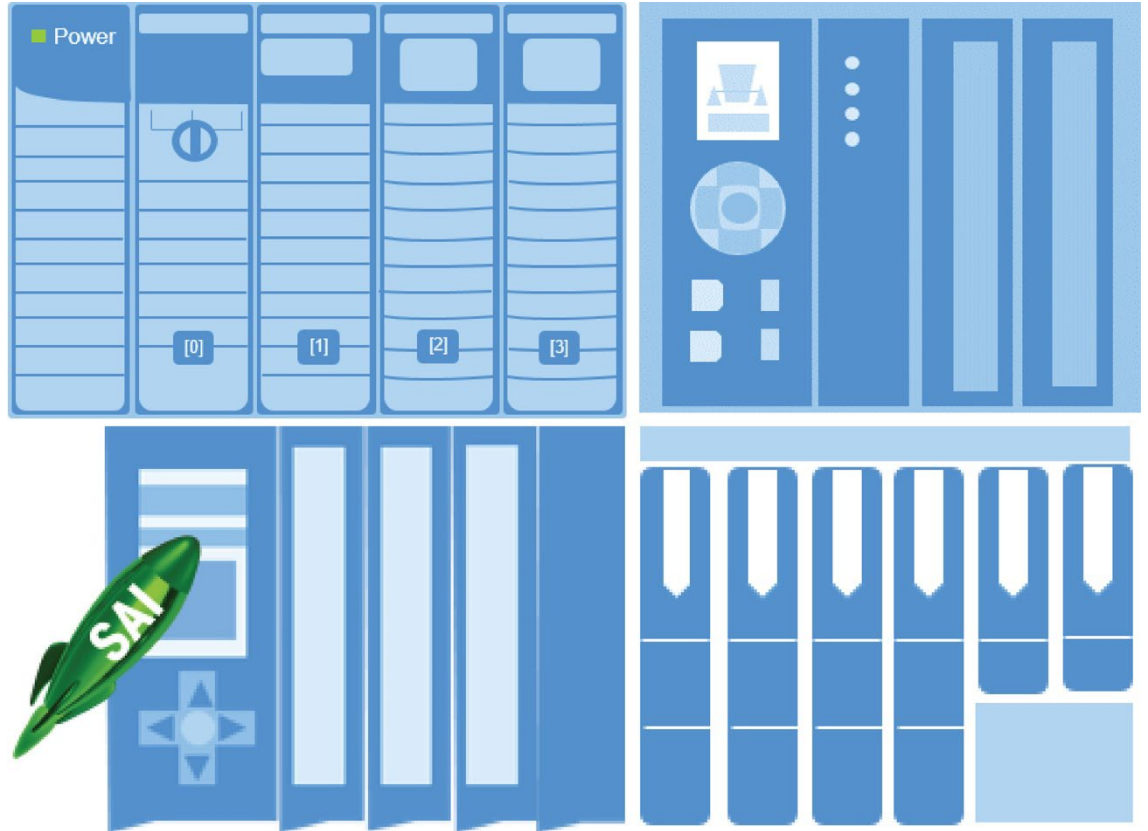


Standard Automation Interface

Version 2.0.00



METTLER TOLEDO

Contents

| | | |
|----------|---|------------|
| 1 | General Overview | 1-1 |
| 1.1. | Introduction | 1-2 |
| 1.2. | General Data Types | 1-2 |
| 2 | Block Formats..... | 2-1 |
| 2.1. | 1 Block Format..... | 2-1 |
| 3 | Block Types | 3-1 |
| 3.1. | Floating Point Block | 3-1 |
| 3.2. | Status/Command Block | 3-5 |
| 4 | Acyclic Format..... | 4-1 |
| 4.1. | Control System Integration..... | 4-1 |
| 4.2. | Access Methods..... | 4-1 |
| 4.3. | Control System Parameters for Direct and Indirect Access..... | 4-2 |
| 4.4. | PROFIBUS/PROFINET Acyclic Messages | 4-3 |
| 4.5. | EIP/ControlNet Acyclic Messages | 4-3 |
| 5 | Commands..... | 5-1 |
| 5.1. | Command Responses | 5-1 |
| 5.2. | Floating Point block Commands | 5-4 |
| 5.3. | Status block Commands | 5-5 |
| 5.4. | Variable Block Commands | 5-5 |
| 6 | Status/Command Block Word Types | 6-1 |
| 6.1. | RedAlert Alarms (default, word 0) | 6-1 |
| 6.2. | Alarm Status | 6-2 |
| 6.3. | Scale Group 2 Status (default word 1)..... | 6-4 |
| 6.4. | Target Status Group | 6-5 |
| 6.5. | Comparator Status Group(s) | 6-6 |
| 6.6. | I/O Status Groups | 6-7 |
| 6.7. | Custom Application Status Group..... | 6-8 |
| 6.8. | Last Error Message Status Group | 6-9 |
| 6.9. | Custom Application Group 2..... | 6-10 |
| 7 | Test Mode | 7-1 |
| 7.1. | Entering Test Mode..... | 7-1 |
| 7.2. | Exit Test Mode | 7-1 |
| 7.3. | Byte Order Selection | 7-2 |

- 7.4. Floating Point Test Commands for "Report" values 7-2
- 7.5. Floating Point Test Commands for Scale Status bits..... 7-3
- 7.6. Status Block Test Commands 7-3
- 7.7. Test Variables for Acyclic and Variable Block Test Commands 7-3
- 7.8. Performance Test Commands 7-3
- A. Cyclic Commands A-1**
- A.1. Special Commands A-4
- A.2. General System Commands A-4
- A.3. Floating Point block Commands A-5
- A.4. Status/Command block Commands A-17
- B. Acyclic (Explicit) Commands B-1**
- B.1. Direct Level 1 B-1
- B.2. Direct Level 2 B-5
- B.3. Indirect Variables B-5

1 General Overview

The Standard Automation Interface (SAI) is a protocol designed to exchange data between METTLER TOLEDO devices and automation systems. The goals of this standard are to provide:

1. A common data layout for load cells, terminals, and other devices regardless of the physical interface or automation network used.
2. A protocol that is optimized for machine-to-machine communication along with meaningful condition monitoring.
3. A protocol covering sensors with a capacity of 10g up to scales in the thousands of kilograms.
4. A single protocol for the convenience of automation integrators, control system programmers, and our automation customers
5. A tiered approach to create a flexible protocol for diverse devices.

In this document, we will refer to the METTLER TOLEDO product using SAI to communicate as "the device". The "controller", or "control system" will be used to designate an automation system such as a Programmable Automation Controller (PAC), a Programmable Logic Controller (PLC), a Distributed Control System (DCS) or an Industrial PC (IPC).

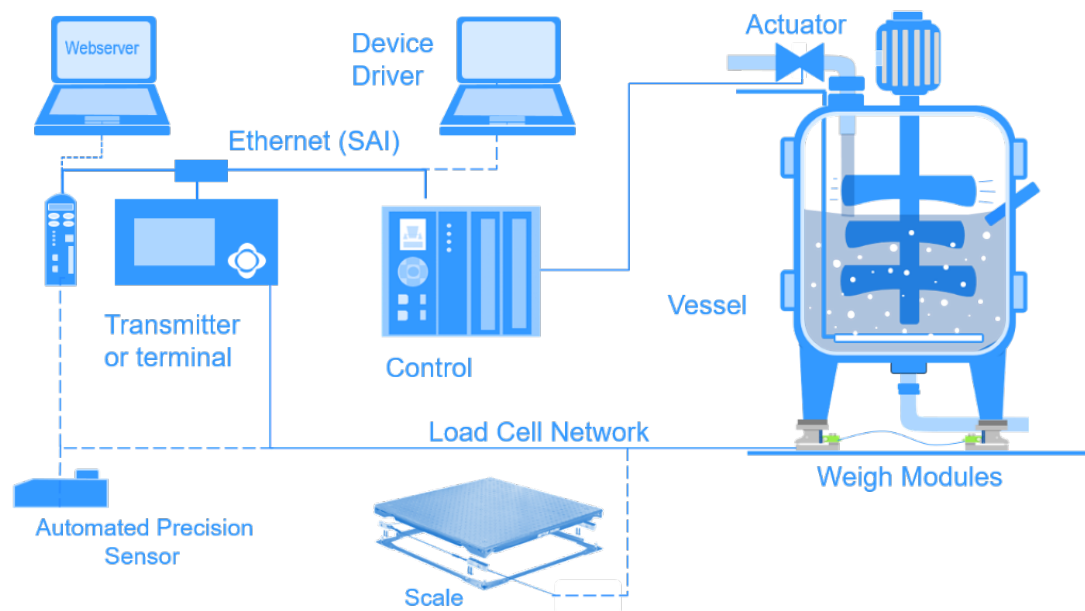


Figure 1-1: Overview of a Control System

1.1. Introduction

The SAI protocol is intended for use with weighing devices connected to control systems. The primary weight data is based on 32-bit floating point; allowing our users to use their same program no matter which sensor, transmitter, terminal, scale or software they select. SAI provides Condition Monitoring, called "status" in each cyclic message. Even though SAI is a standard, METTLER TOLEDO products have more or less features depending on their level of performance. These features are described in each sensor's individual documentation.

Products using the SAI protocol, when configured, will immediately begin transmitting rounded gross weight as a floating-point value. Byte order called Endianness is automatically set for PROFIBUS, PROFINET and EtherNet/IP. For other controller types, there is an automatic order detection feature built-in. This eliminates the necessity to perform byte swapping in the controller.

This manual contains specific information about how SAI is constructed. METTLER TOLEDO recommends that you use the appropriate device drivers available either from ODVA, Rockwell Automation, PROFIBUS International or directly from METTLER TOLEDO. Currently, METTLER TOLEDO supports EDS (EtherNet/IP), Custom AOP (EtherNet/IP), GSD (PROFIBUS DP) and GSDML (PROFINET) device drivers. Sample programming code is available for selected products and control systems.

SAI is managed as a standard at METTLER TOLEDO and this protocol is tested and proven for each product as well as the target network with approved components. METTLER TOLEDO strongly recommends that automation infrastructure components, such as cables and switches, are those recommended by the automation supplier (e.g. Rockwell or Siemens) to avoid startup problems or network performance risks

1.2. General Data Types

The protocol has two primary data types – cyclic data and asynchronous data (also known as acyclic or explicit messaging).

1.2.1. Cyclic Data

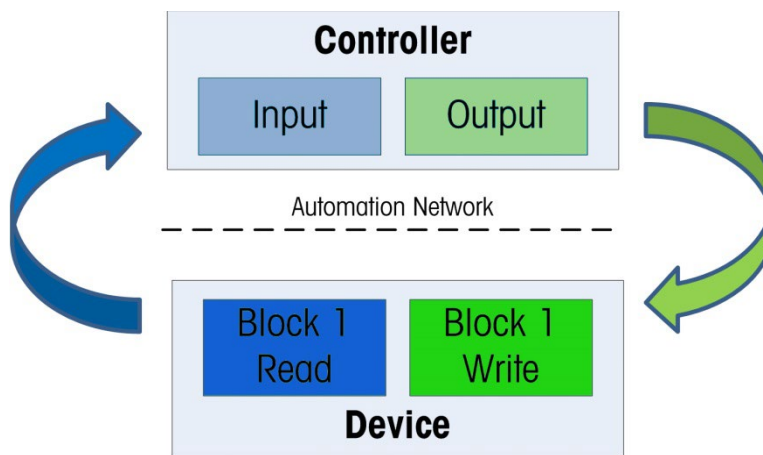


Figure 1-2: Controller/Device Read-Write Overview

Cyclic data is used when the controller requires constantly updating data from the sensor or scale. The cyclic data is refreshed by the scale or sensor at the fastest rate possible. In those instances where the data refresh rate is too high, the controller can send a command to reduce the speed. The cyclic data is broken in sections of data called blocks. Each block of data is 4 words (16 bits per word) in length. The data within these blocks expresses numeric values, string values, or individual bits representing state/command depending on the type of block specified. The number of input words (data sent from the device to the process controller) and output words (data sent from the process controller to the device) always match. This limits the number of configurations in the controller to a reasonable amount.

- Minimum size of data - 1 Block In & 1 Block Out

The three types of cyclic blocks are Floating Point Block, Status Command Block, and Variable Block. Numeric values will be sent via floating-point blocks as opposed to integer or other types because the decimal point and sign are included and do not require special data handling. Status data is grouped together in 16-bit words to conserve space. These are indicated by Boolean variables in the control system.

Fixed formats are provided for applications requiring smaller amounts of data with little to no device configuration. A custom format selection is optionally provided to allow the user to select the types and number of blocks required. See Section 2 for more information on block formats.

1.2.2. Acyclic Data

The operation of the SAI Acyclic Format can be simply described as, the control system sends a request and the device responds to the request. Acyclic communication operates in a significantly different manner compared to the other formats because it was designed for one-time read or write data that occurs outside of the normal controller scan cycle. Acyclic communication is typically used for setup data before the operation "starts" or other special information that is not needed as frequently; for example, zero, tare and clear tare commands. Acyclic messaging is typically not used for "real-time" activities and is generally used for non-repetitive or low repetition requests from a control system.

1.2.3. Acyclic Commands transmitted Cyclically

SAI supports a number of acyclic commands such as zero, tare and clear tare. At the operator's convenience, these commands can also be sent cyclically. A caution here. Commands such as zero and tare shall only be sent once, even when transmitted cyclically, to prevent weighing errors.

2 Block Formats

Block formats are based on device/application functionality. Data type and space required are dependent on the selected format. All devices use either the 1 block format or 2 block format as their default. Any other format, such as an 8 block, is optional based on a device's functional capabilities.

2.1. 1 Block Format

The SAI 1 Block Format is the smallest (lean) format and was designed with three criteria in mind:

1. Some control network interfaces have limited bandwidth.
2. Some devices have limited data and processing power.
3. Some users want limited data.

The objective is to provide basic data with little to no special coding. In addition, data size is limited to ensure high communication performance and to comply with as many automation networks as possible.

The 1 Block Format always consists of a single Read Floating-Point Block and a single Write Floating Point Block.

| Block format | Write data (Control system to device) | | Read data (Device to control system) |
|-----------------------|---------------------------------------|--|---|
| 1 Block Format | Floating Point Block | | Floating Point Block |
| | Word 0 | Floating point value (32 bit), optionally used as parameter with command | Requested floating point value (32 bit) |
| | Word 1 | | |
| | Word 2 | 16 bit channel mask | 16 bit device status |
| | Word 3 | Command | Response |

Figure 2-1: 1 Block Data Format Layout

2.1.1. 2 Block Format

The SAI 2 Block Format builds on the format structure used by the 1 Block Format; providing support for two blocks of input data and two blocks of output data. This format is designed for applications where the weighing device supports additional status information.

