Technical Information

Steering
SASA Sensor
### Revision history

**Table of revisions**

<table>
<thead>
<tr>
<th>Date</th>
<th>Changed</th>
<th>Rev</th>
</tr>
</thead>
<tbody>
<tr>
<td>Nov 2017</td>
<td>Replaced SAK Adapter image on page 20</td>
<td>0102</td>
</tr>
<tr>
<td>March 2016</td>
<td>First edition</td>
<td>0101</td>
</tr>
</tbody>
</table>
## Sensor type SASA general

### Versions, code numbers and weights for SASA sensor

**SASAIIC**
- SASAIIC CAN message protocol: 6
- SASAIIC parameter setup: 7
- SASAIIC technical data: 8

**SASAIID**
- SASAIID CAN message protocol: 9
- SASAIID configuration protocol: 12
- Read data by identifier: 12
- Write data by identifier: 14
- SASAIID configuration data list: 17
- SASAIID technical data: 18

### Dimensions SASA sensor

**SAK**
- Code number and weight, SAK adapter kit: 20
- SAK adapter kit: 20

### Installation

- Wiring guidelines: 22
Sensor type SASA general

The SASA sensor detects the absolute position and speed of the steering wheel. The sensor can be used in electro-hydraulic steering systems using Danfoss OSPE, EHI, or EHPS steering valves with programmable controller or in any other steering system where the rotation of the steering wheel must be detected as an electronic signal.

There are two versions available of SASA sensors:

- **SASAIC** used for PVED-CL actuator. SASAIC has single CAN output signal and “fail silent” concept.
- **SASAID** used for PVED-CLS actuator and general usage. SASAID has dual CAN output signal.

The use of SASA sensor is relevant e.g. for variable steering ratio and closed loop set-ups where steering wheel position and steering angle have to match.

The SASA sensor also can be used for “kick out” of Auto-steering. When the steering valve is in auto-guidance mode, and the driver wants to swap to manual steering, a signal from the SASA sensor will deactivate the Auto-steering and the steering wheel movement has priority.

SASA is based on a non-contact inductive principle giving a very high resolution.

The sensor features a robust design and resists e.g. electro-magnetic radiation.

The output is a CAN signal, which makes it easy to interface to advanced vehicle controllers and to Danfoss PVED-CLS actuators for steering valves.

The steering wheel shaft turns the rotor of the SASA sensor, and the sensor is simply mounted between steering unit and steering column. The shaft of the steering column must be 15 mm longer when using SASA sensor.

In cases where customers want to use the same steering column in applications with and without SASA sensors, Danfoss offers an adapter kit type SAK to built in between column and sensor.

The SASA sensor offers the following features:

- High resolution < 0.1°
- Output CAN signal
- High safety
  - SASAIC: “fail silent” concept
  - SASAID: two output messages concept
- SASAID is PLUS+1® Compliant
- Flanged in between steering unit and column
- Compact design
## Versions, code numbers and weights for SASA sensor

<table>
<thead>
<tr>
<th>Code number</th>
<th>Type</th>
<th>Supply voltage</th>
<th>Termination resistor</th>
<th>Cable length</th>
<th>Connector</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>11088656</td>
<td>SASAIIC, CAN</td>
<td>9 - 32 V&lt;sub&gt;DC&lt;/sub&gt;</td>
<td>No</td>
<td>500 mm</td>
<td>AMP code no. 2-967059-1</td>
<td>0.25 [0.55]</td>
</tr>
<tr>
<td>11099289</td>
<td>SASAIIC, CAN</td>
<td>9 - 32 V&lt;sub&gt;DC&lt;/sub&gt;</td>
<td>No</td>
<td>500 mm</td>
<td>DEUTSCH DT04-4P-CE02</td>
<td>0.25 [0.55]</td>
</tr>
<tr>
<td>11116505</td>
<td>SASAIDI, CAN</td>
<td>6 - 36 V&lt;sub&gt;DC&lt;/sub&gt;</td>
<td>No</td>
<td>500 mm</td>
<td>DEUTSCH DT04-4P-CE02</td>
<td>0.25 [0.55]</td>
</tr>
</tbody>
</table>
SASAIIC CAN message protocol

Interface: CAN 2.0 B
Baud rate: 125 kBaud, 250 kBaud (default), 500 kBaud

SASA returns cyclic the following CAN message every 5, 10 (default) or 20 ms.

