General ................................................................. 2
Electrical Data ................................................................. 2
   General Design ................................................................. 2
   Supply Voltage (SELV) ......................................................... 2
   Intrinsic current consumption (w/o output current) ....................... 2
   Resolution and Accuracy .................................................... 2
Incremental Signals A, B ....................................................... 3
SSI ................................................................. 4
   Electrical Data ................................................................. 4
   Physical interface ............................................................. 4
   Data format ................................................................. 4
   Pinout PCB – Connector 12pin ............................................... 5
SSI - Protocol ........................................................... 6
   Protocol standard SSI .......................................................... 6
   Protocol SSI Extended ........................................................ 8
   High resolution SSI and SSI extended (resolutions > 14 Bit) ......... 9
   Timing SSI .................................................................. 10
BiSS ............................................................ 11
   Electrical Data ................................................................. 11
   Input / Output Signal ....................................................... 11
Outputs: ................................................................. 11
BiSS Protocol ........................................................... 11
   Bidirectional Serial Sensor Interface (BiSS) .............................. 11
   Transmitting sensor data (BiSS-Mode) .................................... 12
   Register – mode (BiSS – Interface) ........................................ 13
   Register mode: read ....................................................... 14
   Register mode: write .................................................... 14
Timing ................................................................. 15
   Timing BiSS Sensor Mode .............................................. 15
   Timing BiSS Register Mode ........................................... 16
Example for read register 78h ............................................ 17
   Hints for PWM Signals .................................................. 18
   CRC - Generation ....................................................... 19
Address MAP ........................................................... 22
   Register Map ............................................................. 23
Recommended Interface .................................................. 25
   SSI Standard with Incremental signals ................................. 25
   BiSS Standard Encoder .................................................. 26
   Electrical behaviour at power up in BiSS Mode ....................... 27
1 General

General information, applications
The "Acuro - Industry" optical absolute encoder is available as a singleturn or multiturn version. The multiturn design is based on a reliable high-speed gear with optical scanning and the latest generation of OptoAsics. The mechanical concept is based on a double ball bearing design, which is available as a solid-shaft or hollow-shaft version in common diameter sizes. The field of application encompasses positioning tasks in all industrial applications.

The electrical concept of the Acuro series is addressing the ever progressing requirements of industrial applications and the state of the art in interface technology. Additional to the widespread absolute encoder interface SSI, the Acuro series features the open and bidirectional high speed sensor interface BiSS. The physical layer of the BiSS interface is backward compatible to SSI. To match with the still often used sine wave analog inputs in motion control applications the Acuro is available also with sine wave output combined with SSI.

Electrical Data

General Design
Protection Degree III
Pollution Degree 2
Over voltage Category II

according DIN EN 61010 part 1 (03.94)/ EN 61010-1/A2 (05.96) (VDE 0411)

Supply Voltage (SELV)
DC 5 V –5%/ +10%
DC 7…30 V

Intrinsic current consumption (w/o output current)

Singleturn: at DC 5 V ≤ 45 mA
Multiturn: at DC 5 V ≤ 85 mA

Resolution and Accuracy

Incremental Signals (A, B)
2048 Periods / Revolution
Incremental Signals A, B

Track A leads B by 90° at rotation and view on shaft end.

\[
A = U_M + \Delta A \sin(z_{\text{mech.}})
\]

\[
A^* = U_M - \Delta A \sin(z_{\text{mech.}})
\]

\[
B = U_M - \Delta B \cos(z_{\text{mech.}})
\]

\[
B^* = U_M + \Delta B \cos(z_{\text{mech.}})
\]

\[z: \text{No. Of signal periods (2048)}\]

Amplitudes:

- \(\Delta (A^*; B^*) = 0,5V - 25\% \) / + 20\% (f \leq 1kHz)
- \(\Delta (A^*; B^*) = 0,35V ... 0,6V\) (f > 1kHz)