The cyclic data of the 2 Block Format supports a single Floating Point Block (Identical to the 1 Block) and a single Status Block for each of the read and write data areas.

| Block format | Write data (Control system to device) | | Read data (Device to control system) |
|----------------|---------------------------------------|--|---|
| 2 Block Format | Floating Point Block | | Floating Point Block |
| | Word 0 | Floating point value (32 bit), optionally used as parameter with command | Requested floating point value (32 bit) |
| | Word 1 | | |
| | Word 2 | 16 bit Channel Mask | 16 bit device status |
| | Word 3 | Command | Response |
| | Status/Command Block | | Status Block |
| | Word 4 | Reserved or optional Status Word selection | Status/Command Group 1 |
| | Word 5 | Reserved or optional Status Word selection | Status Group 2 |
| | Word 6 | Reserved or optional Status Word selection | Status Group 3 |
| | Word 7 | Command | Response |

Figure 2-2: 2 Block Data Format Layout

2.1.2. 8 Block Format

The SAI 8 Block Format builds on the format structure used by the 2 Block format; providing support for eight blocks of input data and eight blocks of output data. This format was designed for applications where the users would prefer more data within one read cycle. For example, reading gross weight, tare weight and net weight all in one cycle.

The cyclic data of the 8 Block Format supports seven instances of a Floating Point Block and one instance of a Status/Command Block for each of the read and write data areas.

| MT 'Device' to 'Control' Words (Read) | | | | | | | | 'Control' to MT 'Device' Words (Write) | | | | | | | |
|--|-----|-----|-----|-----|-----|-----|-----|--|-----|-----|-----|-----|-----|-----|-----|
| 8 Block format (8 blocks in/8 blocks out: 7 sets of Floating Point Blocks and Status/Command Blocks in each) | | | | | | | | | | | | | | | |
| FPB | SCB | FPB | FPB | FPB | FPB | FPB | FPB | FPB | SCB | FPB | FPB | FPB | FPB | FPB | FPB |

Figure 2-3: 8-Block Format Data Layout

3 Block Types

SAI supports two block types – floating point and status/command. Both block formats are composed of one or multiple of these blocks.

3.1. Floating Point Block

The Floating Point Block consists of a 32-bit Floating Point value (2 words) and two 16-bit words used for Boolean status (response) or operation (command):

| Floating Point Block (Write) | |
|------------------------------|---|
| Word 0 | Floating point value (32-bit), optionally used with command |
| Word 1 | |
| Word 2 | Channel mask |
| Word 3 | Command value |

Figure 3-1: Control System to Device

| Floating Point Block (Read) | |
|-----------------------------|---|
| Word 0 | Requested floating point value (32-bit) |
| Word 1 | |
| Word 2 | Device status bits |
| Word 3 | Response value |

Figure 3-2: Device to Control System

3.1.1. Floating Point Block Write (Control System to Device)

This block contains three parts:

1. 32 bits for a Floating Point value that can optionally be used as a “command” argument. (Words 0-1)
2. 16 bits that provide the ability to send a single command (such as zero or tare) to multiple channel scales (word 2)
3. 16 bits for the “Command value” used to identify the data desired by the control system (Word 3). Refer to Appendix A for details

3.1.1.1. Floating Point Value (Word 0 & 1)

The floating point value is single precision. Layout of floating point value bits are described in the byte order section. Note that the 32-bit precision causes truncation of the 8th most significant digit on very precise sensors and balances.

3.1.1.2. Channel Mask (Word 2)

A Channel Mask is used to target a “command” to a particular device in multiple channel devices. If the command shall only be applied to one particular channel, the channel mask should have a “1” in the channel to be affected position all other positions shall remain “zero”. Current devices only support one channel so channel 1 is the only channel that should be modified via the PLC. The actual command that is applied to is defined as part of the Command Word.

| Channel Mask Structure | |
|------------------------|---------------------------------|
| Bit # | Data |
| 0 | 1 = Apply command to Channel 1 |
| 1 | 1 = Apply command to Channel 2 |
| 2 | 1 = Apply command to Channel 3 |
| 3 | 1 = Apply command to Channel 4 |
| 4 | 1 = Apply command to Channel 5 |
| 5 | 1 = Apply command to Channel 6 |
| 6 | 1 = Apply command to Channel 7 |
| 7 | 1 = Apply command to Channel 8 |
| 8 | 1 = Apply command to Channel 9 |
| 9 | 1 = Apply command to Channel 10 |
| 10 | 1 = Apply command to Channel 11 |
| 11 | 1 = Apply command to Channel 12 |
| 12 | 1 = Apply command to Channel 13 |
| 13 | 1 = Apply command to Channel 14 |
| 14 | 1 = Apply command to Channel 15 |
| 15 | 1 = Apply command to Channel 16 |

Figure 3-3: Channel Mask Word

Bits 0-15 independently apply to channels 1-16 of the device as appropriate. If a command is issued for a specific device channel and includes the channel mask, all channels included in both words (the command word AND the mask) shall have the command applied to them.

For example, if a command should apply to Channel 1, Channel 2, and Channel 3, bits 0-2 should be set to 1 and all others would remain set to 0.

3.1.1.3. Command Value (Word 3)

This word is used to send commands to the device on a selected channel. The value used in this word is a combination of the command and the channel value. This combination informs the device where the specified command should be used when more than one measurement channel is present (for example, some terminals support multiple scales). Layout of this word is described in the Command & Response Word section.

| | | |
|---------------------------------------|------------------|--------------------|
| Bits 0-10 | Bits 11-14 | Bit 15 |
| Command Value (Refer to section A.3.) | Device Indicator | 0 = Okay 1 = Fault |

The Command value consists of three sets of bits: bits 0-10 are used to indicate what command is sent. Bits 11-14 indicate which device is providing the data and Bit 15 is always 0 to keep the overall value of this word positive unless the device cannot provide the desired data or comply with the requested command.

| Cycle x | | Cycle x + 1 | |
|-------------------------------------|--------------------------------------|------------------------------------|------------------------------|
| Floating Point Block (Write) | | Floating Point Block (Read) | |
| Word 0 | 0000000000000000 | Word 0 | Rounded Gross Weight |
| Word 1 | | Word 1 | |
| Word 2 | Channel mask = 0000000000000000 | Word 2 | Device status bits (Scale 1) |
| Word 3 | Send Gross Weight = 0000000000000000 | Word 3 | 00000000 = Gross Scale 1 |

Figure 3-4: Example 1 – Selecting Gross Weight (default Condition)

| Cycle x | | Cycle x + 1 | |
|-------------------------------------|---------------------------------|------------------------------------|------------------------------|
| Floating Point Block (Write) | | Floating Point Block (Read) | |
| Word 0 | 0000000000000000 | Word 0 | Rounded Gross Weight |
| Word 1 | | Word 1 | |
| Word 2 | Channel mask = 0000000000000000 | Word 2 | Device status bits (Scale 1) |
| Word 3 | Tare Scale | Word 3 | 00000000 = Gross Scale 1 |

Figure 3-5: Example 2 – Selecting Tare Scale

| Cycle x + 2 | | Cycle x + 3 | |
|-------------------------------------|---------------------------------|------------------------------------|--------------------------------|
| Floating Point Block (Write) | | Floating Point Block (Read) | |
| Word 0 | 0000000000000000 | Word 0 | Rounded Gross Weight |
| Word 1 | | Word 1 | |
| Word 2 | Channel mask = 0000000000000000 | Word 2 | Device status bits (Scale 1) |
| Word 3 | Load Rounded Net Weight | Word 3 | 00000000 = Rounded Net Scale 1 |

| Floating Point Block (Read) | |
|-----------------------------|------------------------------|
| Word 0 | Rounded Net Weight |
| Word 1 | |
| Word 2 | Device status bits (Scale 1) |
| Word 3 | Rounded Net Weight |

Figure 3-6: Example 3 – Selecting Net Weight and Viewing Gross Weight

3.1.2. Floating Point Block Read (Device to Control System)

The block consists of three parts:

1. 32 bits representing the floating point value (Words 0-1) that the Control device requested in the previous cycle
2. 16 bits to provide Scale status bits (Word 2) which are defined below

3. 16 bits for the “response value” (Word 3).

By default, if the control system sends no data in its Floating Point Block Write (the write data is all zeroes), the device will send its default data for channel 1 (rounded gross weight), the scale status word for this channel, and a response value word with all bits set to 0.

3.1.2.1.1. Floating Point Value (Word 0 & 1)

The floating point value is single precision. The order of the data in the floating point value will be determined by byte order configuration of the device (big endian and little endian are also possible). The default setting of these parameters is automatically selected for Profibus, PROFINET and Ethernet/IP. There is also the capability for automatic detection of Endianness.

In certain special cases, the floating-point value may be used to send a different type of numeric data (for example, a long integer). All special cases note the expected data type with the command request.

3.1.2.1.2. Device Status (Word 2)

The device status is a composite status word that contains individual bits to indicate the state of various scale or device specific binary (on/off) values. This includes error and data validation bits to determine when the data received is good and is safely used. It also contains command status bits to provide information on the state of an issued command in the Floating Point (write) Block. The following table explains the bits. For information on how a specific device implements these bits, please see the "Device Status Bits" section of the corresponding SAI device manual.

Table 3-1: Device Status Bits

| | |
|-------|--|
| Bit 0 | <p>Sequence bits: As commands are sent by the control system (and seen by the device), the device increments their value as an indication that a command has been seen and acted upon.</p> <p>These bits are used during a sequence of commands to ensure that there have been no sequencing errors in the request and the response. They are updated on every NEW command</p> |
| Bit 1 | |
| Bit 2 | <p>Heartbeat: Is a security mechanism that ensures that the device is working as expected and updating data in Words 0, 1 and 2, this heartbeat bit is toggled between off and on states at a rate of 1 time per second or less. Should this bit stop toggling, this indicates that the device is no longer operating and the weight value is incorrect.</p> |
| Bit 3 | <p>Data Okay: This bit gets set to 0 when the device is still operational but the value being reported cannot be guaranteed to be valid.</p> <p>The following conditions cause the Data Okay bit to be set to 0:</p> <ul style="list-style-type: none"> Device is powering up Device is in setup mode Device is in test mode Over capacity condition occurs <ul style="list-style-type: none"> When the A/D converter is at its limit Product dependent over capacity that occurs when the device determines it cannot trust the weight Under capacity condition occurs <ul style="list-style-type: none"> When the A/D converter is at its limit Product dependent under capacity that occurs when the device determines it cannot trust the weight |

| | |
|-----------|--|
| Bit 4 | RedAlert Alarm Condition: This bit is true when the weight value is invalid and the weighing process shall be stopped. If this bit is true it is an indication that the control system must stop until the source of the alarm is evaluated and corrected. The control system should evaluate the RedAlert status block to determine the nature of the alarm. Refer to RedAlert Alarms in section 6.1 of this document for more information. |
| Bit 5 | Center of Zero: This bit is true when the gross weight value is at a value of zero +/- one quarter of a weights and measures verification interval denoted as "e" |
| Bit 6 | Motion: This bit is true when the weight reported by the device is unstable the sensitivity of this bit is influenced both by electronic filtering and the motion sensitivity setting of each product - see individual product documentation for details. |
| Bit 7 | Net Mode: This bit is true when the unit is operating in "Net" mode (Gross-Tare). This bit indicates that a tare command is successful. <ul style="list-style-type: none"> ■ Note: Tare is the weight of the container, pallet, or other mass that is excluded from the measurement of the item to be weighed. |
| Bit 8 | Alternate weight unit: This bit indicates when a weight unit other than the primary weight unit is being used by the device (normally for the operator). This bit is set to "1" to indicate that an alternate weight unit is in use; however, the weight value in the Floating Point Block remains in the original unit of measure unless "displayed weight" is selected. |
| Bit 9 -15 | Device Specific bits: These bits are used to provide device specific status information Please refer to the Device Specific Bits section of each device's corresponding SAI device manual for information on the implementation of these bits. |

3.1.2.2.

Response Value (Word 3)

This is the response sent by the device to indicate the status of command execution. The structures of Response Value (in this block) and Command Value in the Write block should match, unless an error has occurred. The device responds as follows:

- Once a command is successfully sent by the control system and completed by the device, this Response value will include exactly the same value, plus an indication of the device address (usually 0000) for a single scale device or sensor.
- In case of an error (unknown command, error condition): the device responds with an error code indicating the nature of the error.

The Response value consists of three sets of bits: bits 0-10 are used to indicate what command is sent. Bits 11-14 indicate which device is providing the data and Bit 15 is always 0 to keep the overall value of this word positive unless the device cannot provide the desired data or comply with the requested command. Then this bit is set to a "1"

3.2. Status/Command Block

The Status blocks are used to provide state (on/off) data. The Status blocks are structured to support three sets of Status words. Examples of data provided through Status blocks include alarms, physical I/O, target control, comparator state, or application-specific states. Refer to chapter 6 for details.

A Status Block consists of four 16-bit words.

| Status Block (Write) | |
|----------------------|---------------|
| Word 0 | Reserved |
| Word 1 | Reserved |
| Word 2 | Reserved |
| Word 3 | Command value |

Figure 3-7: Control System to Device

3.2.1. Status Block Write (Control System to Device)

3.2.1.1. Optional Argument (Word 0 - 2)

Two types of command structures are supported in the Status Block Write data. In the first type, a single command is sent that requests a specific pre-configured combination of status data. In order to maintain a consistent layout (input vs. output block size), words 0 - 2 are not used.

3.2.1.2. Command Value (Word 3)

Send commands to the device on a selected channel. The value used in this word is a combination of the command and the device value. This combination informs the device where the specified command should be used when more than one measurement channel is present (for example, some terminals support multiple scales). Layout of this word is described in the Command & Response Word section.

3.2.2. Status/Command Block Read (Device to Control System)

| Status Block (Read) | |
|---------------------|------------------------|
| Word 0 | Status/Command Group 1 |
| Word 1 | Status Group 2 |
| Word 2 | Status Group 3 |
| Word 3 | Command value |

Figure 3-8: Device to Control System

3.2.2.1. Status Group 1 -3 (Word 0 - 2)

The Status/Command Group 1, Status Group 2, and Status Group 3 words contain specially grouped bits of status information. The Status/Command Value word in the Write Status Block is used by the control system to tell the device what data is desired in these words. By default, the device will send RedAlert, Scale Status Group2, and I/O Group1.

