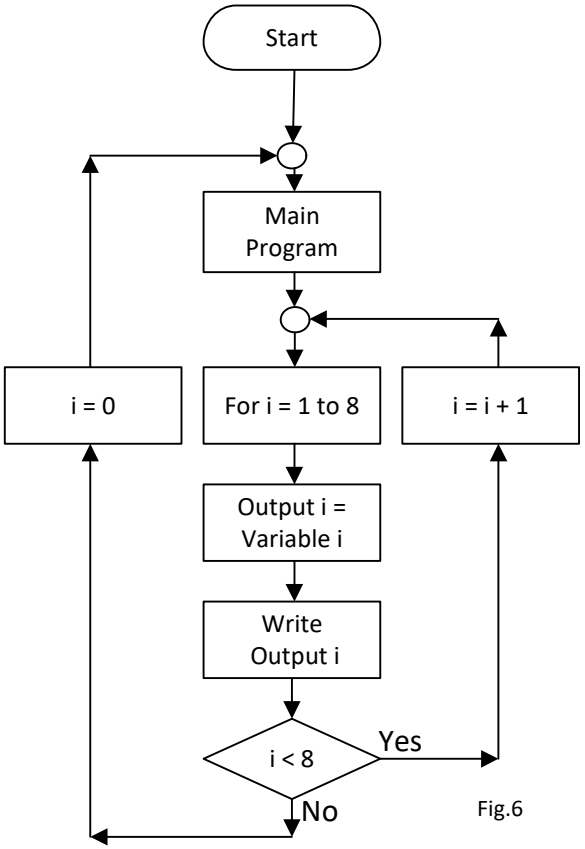
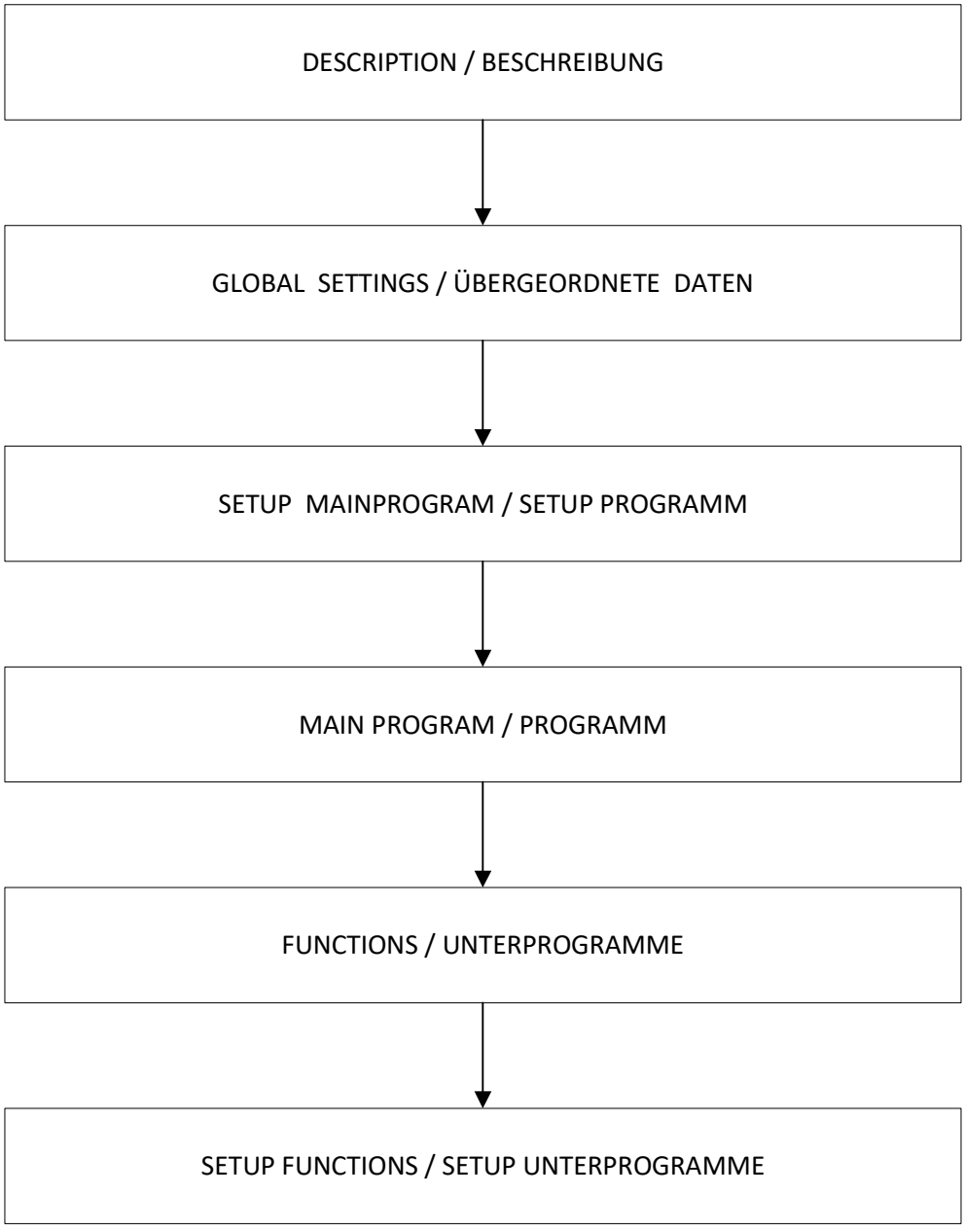


