# Sx-IP

## IP67 versatile

## Xenon Flash

# MODBUS MANUAL

**2.xx software revisions**

# 1   TABLE OF CONTENTS

# 2 MODBUS INTERFACE:

Sx-IP device is fully configurable through a RS485 Modbus interface with following parameters:

- RTU mode
- 19200bps
- Even parity

For Modbus protocol details refer to:

- MODBUS APPLICATION PROTOCOL SPECIFICATION V1.1b3
- Modbus over serial line specification & implementation guide V1.02

Up to 247 Sx-IP devices can be connected on a single RS485 link.

Modbus interface gives write access to configurable parameters like:

- Modbus address
- Energy level
- I/O configuration
- Device status, counters and statistics

Modbus interface also gives read access to many device parameters like:

- Device information (serial number; date code; hardware and software revisions…)
- Device status (flash energy; internal temperature; detailed status and errors…)
- Counters and statistics (Flash counts, min/max temperature, flash health…)

---

*Design notes :*

*Endianness can be a little confusing:*

- *RS485 serial protocol is Little Endian (LSB first).*
- *Modbus protocol is Big Endian (MSB first).*
- *Sx-IP registers' implementation is Little Endian (MSB first)*

| **Serial** protocol | **Modbus** protocol | **Sx-IP** 32bits **registers'** implementation |
|---|---|---|
| bits order in serial frame | Byte order in 16bits words | 16bits words orders in the Modbus frame |
| **Little Endian** | **Big Endian** | **Little Endian** |

*As a result, Sx-IP answer to a Modbus 32bits registers' read request will be as follow:*

| Address | Function | Nb of bytes | LSB of 32bit word | | MSB of 32bit word | | CRC | |
|---|---|---|---|---|---|---|---|---|
| | | | MSB | LSB | MSB | LSB | MSB | LSB |
| 0      7 | 0      7 | 0      7 | 8      15 | 0      7 | 24      31 | 16      23 | 8      15 | 0      7 |

---

## 2.1 REGISTERS

| Register address Dec | Hex | Register name | Register content | Access |
|---|---|---|---|---|
| 0 | 0x00 | SERIAL_NUMBER | Device serial number | R |
| 1 | 0x01 | DATE_CODE | MSB = year, LSB = week | R |
| 2 | 0x02 | HARDWARE_REV | MSB = Major code (1->A), LSB = minor code, 0x0454 -> rev.D.54 | R |
| 3 | 0x03 | PRODUCT_ID | Product ID | R |
| 34 | 0x22 | GLITCH_FILTER | Deglitch time before hardware trigger is accounted (10 to 255) (µs) | R/W |
| 37 | 0x25 | MISS_FLASH_WEIGHT | Weight of a missed flash for flash health supervision | R/W |
| 38 | 0x26 | FLASH_HEALTH_TH | Threshold for the flash health supervision to trigger a failure | R/W |
| 39 | 0x27 | RESET_ON_FAILURE | Time (s) from a failure detection to automatic device reset. Value 0 disable automatic device reset in case of failure | R/W |
| 48 | 0x30 | MODBUS_ADDR_PRESET | Modbus address setting, 1 to 247, 0 for hardware selected address | R/W |
| 49 | 0x31 | ENERGY_LEVEL_PRESET_1 | Energy level settings for the trigger 1. Levels from 0 to 15, 16 for hardware selected energy level. 8bits LSB is primary energy level preset. 8bits MSB is alternate energy level preset | R/W |
| 50 | 0x32 | ENERGY_LEVEL_PRESET_2 | Energy level settings for the trigger 2 | R/W |
| 51 | 0x33 | ENERGY_LEVEL_PRESET_3 | Energy level settings for the trigger 3 | R/W |
| 52 | 0x34 | ENERGY_LEVEL_PRESET_4 | Energy level settings for the trigger 4 | R/W |
| 54 | 0x36 | FLASH_DELAY | Delay from the flash request to the light pulse rising (µs) | R/W |
| 55 | 0x37 | SYNC_DELAY | Delay from the flash request to the synchronization signal activation (µs) | R/W |
| 56 | 0x38 | SYNC_WIDTH | Synchronization pulse width (µs) | R/W |
| 62 | 0x3E | INPUTS_CONFIG | Select hardware inputs polarity and buffer | R/W |
| 63 | 0x3F | OUTPUTS_CONFIG | Select hardware outputs polarity and function | R/W |
| 64 | 0x40 | ENERGY_HW_SEL | Set the combination of inputs generating the hardware selected energy level | R/W |
| 65 | 0x41 | ADDRESS_HW_SEL_L | Set the combination of inputs generating the hardware selected address | R/W |
| 66 | 0x42 | ADDRESS_HW_SEL_H | | R/W |
| 67 | 0x43 | TRIGGERS_INPUT_SEL | Select an input for each of the four triggers | R/W |
| 68 | 0x44 | MISC_INPUT_SEL | Select an input for the alternate command | R/W |
| 256 | 0x100 | STATUS | Status flags | R |
| 257 | 0x101 | ERRORS | Error flags | R |
| 258 | 0x102 | FAILURES | Failure flags | R |
| 259 | 0x103 | COM_ERRORS (LSB) INTERNAL_ERRORS (MSB) | Communication error flags. Internal error flags | R |
| 260 | 0x104 | CONFIG_ERRORS | Configuration error flags | R |
| 262 | 0x106 | IO_STATE | Actual state of hardware inputs and outputs | R |
| 269 | 0x10D | RESET_SOURCE | Source of the last device's reset | R |
| 270 | 0x10E | MODBUS_ADDRESS | Modbus address (1 to 247) | R |
| 271 | 0x10F | ENERGY_LEVEL_T1 ENERGY_LEVEL_T2 | Energy level (0 to 15) associated with trigger 1. Energy level (0 to 15) associated with trigger 2 | R |
| 272 | 0X110 | ENERGY_LEVEL_T3 ENERGY_LEVEL_T4 | Energy level (0 to 15) associated with trigger 3. Energy level (0 to 15) associated with trigger 4 | R |