- Limiting frequency \(f_g = 500\ kHz\)
- Amplitudes difference \(^1\)
- Degree of modulation (mech.) \(^4\)
- Offset \(U_{\text{off}} (A^*; B^*) \leq 0,1 \Delta (A^*; B^*)\)
- Phase A to B \(^6\)
- Harmonic distortion \(^3\)
- DC Offset \(^5\)

\[U_0: \text{Basic Signal, } U_1 ... U_n : \text{harmonics}\]

\[k = \frac{\sqrt{U_1^2 + U_2^2 + \ldots + U_n^2}}{\sqrt{U_0^2 + U_1^2 + \ldots + U_n^2}}\]

\[m = \frac{\Delta u}{u}\]

\[^1\): measured with 120 Ω termination resistor at encoder output

\(^2\): at f = 1 kHz (corresponds to 30 U/min)

\[^3\): \(k < 2\% \) (typ. 1\%)

\[^4\): \(U_M = 2,5V \pm 20\%\)

\[^5\): \(U_M\) same for A and A* and for B and B* signals.

\[^6\): Average\]
SSI

Electrical Data

Physical interface

Number of wires 4 unidirectional
  Data and /Data: RS485-Level
  Clock and /Clock: RS485-Level
  Driver Output current: max.60 mA
  Short circuit output current: max.250 mA

Transmission speed 70 kHz – 1 MHz according to SSI definition

Data format:

The data alignment is MSB left flush. That means with the first rising edge the MSB bit is on the output. The output is programmable for binary or gray code.

Resolution standard SSI (25 data bits)
  Data format MSB - left- flush
  Singleturn: 13 Bit
  Multiturn: 12 Bit
  Binary Code

Optional: Resolution SSI with extendable data length (> 25 data bits)
  Singleturn: max 19 Bit
  Multiturn: 12 Bit
  *Singleturn in 25-Bit MT-Model: programmable up to 19 Bit

Monoflop – timeout 10 < tm < 30µs
Cable

For clock and data should be twisted in pairs and shielded

**Baud rate / Cable length**

<table>
<thead>
<tr>
<th>Cable length</th>
<th>Baud rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt; 25 m</td>
<td>&lt; 1 MHz</td>
</tr>
<tr>
<td>&lt; 50 m</td>
<td>&lt; 400 kHz</td>
</tr>
<tr>
<td>&lt; 100 m</td>
<td>&lt; 300 kHz</td>
</tr>
<tr>
<td>&lt; 200 m</td>
<td>&lt; 200 kHz</td>
</tr>
<tr>
<td>&lt; 400 m</td>
<td>&lt; 100 kHz</td>
</tr>
</tbody>
</table>

**Pinout PCB – Connector 12pin**

<table>
<thead>
<tr>
<th>PIN</th>
<th>5 or 7-30 V (UB) gr/pk</th>
<th>Clock wt</th>
<th>B-rd</th>
<th>0V (UN) wt/gn</th>
<th>A-ye</th>
<th>Data bk</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Data vio</td>
<td>A+ gn</td>
<td>0V Sens bn/gn</td>
<td>B+ bl</td>
<td>Clock/ bn</td>
<td>5 V Sens rd/bl</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Connection on encoder side over 12-pin PCB connector, Manufacturer Berg, Type: Minitek.
SSI - Protocol

Protocol standard SSI
The SSI data transmission of an absolute encoder position value is based on a shift register, where the shift clock is provided by the external control. The encoder provides its position data and depending on configuration also status information synchronous to the external clock on the data line. Both lines are physically according to RS422 specification.

Clock pulse diagram

Transmission Cycle

For correct transfer of the data a defined number of impulses (clock pulse brush) must be applied to the clock input of the absolute shaft encoder. Next, a pause TP must be observed. As soon as a clock pulse brush is applied to the clock pulse input, the actual angle information will be latched.

With the first shift of the clock signal from low to high ② the most significant bit (MSB) of the angular data is applied to the shaft encoder’s serial output. With each succeeding rising edge, the next less significant bit is shifted to the data output. After transmission of the least significant bit (LSB) the Alarm bit or other special bits are transferred, depending on configuration. Then the data line switches to low ③ until the time tm has passed. A further transfer of data cannot be started until the data line switches to high ④ again. If the clock pulse sequence is not interrupted at point ③, the ring-register mode is activated automatically. This means that the data stored at the first clock pulse transition ② are returned to the serial input Si via the terminal SO. As long as the clock pulse is not interrupted at ④, the data can be read out as often as wanted (multiple transfer). The number of clock pulses necessary for data transfer is independent of the resolution of the absolute shaft encoder. The clock signal can be interrupted at any point, or continued in ring-register mode for repeated polling.
With a data length of 25 Bit (simple transmission) the transmission after the lowest Bit (LSB), the data line holds on low, till the time \( t_m \) is elapsed.