Fig.6

C O D E / P R O G R A M M



DESCRIPTION / BESCHREIBUNG

```
// ================================================
// A I N S H I E L D
// ================================================
// Additional Input PIN's for
// Arduino Uno, Mega, Due...
// AINSHIELD : MULTIPLEX using 74HC4051
// Library       : SPI
// SCADA         : SerialComInstruments  Version 4.1.0.0
//                 © Software from U.Maassen
// http://www.serialcominstruments.com/download.php
// ================================================
```

GLOBAL SETTINGS / ÜBERGEORDNETE DATEN

```
// ================================================
// G L O B A L  S E T T I N G S  A I N S H I E L D
   #include <SPI.h>
   unsigned int t0 = 0;//-----------------------Cicle Time Start
   unsigned int t1 = 0;//-----------------------Cicle Time
   int synClock = digitalRead(13);//----------SCK, Clock SPI
   int bit0 =0;//-----------------------------------Bit for Steering Bus MUX (S0)
   int bit1 =0;//-----------------------------------Bit for Steering Bus MUX (S1)
   int bit2 =0;//-----------------------------------Bit for Steering Bus MUX (S2)
   int sigNo = 0;//---------------------------------Signal Number (0...7)
   int inAn0[8];//----------------------------------input Analogue No. 0, Signal 0 to 7
   int inAn1[8];//----------------------------------input Analogue No. 1, Signal 0 to 7
   int inAn2[8];//----------------------------------input Analogue No. 2, Signal 0 to 7
   int inAn3[8];//----------------------------------input Analogue No. 3, Signal 0 to 7
// ================================================
```

SETUP MAINPROGRAM / SETUP PROGRAMM

```
// ================================================
// S E T U P   M A I N P R O G R A M
   void setup()
   {
   AINsetup();//-----------------------------------Function
   Serial.begin(57600);//-----------------------Scada seriell
   }
// ================================================
```

MAIN PROGRAM / PROGRAMM

```
// ================================================
// M A I N   P R O G R A M
// ================================================
   void loop()
   {
   AIN();//-----------------------------------------Function
   SCADA_SER();//-----------------------------------Function
   }
// ================================================
```