<table>
<thead>
<tr>
<th>301h</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Low byte</td>
</tr>
<tr>
<td>1</td>
<td>High byte</td>
</tr>
<tr>
<td>2</td>
<td>Low byte</td>
</tr>
<tr>
<td>3</td>
<td>High byte</td>
</tr>
<tr>
<td>4</td>
<td>Low byte</td>
</tr>
<tr>
<td>5</td>
<td>High byte</td>
</tr>
</tbody>
</table>

**ID**
- Steering angle (12 bit word)
  - 0 = 0 degrees
  - 4095 = 359,912 degrees
  - Overflow at 4095 for CW activation shall increment 0
  - Underflow at 0 for CCW activation shall decrement 4095

**Count**
- Byte (0-255)
- Increments 1 for each message

**Steering angle change**
- Difference between 2 transmitted position values in succession.
  - 16 bit integer with 2's complementary encoding for negative values (-32768 to 32767).
  - -4095 = -359,912 degrees
  - 0 = 0 degrees
  - 4095 = 359,912 degrees

**Status**

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>Programming mode</td>
</tr>
</tbody>
</table>

**Programming mode**
- Normal state is 1
- Response with a 0 when starting the programming sequence (See the programming sequence described below under setup message)

**CRC-16**
- The standard CRC16 polynomial is used \((x^{16} + x^{15} + x^2 + 1)\)
SASAIIC parameter setup

Setup message: sensor can be programmed as shown in the CAN setup message below.

<table>
<thead>
<tr>
<th>ID</th>
<th>Baud rate</th>
<th>Data rate</th>
<th>Set 0-index</th>
<th>Programming sequence</th>
<th>CRC-16</th>
</tr>
</thead>
<tbody>
<tr>
<td>0C0h</td>
<td>02h for 125 kBaud</td>
<td>02h for 5 ms</td>
<td>02h for 125 kBaud</td>
<td>02h for 5 ms</td>
<td>02h for 125 kBaud</td>
</tr>
</tbody>
</table>

Identifier

0C0h (11 bit)

Baud rate

During the programming sequence, byte 0 is set to:

- 02h for 125 kBaud
- 03h for 250 kBaud (default)
- 04h for 500 kBaud

Data rate

During the programming sequence, byte 1 is set to:

- 02h for 5 ms
- 03h for 10 ms (default)
- 04h for 20 ms

Set 0-index

If byte 2 is set to AAh during the programming sequence, the actual angle will be stored as a reference value (0 degree) in persistent memory.

Programming sequence

The following sequence is used when programming the sensor.

1. The controller unit sends a setup message where byte 4 is set to AAh and byte 5 is set to 55h.
2. The sensor answers with a 0 on the status byte (bit 0).
3. The control unit then sends a setup message where byte 4 is set to 0Fh and byte 5 is set to F0h.

The first and second message shall match.

4. After receiving the last message the programming takes place in the sensor if the parameters are in the defined range, the timeout period has not been exceeded and the CRC-16 check is correct in both messages.

5. After programming the status bit in the output message changes back from 0 to 1.

6. If the controller's second message does not come within 1 second after the controller's first message, the programming sequence will be aborted.

CRC-16

The standard CRC16 polynomial is used \(x^{16}+x^{15}+x^{2}+1\)
SASAIIC technical data

**Mechanical**
- **Input range:** Continuous 360° rotation
- **Rotor torque:** ≤ 0.2 N•m
- **Expected life:** > 10 million cycles

**Electrical**
- **Supply voltage:** 9 - 32 V_{DC}
- **Power consumption:** < 1 W

**Output**
- **CAN V2.0B, (compatible to J1939)**
- **Termination resistor:** 120 ohm (optional)
- **Baud rate:** 125, 250 or 500 kb/s
- **Angle:** 12-bit word (0 - 4095) relative to a programmable 0-index point.
- **Resolution:** < 0.1°
- **Linearity:** 1% of full scale
- **Angle change:** 16 bit integer with 2's complementary encoding for negative values (-32768 to 32767).