| Register address | | Register name | Register content | Access |
|---|---|---|---|---|
| Dec | Hex | | | |
| 273 | 0x111 | INPUT_VOLTAGE | Input voltage (mV) | R |
| 274 | 0x112 | TEMPERATURE | Internal temperature (°C) | R |
| 276 | 0x114 | HT_VOLTAGE | Flash capacitor voltage (V/10) | R |
| 279 | 0x117 | FLASH_ENERGY | Last flash energy (J/100) | R |
| 280 | 0x118 | AVERAGE_POWER | Average device power (W) (8s average) | R |
| 281 | 0x119 | NOT_READY_COUNT | Count of flash requests that were not fulfilled because the flash was not ready | R |
| 282 | 0x11A | FLASH_ERROR_COUNT | Flash error count since last reset | R |
| 283 | 0x11B | HEALTH_INDICATOR | Flash health indicator (0 if no error) | R |
| 285 | 0x11D | FAN_RPM | Fan speed (rpm) | A |
| 286 | 0x11E | MIN_INPUT_VOLTAGE | Minimum voltage seen on input since last reset (mV) | R |
| 287 | 0x11F | TARGET_ENERGY | Last flash target energy (J/100) | R |
| 288 | 0x120 | MAX_TEMP (LSB) MIN_TEMP (MSB) | Maximum recorded temperature Minimum recorded temperature | R |
| 289 | 0x121 | START_COUNT_L | Number of device power-up | R |
| 290 | 0x122 | START_COUNT_H | | R |
| 291 | 0x123 | TIME_COUNT_L | Device up-time since last power-up (minutes) | R |
| 292 | 0x124 | TIME_COUNT_H | | R |
| 293 | 0x125 | TOTAL_TIME_COUNT_L | Total device up-time (minutes) | R |
| 294 | 0x126 | TOTAL_TIME_COUNT_H | | R |
| 295 | 0x127 | FLASH_COUNT_L | Flash count since last power-up | R |
| 296 | 0x128 | FLASH_COUNT_H | | R |
| 297 | 0x129 | TOTAL_ FLASH_COUNT_L | Total flash count | R |
| 298 | 0x12A | TOTAL_ FLASH_COUNT_H | | R |
| 301 | 0x12D | TUBE_COUNT_L | Flash count since last flash tube replacement | R |
| 302 | 0x12E | TUBE_COUNT_H | | R |
| 304 | 0x130 | GLITCH_COUNT | Count of glitches detected on triggers' inputs | R |
| 305 | 0x131 | FTIME_1 | Flash time associated with trigger 1 | R |
| 306 | 0x132 | FTIME_2 | Flash time associated with trigger 2 | R |
| 307 | 0x133 | FTIME_3 | Flash time associated with trigger 3 | R |
| 308 | 0x134 | FTIME_4 | Flash time associated with trigger 4 | R |
| 309 | 0x135 | HW_SEL_ENERGY (LSB) HW_SEL_ADDR (MSB) | Hardware selected energy value Hardware selected address value | R |
| 310 | 0x136 | ENERGY_1 | Energy associated with trigger 1 | R |
| 311 | 0x137 | ENERGY _2 | Energy associated with trigger 2 | R |
| 312 | 0x138 | ENERGY _3 | Energy associated with trigger 3 | R |
| 313 | 0x139 | ENERGY _4 | Energy associated with trigger 4 | R |