Clock frequency: \( 100 \text{ kHz} \ldots 1,5 \text{ MHz} \)
Monoflop time \( t_m \): \( 12 \mu s = t_m = 20 \mu s \)
Clock pulse brush: 25 Clock cycles for Multiturn
13 Clock cycles for Singeltum
Protocol SSI Extended

SSI interface: extended SSI formats

In extended SSI mode singleturn data of up to 21 bits and an additional 12 to 24 bits of multiturn data can be transmitted. This is followed by the 8-bit temperature value.

Expansions compared to SSI standard format:

After the position data there are two additional bits that indicate the status of the encoder. First the alarm bit and then the warning bit.

The reports from the LED current control or the temperature control can be assigned freely between the two bits (alarm, warning).

In the standard version the alarm bit is assigned to the LED current control and the warning bit is assigned to the internal temperature sensor.

Function: If one of the fault conditions occurs, the alarm bit is set “high” (.1’bit in the protocol). After reading a new position value the alarm bit is cleared automatically again. If the cause for the alarm is not longer present, the alarm bit is set to ‘0’ (in the protocol is like ‘0’). Should the cause still be present, the alarm bit is set once more (‘1’).

6 Bit CRC Checksum
To guarantee a safe data transmission, there is a 6 bit CRC checksum over data and status bits. The start value of the checksum is 43h (1000011b). The checksum is formed and transferred in inverted format. First of the 6 Bit CRC transmitted is CRC5 CRC0.

Temperature
A temperature sensor with a resolution of 1 °C (LSB) within a range of -64 °C to +191 °C has been included on the internal OptoAsic chip for the monitoring of the operating temperature. The current temperature is stored as an 8-bit value. The sensor is calibrated in such a way that the value "0100 0000" is produced at 0 °C. The sensor sets error bit when either the upper or lower alarm thresholds have been exceeded.

Example temperature value = 59h (59h –40h = 19h) temperature 25° C

4 Bit CRC Checksum
To guarantee a safe temperature transmission, there is a 4 bit CRC checksum
The start value of the checksum is 13h (10011b). The checksum is formed and transferred inverted format. First of the 4 Bit CRC transmitted is CRC3 CRC0.
High resolution SSI and SSI extended (resolutions > 14 Bit)

For Singleturn resolutions > 14 Bit either the Clock frequency must not exceed 100 kHz, or with higher clock frequencies the first negative Clock pulse needs to be on low level for minimum $t_{\text{SAR}}$ time. This is due to the time needed for internal A/D conversion.

Example how to calculate $t_{\text{SAR}}$: The resolution of the encoder shall be 1217, so the single turn part is 17 bit. The 17 bit consists internally of 11 bit digital information and 6 bit interpolated information. So we need time for 6 bits interpolation. If $n$ is the number of interpolation bits we need (worst case) $600 \times (n+1)$ ns calculation time. Now you can calculate the delay: it is $(6\text{bit} + 1) \times 600\text{ns}= 4.2 \mu\text{s}$. After this time the output value is ready to be transmitted.
Timing SSI

![Timing SSI Diagram]

*needed only for Singleturn resolutions > 14 Bit

<table>
<thead>
<tr>
<th>SSI Mode</th>
<th>Parameter</th>
<th>Conditions</th>
<th>min</th>
<th>typ.</th>
<th>max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>timeout sens (Ttos)*</td>
<td></td>
<td></td>
<td>9.9</td>
<td>12.4</td>
<td>14.9</td>
<td>µs</td>
</tr>
<tr>
<td>TMAS</td>
<td>Permissible Clock Period</td>
<td></td>
<td>250 ns</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>tMASH</td>
<td>Clock Signal Hi Level Duration</td>
<td></td>
<td>125</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>tMASI</td>
<td>Clock Signal Lo Level Duration</td>
<td></td>
<td>125</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>treq</td>
<td>Data Request Lo Level Duration</td>
<td>only with SAR converter</td>
<td>tSAR</td>
<td></td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>fclk</td>
<td>Clock Frequency</td>
<td></td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>MHz</td>
</tr>
<tr>
<td>tSAR</td>
<td>Conversion Time SAR Converter</td>
<td>n = resolution of SAR converter</td>
<td>2(n+1)fclk</td>
<td></td>
<td>µS</td>
<td></td>
</tr>
</tbody>
</table>