**Safety function**
If a failure occurs, the SASA CAN transceiver will be disabled.

**Environmental**
- **Operating temperature:** -30° to 85°C [-22 to 185°F]
- **Storage temperature:** -40° to 105°C
- **Sealing:** IP65
- **EMI/RFI Rating:** 100 V/m
- **Vibration:** Meets IEC 60068-2-64
- **Shock:** Meets IEC 60068-2-27 test Ea
SASAIID CAN message protocol

Dual CAN output

The PVED-CLS (valve controller) can interface to a SASA with redundant output. A safety function that compares the two output messages from SASA is implemented in the PVED-CLS. There is a 180-degree phase shift between the two angles transmitted in the messages. Steering angle 1 and steering angle 2 are sampled with a delay < 2ms. The ID1 and ID2 messages are transmitted with a delay < 5ms. It is possible to configure the sensor to transmit cyclic 1 or 2 messages.

Interface: CAN 2.0 B
Baud rate: 250 kBaud
Transmit rate: 10, 20, 50 (default), 100, 200 ms (cyclic message transmission, 1 or 2 messages, tolerance: ±10%)

<table>
<thead>
<tr>
<th>Proprietary B 29-bit</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>ID1</td>
<td>Steering angle 1</td>
</tr>
</tbody>
</table>

Identifier

J1939 proprietary B. Programmable 29 bit message id.
ID1 = $0CFFYYXX
PGN-offset1 YY is programmable, default value is $10
Source Address XX is programmable, default value is $4D

Steering angle 1

Data byte 0-1: Value measured by angle sensor 1 relative to the 0-index point.
16-bit integer
0 corresponds to 0 degrees
4095 corresponds to 359.912 degrees
Overflow at 4095 for CW activation shall increment 0
Underflow at 0 for CCW activation shall decrement 4095
4096 – 65534: Not available
65535: Sensor failure

Steering angle velocity 1

Data byte 2-3: Velocity of the rotor measured by sensor 1.
16-bit integer
0 corresponds to -300 RPM (CCW)
20480 corresponds to 0 RPM
40960 corresponds to 300 RPM (CW)
40961 – 65534: Not available
65535: Sensor failure

Reserved

Data byte 4: $FF

Status

Data byte 5
SASAIID

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error Code</td>
<td>Sequence number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Error Code**
- Bit 4-7
- $0$ - Reserved
- $1$ - Sensor Chip Error
- $2$ - Steering angle failure
- $3$ - CAN input message failure
- $4$ - Power failure
- $5$ - CPU failure
- $6$ - Memory failure
- $7$-$D$ - Reserved
- $E$ - Temperature warning
- $F$ - No error

**Sequence number**
- Bit 0-3
- Increments 1 for each message.
- Valid range is $0$ - $F$

**CRC-16**
Data byte 6-7: The CRC16 polynomial 0xC86C is used to calculate a checksum for byte 0-5.

- Width - 16
- Init - 0x0000
- Ref-In - False
- Ref-Out - False
- Xor-Out - 0x000
- Check - 0x6774

'Check' is the CRC result for UTF-8 string "123456789".
The above parameters conform to the Rocksoft™ Model CRC Algorithm.

<table>
<thead>
<tr>
<th>Proprietary B 29-bit</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>ID2</td>
<td>Steering angle 2</td>
</tr>
</tbody>
</table>

**Identifier 2 (Optional)**
- J1939 proprietary B. Programmable 29 bit message id.
- ID2 = $0CFFZZXX$
- PGN-offset2 ZZ is programmable, default value is $11$
- Source Address XX is programmable, default value is $4D$
### SASAIID

**Steering angle 2**

Data byte 0-1: Value measured by angle sensor 2 (with a 180 degree offset to Steering angle 1).

- 16-bit integer
- 0 corresponds to 180 degrees
- 4095 corresponds to 179.912 degrees
- Overflow at 4095 for CW activation shall increment 0
- Underflow at 0 for CCW activation shall decrement 4095
- 4096 – 65534: Not available
- 65535: Sensor failure

**Steering angle velocity 2**

Data byte 2-3: Velocity of the rotor measured by sensor 2.