## 2.2 IMPLEMENTED STANDARD FUNCTIONS

### 2.2.1 FUNCTION 03: READ HOLDING REGISTERS

Example of serial number (0x0000) register read operation:

| Request | 8 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x03 | Function code | 1 byte | 0x03 |
| Starting address | 2 bytes | 0x0000 | Number of bytes | 1 byte | 0x02 |
| Quantity of registers | 2 bytes | 0x0001 | Register content | 2 bytes | 0x0001 |
| CRC | 2 bytes | 0x840A | CRC | 2 bytes | 0x7984 |

Example of out of range (0x0320) register read operation:

| Request | 8 bytes | | Answer | 5 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x03 | Error code | 1 byte | 0x83 |
| Starting address | 2 bytes | 0x0320 | Exception code | 1 byte | 0x02 |
| Quantity of registers | 2 bytes | 0x0001 | CRC | 2 bytes | 0xC0F1 |
| CRC | 2 bytes | 0x8584 | | | |

Example of multiple register read (0x0005 to 0x0008) operation:

| Request | 8 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x03 | Function code | 1 byte | 0x03 |
| Starting address | 2 bytes | 0x0000 | Number of bytes | 1 byte | 0x06 |
| Quantity of registers | 2 bytes | 0x0003 | Register content | 6 bytes | 0x0001 170E 0200 |
| CRC | 2 bytes | 0x05CB | CRC | 2 bytes | 0x79A2 |

### 2.2.2 FUNCTION 06: PRESET SINGLE REGISTER

Example of energy level preset (0x0031) register write operation:

| Request | 8 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x06 | Function code | 1 byte | 0x06 |
| Register address | 2 bytes | 0x0031 | Register address | 2 bytes | 0x0031 |
| Value to write | 2 bytes | 0x0005 | Value to be write | 2 bytes | 0x0005 |
| CRC | 2 bytes | 0x1806 | CRC | 2 bytes | 0x1806 |

Example of out of range (serial number = 0x0000) register write operation:

| Request | 8 bytes | | Answer | 5 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x03 | Error code | 1 byte | 0x86 |
| Register address | 2 bytes | 0x0000 | Exception code | 1 byte | 0x02 |
| Value to write | 2 bytes | 0x0001 | CRC | 2 bytes | 0xC3A1 |
| CRC | 2 bytes | 0x480A | | | |

### 2.2.3 FUNCTION 05: WRITE SINGLE COIL

Write single coil function is used to send specific commands to the Sx-IP.

| Command code | | Command name | Purpose | Execution time |
|---|---|---|---|---|
| **Dec** | **Hex** | | | |
| 1 | 0x01 | RESET | Sx-IP software reset request | <10ms |
| 2 | 0x02 | INIT_DONE | Set the INIT_DONE flag | <10ms |
| 3 | 0x03 | CLEAR | Clear errors and fault counters | <10ms |
| 4 | 0x04 | FLASH | Software flash request | <10ms |
| 5 | 0x05 | SAVE_SETTINGS | Save setting registers' content | **Up to 2s** |
| 6 | 0x06 | CLR_TUBE_COUNT | Clear flash tube count (at flash tube replacement) | Up to 200ms |