* Ttos = is programmable Time
BiSS

Electrical Data

Input / Output Signal
- Clock and /Clock: RS485 (Input)
- Data and /Data: RS485 (Output)
- Clock frequency: 100 kHz ...10 MHz (13 Bit or more: clock frequency max. 9.1 MHz)
- Timeout_{SENS}: 12µs
- Timeout_{reg}: 51µs
- * Timeout_{SENS} and timeout_{reg} are programmable

Outputs:
- Driver output current: max.60 mA
- Short circuit output current: ± 250 mA

Cable
- Leads for clock and data should be twisted in pairs
- Entire cable shielded and according to CAT 5
- Cable capacity ≤ 100 pF/m
- Cable length max 100 m
- Baud rate < 10 MHz

BiSS Protocol

Bidirectional Serial Sensor Interface (BiSS)

The Serial BiSS communication differentiates between the fast transmission of sensor data and the slower transmission of register data. The transmission of sensor data is unidirectional; here, ACURO can only output data, whereas the bidirectional transmission of register data can include read and write access.

The BiSS sensor interface can be operated in an SSI compatible mode, in which only a lower transmission speed is possible and ACURO may not demand processing time for procedures such as interpolation, for example.
Transmission is initiated by a falling edge on the master line (MA). The master then again ramps the master line up to high within a stipulated period (<timeout_SENS>) and continues the clock pulse. ACURO acknowledges the request for sensor data on the second rising MA edge with a low signal at SLO (see description of the BISS protocol). The next rising edge gives the validity of the position data and is interpreted as a start bit by the master.

Depending on the configuration the length of ACURO’s position data varies between 9 and 45 bits, plus an error bit and a warning bit. With a maximum length of 47 bits this data is protected by a 6-bit cyclic redundancy check value or CRC (polynomial 0x3 = “1000011b”) which directly follows the data. **MCD: Multicycle data is not supported!**

Transmitting sensor data in BISS mode.

**The Warning – Bit (War)** is coupled to the internal temperature sensor of the OptoAsic. It is high, when the following temperature limits are exceeded or under - run:

<table>
<thead>
<tr>
<th>Series</th>
<th>Operating temperature</th>
<th>Internal Warning tresholds</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACURO Industry (AC)</td>
<td>-40° ... +100°C</td>
<td>-45° .. +105°C</td>
</tr>
<tr>
<td>ACURO Drive (AD)</td>
<td>-15° ... +120°C</td>
<td>-20° .. +125°C</td>
</tr>
</tbody>
</table>

**The Error – Bit (Err)** is coupled to the LED – current. It is high, when an factory defined threshold is exceeded. An excess LED current can indicate Pollution; Condensation, Over temperature or Ageing of the LED

**Chart: Alignment Bits**

<table>
<thead>
<tr>
<th>MT</th>
<th>ST</th>
<th>Alignment Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>9</td>
<td>0</td>
</tr>
<tr>
<td>12</td>
<td>10</td>
<td>0</td>
</tr>
<tr>
<td>16</td>
<td>11</td>
<td>0</td>
</tr>
<tr>
<td>20</td>
<td>12</td>
<td>0</td>
</tr>
<tr>
<td>24</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>14</td>
<td>14</td>
<td>0</td>
</tr>
<tr>
<td>15</td>
<td>15</td>
<td>2</td>
</tr>
<tr>
<td>16</td>
<td>16</td>
<td>1</td>
</tr>
<tr>
<td>17</td>
<td>17</td>
<td>0</td>
</tr>
<tr>
<td>18</td>
<td>18</td>
<td>6</td>
</tr>
<tr>
<td>19</td>
<td>19</td>
<td>5</td>
</tr>
<tr>
<td>20</td>
<td>20</td>
<td>4</td>
</tr>
<tr>
<td>21</td>
<td>21</td>
<td>3</td>
</tr>
<tr>
<td>22</td>
<td>22</td>
<td>2</td>
</tr>
<tr>
<td>23</td>
<td>23</td>
<td>1</td>
</tr>
<tr>
<td>24</td>
<td>24</td>
<td>0</td>
</tr>
</tbody>
</table>

Values in columns: “Length of Data bits”
Register – mode (BiSS – Interface)