Each command used to request status data in these words represent a particular combination (refer to Appendix A, **Cyclic Commands**).

3.2.2.2. Response Value (Word 3)

Command response sent by the device to indicate the status of command execution. The structures of Response Value in the Read block and Command Value in the Write block match. The device responds as follows:

- Once the command is successfully completed by the device: command issued in the Write will have the same exact value in the Read response (command value and channel). The device begins sending data.
- In case of an error (unknown command, error condition): device responds with error code.

3.2.3. Command/Response Word

The command word and the command response word for all blocks follow the same structure:

- Bit 15 is always 0 to keep the overall value of this word positive unless the device cannot provide the desired data or comply with the requested command. Then this bit is set to a "1" to indicate an error. A full description of the possible error codes is described in the Commands section.
- Bits 11-14, used to indicate which channel of the device is providing the data. The channel indicator bits are intended for devices that support multiple sensors or scales, allowing an individual device to support up to a maximum of 16 channels. Single channel (scale or load cell) devices will always use 0000 as their default channel value.
- Bits 0-10, used to indicate the command or response value.

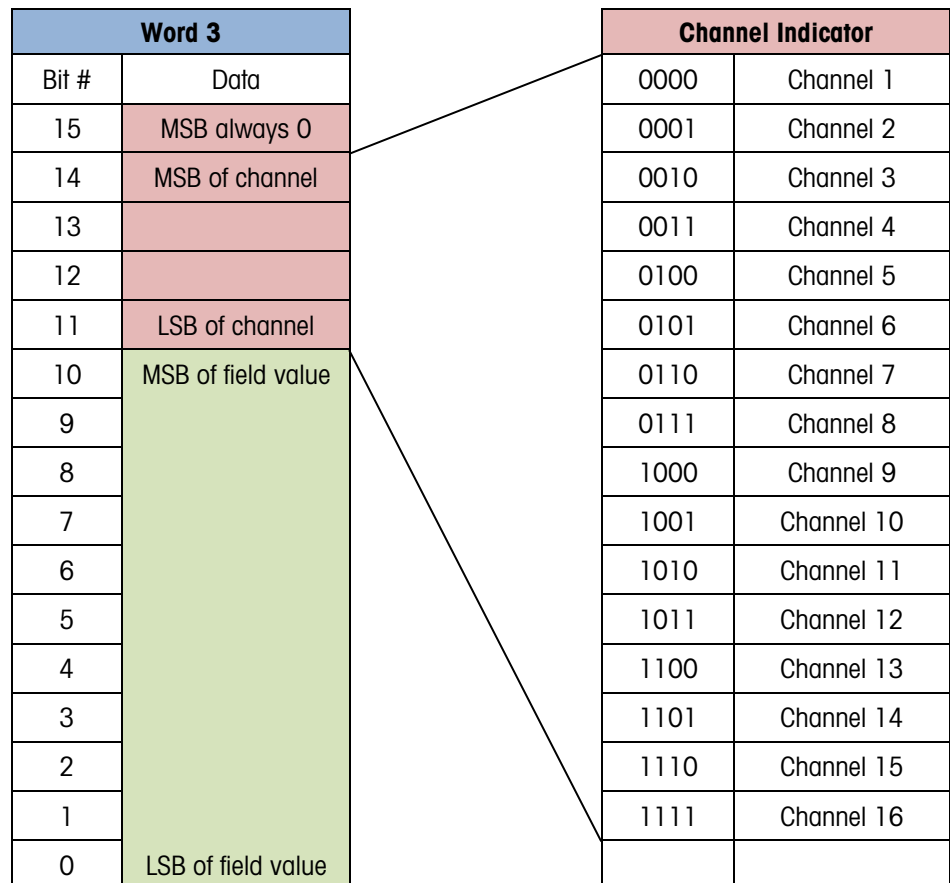


Figure 3-9: Response Value Word Layout

The Channel bits are mapped as shown above. The Command bits 0-10 are used by the control system to select the reported data from the device data in the Floating Point (Read) Block, change device data values, or issue operational commands such as tare, zero, etc. Details of this structure

are provided in the command section of this document. The Response bits 0-10 are sent by the device to indicate the status of the command or to identify the data being sent in the read block.

The value used in this word is a combination of the command/response and the channel value. This combination informs the device where the specified command should be used when more than one measurement channel is present (for example, some terminals support multiple scales) or which measurement channel is providing the data in the response.

To calculate the word's complete integer value, the command/response value should be added to the channel value:

$$\text{Command or Response Word} = \text{Command or Response Value} + \text{Channel Value}$$

3.2.3.1. Example

Send command "report rounded net weight" on channel number 3 to the device. The resulting 16-bit command word is "4098" and calculated as follows:

| | Dec | Command words bits | | | | | | | | | | | | | | | | |
|------------------|------|--------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|---|
| | | 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| Command | 2 | | | | | | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| Channel number 3 | 4096 | | 0 | 0 | 1 | 0 | | | | | | | | | | | | |
| Command word | 4098 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | |

4 Acyclic Format

The SAI Acyclic Format operates significantly different from cyclic data exchange. This format was designed for one-time read or write data that occurs outside of the normal scan cycle. This is typically used for setup data before the operation “starts” or other special information that is not needed as frequently. Acyclic messaging is typically not used for “real-time” activities and is generally used for non-repetitive or low repetition requests, from a control system. Simply stated, the control system sends a request and the device responds to the request. A device can support this format along with one or more of the cyclic data formats.

4.1. Control System Integration

The asynchronous format uses special message instructions (acyclic parameter access). The actual implementation is control system dependent. Within the configuration of the instruction, several parameters are used:

- Command type: read or write instruction
- Address: according to addressing schema of control system
Ethernet/IP: class/instance/attribute
PROFINET: slot/subslot/index
- Buffer: control system memory space or variable used with the instruction (for write value or read response)

4.2. Access Methods

There are two methods of sending/receiving data through acyclic messages:

- Direct access through a unique name or number defined by the control system's acyclic message block
Direct Access Level 1: mandatory instruction set required by devices supporting asynchronous data exchange. Mostly weight related data
Direct Access Level 2: optional instruction set extending direct access level 1. Adds application data (e.g. piece counting)
- Indirect access through a variable name provided in the data structure of two generic message blocks reserved for this purpose (one for read, one for write).

4.2.1. Direct Access

Each variable is assigned its own unique address inside the address space of the control system. Example:

| Variable | Ethernet/IP (hex) Class / instance / attribute | PROFINET (hex) Slot / subslot / index |
|---------------------|---|--|
| Report gross weight | 300 / 1 / 02 | 0 / 1 / 2001 |
| Report net weight | 300 / 1 / 04 | 0 / 1 / 2003 |

The data returned by the device to the control system is payload only, i.e. there is no special header. The control system can convert the returned data to the appropriate data type and directly use the data. See Appendix B of this document or the "Acyclic Command List" section of the corresponding product specific SAI manual for a full list of acyclic, direct access commands

4.2.2. Indirect Access

The indirect access method uses two acyclic messages with a dedicated address, one for writing and one for reading. Compared to direct access, the data point to be accessed on the device is encoded in the payload of the acyclic message. This way, access to every variable available from the device is supported. This method specifically suited for devices with a large number of variables. It uses the same structure for naming the variable as the variable block in the cyclic data's Custom Format.

| | | |
|----------|-------------------|-----------------|
| Word 0 | Read Length | |
| Word 1 | Variable Group | |
| Word 2 | Variable SubGroup | |
| Word 3 | Variable Item | |
| Word 4 | Byte 2 | Byte 1 |
| Word 5 | Byte 4 | Byte 2 |
| Word 6 | Byte 6 | Byte 3 |
| Word 7 | Byte 8 | Byte 4 |
| Word n | Byte $(2n + 2)$ | Byte $(2n + 1)$ |

Figure 4-1: Acyclic Indirect Command Response

4.3. Control System Parameters for Direct and Indirect Access

Because control systems use different methods for providing acyclic messages, the parameter used in the message block may also differ and is based on their requirements.

4.4. PROFIBUS/PROFINET Acyclic Messages

For most control systems using PROFIBUS or PROFINET networks, there are two types of commands that can be used to execute acyclic messages: RDREC (SFB52) and WRREC (SFB53). The index and length parameters within these blocks are used to specify what variable (for Direct Access) or what command (for Indirect Access) is required. To read a variable, a combination of RDREC and WRREC commands are used. To write a value to a variable, the WRREC command is used. Index numbers used are interface dependent due to the restrictions of the particular network type.

4.5. EIP/ControlNet Acyclic Messages

For control systems using EthernetIP or ControlNet networks, there are two types of commands that can be used to execute asynchronous messages: CIP Generic Message Instruction for Get Attribute Single ("e") or CIP Generic Message Instruction for Set Attribute Single ("lO"). These commands use parameters called class code, instance number, attribute number and length to be configured in the message block in order to specify what variable (for Direct Access) or what command (for Indirect Access) is required.

5 Commands

The commands available for the control system to use in Write blocks are grouped according to block type and command type. All commands are considered "one-shot" – meaning that the command is only triggered once even if it remains in the command word for multiple scans until another command is sent. In order to permit a command to be issued a second time, a no-op command is provided.

No more than one command per block is permitted (the only exception is a cancel operation command to abort the previously sent command). All other commands to this block will be ignored until its active command is completed (either successfully or not). Specifically, additional commands sent before completion are not "stored" to be acted upon completion of the prior command.

All responses to commands for the cyclic data blocks are provided through the command word. Several command values are reserved – these are used for all command types to indicate special state responses to any given command or for special sequencing commands when multiple steps are required. Bit 15 of the response word is used to indicate a failure or error in the command sent. This bit along with the value in the response bits (0-10) should be used to determine if the command was successfully executed or not; and if not, why the command failed.

The table below shows which responses are possible for each command type.

| Command Type | Valid Command Response | Invalid Command Response | Unknown Command Response | Invalid Command Data Response | Timeout Command Data Response | Aborted Command Response | Step Successful Command Response | Successful / Next Value Command Response | Step Failed Command Response |
|--------------------------|------------------------|--------------------------|--------------------------|-------------------------------|-------------------------------|--------------------------|----------------------------------|--|------------------------------|
| Floating Point Report | X | X | | | | X | X | X | X |
| Floating Point Write | X | X | X | X | | X | X | X | X |
| Floating Point Operation | X | X | X | X | X | X | X | X | X |
| Status Block | X | X | | | | X | X | X | X |
| Variable Block | X | X | X | X | X | X | X | X | X |

5.1. Command Responses

5.1.1. Valid Command Response

When a valid command is received, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command.
2. Execute the command (report the data, write the value, etc.).

3. For multiple step commands only, report "step complete" when current step is finished successfully. Report "step failed" when current step is unsuccessful. Wait for Continue or Abort command from control system to start next step. Repeat as many times as required. If the process is complete, continue to step 4.
4. Update the response word to indicate successful completion of the command.
5. Increment the sequence bits in the status word.
6. For floating point report and status block commands only, continue to update the floating point value and/or the status bits until another command is received.

5.1.2. Invalid Command Response

An invalid command response is sent when the device determines that the command is known but cannot be executed. This might occur due state restrictions – for example attempting to zero when the scale is outside of acceptable zero range. When an invalid command is received, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate invalid command (Bit 15 = 1, error value = 1, Total Response = 0x8001)
3. Increment the sequence bits in the status word.

5.1.3. Unknown Command Response

An unknown command response is sent when the device does not support this information (for example requesting rate values from a device that does not provide rate functionality). When an unsupported command is received, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate unsupported command (Bit 15 = 1, error value = 4, Total Response = 0x8004)
3. Increment the sequence bits in the status word.

5.1.4. Invalid Command Data Response

An invalid command data response is sent when a valid write command is received with an invalid argument (for example one that is smaller or larger than the allowed value). When invalid command data is received, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate invalid command value (Bit 15 = 1, error value = 8, Total Response = 0x8008)
3. Increment the sequence bits in the status word.

5.1.5. Timeout Command Response

A timeout command response is sent when the valid command that is received by the device is unable to execute within a pre-determined time. This might occur for commands that require stable weight before execution, for example. When the command timeout occurs, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate command timeout failure value (Bit 15 = 1, error value = 2, Total Response = 0x8002)
3. Increment the sequence bits in the status word.

5.1.6. Aborted Command Response

If the control system chooses to abort the previous command, it can issue a special command to cancel operation of this previous command as long as its status is in process or during any step of a multiple step sequence. The device will provide a response to indicate the initial command has been aborted.

A command failed - aborted response is sent once a second command to cancel the prior command has been received and processed. This can only occur if the original command 1) permits cancellation, 2) has not already completed successfully, and 3) has not already failed. When the command abort occurs, the device should do the following:

1. Place the "abort in process" response (2004) in the response word and begin processing the command ...
2. Update the response word to indicate command abort failure value (Bit 15 = 1, error value = 16, Total Response = 0x8010)
3. Increment the sequence bits in the status word.

Refer to Appendix A for a flow chart of the abort process and Appendix B for an example of this command sequence..

Commands for procedures that have multiple steps have some special responses:

5.1.7. Step Successful Command Response

A step successful command response is sent when the device determines that the command's current step has been successfully executed and requires acknowledgement to start the next step in the process. When a previous step completes and the next step in the sequence must be "started", the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate step successful when ready to execute next step (value = 2046)
3. Increment the sequence bits in the status word.

The control system should send one of the next step commands to continue with the next step in the process.

5.1.8. Step Successful / Next Value Command Response

A step successful / next value command response is sent when the device determines that the command's current step has been successfully executed and requires the next value and acknowledgement to start the next step in the process. When a previous step completes and the next step in the sequence requires another value to execute the next step in the process, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate step successful and next value is needed when ready to execute next step (value = 2045)
3. Increment the sequence bits in the status word.

The control system should send one of the next step commands with the required value in the floating point words to continue with the next step in the process.

5.1.9. Step Failed Command Response

A step failed command response is sent when the device determines that the command's current step has failed. Once the current step executes and fails, the device should do the following:

1. Place the "in process" response in the response word and begin processing the command ...
2. Update the response word to indicate step failed (Bit 15 = 1, error value = 32, Total Response = 0x8020)
3. Increment the sequence bits in the status word.