- 16-bit integer
- 0 corresponds to -300 RPM (CCW)
- 20480 corresponds to 0 RPM
- 40960 corresponds to 300 RPM (CW)
- 40961 – 65534: Not available
- 65535: Sensor failure

**Reserved**

Data byte 4: $FF

**Status**

Data byte 5

<table>
<thead>
<tr>
<th>7</th>
<th>6</th>
<th>5</th>
<th>4</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
</tr>
</thead>
<tbody>
<tr>
<td>Error Code</td>
<td>Sequence number</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Error Code**

- Bit 4-7
- $0 - Reserved
- $1 - Sensor Chip Error
- $2 - Steering angle failure
- $3 - CAN input message failure
- $4 - Power failure
- $5 - CPU failure
- $6 - Memory failure
- $7-$D - Reserved
- $E - Temperature warning
- $F - No error

**Sequence number**

- Bit 0-3
- Increments 1 for each message.
- Valid range is $0 - $F

**CRC-16**

Data byte 6-7: The CRC16 polynomial 0xC86C is used to calculate a checksum for byte 0-5.
SASAIID

Width - 16
Init - 0x0000
Ref-In - False
Ref-Out - False
Xor-Out - 0x000
Check - 0x6774

'Check' is the CRC result for UTF-8 string “123456789”.
The above parameters conform to the Rocksoft™ Model CRC Algorithm.

SASAIID configuration protocol

The sensor complies with the PLUS+1® Lite Diagnostic Communication Protocol – UDS.

The services to configure the sensor are:
• Read data by identifier on page 12
• Write data by identifier on page 14

Read data by identifier

This service is used for retrieving the value of a given parameter.

<table>
<thead>
<tr>
<th>29 bit CAN identifier</th>
<th>PCI and Frame data bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Byte #0</td>
</tr>
<tr>
<td>CAN ID</td>
<td>PCI</td>
</tr>
</tbody>
</table>

CAN ID

ID = $1BC788XX (The sensor node address XX is configurable, default value is $4D, and the Danfoss diagnostic tester node address is $F1)

ID = $1BC78FFF (Used for broadcast requests, i.e. when the Danfoss diagnostic tester is requesting the configuration address format)
SASAIID

**PCI**
$\text{PCI} = \$03$ (PCI type = 0, SF Data Length = 3)

**Service ID**
$\text{SID} = \$22$ – Read Data by Identifier

**ID - Sub function**
See *SASAIID configuration data list* on page 17

**Positive response**
This message is used as positive response to a Read Data by Identifier request.

<table>
<thead>
<tr>
<th>29 bit CAN identifier</th>
<th>PCI and Frame data bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte #0</td>
<td>Byte #1</td>
</tr>
<tr>
<td>Byte #2</td>
<td>Byte #3</td>
</tr>
<tr>
<td>Byte #4</td>
<td>Byte #5</td>
</tr>
<tr>
<td>Byte #6</td>
<td>Byte #7</td>
</tr>
<tr>
<td>CAN ID</td>
<td>PCI</td>
</tr>
<tr>
<td>Service ID</td>
<td>ID – Sub function (MSB first)</td>
</tr>
<tr>
<td>Value</td>
<td></td>
</tr>
</tbody>
</table>

**CAN ID**
$\text{ID} = \$1BC268F1$ (Here shown with the default sensor node address $\$4D$, and the Danfoss diagnostic tester node address $\$F1$)

**Negative response**
This message is used as negative response to a Read Data by Identifier request.

<table>
<thead>
<tr>
<th>29 bit CAN identifier</th>
<th>PCI and Frame data bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte #0</td>
<td>Byte #1</td>
</tr>
<tr>
<td>Byte #2</td>
<td>Byte #3</td>
</tr>
<tr>
<td>CAN ID</td>
<td>PCI</td>
</tr>
<tr>
<td>Service ID</td>
<td>ID – Sub function (MSB first)</td>
</tr>
</tbody>
</table>