The register communication is initiated by a low signal following the first falling edge from the master on the clock line. The master keeps the clock line on low until the ACURO reacts with a falling edge on the data line and thus signaled the change over to register mode. After this has happened the master transmits the addressing data coded as a PWM signal (pulse width modulated clock signal). The individual sensors (slaves) are addressed by slave IDs which are generated automatically according to the order of the slaves in the sequential circuit. ACURO uses two slave IDs (e.g. ID "000" and "001") so that it can extend the available addressing range from 7 to 8 bits.
Register mode: read

Once ACURO has signaled the changeover to register mode the master transmits the start bit, the 3-bit slave ID and the 7-bit register address for the addressing sequence, followed by the WNR bit ("0") and the 6-bit CRC. Each bit is coded by the duty cycle (PWM), including the start bit. The generator polynomial for the 4-bit CRC is \( 0x13 = 10011 \) (see the definitions in the description of the BiSS protocol). The ACURO does not require any processing time to read the internal registers and answers immediately with the data of the addressed registers. When reading the external EEPROM registers the output is delayed until the data from the EEPROM has been made available. All 8-bit read data can also be checked for transmission errors by the 4-bit CRC 0x13.

Register mode: write

When data is being written to a register, after the ACURO has confirmed the mode changeover the same addressing sequence as for read access is used (with the WNR bit at "1"). Following the second start bit the master transmits the data to be written which ACURO returns for verification, bit by bit one clock pulse later. As in the above, a 4-bit CRC have to follow the 8-bit write data which is returned by ACURO in the same manner, however not in PWM format. A transfer to the EEPROM registers is processed in the background and can be validated by a read access once transmission is over.
Timing

Timing BiSS Sensor Mode

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Conditions</th>
<th>min</th>
<th>typ</th>
<th>max.</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>timeout sens</td>
<td>(Ttos)*</td>
<td></td>
<td>9,9</td>
<td>12,4</td>
<td>14,9</td>
<td>µs</td>
</tr>
<tr>
<td>TMAS</td>
<td>Permissible Clock Period</td>
<td></td>
<td>100</td>
<td>2* Ttos</td>
<td></td>
<td></td>
</tr>
<tr>
<td>tMASH</td>
<td>Clock Signal Hi Level Duration</td>
<td></td>
<td>50</td>
<td>Ttos</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>tMASI</td>
<td>Clock Signal Lo Level Duration</td>
<td></td>
<td>50</td>
<td>Ttos</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>Treq</td>
<td>Data Request Lo Level Duration</td>
<td>only with SAR converter</td>
<td>50</td>
<td>Ttos</td>
<td></td>
<td>ns</td>
</tr>
</tbody>
</table>

* Ttos = is programmable Time
Timing BiSS Register Mode

### BiSS Register Mode

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Parameter</th>
<th>Conditions</th>
<th>Min</th>
<th>Max</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMAR</td>
<td>Permissible Clock Period</td>
<td>CFGTOR = ZEh</td>
<td>4</td>
<td>52</td>
<td>(\mu s)</td>
</tr>
<tr>
<td>tidle</td>
<td>Permissible Clock Halt (idle)</td>
<td></td>
<td>0</td>
<td>Indefinite</td>
<td></td>
</tr>
<tr>
<td>tMARh</td>
<td>Clock Signal Hi Level Duration</td>
<td>read out of register data</td>
<td>50 %</td>
<td></td>
<td>% TMAR</td>
</tr>
<tr>
<td>tMARl</td>
<td>Clock Signal Lo Level Duration</td>
<td></td>
<td>Tlor</td>
<td></td>
<td>ns</td>
</tr>
<tr>
<td>tMA0h</td>
<td>Logic 0* Hi Level Duration</td>
<td></td>
<td>10</td>
<td>30</td>
<td>% TMAR</td>
</tr>
<tr>
<td>tMA1h</td>
<td>Logic 1* Hi Level Duration</td>
<td></td>
<td>70</td>
<td>90</td>
<td>% TMAR</td>
</tr>
</tbody>
</table>
Example for read register 78h

Keep clock active until start bit is sent by encoder. Approx. time ~ 416 µs

See Figure: The clock should be applied until encoder sends ACK (~ 416 µs). This time is needed because the ASIC has to read the EEPROM internally before sending the data. There are different times for different registers because registers are mapped either directly in the ASIC or externally to an EEPROM value (takes more time).
Hints for PWM Signals