FUNCTIONS / UNTERPROGRAMME

```
// ================================================
// F U N C T I O N  A I N S H I E L D
// ================================================
   void AIN()
   {
   t0 = micros();//-------------------------------------Start Cycle Time
   if(synClock =1)
   sigNo = sigNo+synClock;
   if(sigNo > 7)
   sigNo = 0;
   bit0= bitRead(sigNo, 0);
   digitalWrite(3, bit0);//----------------------------Set Bit Steering Bus MUX S0
   bit1= bitRead(sigNo, 1);
   digitalWrite(4, bit1);//----------------------------Set Bit Steering Bus MUX S1
   bit2= bitRead(sigNo, 2);
   digitalWrite(5, bit2);//----------------------------Set Bit Steering Bus MUX S2
   inAn0[sigNo]=analogRead(0);//----------------Set Signals 0...7 to variables
   inAn1[sigNo]=analogRead(1);//----------------Set Signals 0...7 to variables
   inAn2[sigNo]=analogRead(2);//----------------Set Signals 0...7 to variables
   inAn3[sigNo]=analogRead(3);//----------------Set Signals 0...7 to variables
   t1 = micros()-t0;//---------------------------------Cicle Time
  }
// ================================================
```

SETUP FUNCTIONS / SETUP UNTERPROGRAMME

```
// ================================================
// S E T U P   A I N S H I E L D
// ================================================
   void AINsetup()
   {
   pinMode(3, OUTPUT);//----------------------------Set MUX, PIN3 to Output
   pinMode(4, OUTPUT);//----------------------------Set MUX, PIN4 to Output
   pinMode(5, OUTPUT);//----------------------------Set MUX, PIN5 to Output
   }
// ================================================
```

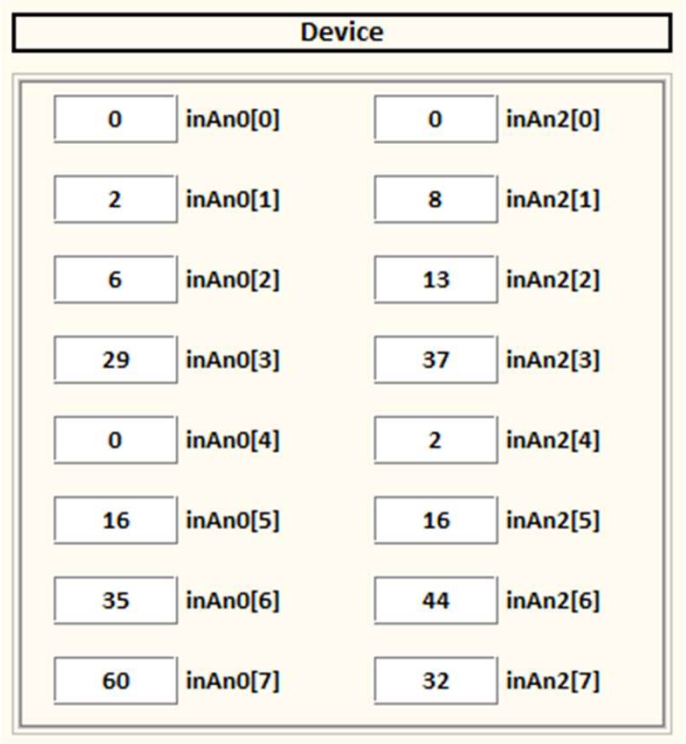
FUNCTIONS / UNTERPROGRAMME

```
// ================================================
// F U N C T I O N   S C A D A
// ================================================
   void SCADA_SER()//--------------------------------only for Test
   {
   Serial.print("#");
   Serial.print(80);
   Serial.print("M");
   Serial.print("26");
   Serial.print(„Device“);
   Serial.println("<");

   for (int i=0; i <= 7; i++)
   {
   Serial.print("#");
   Serial.print(i+1);
   Serial.print("M");
   Serial.print(inAn0[i]);
   Serial.println("<");
   }
   for (int i=0; i <= 7; i++)
   {
   Serial.print("#");
   Serial.print(i+9);
   Serial.print("M");
   Serial.print(inAn2[i]);
   Serial.println("<");
   }
   }
// ================================================
```

S C A D A

Supervisory Control And Data Acquisition



A pretty tool for testing.

Ein gutes Werkzeug zum Testen.