At this point the control system will need to decide whether to abort the sequence (command = 2004), to retry the step (command = 2005), or to skip this step (command = 2006) and try to perform the next step. Not all processes will allow all three of these options – those not permitted will have an invalid command response when the next step command is sent.

5.2. Floating Point block Commands

There are three types of floating point commands: report, write or operation. Each type is based on the command use and the expected response from the device. See Appendix A for the full list of possible commands.

5.2.1. Floating Point Report Commands

Floating Point Report Commands are used to request the data that is sent by the device in its read floating point block. These commands are used to get continuously updated data for information such as gross weight, net weight, raw weight, rate, or other application numeric data values that are time critical.

5.2.2. Floating Point Write Commands

Floating Point Write Commands are used to write a value provided by the Floating Point (write) Block to selected device functions. These commands are used to set common process values such as tare, target, tolerance, or other application numeric data values that are user-specified during operation. Due to speed issues, the actions triggered by these commands may, or may not, execute immediately. This means that the process response could be seen by the control system, within the cyclic message (data) before the "commanded" action is complete.

5.2.3. Floating Point Operation Commands

Floating Point Operation Commands are used to trigger an operation. These commands may or may not require a sequence of responses. Simple operations may execute immediately and require

no additional sequence steps. More complex operations may take multiple steps to execute and require additional commands from the control system to continue through the entire process.

5.3. Status block Commands

Status block commands are only report type. They are used to request the data that is sent by the device in its status blocks. These commands are used to get continuously updated status data for scale, target, or physical discrete input/output information. If a status block command is issued and an error response is returned, best practice is to issue a no-op command for the status block so the PLC program has visibility for when the next command completes. This is necessary for the status block and not the floating block because the status block does not have sequence bits to increment each time a command is executed.

Refer to Appendix A for the full list of possible commands.

5.4. Variable Block Commands

There are only two types of variable block commands: report or write. They are both used to access information directly from the device's variables and require specific knowledge about the variable (group, subgroup, item, and usage).

6 Status/Command Block Word Types

6.1. RedAlert Alarms (default, word 0)

These status bits are critical alarm status bits sent as part of the default status block when a status block command "0" is sent. If the control system does not place any data in the Write Status/Command word, the device will send this data in word 0. For information on how a specific device implements these bits, please refer to the **RedAlert Alarms** section of the corresponding SAI device manual.

Table 6-1: RedAlert Status Bits

| Bit # | Red Alert Failures |
|-------|-------------------------------------|
| 0 | Calibration error |
| 1 | Out of A/D range over/under |
| 2 | Checksum failure |
| 3 | Weight blocked |
| 4 | Single sensor communication failure |
| 5 | Customer-defined overload |
| 6 | Customer-defined underload |
| 7 | Network failure (all cells) |
| 8 | Zero out of range |
| 9 | Symmetry errors |
| 10 | Temperature error normal range |
| 11 | Weights and measures failure |
| 12 | Foreign device detected |
| 13 | Test mode |
| 14 | Temperature error operation range |
| 15 | LC parameters block checksum error |

Bit 0 **Calibration Error:** Indicates that the weight data can no longer be trusted, could be a result of loss of calibration data or an algorithm running in the product to detect weighing irregularities

Bit 1 **Out of A/D range over/or under:** A "1" state occurs when A/D value is at or beyond its absolute maximum value or at or below its minimum value.

| | |
|--------|---|
| Bit 2 | Checksum Failure: A checksum analysis of memory does not yield the expected result. |
| Bit 3 | Weight Blocked: The weight data does not change appreciably over a defined period. The assumption is that all scales have drift and noise from the environment, a non-changing condition would indicate that the scale is not moving as expected because it is either blocked or is not responding to the MCU (under conditions when there are no proactive diagnostics available to discover the root cause). |
| Bit 4 | Load Cell / Sensor Comm Failure: Many MT products contain "Smart" or networked load cells which operate independently from the MCU. This bit changes state when one or more sensors fail to respond to the MCU. |
| Bit 5 | Customer-Defined Overload: The weight is equal to or greater than a "customer-programmed" limit either on the scale or individual sensor's capacity (in a multi-sensor system). Overload is a conditional limit but in many cases can lead to catastrophic errors such as mechanical breakage or personal injury |
| Bit 6 | Customer-Defined Under load: The weight is under the "customer-programmed" limit on the scale / sensor (under zero but still within A/D range) |
| Bit 7 | Network Failure: On multi-cell networks indicates failure of the entire network. No Cells are responding |
| Bit 8 | Zero Out of Range: A control system, or an operator attempts a zero command and the device does not accept the command because the weight is outside of the specified (setup) limits or the weights and measures limits. This typically occurs when the user inadvertently attempts to zero the scale when the object being measured has not been removed. |
| Bit 9 | Symmetry errors: TraxDSP function that detects significant errors between load cells and their peers. |
| Bit 10 | Temperature error normal range: Many sensors and scales include sensors for temperature compensation of the weight value. These sensors can also indicate that the weight value is outside of acceptable tolerances and either the weight value could be affected or the components could prematurely fall (in extreme cases) |
| Bit 11 | Weights and Measures failure: An algorithm in the sensor or scale detects that the product is no longer in compliance with weights and measures regulations |
| Bit 12 | Incompatible Device Detected: A non-matching MT device or a foreign device is attached to the system. |
| Bit 13 | Test Mode: This bit is set to 1 when the device is in a mode in which live data is replaced with special test data. |
| Bit 14 | Temperature error operation range: This bit is set to 1 when at least one loadcell has a temperature outside the rated operation temperature range |
| Bit 15 | LC parameters block checksum error: This bit is set to 1 if a checksum error occurs with the loadcell parameters block. |

6.2. Alarm Status

These status bits are application alarm bits sent when a status block command that contains this status word is sent in the Write Status word. For information on how a specific device implements

these bits, please refer to the **Status Group 2 - Alarms** section of the corresponding SAI device manual.

Table 6-2: Application Alarm Status Bits

| Bit # | Red Alert Failures |
|-------|-------------------------|
| 0 | Rate of change error |
| 1 | Communication error |
| 2 | Over / under voltage(s) |
| 3 | Weight drift |
| 4 | Breach |
| 5 | Calibration expired |
| 6 | Application-defined |
| 7 | Application-defined |
| 8 | Application-defined |
| 9 | Application-defined |
| 1 | Application-defined |
| 1 | Application-defined |
| 1 | Application-defined |
| 1 | Application-defined |
| 1 | Application-defined |

- Bit 0 **Rate of Change:** product, application or user defines a weight / time scenario as a method of assurance that the scale is detecting weight. A typical application would be a filling system where either the material is not available for filling or a feeding system is not transporting material to the scale (slow fill timeout)
- Bit 1 **Communication errors:** relates to a device that is connected to a sensor or terminal and the necessary communication is not functioning according to specification
- Bit 2 **Over or Under Voltage (s):** relates to a device that supports dynamic measurement of system power. For example - POWERCELL CAN network voltages
- Bit 3 **Weight Drift:** relates typically to a strain gage sensor that either has a broken bridge or is damaged by water or lightning. The drift weight vs. time is outside of acceptable tolerances.
- Bit 4 **Breach:** the sensor's enclosure has been compromised and therefore vulnerable to outside influences such as moisture / water. In most cases, a failure will occur if the breach is not corrected or the sensor is not replaced
- Bit 5 **Calibration Expired:** User / technician determines the maximum number of transactions, or a time limit to occur before a preventative service or recalibration – the alarm will toggle on N+1 weighing transactions
- Bit 6 – 15 Open (application defined if needed, 0 if not used). Refer to the **Application Specific Alarms** section of the corresponding SAI device manual for more information on these application specific alarms.

6.3. Scale Group 2 Status (default word 1)

These status bits are sent as part of the default status block when a status block command "0" is sent. If the control system does not place any data in the Write Status command word, the device will send this data in word 1. For information on how a specific device implements these bits, please see the "Scale Group 2" section of the corresponding SAI device manual.

Table 6-3: Scale Group 2 Status Bits

| Bit # | Red Alert Failures |
|-------|-----------------------|
| 0 | Unit bit 1 |
| 1 | Unit bit 2 |
| 2 | Unit bit 3 |
| 3 | Unit bit 4 |
| 4 | MinWeigh Error |
| 5 | Range bit 1 |
| 6 | Range bit 2 |
| 7 | In Setup |
| 8 | Power Up Zero Failure |
| 9 | GWP out of tolerance |
| 10 | Selected scale |
| 11 | Open (always 0) |
| 12 | Open (always 0) |
| 13 | Open (always 0) |
| 14 | Open (always 0) |
| 15 | Open (always 0) |

Bit 0-3 **Unit bits 1-4:** these bits are used to indicate the weight unit based on this chart:

Table 6-4: Weight Unit Bits 1 to 4

| Unit Bit 3 | Unit Bit 2 | Unit Bit 1 | Unit Bit 0 | Value |
|-------------|------------|------------|------------|----------------|
| 0 | 0 | 0 | 0 | g |
| 0 | 0 | 0 | 1 | kg |
| 0 | 0 | 1 | 0 | lb |
| 0 | 0 | 1 | 1 | t |
| 0 | 1 | 0 | 0 | ton |
| 0 | 1 | 0 | 1 | Mg |
| 0 | 1 | 1 | 0 | µg |
| 0 | 1 | 1 | 1 | Special/custom |
| 1 | 0 | 0 | 0 | oz |
| 1 | 0 | 0 | 1 | dwt |
| 1 | 0 | 1 | 0 | ozt |
| 1000 - 1111 | | | | Reserved |

- Bit 4 **MinWeigh Error:** 1 = scale below acceptable minimum weighing range
- Bit 5 & 6 **Range bits 1 & 2:** these bits are used to indicate the weight range or interval based on this chart:

Table 6-5: Weight Range Bits 1 to 2

| Bit 6 | Bit 7 | Value |
|-------|-------|------------------|
| 0 | 0 | Range/Interval 1 |
| 0 | 1 | Range/Interval 2 |
| 1 | 0 | Range/Interval 3 |
| 1 | 1 | Reserved |

- Bit 7 **In Setup:** Used to indicate when scale is in setup mode
- Bit 8 **Power up zero failure:** Used to indicate when scale has not been able to complete its power- up restore/reset of zero
- Bit 9 **GWP out of tolerance:** Used to indicate when scale has a GWP out of tolerance error
- Bit 10 **Selected Scale:** Used to indicate which scale is "selected" and therefore in focus or seen on the device display for multi-scale devices (a single channel device will always have its scale selected)
- Bit 11-15 Unused (always 0)

6.4. Target Status Group

These status bits are target application bits sent when a status block command that contains this status word in its combination is sent in the Write Status command word. For information on how a specific device implements these bits, please refer to the **Target Status Group** section of the corresponding SAI device manual.

Table 6-6: Target Status Bits

| Bit # | Data |
|-------|--------------|
| 0 | Feed |
| 1 | Fast Feed |
| 2 | Coarse Feed |
| 3 | Feed stage 2 |
| 4 | Feed stage 1 |
| 5 | Tolerance OK |
| 6 | Over Zone |
| 7 | Under Zone |
| 8 | Heavy Zone |
| 9 | Light Zone |
| 10 | Open |

Table 6-7: Target Status Bit Description

| Bit # | Description |
|---------|---|
| 0 | Feed bit: turns ON when target feed is active (before final cutoff) |
| 1 | Fast Feed bit: turns ON when target fast feed is active |
| 2 | Coarse Feed bit: optional 3rd speed, turns ON when coarse feed is active (prior to fast feed) |
| 3 | Feed Stage 2 |
| 4 | Feed Stage 1 |
| 5 | Tolerance OK: turns ON when weight value is within tolerance of target |
| 6 | Over: turns ON when weight value is in Over Zone (too high) or optionally when weight value is outside of + TOL |
| 7 | Under: turns ON when weight value is in Under Zone (too low) or optionally when weight value is outside of – TOL |
| 8 | Heavy: turns ON when weight value is in Heavy Zone (acceptable high) |
| 9 | Light: turns ON when weight value is in Light Zone (acceptable low) |
| 10 - 15 | Unused (always 0) |

6.5. Comparator Status Group(s)

These status bits are comparator application bits sent when a status block command that contains this status word in its combination is sent in the Write Status command word. For information on how a specific device implements these bits, please refer to the **Comparator Status Groups** section of the corresponding SAI device manual.

Table 6-8: Comparator Group 1 Status Bits

| Bit # | Data |
|-------|---------------|
| 0 | Comparator 1 |
| 1 | Comparator 2 |
| 2 | Comparator 3 |
| 3 | Comparator 4 |
| 4 | Comparator 5 |
| 5 | Comparator 6 |
| 6 | Comparator 7 |
| 7 | Comparator 8 |
| 8 | Comparator 9 |
| 9 | Comparator 10 |
| 10 | Comparator 11 |
| 11 | Comparator 12 |
| 12 | Comparator 13 |
| 13 | Comparator 14 |
| 14 | Comparator 15 |

| Bit # | Data |
|-------|---------------|
| 15 | Comparator 16 |

There are two groups of these bits. The second group contains the status of the remaining comparators.

Table 6-9: Comparator Group 2 Status Bits

| Bit # | Data |
|--------|---------------|
| 0 | Comparator 17 |
| 1 | Comparator 18 |
| 2 | Comparator 19 |
| 3 | Comparator 20 |
| 4 - 15 | Open |

The comparator status bits turn ON for the listed comparator when the assigned comparison logic is true. For example, if a comparator were configured for less than 100 kg, when the weight value is less than 100 kg the bit would be ON.

6.6. I/O Status Groups

There are multiple commands for input and output status words. From the device perspective, these can be separated into two categories: physical and virtual. Physical I/O can be internal or external. Devices that have no physical I/O can still have variables and logic to virtually represent inputs and outputs within the device. The I/O status groups are used to contain a combination of input and output status bits for all of these types of I/O. Since the amount of I/O varies by device, devices will provide an invalid command response to a command for any unsupported I/O groups. The status bits for group 1 are sent as part of the default status block (in word 2). There are several other commands that can be used when various combinations of these groups are needed. The Response Value should match the Command Value sent and these status bits should be sent in one of the Read Status words. For information on how a specific device implements these bits, please refer to the **I/O Status Groups** section of the corresponding SAI device manual.