Code sample for PWM with port bit

```plaintext
trans_0:
  set portx ; 1 time high 3 time low
  clear portx ; Output = high
  clear portx ; Output low
  clear portx ; Output low
  ret ; end send '0'

trans_1:
  set portx ; 3 time high 1 time low
  set portx ; Output = high
  set portx ; Output = high
  set portx ; Output = high
  clear portx ; Output low
  ret ; end send '1'

x = Output Pin MA
```

Code sample for PWM with SPI

```plaintext
trans_0: ; 1 time high 3 time low
  transmit '0'

trans_1: ; 3 time high 1 time low
  transmit '1'
```

```
1 0 0 0 1 1 1 0
```

Clock Clock

TMA TMA

transmit '0' transmit '1'

set portx ; Output = high
set portx ; Output = high
set portx ; Output = high
set portx ; Output = high
set portx ; Output = high
set portx ; Output low
set portx ; Output low
set portx ; Output low
set portx ; Output low
set portx ; Output low
```
CRC - Generation

Depending on the configuration the length of the position data varies between 9 and 45 bits, plus an error bit and a warning bit. With a maximum length of 47 bits this data is protected by a 6-bit cyclic redundancy check value or CRC (polynomial 0x43 = "1000011") which directly follows the data.

A CRC "checksum" is the remainder of a binary division with no bit carry (XOR used instead of subtraction), of the message bit stream, by a predefined (short) bit stream of length \( n \), which represent the coefficients of a polynomial. Before the division, \( n \) zeros are appended to the message stream.

Example:
The Bit stream 1000011 is equivalent to the Polynom \( 1x^6 + 0x^5 + 0x^4 + 0x^3 + 0x^2 + 1x^1 + 1x^0 \)

\[ = x^6 + x^1 + 1 \]

Hardware

Process:
1. Shift register set to 0
2. Shift Data
3. Shift CRC
/**** function for calculating a new CRC *****/

text byte bitString[tmpx]; // contains the data, one data bit per byte

int calcCRCnew (byte bitPolynom) // parameter = 4 or 6 bit CRC
{
    // CRC calculating for 4 bit polynom 1 0011 and 6 bit polynom 1 100 0011
    // XOR - function only by MSB = high of the working bytes!

    // Variable byte crcByte, polynom; // resulting crc byte
    int tmpx, msb, zB, tmpy; // temporary variables

    if(bitPolynom==4) // calculate 4 Bit CRC or 6 Bit CRC
        zB=4;
    polynom = 0x13;
    else if(bitPolynom == 6) // calculate 4 Bit CRC or 6 Bit CRC
        zB=6;
    polynom = 0x63;

    crcByte = 0; // start value

    // BitString mit 4 oder 6 Nullen füllen
    // clear BitString for 4 or 6 bits
    for (tmpx = bitZ; bitZ < tmpx +bitPolynom;bitZ++) {
        bitString[tmpx] = 0;
    }

    // first fill up crc byte up to polynom length
    for (tmpx=0,tmpy = bitPolynom;tmpx <= bitPolynom; ++ tmpx, --tmpy) {
        crcByte = crcByte + (bitString[tmpx] << tmpy);
    }

    // do the shift and xor operations
    for(;;) {
        // EXOR if MSB high
        if ((crcByte >> 7) & 1) \{ // check if MSB is 1
            crcByte = (crcByte ^ polynom);
        \}

        // shift
        ++ zB;
        if (zB == bitZ) break; // finished
        else {
            crcByte = (crcByte << 1); // else continue shifting data
            if (bitPolynom == 4) // limit crc value to polynom length by masking
                crcByte = crcByte & 0x1F;
            else
                crcByte = crcByte&0x7F; // 7 Bit Maske!
            crcByte = crcByte+(bitString[zB]); // add next bit
        }
    }

    if (bitPolynom == 4) // transmit inverted crc value
        return (~crcByte)&0x0F;
    else
        return (~crcByte)&0x3F;
}
// end function calcCRCnew
Sample: Set Preset
### Address MAP

<table>
<thead>
<tr>
<th>Address</th>
<th>Access Protection</th>
<th>Bank 0</th>
<th>Bank 1</th>
<th>Bank 2 … 7</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x00</td>
<td>security</td>
<td>Configuration Data</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x5F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x60</td>