**CAN ID**
$\text{ID} = \$1BC268F1$ (Here shown with the default sensor node address $\$4D$, and the Danfoss diagnostic tester node address $\$F1$)
**SASA Sensor**

### Write data by identifier

This service is used for writing configuration data.

```
<table>
<thead>
<tr>
<th>29 bit CAN identifier</th>
<th>PCI and Frame data bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte #0</td>
<td>Byte #1</td>
</tr>
<tr>
<td>CAN ID</td>
<td>PCI</td>
</tr>
</tbody>
</table>
```

**CAN ID**

ID = $1BC788XX (The sensor node address XX is configurable, default value is $4D, and the Danfoss diagnostic tester node address is $F1)
SASAIID

<table>
<thead>
<tr>
<th>29 bit CAN identifier</th>
<th>PCI and Frame data bytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Byte #0</td>
<td>Byte #1</td>
</tr>
<tr>
<td>CAN ID</td>
<td>PCI</td>
</tr>
</tbody>
</table>

**PCI**  
PCI = $0X$ (PCI type = 0, SF Data Length = X)

**Service ID**  
SID = $2E$ – Write Data by Identifier

**ID - Sub function**  
See SASAIID configuration data list on page 17

**Value**  
See SASAIID configuration data list on page 17

**Positive response**

This message is used as positive response to a Write Data by Identifier request.

**Negative response**

This message is used as negative response to a Write Data by Identifier request.
ID = $1BC268F1 (Here shown with the default sensor node address $4D, and the Danfoss diagnostic tester node address $F1)

<table>
<thead>
<tr>
<th>Priority</th>
<th>Extended data page</th>
<th>Data page</th>
<th>Type of service (TOS)</th>
<th>Source address</th>
<th>Destination address</th>
</tr>
</thead>
<tbody>
<tr>
<td>28</td>
<td>27</td>
<td>26</td>
<td>25</td>
<td>24</td>
<td>23</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>01</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

PCI = $03 (PCI type = 0, SF Data Length = 3)

Service ID SID = $7F –Negative response ID

ID - Sub function $2EXX (For XX, see the following table)

<table>
<thead>
<tr>
<th>Byte 2</th>
<th>Error message</th>
<th>Error cause</th>
<th>Service Tool response</th>
</tr>
</thead>
<tbody>
<tr>
<td>$13</td>
<td>Incorrect Message Length Or Invalid Format</td>
<td>The length of the message is wrong</td>
<td>Resend correct command</td>
</tr>
<tr>
<td>$22</td>
<td>Conditions Not Correct</td>
<td>This code shall be returned if operating conditions of the server for performing the required action are not met.</td>
<td>Resend command later</td>
</tr>
<tr>
<td>$31</td>
<td>Request Out Of Range</td>
<td>This code shall be sent if: 1. none of the requested data identifier values are supported by the device 2. the client exceeded the maximum number of data identifiers allowed to be requested at a time</td>
<td>Resend with correct ID / data</td>
</tr>
<tr>
<td>$33</td>
<td>Security Access Denied</td>
<td>This code shall be sent if at least one of the data identifiers is secured and the server is not in an unlocked state</td>
<td>Perform security access routine</td>
</tr>
<tr>
<td>$72</td>
<td>General Programming Failure</td>
<td>This return code shall be sent if the server detects an error when writing to a memory location</td>
<td>Resend command</td>
</tr>
<tr>
<td>$78</td>
<td>Response pending</td>
<td>Routine needs more time</td>
<td>Extend P2_CAN_Server_max to P2*_CAN_Server_max</td>
</tr>
</tbody>
</table>
### SASAIID configuration data list

This is a list of all configurable parameters.