Table 6-10: I/O Group 1 Status Bits

| Bit # | Data |
|-------|--------|
| 0 | In 1 |
| 1 | In 2 |
| 2 | In 3 |
| 3 | In 4 |
| 4 | In 5 |
| 5 | In 6 |
| 6 | In 7 |
| 7 | In 8 |
| 8 | Out 9 |
| 9 | Out 10 |

| Bit # | Data |
|-------|--------|
| 10 | Out 11 |
| 11 | Out 12 |
| 12 | Out 13 |
| 13 | Out 14 |
| 14 | Out 15 |
| 15 | Out 16 |

The Input status bits reflect the state of the associated input (ON when ON, OFF when OFF). The output status bits reflect the state of the associated output (ON when ON, OFF when OFF).

The layout of Groups 2 - 14 is the same as Group 1.

6.7. Custom Application Status Group

There are two commands for custom application status words. The first command requests application bits 1-16 and the second requests application bits 17-32. Devices that support 16 or fewer application bits will provide an invalid command response to a command for the second group or an empty word (all zeroes). The usage of these status bits may differ from device to device and application to application even in the same device. Each device application will provide its own documentation on the operation of these status bits.

The custom application status bits are sent when a status block command that contains one of these status words in its combination is sent in the Write Status command word. For information on how a specific device implements these bits, please refer to the **Custom Application Status Group** section of the corresponding SAI device manual.

Table 6-11: I/O Group 1 Status Bits

| Bit # | Data |
|-------|---------------|
| 0 | Custom Bit 1 |
| 1 | Custom Bit 2 |
| 2 | Custom Bit 3 |
| 3 | Custom Bit 4 |
| 4 | Custom Bit 5 |
| 5 | Custom Bit 6 |
| 6 | Custom Bit 7 |
| 7 | Custom Bit 8 |
| 8 | Custom Bit 9 |
| 9 | Custom Bit 10 |
| 10 | Custom Bit 11 |
| 11 | Custom Bit 12 |
| 12 | Custom Bit 13 |
| 13 | Custom Bit 14 |

| Bit # | Data |
|-------|---------------|
| 14 | Custom Bit 15 |
| 15 | Custom Bit 16 |

Table 6-12: I/O Group 2 Status Bits

| Bit # | Data |
|-------|---------------|
| 0 | Custom Bit 17 |
| 1 | Custom Bit 18 |
| 2 | Custom Bit 19 |
| 3 | Custom Bit 20 |
| 4 | Custom Bit 21 |
| 5 | Custom Bit 22 |
| 6 | Custom Bit 23 |
| 7 | Custom Bit 24 |
| 8 | Custom Bit 25 |
| 9 | Custom Bit 26 |
| 10 | Custom Bit 27 |
| 11 | Custom Bit 28 |
| 12 | Custom Bit 29 |
| 13 | Custom Bit 30 |
| 14 | Custom Bit 31 |
| 15 | Custom Bit 32 |

Refer to the product documentation for mapping and operation of these bits.

6.8. Last Error Message Status Group

There is a command to request the last error code message from the device. This command uses all three status bit words in the status block to report device specific error code information to the control system.

Table 6-13: Last Error Message Status Group

| | |
|--------|------------------------|
| Word 4 | Device Type |
| Word 5 | Error Type |
| Word 6 | Error Code |
| Word 7 | Response value (= 100) |

In devices that support buffering more than one error message, this command also uses optional status word 4 to indicate which error message is requested. The default value (0) in word 4 and devices that only support a single error message will report the last error code. For devices that support an error code buffer, the word 4 parameter is used to indicate which message in the buffer to report. If word 4 is set to 1, the device will report the previous error code. If word 4 is set to 2, it will send the next previous error code and continue in this manner as the value is incremented. For

information on how a specific device implements these bits, please refer to the **Last Error Message Status Group** section of the corresponding SAI device manual.

6.9. Custom Application Group 2

This group of status bits is for critical errors for individual attached devices, such as POWERCELL® load cells, that can provide separate status information.

If fewer than the maximum possible number of attached devices are in use, any unused values should be set to 0. For information on how a specific device implements these bits, please refer to the **Custom Application Group 2** section of the corresponding SAI device manual.

Each status bit should be 0 when no error has occurred and 1 when an error is present.

Table 6-14: Custom Application Group 2

| Bit # | Attached Device (Load Cell) Error |
|-------|-----------------------------------|
| 0 | Device 1 |
| 1 | Device 2 |
| 2 | Device 3 |
| 3 | Device 4 |
| 4 | Device 5 |
| 5 | Device 6 |
| 6 | Device 7 |
| 7 | Device 8 |
| 8 | Device 9 |
| 9 | Device 10 |
| 10 | Device 11 |
| 11 | Device 12 |
| 12 | Device 13 |
| 13 | Device 14 |
| 14 | Device 15 |
| 15 | Device 16 |

7 Test Mode

In test mode, the device sends predefined data and allows toggling certain bits for verification purposes. The test mode can be also used to confirm that the device is communicating with the control system even when other parts of the device may not be configured or operational (for example, the scale not yet calibrated).

The Test Mode bit (13) of the RedAlert status bits will turn on and the Data OK bit will turn off when the device is in the test mode. In this state, the device will also accept the other test commands and respond with pre-defined test data instead of normal operational data. For example, when in test mode, the device will respond with an expected fixed weight value instead of the actual weight so that test scripts can be written against these known values. This also provides methods for the control system to force status bits to a specific state for the same reason.

7.1. Entering Test Mode

To enter test mode, the control system must place the value 2.76 (0x4030a3d7) in the floating point write value and send 80h in all four bytes of word 2 and word 3:

| | | |
|--------|-----------------|-------------|
| Word 0 | FP value = 2.76 | |
| Word 1 | | |
| Word 2 | Byte 2 = 80 | Byte 1 = 80 |
| Word 3 | Byte 4 = 80 | Byte 3 = 80 |

Figure 7-1: Test Mode Entry

7.2. Exit Test Mode

Once the command to exit test mode is received or the device's power is cycled, it will revert to sending operational data instead of test data.

To exit test mode, the control system must send the Exit Test Mode command by sending 88h in both bytes of Word 3.

| | | |
|--------|-----------------|--------------|
| Word 0 | FP value = 2.76 | |
| Word 1 | | |
| Word 2 | Byte 2 = N/A | Byte 1 = N/A |
| Word 3 | Byte 4 = 88h | Byte 3 = 88 |

Figure 7-2: Test Mode Exit

If the Custom Format is used and there is no Floating Point Block, the Test Command can be sent through the Write Status Block. If both block types are present, only one command in one of the blocks is required to place the device in or out of test mode.

7.3. Byte Order Selection

If the device is configured to use automatic byte order selection, it will automatically determine which byte order the control system is using. Once the byte order has been determined through a patented process ([US10025555B2](#)) it changes its configuration to match and should then respond to the command in same order as the command data. The response will include the sent value (2.76) so that the control system can verify that the device is sending data in the expected order. All SAI devices implementing this test feature must include patent marking information with reference to this patent number.

If a fixed byte order is selected, the control system is responsible for sending the data in the correct byte order. If there is a byte order mismatch, communication in test mode may not be possible.

7.4. Floating Point Test Commands for "Report" values

If the control system sends a floating point command to report a value when in test mode, the device will respond with fixed data instead of valid data (it may be unable to provide valid data during the test). A simple formula will be used to calculate what this fixed value should be:

$$5000.11 + \text{Command Value} = \text{Reported data value for command}$$

For example, if the command to report gross weight is sent after the device is placed in test mode, the device should provide the proper response value and its floating point value for gross weight should be 5000.11 (5000.11 + 0 = 5000.11). If Net weight is requested, the device should send 5003.11 (5000.11 + 3 = 5003.11).

7.5. Floating Point Test Commands for Scale Status bits

Special test commands are available to turn on and off the scale status bits provided in the floating point blocks when the device has been placed in test mode. See the list provided in the Commands section.

7.6. Status Block Test Commands

Special test commands are available to turn on and off the status bits provided in the status blocks when the device has been placed in test mode. A special error response is provided when a test command is sent for status bits that have not been requested. See the list provided in the Commands section.

7.7. Test Variables for Acyclic and Variable Block Test Commands

Acyclic and variable block test commands do not require the device to be in test mode. To test the acyclic Direct Access Level 1 variables, a reserved index or class/instance/attribute is used for each type of variable (See Commands). Each read variable always returns the same value and does not permit a new value to be written to it. Each write variable expects a specific value to be sent and will return an error state if any other value is sent. No special test command is assigned for Direct Access Level 2 variables.

For Indirect Access and Variable Block tests, two reserved variable names are assigned (see Commands). These variables have no other functional purpose except to respond to test read and test write commands. Each read variable always returns the same value and does not permit a new value to be written to it. Each write variable expects a specific value to be sent and will return an error state if any other value is sent.

7.8. Performance Test Commands

A special command to permit performance testing of the device's data update through the fieldbus interface is provided through the floating point data block. This command (1912) switches the device into a special mode to send a timer count at the rate specified by the value in the floating point write (0 = at A/D rate, 1 = at 1 msec rate, n = n msec rate). If the device is generally not capable of providing this data or data at a particular speed, it will return a response of invalid command. It still may be possible that the device cannot reach the desired speed due to other system activity.

Once in this mode, the device will then report the updated count value in the floating point read block at the specified rate.

A. Cyclic Commands

The commands available for the control system to use in Write blocks are grouped according to block type and command type. All commands are considered "one-shot" – meaning that the command is only triggered once even if it remains in the command word for multiple scans until another command is sent. In order to permit a command to be issued a second time, a no-op command is provided.

All responses to commands for the cyclic data blocks are provided in the same manner – through the field indicator / channel value. During the process of receiving and executing the operation required for a command, the device will follow a defined sequence, as shown in Figure A-1.

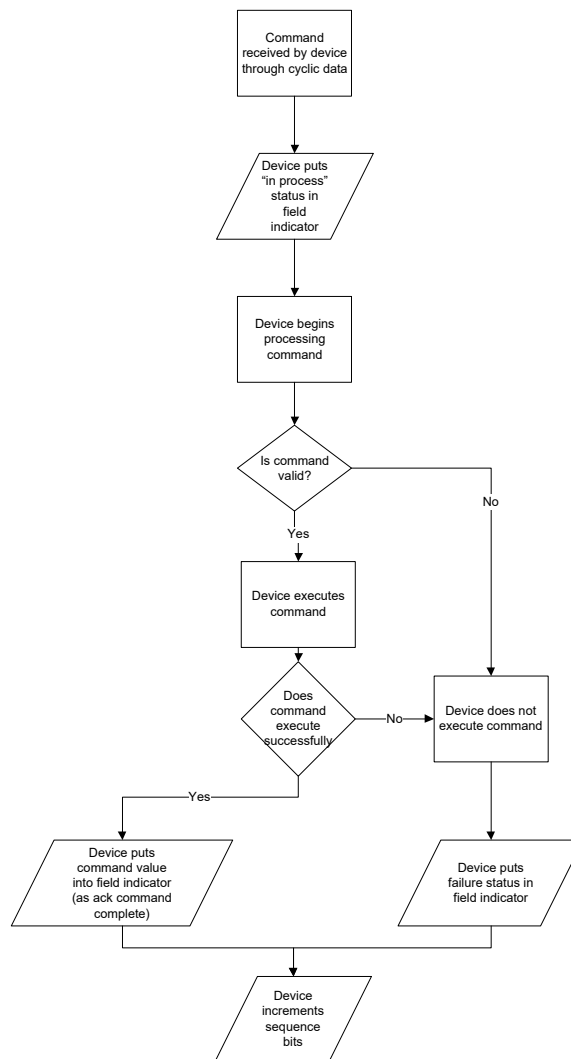


Figure A-1: Receiving and Executing a Command

If a command is aborted, it will follow the sequence shown in Figure A-2.

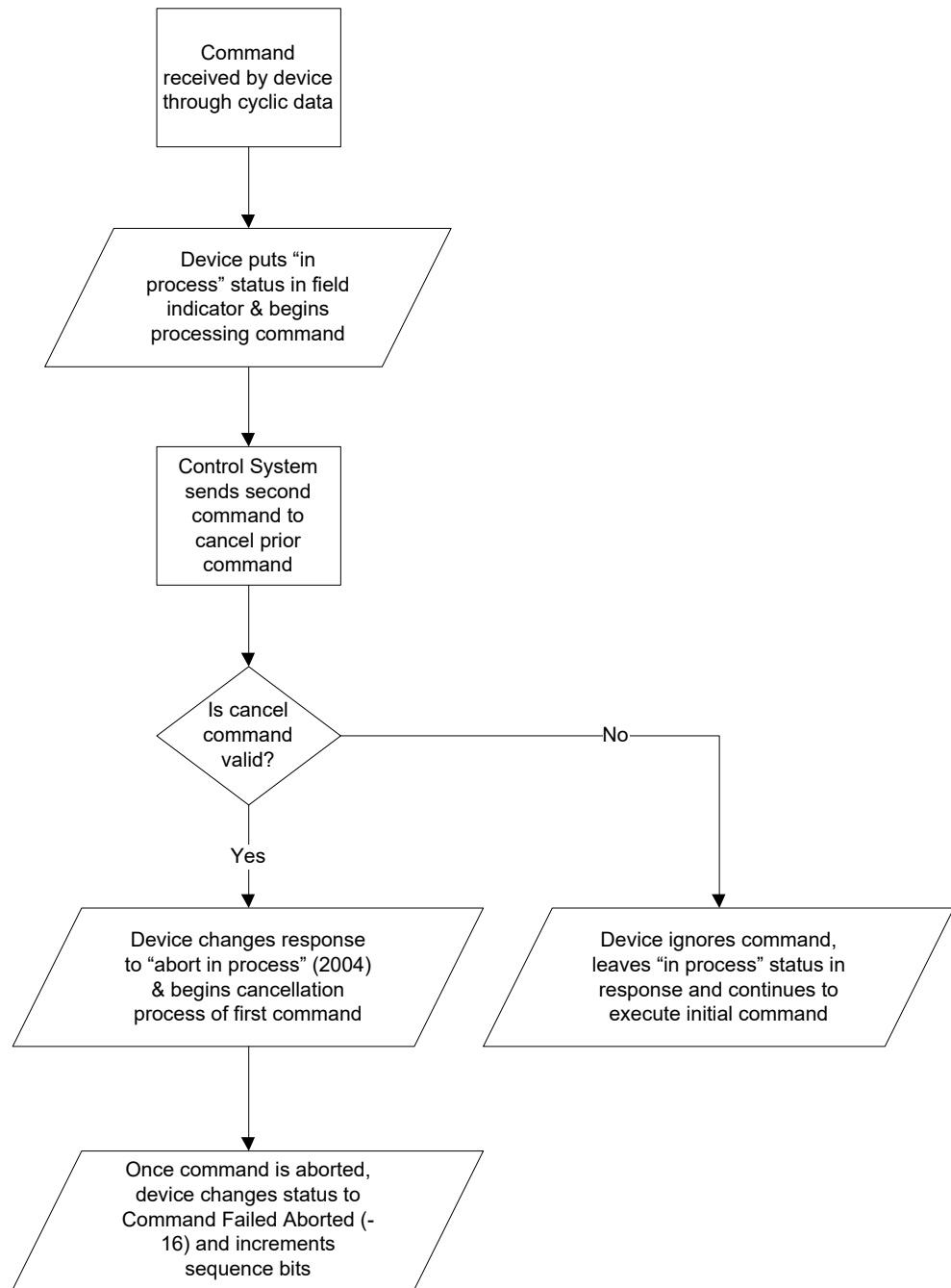


Figure A-2: Aborted Command Sequence

For multi-step commands, special commands are used to advance to the next step, to retry a failed step, or to abort the entire process. There are also corresponding status values to indicate when the device is ready to execute the next step or when the step has failed.