Example of clear command (0x0003):

| Request | 8 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x05 | Function code | 1 byte | 0x05 |
| Coil address | 2 bytes | 0x0003 | Coil address | 2 bytes | 0x0003 |
| Value to be write | 2 bytes | 0xFF00 | Value to be write | 2 bytes | 0xFF00 |
| CRC | 2 bytes | 0x7C3A | CRC | 2 bytes | 0x7C3A |

Example of unauthorized command (0x0011):

| Request | 8 bytes | | Answer | 5 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x05 | Error code | 1 byte | 0x85 |
| Coil address | 2 bytes | 0x000B | Exception code | 1 byte | 0x03 |
| Value to be write | 2 bytes | 0x0000 | CRC | 2 bytes | 0x0291 |
| CRC | 2 bytes | 0x3DCA | | | |

---

*Design notes:*

*The Sx-IP will answer a request only after completion of the process (after the execution delay).*

*For long processes like «Save setting», a suitable response timeout shall be set at client side.*

### 2.2.4   SERIAL LINE DIAGNOSTIC COUNTERS

Serial line diagnostic counters are available as defined by the Modbus organization.

Example of Bus message counter read:

| Request | 8 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x08 | Function code | 1 byte | 0x08 |
| Sub function code | 2 bytes | 0x000B | Sub function code | 2 bytes | 0x000B |
| Value to write | 2 bytes | 0x0000 | Value to be write | 2 bytes | 0x24B2 |
| CRC | 2 bytes | 0x91C9 | CRC | 2 bytes | 0x0ABC |

### 2.2.5   READ DEVICE IDENTIFICATION

Basic device identification is available as defined by the Modbus organization.

Example of Read Modbus ID request:

| Request | 7 bytes | | Answer | 7 bytes | |
|---|---|---|---|---|---|
| Device address | 1 byte | 0x01 | Device address | 1 byte | 0x01 |
| Function code | 1 byte | 0x2B | Function code | 1 byte | 0x2B |
| MEI type | 1 byte | 0x0E | MEI type | 1 byte | 0x0E |
| Read Dev Id code | 1 byte | 0x02 | Read Dev Id code | 1 byte | 0x02 |
| Object Id | 1 byte | 0x00 | Conformity level | 1 byte | 0x82 |
| CRC | 2 bytes | 0x7087 | More follows | 1 byte | 0x00 |
| | | | Next object Id | 1 byte | 0x00 |
| | | | Nb of objects | 1 byte | 0x05 |
| | | | Object Id | 1 byte | 0x00 |
| | | | Object length | 1 byte | 0x07 |
| | | | Object value | 7 bytes | "PHOXENE" |
| | | | Object Id | 1 byte | 0x01 |
| | | | Object length | 1 byte | 0x05 |
| | | | Object value | 4 bytes | "0953U" |
| | | | Object Id | 1 byte | 0x02 |
| | | | Object length | 1 byte | 0x04 |
| | | | Object value | 4 bytes | "C.00" |
| | | | Object Id | 1 byte | 0x03 |
| | | | Object length | 1 byte | 0x0B |
| | | | Object value | 11 bytes | "phoxene.com" |
| | | | Object Id | 1 byte | 0x04 |
| | | | Object length | 1 byte | 0x04 |
| | | | Object value | 4 bytes | "Sx-IP" |
| | | | CRC | 2 bytes | 0x4C10 |

Modbus identification information is then:

|  |  |
|---|---|
| Vendor name | PHOXENE |
| Product code | 0953U |
| Revision | C00 |
| Vendor Url | phoxene.com |
| Product name | Sx-IP |

# 3    SX-IP CONFIGURATION

## 3.1    SX-IP OVERVIEW

### 3.1.1    OPERATING MODES

Sx-IP operating mode is configurable. Two modes are available:

- Energy regulation: The Sx-IP adjust the flash time to the expected flash energy according to the last flash energy. This mode is intended to be used when the flashes are requested at constant frequency.
- Burst mode: The Sx-IP use reference tables to adapt the flash time to the capacitors' voltage. It allows to chain flashes of constant energy in high frequency bursts without recharging.