<td></td>
<td>Command (wr) &amp; Status (rd) Register</td>
<td>OEM</td>
<td>OEM</td>
</tr>
<tr>
<td>0x61</td>
<td></td>
<td>Position and Status Data</td>
<td>128 Byte</td>
<td></td>
</tr>
<tr>
<td>0x69</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x6A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x77</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x78</td>
<td></td>
<td>BiSS – Device ID</td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x7F</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x80</td>
<td></td>
<td></td>
<td>OEM</td>
<td>128 Byte</td>
</tr>
<tr>
<td>0xFE</td>
<td></td>
<td>Bank select</td>
<td>Bank select</td>
<td></td>
</tr>
<tr>
<td>0xFF</td>
<td></td>
<td></td>
<td>Bank select</td>
<td></td>
</tr>
</tbody>
</table>

Bank 0: 128 Byte - OEM useable Memory  
Bank 1: 256 Byte - OEM useable Memory  
Bank 2…7: optional
Register Map

0x60  Command register

Preset function

When the command "Implement PRESET" (data0x02) is written to the command register (address 0x60), the current position is written to the external configuration EEPROM as an OFFSET value. At the same time the relevant values are written to the six OFFSET registers, one after another.

Sequence activity is signaled for a few microseconds after the start of the sequence with a "1" in PRES in the status register (address 0x60, bit 7). The bit switches back to "0" while the sequence is still running.

The entire preset sequence ends after the sixth BUSY "1>0" change (address 0x60, bit 2: serial communication active).

0x67  Temperature Data Register (read only)

Bit 7...0 absolute temperature as 8 bit data

0x68  Error register

Bit 7 = Temperature out of defined range  default temperature range (see chapter 4.3)
Bit 6 = External failure over NERR  not necessary in BiSS mode
Bit 5 = Serial interface failure  not necessary in BiSS mode
Bit 4 = Position data not valid  not necessary in BiSS mode
Bit 3 = Failure configuration interface  not necessary in BiSS mode
Bit 2 = Position Code Error  Controls the binary code single step by step
Bit 1 = External Multiturn Error  Controls the communication between the gear PCB and singleturn PCB
Bit 0 = LED current out of control range  Pollution; Condensation, Over temperature, Ageing of LED

0x78..0x7F  BiSS Device ID

0x78  0x41  A  Product ID  e.g. AC or AD
0x79  0x43  C
0x7A  0x3A  58 for ACURO
0x7B  ........................................................................................................... resolution
0x7C  ........................................................................................................... timeout
0x7D  ........................................................................................................... free
0x7E  0x48  H  manufacturer code
0x7F  0x45  E  Hengstler
Encoder Characteristics (Produce datas, Resolutions)

<table>
<thead>
<tr>
<th>Reg. Adr.</th>
<th>Description</th>
<th>Length</th>
<th>Format</th>
<th>1.</th>
<th>2.</th>
<th>3.</th>
<th>4.</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x34</td>
<td>Serial No.</td>
<td>4 Byte</td>
<td>Ser. – No. BCD format</td>
<td>SS</td>
<td>SS</td>
<td>SS</td>
<td>LL</td>
</tr>
<tr>
<td>0x38</td>
<td>Production- Date</td>
<td>4 Byte</td>
<td>Date BCD format</td>
<td>DD</td>
<td>MM</td>
<td>JJ</td>
<td>JJ</td>
</tr>
<tr>
<td>0x3C</td>
<td>Article No.</td>
<td>4 Byte</td>
<td>Article No. BCD format</td>
<td>XX</td>
<td>XX</td>
<td>XX</td>
<td>0</td>
</tr>
<tr>
<td>0x40</td>
<td>MT- Resolution</td>
<td>1 Byte</td>
<td>MT BCD format (0 / 12-Bit)</td>
<td>12</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x41</td>
<td>ST-Resolution</td>
<td>1 Byte</td>
<td>ST BCD format (9...22-Bit)</td>
<td>19</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x42</td>