During a stepped procedure, the commands follow the procedure shown in Figure A-3.

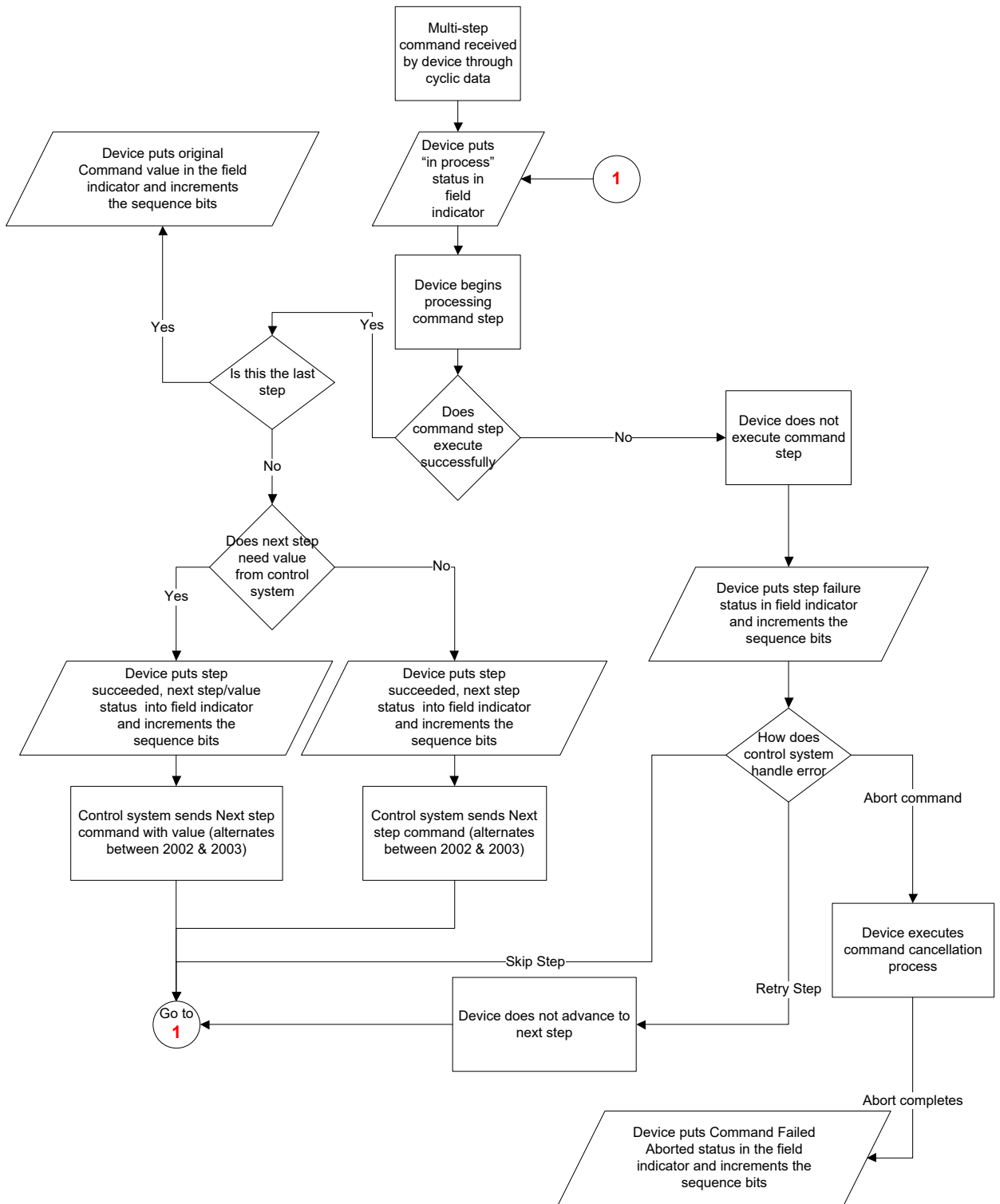


Figure A-3: Stepped Command Procedure

A.1. Special Commands

For all block types, there are some reserved commands used for unique status or command sequencing.

| Command | Description | Value |
|----------------|---|---|
| Test Command | Used to determine communication works and byte/word order & enter test mode | FP = 2.76 Mask = 80h, 80h Cmd = 80h, 80h Bit 15 = 1 + ... 2185 for both words |
| Exit Test Mode | Used to return to normal operation | FP & Mask = 0 Cmd = 88h, 88h Bit 15 = 1 + ... 6553 |
| Response only | These are used to indicate failed command status through the floating point field ... each of these responses turns on bit 15 and the bit value shown in () | Bit 15 = 1 + ... |
| Response only | Command Failed Invalid | (1) |
| Response only | Command Failed Timeout | (2) |
| Response only | Command Failed Unknown | (4) |
| Response only | Command Failed Value Invalid | (8) |
| Response only | Command Failed Aborted | (16) |
| Response only | Command Step Failed | (32) |
| Response only | Test Command Failed | (64) |

A.2. General System Commands

| Command | Description | Value |
|------------------|---|-------|
| Response only | Command has been received and is being evaluated (in process) | 2047 |
| Response only | Step Successful | 2046 |
| Response only | Step Successful, Next Value | 2045 |
| Response only | Calibration successful but unstable | 2044 |
| NOOP 1 | No operation command – used to test command response, clear prior command, etc. | 2000 |
| Next step 1 | Continue to next step in sequence | 2002 |
| Next step 2 | Continue to next step in sequence | 2003 |
| Cancel operation | Abort sequence ... response value means abort in process | 2004 |
| Retry Step | After step failure, retries previous step in sequence | 2005 |
| Skip Step | After step failure, skips step and advances to next in sequence | 2006 |

A.3. Floating Point block Commands

A.3.1. Weight Report Commands

| Command | Description | Value |
|-----------------------------|---|-------|
| Report Default Value | For terminals & sensors this is Gross weight data in displayed resolution | 0 |
| Report Rounded Gross Weight | Gross Weight data is displayed resolution | 1 |
| Report Rounded Tare Weight | Tare weight data in displayed resolution | 2 |
| Report Rounded Net Weight | Net weight data in displayed resolution | 3 |
| Report Rounded Rate | Rate (change in gross weight over time) in displayed resolution | 4 |
| Report Gross Weight | Gross weight data in internal resolution | 5 |
| Report Tare Weight | Tare weight data in internal resolution | 6 |
| Report Net Weight | Net weight data in internal resolution | 7 |
| Report Rate | Rate (change in gross weight over time) in internal resolution | 8 |
| Report Weight Units | | 9 |
| Report raw counts | Unprocessed weight data (no filter or unit calculation) | 10 |
| Report Peak Weight | Peak Weight application value in displayed resolution | 11 |
| Report raw load cell counts | Unprocessed load cell weight data in counts. The channel value in the command is issued to indicate the load cell number. | 12 |
| Report Fast Gross Weight | Gross weight at fast update | 13 |
| Report Fast Net Weight | Net weight at fast update | 14 |

A.3.2. Target Report Commands

| Command | Description | Value |
|-----------------------------|-------------------------------|-------|
| Report Target Weight | | 20 |
| Report Fine Feed Weight | | 21 |
| Report Coarse Feed | Coarse Feed Time/Weight value | 22 |
| Report Stage 2 Weight | | 23 |
| Report Stage 1 Weight | | 24 |
| Report Spill Weight | | 25 |
| Report (-) Tolerance Weight | | 26 |
| Report (+) Tolerance Weight | | 27 |
| Report (-) Tolerance % | | 28 |
| Report (+) Tolerance % | | 29 |

A.3.3. Zone Report Commands

| Command | Description | Value |
|--------------------------|-------------|-------|
| Report Over Zone Weight | | 30 |
| Report Heavy Zone Weight | | 31 |
| Report Light Zone Weight | | 32 |
| Report Under Zone Weight | | 33 |

A.3.4. Comparator Report Commands

| Command | Description | Value |
|---------------------------------|-------------|-------|
| Report Comparator 1 Limit | | 40 |
| Report Comparator 1 High Limit | | 41 |
| Report Comparator 2 Limit | | 42 |
| Report Comparator 2 High Limit | | 43 |
| Report Comparator 3 Limit | | 44 |
| Report Comparator 3 High Limit | | 45 |
| Report Comparator 4 Limit | | 46 |
| Report Comparator 4 High Limit | | 47 |
| Report Comparator 5 Limit | | 48 |
| Report Comparator 5 High Limit | | 49 |
| Report Comparator 6 Limit | | 50 |
| Report Comparator 6 High Limit | | 51 |
| Report Comparator 7 Limit | | 52 |
| Report Comparator 7 High Limit | | 53 |
| Report Comparator 8 Limit | | 54 |
| Report Comparator 8 High Limit | | 55 |
| Report Comparator 9 Limit | | 56 |
| Report Comparator 9 High Limit | | 57 |
| Report Comparator 10 Limit | | 58 |
| Report Comparator 10 High Limit | | 59 |
| Report Comparator 11 Limit | | 60 |
| Report Comparator 11 High Limit | | 61 |
| Report Comparator 12 Limit | | 62 |
| Report Comparator 12 High Limit | | 63 |
| Report Comparator 13 Limit | | 64 |
| Report Comparator 13 High Limit | | 65 |
| Report Comparator 14 Limit | | 66 |
| Report Comparator 14 High Limit | | 67 |
| Report Comparator 15 Limit | | 68 |
| Report Comparator 15 High Limit | | 69 |

| Command | Description | Value |
|---------------------------------|-------------|-------|
| Report Comparator 16 Limit | | 70 |
| Report Comparator 16 High Limit | | 71 |
| Report Comparator 17 Limit | | 72 |
| Report Comparator 17 High Limit | | 73 |
| Report Comparator 18 Limit | | 74 |
| Report Comparator 18 High Limit | | 75 |
| Report Comparator 19 Limit | | 76 |
| Report Comparator 19 High Limit | | 77 |
| Report Comparator 20 Limit | | 78 |
| Report Comparator 20 High Limit | | 79 |

A.3.5. Miscellaneous Report Commands

| Command | Description | Value |
|-------------------------------------|--|-------|
| Report Transaction # | | 80 |
| Report Sequential # | | 81 |
| Report Filter Mode | | 90 |
| Report Filter Environment | | 91 |
| Report Low Pass Frequency | | 92 |
| Report Notch Filter | | 93 |
| Report Stability Filter | | 94 |
| Report Zero register | | 95 |
| Report current weight readability | | 96 |
| Report temperature | Temperature of sensor relevant for weight result | 97 |
| Report Fast Weight Filter Frequency | Refer to SICS FCUTD | 98 |

A.3.6. Custom Application Report Commands

| Command | Description | Value |
|------------------------|---------------------------|-------|
| Report Custom Field 1 | Application Data Field 1 | 101 |
| Report Custom Field 2 | Application Data Field 2 | 102 |
| Report Custom Field 3 | Application Data Field 3 | 103 |
| Report Custom Field 4 | Application Data Field 4 | 104 |
| Report Custom Field 5 | Application Data Field 5 | 105 |
| Report Custom Field 6 | Application Data Field 6 | 106 |
| Report Custom Field 7 | Application Data Field 7 | 107 |
| Report Custom Field 8 | Application Data Field 8 | 108 |
| Report Custom Field 9 | Application Data Field 9 | 109 |
| Report Custom Field 10 | Application Data Field 10 | 110 |
| Report Custom Field 11 | Application Data Field 11 | 111 |

| Command | Description | Value |
|------------------------|---------------------------|-------|
| Report Custom Field 12 | Application Data Field 12 | 112 |
| Report Custom Field 13 | Application Data Field 13 | 113 |
| Report Custom Field 14 | Application Data Field 14 | 114 |
| Report Custom Field 15 | Application Data Field 15 | 115 |
| Report Custom Field 16 | Application Data Field 16 | 116 |
| Report Custom Field 17 | Application Data Field 17 | 117 |
| Report Custom Field 18 | Application Data Field 18 | 118 |
| Report Custom Field 19 | Application Data Field 19 | 119 |
| Report Custom Field 20 | Application Data Field 20 | 120 |

A.3.7. Weight Write Immediate Commands

| Command | Description | Value |
|--------------------------|------------------------------------|-------|
| Write Preset Tare Weight | Sets Preset Tare to Value provided | 201 |

A.3.8. Target Write Immediate Commands

| Command | Description | Value |
|----------------------------|-------------------------------|-------|
| Write Target Weight | | 220 |
| Write Fine Feed Weight | | 221 |
| Write Coarse Feed | Coarse Feed Time/Weight value | 222 |
| Write Stage 2 Weight | | 223 |
| Write Stage 1 Weight | | 224 |
| Write Spill Weight | | 225 |
| Write (-) Tolerance Weight | | 226 |
| Write (+) Tolerance Weight | | 227 |
| Write (-) Tolerance % | | 228 |
| Write (+) Tolerance % | | 229 |

A.3.9. Zone Write Immediate Commands

| Command | Description | Value |
|-------------------------|-------------|-------|
| Write Over Zone Weight | | 230 |
| Write Heavy Zone Weight | | 231 |
| Write Light Zone Weight | | 232 |
| Write Under Zone Weight | | 233 |