<table>
<thead>
<tr>
<th>Identifier</th>
<th>Name</th>
<th>Access</th>
<th>Value/Range</th>
<th>Default</th>
</tr>
</thead>
<tbody>
<tr>
<td>$0101</td>
<td>Node Address</td>
<td>R/W</td>
<td>$01 – $FD</td>
<td>$4D</td>
</tr>
<tr>
<td>$0102</td>
<td>Output mode</td>
<td>R/W</td>
<td>$01 (Single message) $02 (Dual message)</td>
<td>$02</td>
</tr>
<tr>
<td>$0103</td>
<td>Message transmission rate (CAN mode)</td>
<td>R/W</td>
<td>$03 (10 ms) $04 (20 ms) $05 (50 ms) $06 (100 ms) $07 (200 ms)</td>
<td>$05</td>
</tr>
<tr>
<td>$0104</td>
<td>Zero-index</td>
<td>W</td>
<td>$AA – just a trigger value</td>
<td>$00</td>
</tr>
<tr>
<td>$0109</td>
<td>Temperature</td>
<td>R</td>
<td>$XX (8 bit integer),[°C]</td>
<td></td>
</tr>
<tr>
<td>$010A</td>
<td>PGN-offset1</td>
<td>R/W</td>
<td>$01 – $FF</td>
<td>$10</td>
</tr>
<tr>
<td>$010B</td>
<td>PGN-offset2</td>
<td>R/W</td>
<td>$01 – $FF</td>
<td>$11</td>
</tr>
<tr>
<td>$0201 - $021F</td>
<td>Reserved for supplier</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>$F010</td>
<td>Address Format</td>
<td>R</td>
<td>$A5</td>
<td></td>
</tr>
<tr>
<td>$F1FA</td>
<td>Diagnostic File ID Part A</td>
<td>R</td>
<td>$350A706F (MSB first)</td>
<td></td>
</tr>
<tr>
<td>$F1FB</td>
<td>Diagnostic File ID Part B</td>
<td>R</td>
<td>$0D164B57 (MSB first)</td>
<td></td>
</tr>
<tr>
<td>$F1FC</td>
<td>Diagnostic File ID Part C</td>
<td>R</td>
<td>$B59E069 (MSB first)</td>
<td></td>
</tr>
<tr>
<td>$F1FD</td>
<td>Diagnostic File ID Part D</td>
<td>R</td>
<td>$4ACFFEE7 (MSB first)</td>
<td></td>
</tr>
<tr>
<td>$F193</td>
<td>Hardware version</td>
<td>R</td>
<td>XXXXXXXX (4 characters)</td>
<td></td>
</tr>
<tr>
<td>$F195</td>
<td>Software version</td>
<td>R</td>
<td>XXXXXXXX (4 characters)</td>
<td></td>
</tr>
<tr>
<td>$F18C</td>
<td>Serial Number</td>
<td>R</td>
<td>XXXXXXXX (32 bit integer, MSB first)</td>
<td></td>
</tr>
<tr>
<td>$F188</td>
<td>Manufacturing date</td>
<td>R</td>
<td>YY,MM,DD in BCD</td>
<td></td>
</tr>
</tbody>
</table>
SASAIID technical data

**Mechanical**
- **Input range:** Continuous 360° rotation
- **Rotor torque:** ≤ 0.2 N•m
- **Expected life:** > 10 million cycles

**Electrical**
- **Supply voltage:** 6 - 36 V\(_{\text{DC}}\)
- **Power consumption:** <2 W

**Output**
- **CAN V2.0B, (compatible to J1939)**
  - **Termination resistor:** 120 ohm (optional)
  - **Baud rate:** 250 kb/s
  - **Angle:** 12-bit word (0 - 4095) relative to a programmable 0-index point.
  - **Resolution:** < 0.1°
  - **Linearity:** <1% of full scale
  - **Angle change:** 16 bit integer. Values: primary message from 0–40960. Redundant message: the same but with 20480 offset.

**Safety function**
Dual CAN output signal (the CAN signals are supervised by the PVED-CLS actuator or alternative external safety controller).
However, the SASAIID has built in these safety and monitoring functions:
- Sensor chip diagnostics
- Steering angle cross-check
- MCU heartbeat check: the two microcontrollers check each other’s aliveness
- Power-on memory check

**Environmental**
- **Operating temperature:** -40° to 105°C [-40° to 221°F]
- **Storage temperature:** -55° to 105°C [-67° to 221°F]
- **Sealing:** IP65
- **EMI/RFI Rating:** 150 V/m
- **Vibration:** Meets IEC 60068-2-64
- **Shock:** Meets IEC 60068-2-27 test Ea
Dimensions SASA sensor
SAK