<td>Alignment Bits</td>
<td>1 Byte</td>
<td>BCD format (0...11-Bit)</td>
<td>5</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0x43</td>
<td>SinCos-Periods</td>
<td>2 Byte</td>
<td>SinCos BCD format</td>
<td>20</td>
<td>48</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Example Data Output BiSS Protocol

Resolution AD36 1219

<table>
<thead>
<tr>
<th>Reg.</th>
<th>Description</th>
<th>Length</th>
<th>Format</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x40</td>
<td>MT-Resolution</td>
<td>1 Byte</td>
<td>MT BCD</td>
<td>12</td>
</tr>
<tr>
<td>0x41</td>
<td>ST-Resolution</td>
<td>1 Byte</td>
<td>ST BCD</td>
<td>19</td>
</tr>
<tr>
<td>0x42</td>
<td>Alignment Bits</td>
<td>1 Byte</td>
<td>BCD</td>
<td>5</td>
</tr>
</tbody>
</table>

Resolution AD36 0019

<table>
<thead>
<tr>
<th>Reg.</th>
<th>Description</th>
<th>Length</th>
<th>Format</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x40</td>
<td>MT-Resolution</td>
<td>1 Byte</td>
<td>MT BCD</td>
<td>0</td>
</tr>
<tr>
<td>0x41</td>
<td>ST-Resolution</td>
<td>1 Byte</td>
<td>ST BCD</td>
<td>19</td>
</tr>
<tr>
<td>0x42</td>
<td>Alignment Bits</td>
<td>1 Byte</td>
<td>BCD</td>
<td>5</td>
</tr>
</tbody>
</table>

Resolution AD36 1214

<table>
<thead>
<tr>
<th>Reg.</th>
<th>Description</th>
<th>Length</th>
<th>Format</th>
<th>Bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>0x40</td>
<td>MT-Resolution</td>
<td>1 Byte</td>
<td>MT BCD</td>
<td>12</td>
</tr>
<tr>
<td>0x41</td>
<td>ST-Resolution</td>
<td>1 Byte</td>
<td>ST BCD</td>
<td>14</td>
</tr>
<tr>
<td>0x42</td>
<td>Alignment Bits</td>
<td>1 Byte</td>
<td>BCD</td>
<td>0</td>
</tr>
</tbody>
</table>

©Hengstler GmbH Postfach 1151 78550 Aldingen
☎ 07424/89-0 Fax 07424/89-500
e-mail: info@hengstler.com Internet: www.hengstler.com
Recommended Interface

SSI Standard with Incremental signals

Dimensioning:
R1 = 91 Ω, R2 = 100 Ω, R3 = 10 Ω, R4 = 10kΩ, R5 = R4* desired adjustment, Z0 = 120 Ω
C1 = 1nF
U1 = 2.5 V ± 0.5V (referred to operating voltage)

*1) Alternative population for high transmission rates (> 2MHz) and simultaneous operation of several encoders (i.e. common clock, separate data lines).
BiSS Standard Encoder
Supply 10...30 Volt
Electrical behaviour at power up in BiSS Mode

After 500µs the Data driver output (Data +) switches to 4.5 V level.

After 7ms it is possible to read out data.
Cable length and clock frequency

**SSI interface**
For SSI the maximum data transmission rate depends on the length of the cable. The clock frequency is variable between 100 kHz and 1.5 MHz. That means a long cable and a high clock frequency that can disturb the data signal due to propagation delay of the signals over copper wires. So it is necessary to reduce the clock frequency or the cable length.

**BiSS interface**
Due to the built-in propagation delay compensation of the BiSS interface (ACURO and BiSS - Master) the clock frequency can be up to 10 MHz and simultaneously the cable length up to a maximum of 100 m. The maximum clock frequency is mainly determined by the cable and connecting elements that are used. For 10 MHz the cable should be compliant with CAT 5.

---

**Recommended cable length without delay compensation (SSI) and with delay compensation (BiSS).**

The cable must be twisted pair and shielded.
© by HENGSTLER

HENGSTLER claims copyright protection for this documentation.

This documentation must not be modified, amended, copied or given to third parties without prior written approval by HENGSTLER.

ACURO® is a registered trademark of HENGSTLER

We reserve the right to make technical modifications and improvements that provide technical progress to our products.

HENGSTLER

HENGSTLER GmbH
Uhlandstr. 49
78554 Aldingen / Germany
Tel. +49 (0) 7424-89 0
Fax +49 (0) 7424-89 500
E-Mail: info@hengstler.com
www.hengstler.com