A.3.10. Comparator Write Immediate Commands

| Command | Description | Value |
|-------------------------------|-------------|-------|
| Write Comparator 1 Limit | | 240 |
| Write Comparator 1 High Limit | | 241 |
| Write Comparator 2 Limit | | 242 |

| Command | Description | Value |
|--------------------------------|--------------------|--------------|
| Write Comparator 2 High Limit | | 243 |
| Write Comparator 3 Limit | | 244 |
| Write Comparator 3 High Limit | | 245 |
| Write Comparator 4 Limit | | 246 |
| Write Comparator 4 High Limit | | 247 |
| Write Comparator 5 Limit | | 248 |
| Write Comparator 5 High Limit | | 249 |
| Write Comparator 6 Limit | | 250 |
| Write Comparator 6 High Limit | | 251 |
| Write Comparator 7 Limit | | 252 |
| Write Comparator 7 High Limit | | 253 |
| Write Comparator 8 Limit | | 254 |
| Write Comparator 8 High Limit | | 255 |
| Write Comparator 9 Limit | | 256 |
| Write Comparator 9 High Limit | | 257 |
| Write Comparator 10 Limit | | 258 |
| Write Comparator 10 High Limit | | 259 |
| Write Comparator 11 Limit | | 260 |
| Write Comparator 11 High Limit | | 261 |
| Write Comparator 12 Limit | | 262 |
| Write Comparator 12 High Limit | | 263 |
| Write Comparator 13 Limit | | 264 |
| Write Comparator 13 High Limit | | 265 |
| Write Comparator 14 Limit | | 266 |
| Write Comparator 14 High Limit | | 267 |
| Write Comparator 15 Limit | | 268 |
| Write Comparator 15 High Limit | | 269 |
| Write Comparator 16 Limit | | 270 |
| Write Comparator 16 High Limit | | 271 |
| Write Comparator 17 Limit | | 272 |
| Write Comparator 17 High Limit | | 273 |
| Write Comparator 18 Limit | | 274 |
| Write Comparator 18 High Limit | | 275 |
| Write Comparator 19 Limit | | 276 |
| Write Comparator 19 High Limit | | 277 |
| Write Comparator 20 Limit | | 278 |
| Write Comparator 20 High Limit | | 279 |

A.3.11. Miscellaneous Write Immediate Commands

| Command | Description | Value |
|------------------------------------|----------------------|-------|
| Write Transaction # | | 280 |
| Write Sequential # | | 281 |
| Write Filter Mode | | 290 |
| Write Filter Environment | | 291 |
| Write Low Pass Frequency | | 292 |
| Write Notch Filter | | 293 |
| Write Stability Filter | | 294 |
| Write Zero register | | 295 |
| Write "Weight Readability" | Refer to MT-SICS M23 | 296 |
| Write Fast Weight Filter Frequency | Refer to SICS FCUTD | 298 |

A.3.12. Custom Application Write Immediate Commands

| Command | Description | Value |
|-----------------------|---------------------------|-------|
| Write Custom Field 1 | Application Data Field 1 | 301 |
| Write Custom Field 2 | Application Data Field 2 | 302 |
| Write Custom Field 3 | Application Data Field 3 | 303 |
| Write Custom Field 4 | Application Data Field 4 | 304 |
| Write Custom Field 5 | Application Data Field 5 | 305 |
| Write Custom Field 6 | Application Data Field 6 | 306 |
| Write Custom Field 7 | Application Data Field 7 | 307 |
| Write Custom Field 8 | Application Data Field 8 | 308 |
| Write Custom Field 9 | Application Data Field 9 | 309 |
| Write Custom Field 10 | Application Data Field 10 | 310 |
| Write Custom Field 11 | Application Data Field 11 | 311 |
| Write Custom Field 12 | Application Data Field 12 | 312 |
| Write Custom Field 13 | Application Data Field 13 | 313 |
| Write Custom Field 14 | Application Data Field 14 | 314 |
| Write Custom Field 15 | Application Data Field 15 | 315 |
| Write Custom Field 16 | Application Data Field 16 | 316 |
| Write Custom Field 17 | Application Data Field 17 | 317 |
| Write Custom Field 18 | Application Data Field 18 | 318 |
| Write Custom Field 19 | Application Data Field 19 | 319 |
| Write Custom Field 20 | Application Data Field 20 | 320 |

A.3.13. Weight Operation Immediate Commands

| Command | Description | Value |
|---------|---------------------------------|-------|
| Tare | Tare executed with motion check | 400 |

| Command | Description | Value |
|----------------|---|-------|
| Zero | Zero executed with motion check | 401 |
| Clear Tare | Motion not checked, clear tare executed | 402 |
| Tare Immediate | Motion not checked, tare executed | 403 |
| Zero Immediate | Motion not checked, zero executed | 404 |

A.3.14. Print / Communication Operation Immediate Commands

| Command | Description | Value |
|------------|--|-------|
| Print | Demand Print executed | 410 |
| Trigger 1 | Custom communication trigger 1 executed | 411 |
| Trigger 2 | Custom communication trigger 2 executed | 412 |
| Trigger 3 | Custom communication trigger 3 executed | 413 |
| Trigger 4 | Custom communication trigger 4 executed | 414 |
| Trigger 5 | Custom communication trigger 5 executed | 415 |
| Trigger 6 | Custom communication trigger 6 executed | 416 |
| Trigger 7 | Custom communication trigger 7 executed | 417 |
| Trigger 8 | Custom communication trigger 8 executed | 418 |
| Trigger 9 | Custom communication trigger 9 executed | 419 |
| Trigger 10 | Custom communication trigger 10 executed | 420 |

A.3.15. Miscellaneous Operation Immediate Commands

| Command | Description | Value |
|---------------------|-------------|-------|
| Clear Alarm | | 430 |
| Clear Transaction # | | 431 |
| Clear Sequential # | | 432 |
| Clear Peak Weight | | 433 |
| Apply Filter | | 434 |

A.3.16. Target Operation Immediate Commands

| Command | Description | Value |
|-------------------|-------------|-------|
| Start Target | | 500 |
| Pause Target | | 501 |
| Resume Target | | 502 |
| Abort Target | | 503 |
| Apply Target | | 504 |
| Apply Comparators | | 510 |

A.3.17. Display / Keyboard Operation Immediate Commands

| Command | Description | Value |
|-----------------------|-------------|-------|
| Clear Display Message | | 600 |

| Command | Description | Value |
|------------------------|---|-------|
| Display Message 1 | | 601 |
| Display Message 2 | | 602 |
| Display Message 3 | | 603 |
| Display Message 4 | | 604 |
| Display Message 5 | | 605 |
| Display Message 6 | | 606 |
| Display Message 7 | | 607 |
| Display Message 8 | | 608 |
| Display Message 9 | | 609 |
| Display Message 10 | | 610 |
| Disable Weight Display | | 630 |
| Enable Weight Display | | 631 |
| Disable Keypad | | 632 |
| Enable Keypad | | 633 |
| Reset Enter key bit | | 634 |
| Select Scale | Changes the weight display focus and "selects" scale (bit in scale status becomes true) – uses channel in command to indicate which scale is selected | 640 |

A.3.18. Select Weight Unit Commands

| Command | Description | Value |
|-----------------------|---|-------|
| Select Unit | Switches to unit selected in parameter (uses same values as found in report weight units command 9) | 700 |
| Switch to first unit | Changes to primary unit | 701 |
| Switch to second unit | Changes to secondary unit | 702 |
| Switch to third unit | Changes to third unit | 703 |

A.3.19. Dynamic Application Commands

| Command | Description | Value |
|---------------------------|--|-------|
| Last ProcessWeight | Last dynamic weight value | 800 |
| Object length | Calculated object length | 801 |
| Report Dynamic weighments | Calculated number of dynamic weighments (counts integer) | 802 |
| Report Max Dynamic | Maximum dynamic weight value | 803 |
| Report Min Dynamic | Minimum dynamic weight value | 804 |
| Report Average Dynamic | Mean dynamic weight value | 805 |
| Report Standard Deviation | Standard deviation of last 20 dynamic weighments | 806 |

A.3.20. Discrete Output Operation Immediate Commands

| Command | Description | Value |
|--|------------------------|-------|
| Turn all internal & external outputs OFF | Forces all outputs OFF | 1000 |
| Turn 1st internal Output 1 OFF | | 1001 |
| Turn 1st internal Output 2 OFF | | 1002 |
| Turn 1st internal Output 3 OFF | | 1003 |
| Turn 1st internal Output 4 OFF | | 1004 |
| Turn 1st internal Output 5 OFF | | 1005 |
| Turn 1st internal Output 6 OFF | | 1006 |
| Turn 1st internal Output 7 OFF | | 1007 |
| Turn 1st internal Output 8 OFF | | 1008 |
| Turn 2nd internal Output 1 OFF | | 1009 |
| Turn 2nd internal Output 2 OFF | | 1010 |
| Turn 2nd internal Output 3 OFF | | 1011 |
| Turn 2nd internal Output 4 OFF | | 1012 |
| Turn 2nd internal Output 5 OFF | | 1013 |
| Turn 2nd internal Output 6 OFF | | 1014 |
| Turn 2nd internal Output 7 OFF | | 1015 |
| Turn 2nd internal Output 8 OFF | | 1016 |
| Turn 1st external Output 1 OFF | | 1017 |
| Turn 1st external Output 2 OFF | | 1018 |
| Turn 1st external Output 3 OFF | | 1019 |
| Turn 1st external Output 4 OFF | | 1020 |
| Turn 1st external Output 5 OFF | | 1021 |
| Turn 1st external Output 6 OFF | | 1022 |
| Turn 1st external Output 7 OFF | | 1023 |
| Turn 1st external Output 8 OFF | | 1024 |
| Turn 2nd external Output 1 OFF | | 1025 |
| Turn 2nd external Output 2 OFF | | 1026 |
| Turn 2nd external Output 3 OFF | | 1027 |
| Turn 2nd external Output 4 OFF | | 1028 |
| Turn 2nd external Output 5 OFF | | 1029 |
| Turn 2nd external Output 6 OFF | | 1030 |
| Turn 2nd external Output 7 OFF | | 1031 |
| Turn 2nd external Output 8 OFF | | 1032 |
| Turn 1st internal Output 1 ON | | 1101 |
| Turn 1st internal Output 2 ON | | 1102 |
| Turn 1st internal Output 3 ON | | 1103 |

| Command | Description | Value |
|-------------------------------|-------------|-------|
| Turn 1st internal Output 4 ON | | 1104 |
| Turn 1st internal Output 5 ON | | 1105 |
| Turn 1st internal Output 6 ON | | 1106 |
| Turn 1st internal Output 7 ON | | 1107 |
| Turn 1st internal Output 8 ON | | 1108 |
| Turn 2nd internal Output 1 ON | | 1109 |
| Turn 2nd internal Output 2 ON | | 1110 |
| Turn 2nd internal Output 3 ON | | 1111 |
| Turn 2nd internal Output 4 ON | | 1112 |
| Turn 2nd internal Output 5 ON | | 1113 |
| Turn 2nd internal Output 6 ON | | 1114 |
| Turn 2nd internal Output 7 ON | | 1115 |
| Turn 2nd internal Output 8 ON | | 1116 |
| Turn 1st external Output 1 ON | | 1117 |
| Turn 1st external Output 2 ON | | 1118 |
| Turn 1st external Output 3 ON | | 1119 |
| Turn 1st external Output 4 ON | | 1120 |
| Turn 1st external Output 5 ON | | 1121 |
| Turn 1st external Output 6 ON | | 1122 |
| Turn 1st external Output 7 ON | | 1123 |
| Turn 1st external Output 8 ON | | 1124 |
| Turn 2nd external Output 1 ON | | 1125 |
| Turn 2nd external Output 2 ON | | 1126 |
| Turn 2nd external Output 3 ON | | 1128 |
| Turn 2nd external Output 4 ON | | 1129 |
| Turn 2nd external Output 5 ON | | 1130 |
| Turn 2nd external Output 6 ON | | 1131 |
| Turn 2nd external Output 7 ON | | 1132 |
| Turn 2nd external Output 8 ON | | 1133 |

A.3.21. Custom Application Operation Immediate Commands

| Command | Description | Value |
|------------------------------|-------------------------------|-------|
| Custom Application Command 1 | Application Command trigger 1 | 1201 |
| Custom Application Command 2 | Application Command trigger 2 | 1202 |
| Custom Application Command 3 | Application Command trigger 3 | 1203 |
| Custom Application Command 4 | Application Command trigger 4 | 1204 |
| Custom Application Command 5 | Application Command trigger 5 | 1205 |
| Custom Application Command 6 | Application Command trigger 6 | 1206 |

| Command | Description | Value |
|-------------------------------|--------------------------------|-------|
| Custom Application Command 7 | Application Command trigger 7 | 1207 |
| Custom Application Command 8 | Application Command trigger 8 | 1208 |
| Custom Application Command 9 | Application Command trigger 9 | 1209 |
| Custom Application Command 10 | Application Command trigger 10 | 1210 |
| Custom Application Command 11 | Application Command trigger 11 | 1211 |
| Custom Application Command 12 | Application Command trigger 12 | 1212 |
| Custom Application Command 13 | Application Command trigger 13 | 1213 |
| Custom Application Command 14 | Application Command trigger 14 | 1214 |
| Custom Application Command 15 | Application Command trigger 15 | 1215 |
| Custom Application Command 16 | Application Command trigger 16 | 1216 |
| Custom Application Command 17 | Application Command trigger 17 | 1217 |
| Custom Application Command 18 | Application Command trigger 18 | 1218 |
| Custom Application Command 19 | Application Command trigger 19 | 1219 |
| Custom Application Command 20 | Application Command trigger 20 | 1220 |

A.3.22. Calibration Setup Commands

| Command | Description | Value |
|------------------------------|--|-------|
| Primary Unit | Configures primary weight unit – parameter value selects weight unit (same values as report command 9) | 1700 |
| Second Unit | Configures second weight unit – parameter value selects weight unit (same values as report command 9) | 1701 |
| Third Unit | Configures third weight unit – parameter value selects weight unit (same values as report command 9) | 1702 |
| Calibration Unit | Configures calibration weight unit – parameter value selects weight unit (same values as report command 9 but some restrictions may apply) | 1703 |
| Geo code | Configures Geo Code value (value between 0-31) | 1704 |
| Linearity Adjust | Configure linearity adjust setting – 0=none , 1= 3 pt, 2= 4 pt, 3= 5 pt | 1705 |
| Span Adjustment value (Xlow) | Only used for 5 pt linearity adjust | 1706 |
| Span adjustment value (Low) | Only used for 4 pt and 5 pt linearity adjust | 1707 |
| Span adjustment value (Mid) | Only used for when any linearity adjust is enabled | 1708 |
| Span adjustment value (High) | Sets highest span value – used for all linearity and span adjust procedures | 1709 |
| Span counts value (Xlow) | Used with span value xlow | 1710 |
| Span counts value (Low) | Used with span value low | 1711 |
| Span counts value (Mid) | Used with span value mid | 1712 |
| Span counts value (High) | Used with span value high | 1713 |