Code number and weight, SAK adapter kit

<table>
<thead>
<tr>
<th>Code number</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>150Z6000</td>
<td>0.8</td>
</tr>
</tbody>
</table>

SAK adapter kit

1. Flexible teeth, 12 pieces to interact with splines on steering column
2. Cable, 500 mm with connector. See “Code numbers” for type of connector

1. Distance plate
2. Shaft
Installation

SASA has to be mounted between steering column and steering unit (OSP) with 4 bolts max 30 N-m [265.5 lbf-in]. Shaft in column must be 15 mm [0.59 in] longer when using SASA.

Assembly: SASA sensor and OSP steering unit

1. SASA sensor
2. OSP steering unit

Caution

Make sure that the spline profile of the SASA sensor is aligned to the spline profile of the steering column shaft. A safe method of assembly is to place SASA sensor on the steering column spline shaft first – and not opposite! In case of using force, there is a risk of bending the spline profile of SASA sensor.

For use of original steering column, use adapter kit type SAK, see sketch below.

Assembly: SAK adapter kit, SASA sensor and OSP steering unit

1. Shaft of SAK adapter kit
2. Distance plate of SAK adapter kit
3. SASA sensor
4. OSP steering unit
Installation

Wiring guidelines

- Protect wires from mechanical abuse, run wires in flexible metal or plastic conduits.
- Use 85˚ C (185˚ F) wire with abrasion resistant insulation and 105˚ C (221˚ F) wire should be considered near hot surfaces.
- Use a wire size that is appropriate for the module connector.
- Separate high current wires such as solenoids, lights, alternators or fuel pumps from sensor and other noise-sensitive input wires.
- Run wires along the inside of, or close to, metal machine surfaces where possible, this simulates a shield which will minimize the effects of EMI/RFI radiation.
- Do not run wires near sharp metal corners, consider running wires through a grommet when rounding a corner.
- Do not run wires near hot machine members.
- Provide strain relief for all wires.
- Avoid running wires near moving or vibrating components.
- Avoid long, unsupported wire spans.
- Ground electronic modules to a dedicated conductor of sufficient size that is connected to the battery (-).
- Power the sensors and valve drive circuits by their dedicated wired power sources and ground returns.
- Twist sensor lines about one turn every 10 cm (4 in).
- Use wire harness anchors that will allow wires to float with respect to the machine rather than rigid anchors.
Danfoss Power Solutions is a global manufacturer and supplier of high-quality hydraulic and electronic components. We specialize in providing state-of-the-art technology and solutions that excel in the harsh operating conditions of the mobile off-highway market. Building on our extensive applications expertise, we work closely with our customers to ensure exceptional performance for a broad range of off-highway vehicles.

We help OEMs around the world speed up system development, reduce costs and bring vehicles to market faster.

Danfoss – Your Strongest Partner in Mobile Hydraulics.

Go to www.powersolutions.danfoss.com for further product information.

Wherever off-highway vehicles are at work, so is Danfoss. We offer expert worldwide support for our customers, ensuring the best possible solutions for outstanding performance. And with an extensive network of Global Service Partners, we also provide comprehensive global service for all of our components.

Please contact the Danfoss Power Solution representative nearest you.

Products we offer:
- Bent Axis Motors
- Closed Circuit Axial Piston Pumps and Motors
- Displays
- Electrohydraulic Power Steering
- Electrohydraulics
- Hydraulic Power Steering
- Integrated Systems
- Joysticks and Control Handles
- Microcontrollers and Software
- Open Circuit Axial Piston Pumps
- Orbital Motors
- PLUS+1® GUIDE
- Proportional Valves
- Sensors
- Steering
- Transit Mixer Drives

Danfoss can accept no responsibility for possible errors in catalogues, brochures and other printed material. Danfoss reserves the right to alter its products without notice. This also applies to products already on order provided that such alterations can be made without changes being necessary in specifications already agreed.

All trademarks in this material are property of the respective companies. Danfoss and the Danfoss logotype are trademarks of Danfoss A/S. All rights reserved.