A.3.23. CalFREE Setup Commands

| Command | Description | Value |
|-----------------------|--|-------|
| CalFREE Cell capacity | Cell capacity value used for CalFree calibration | 1720 |
| CalFREE unit | CalFREE calibration unit | 1721 |
| CalFFREE cell output | CalFREE cell output rating | 1722 |

A.3.24. Calibration Commands

| Command | Description | Value |
|----------------------------|---|-------|
| Zero Calibration | | 1500 |
| Span Calibration | | 1501 |
| Step Calibration | Triggers start of step calibration – parameter sent includes number of calibration steps to execute | 1502 |
| Step Adjustment | Triggers step adjustment – parameter value included to indicate span value for step | 1503 |
| CalFREE Calibration | Triggers CalFREE (parameters must have already been loaded via setup commands) | 1504 |
| CalFREE Plus Calibration | Triggers CalFREE Plus calibration process | 1505 |
| Save / Confirm calibration | Saves calibration | 1506 |

A.3.25. Floating Point Status Bit Test Commands

| Command | Description | Value |
|--------------------|--|-------|
| Alarm Bit | Turn on or off bit in scale status word when in test mode | 1900 |
| Motion Bit | ... | 1901 |
| Net Mode bit | FP = 1 to set ON or FP = 0 to set OFF | 1902 |
| Center of zero bit | | 1903 |
| Alt weight bit | | 1904 |
| Device bit 1 | | 1905 |
| Device bit 2 | | 1906 |
| Device bit 3 | | 1907 |
| Device bit 4 | | 1908 |
| Device bit 5 | | 1909 |
| Device bit 6 | | 1910 |
| Device bit 7 | | 1911 |
| Performance Mode | 0 = send performance count at A/D rate 1 = send performance count at 1 msec interval <i>n</i> = send performance count at <i>n</i> msec interval | 1912 |

A.4. Status/Command block Commands

Status/Command block commands are unique to this block type and not ALL block types. So for example, a command value of 0 in the status/command block will report the default status words (RedAlert, I/O Group 1, Scale Group 2 Status bits).

A.4.1. Status Commands

| Command | Description | Value |
|---|--|-------------|
| Report Default Status words | Default status words (see below) | 0 |
| Report RedAlert Alarm, Scale Group 2, I/O group 1 | RedAlert Alarm, Scale group 2, I/O group 1 | 1 |
| Report Target / Comparator Status | Target 1, Comparator group 1, Comparator group 2 | 2 |
| Report Target mix 1 | Target 1, I/O group 1, custom group 1 | 3 |
| Report Target mix 2 | Target 1, I/O group 2, custom group 2 | 4 |
| Report Target mix 3 | Target 1, I/O group 3, I/O group 4 | 5 |
| Report Target mix 4 | Target 1, I/O group 5, I/O group 6 | 6 |
| Report Target mix 5 | Target 1, I/O group 7, I/O group 8 | 7 |
| Report Target mix 6 | Target 1, I/O group 9, I/O group 10 | 8 |
| Report I/O mix 1 | I/O group 2, I/O group 3, I/O group 4 | 9 |
| Report I/O mix 2 | I/O group 5, I/O group 6, I/O group 7 | 10 |
| Report I/O mix 3 | I/O group 8, I/O group 9, I/O group 10 | 11 |
| Report Custom / I/O mix 1 | Custom group 1, custom group 2, I/O group 1 | 12 |
| Report Custom / I/O mix 2 | Custom group 1, custom group 2, I/O group 2 | 13 |
| Report Custom / I/O mix 3 | Custom group 1, I/O group 1, I/O group 2 | 14 |
| Report Custom / I/O mix 4 | Custom group 2, I/O group 3, I/O group 4 | 15 |
| Report Comparator mix 1 | Comparator group 1, comparator group 2, I/O group 1 | 16 |
| Report Comparator mix 2 | Comparator group 1, comparator group 2, I/O group 2 | 17 |
| Report Comparator mix 3 | Comparator group 1, I/O group 1, I/O group 2 | 18 |
| Report Targets group 1 | Target 1, Target 2, Target 3 | 19 |
| Report Targets group 2 | Target 4, Target 5, Target6 | 20 |
| Report Alarms & Scale Group 2 | RedAlert Alarm, Alarm, Scale Group2 | 21 |
| RedAlert, / Comparator | RedAlert,, Comparator 1, Comparator 2 | 22 |
| Alarm / Custom mix | Alarm, I/O group 1, Custom group 1 | 23 |
| Load cell error / custom mix | Load cell group 1, load cell group 2, Custom group 1 | 24 |
| Report Last Error Code* | Error code structure | 100 |
| User selected status words | Command to indicate Words 0-2 should be read to determine which status words to send | 256 and 257 |

The default for terminals and sensors is the same as command 1. This allows devices which need to support a different default to provide it at a later date.

- * When the Report Last Error Code command is sent, an optional value in word 4 is used to indicate which error message is desired when the device supports buffering more than one error code message. The default value (0) will report the last error message. Any other value will request the message at $n +$ the value's position in the buffer. For example, if 3 is placed in the word 4 value, the device should respond with the error code that occurred three spots ahead of the last error message.

When the user selected status word commands are sent, the values below are placed in Words 0 – 2 of the Write block to indicate which status words are wanted:

| Optional status word selection value | Description |
|--------------------------------------|----------------------|
| 0 | No selection |
| 1 | Red Alert |
| 2 | Alarms |
| 3 | Scale Status Group 2 |
| 11 | I/O Group 1 |
| 12 | I/O Group 2 |
| 13 | I/O Group 3 |
| 14 | I/O Group 4 |
| 15 | I/O Group 5 |
| 16 | I/O Group 6 |
| 17 | I/O Group 7 |
| 18 | I/O Group 8 |
| 19 | I/O Group 9 |
| 20 | I/O Group 10 |
| 21 | I/O Group 11 |
| 22 | I/O Group 12 |
| 23 | I/O Group 13 |
| 24 | I/O Group 14 |
| 31 | Comparator Group 1 |
| 32 | Comparator Group 2 |
| 33 | Comparator Group 3 |
| 34 | Comparator Group 4 |
| 35 | Comparator Group 5 |
| 36 | Comparator Group 6 |
| 51 | Target Group 1 |
| 52 | Target Group 2 |
| 53 | Target Group 3 |
| 54 | Target Group 4 |
| 55 | Target Group 5 |
| 56 | Target Group 6 |
| 71 | Custom App Group 1 |

| Optional status word selection value | Description |
|--------------------------------------|--------------------------|
| 72 | Custom App Group 2 |
| 73 | Load Cell Errors Group 1 |
| 74 | Load Cell Errors Group 2 |

A.4.2. Variable block Commands

Variable block requires only two types of commands – a read (report data) command and a write (write data) command. In addition to the command, the block includes data to refer to a specific variable within the device.

| Command | Description | Value |
|-------------|---|---------|
| Report Data | Reads the information from the variable specified in the class, node, instance, and attribute included in the block | 82, 114 |
| Write data | Writes the value supplied to the variable specified in the class, node, instance, and attribute included in the block | 87, 119 |
| No-op | Default command, no variable specified | 0 |

B. Acyclic (Explicit) Commands

Asynchronous message commands are similar in structure to variable block commands. However, these are control system interface dependent. Typically these support only two types of commands: a read (or get) and a write (or set).

B.1. Direct Level 1

The Direct Level 1 variables have different index or class/instance/attribute codes based on the type of control system fieldbus used. The values for PROFIBUS, PROFINET, and EIP systems are as shown in the following tables.

B.1.1. Report Weight Values

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|-----------------------------|---|-----------------------------|---|--------------------------------------|----------------|
| Report Default Data | For terminals & sensors this is Gross weight data in displayed resolution | 14 | 2000 | 300 , 001 | Floating Point |
| Report Rounded Gross Weight | Gross Weight data is displayed resolution | 15 | 2001 | 300 , 002 | Floating Point |
| Report Rounded Tare Weight | Tare weight data in displayed resolution | 16 | 2002 | 300 , 003 | Floating Point |
| Report Rounded Net Weight | Net weight data in displayed resolution | 17 | 2003 | 300 , 004 | Floating Point |
| Report Gross Weight | Gross weight data in internal resolution | 18 | 2004 | 300 , 005 | Floating Point |
| Report Tare Weight | Tare weight data in internal resolution | 19 | 2005 | 300 , 006 | Floating Point |
| Report Net Weight | Net weight data in internal resolution | 1A | 2006 | 300 , 007 | Floating Point |
| Report Tare Status | State of tare command progress | 1F | 2008 | 300, 016 | Floating Point |
| Report Zero Status | State of zero command progress | 24 | 2009 | 300 , 017 | Floating Point |

B.1.2. General Weighing Commands

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|--------------------------|--|-----------------------------|---|--------------------------------------|----------------|
| Write Tare (Preset tare) | Writes value provided as tare | 1B | 2020 | 300 , 008 | Floating Point |
| Tare (with motion check) | Tare of current scale value after motion check | 1C | 2010 | 300 , 009 | Binary |
| Clear Tare | Clear tare value (no motion check) | 1D | 2012 | 300 , 011 | Binary |
| Tare Immediate | Tare of current scale value (no motion check) | 1E | 2011 | 300 , 010 | Binary |
| Read Zero register | Read current zero value | 20 | 2007 | 300 , 012 | Floating Point |
| Write Zero register | Write value in to zero value | 21 | 2021 | 300 , 013 | Floating Point |
| Zero (with motion check) | Zero scale after motion check | 22 | 2013 | 300 , 014 | Binary |

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|------------------------|--|-----------------------------|---|--------------------------------------|-----------|
| Zero Immediate | Zero scale (no motion check) | 23 | 2014 | 300 , 015 | Binary |
| Select Scale (Channel) | Selects scale/channel if device supports multi-channel/multi-scale operation | 32 | 200B | 300, 019 | Integer |

B.1.3. General Device Information

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|-----------------------------|--------------------------------------|-----------------------------|---|--------------------------------------|-----------|
| Report ID1 | Device identification information #1 | 2B | 2050 | 303 , 001 | String |
| Report ID2 | Device identification information #2 | 2C | 2051 | 303 , 002 | String |
| Report ID3 | Device identification information #3 | 2D | 2052 | 303 , 003 | String |
| Report Software version | Device software OS version | 2E | 2053 | 303 , 004 | String |
| Report Fieldbus version | Device fieldbus stack version | 2F | 2054 | 303 , 005 | String |
| Report Software app version | Device application software version | 30 | 2055 | 303 , 006 | String |
| Report SAI version | Device supported SAI version | 31 | 2056 | 303 , 007 | String |

B.1.4. Miscellaneous Commands

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|----------------------|---|-----------------------------|---|--------------------------------------|-----------|
| Clear Alarm | Resets alarms | 25 | 2030 | 30001 | Binary |
| Turn OFF all outputs | Disables all outputs (turns to OFF state) | 26 | 2031 | 30002 | Binary |

B.1.5. Status Group Information

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|------------------------------|-------------------------------------|-----------------------------|---|--------------------------------------|-----------|
| Report Scale Status Group 1 | Reports scale status bits (group 1) | 27 | 2040 | 302 , 001 | Bit Level |
| Report Alarm Status Group | Reports alarm status bits | 28 | 2041 | 302 , 002 | Bit Level |
| Report RedAlert Status Group | Reports RedAlert status bits | 29 | 2042 | 302 , 003 | Bit Level |
| Report Scale Status Group 2 | Reports scale status bits (group 2) | 2A | 2043 | 302 , 004 | Bit Level |

B.1.6. Test Commands

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|----------------------|---|-----------------------------|---|--------------------------------------|----------------|
| Read Floating Point | Reports test floating point value (123.45) no write | A | 5000 | 30F , 001 | Floating Point |
| Write Floating Point | Allows write and read for test purposes - FP | B | 5001 | 30F , 002 | Floating Point |
| Read Integer | Reports integer value (9876) | C | 5002 | 30F , 003 | Integer |
| Write Integer | Allows write and read for test purposes - INT | D | 5003 | 30F , 004 | Integer |
| Read String | Reports string value (ABCD) | E | 5004 | 30F , 005 | String |
| Write String | Allows write and read for test purposes - STR | F | 5005 | 30F , 006 | String |
| Read Long Integer | Reports long integer value (98765) | 10 | 5006 | 30F , 007 | Long |
| Write Long Integer | Allows write and read for test purposes – Long INT | 11 | 5007 | 30F , 008 | Long |
| Read Byte | Reports byte value (56h) | 12 | 5008 | 30F , 009 | Byte |
| Write Byte | Allows write and read for test purposes - byte | 13 | 5009 | 30F , 010 | Byte |

B.2. Direct Level 2

The direct level variables are application and device specific and provided as part of the device documentation.

B.3. Indirect Variables

The indirect access method uses a small set of index and class/instance/attribute codes to provide access to all variables. The variable used is "named" within the data sent to the device.

B.3.1. Indirect Commands

| Command | Description | Profibus Slot 1 index (hex) | ProfiNET Slot,subslot = 0,1 index (hex) | EIP class, instance, attribute (hex) | Data Type |
|-------------------------------|--|-----------------------------|---|--------------------------------------|----------------------------|
| Read Variable | Read a variable specified by prior Request Read | 0 | 0 and 1, 3000 | 310 , 001 | Varies by type of variable |
| Write Variable | Write value into variable specified | 1 | 1 and 3001 | 310 , 002 | Varies by type of variable |
| (Write) Request Read Variable | Specify variable that will be used in Read command | 2 | 2 and 3003 | 310 , 003 | Varies by type of variable |

www.mt.com

For more information

Mettler-Toledo, LLC

1900 Polaris Parkway
Columbus, OH 43240
Phone 800 438 4511
Fax 614 438 4900

© 2023 Mettler-Toledo, LLC
30588288 Rev. 01, 02/2023



30588288