> *Design notes:*
>
> *Operating mode is a Factory setting.*

### 3.1.1    SX-IP FLASH SEQUENCE OVERVIEW

Sx-IP flash sequence is initiated with an unactive to active edge on one of the trigger's inputs.

Sequence starts with a deglitch time that can be configured from 10µs to 255µs (refer to §3.6).
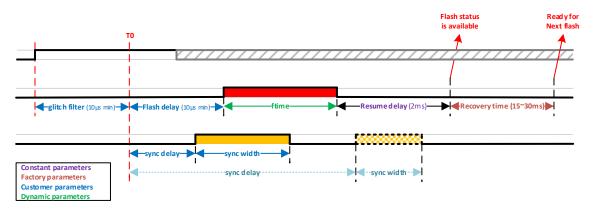
The end of the deglitch time is referenced as T0.

The light emission does not start at T0 but after a minimum 10µs delay. This delay is referenced as "flash delay" and can be configured from 10µs to 60ms (refer to §3.7).

The flash duration is referenced as "ftime". The ftime parameter is computed accordingly to the requested flash energy and the operating mode.

Flash status (energy, errors…) is available after a 2ms resume delay.

The Sx-IP is ready for the next flash after observing a recovery time that is a factory setting (usually 30ms).



Sx-IP is also able to generate a synchronization signal with configurable delay and pulse with (refer to §3.8).

### 3.1.2    SX-IP TRIGGER MANAGEMENT OVERVIEW

Sx-IP include **four logical triggers**. Each of this logical trigger is associated with:

- **A primary and alternate energy level** that is selected from the corresponding *ENERGY_LEVEL_PRESET* register
- **A physical trigger source** (input) that is selected from the *TRIGGER_INPUT_SEL* register

Each logical trigger is armed with energy (*ENERGY[1:4]*) and flash time (*FTIME[1:4]*) values that are read from tables according to:

- The operating mode
- The alternate energy flag that can be associated to an input state using the *MISC_INPUT_SEL* register
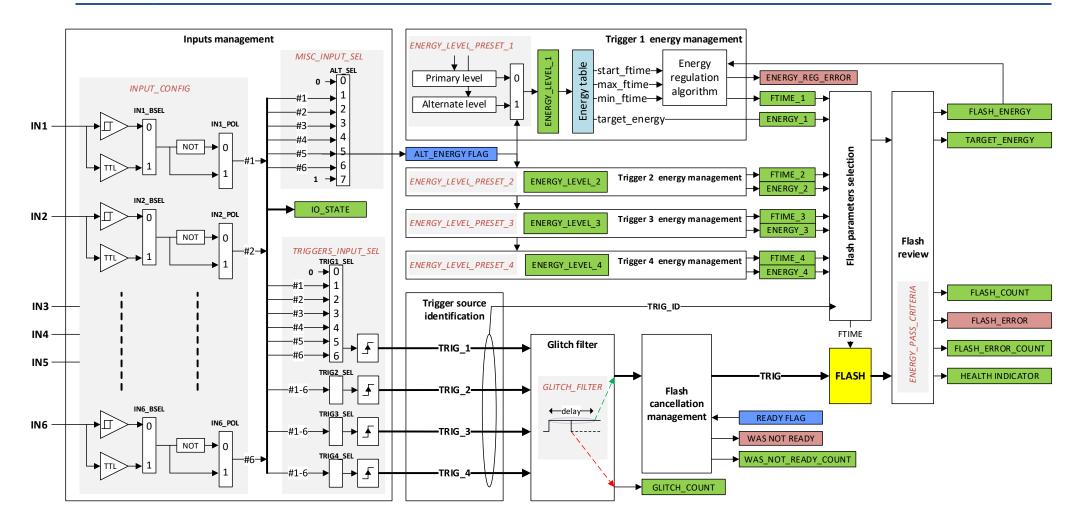- The energy storage voltage


When one of the inputs configured as trigger source is activated:

- The input is identified to select the flash parameters (energy and flash time) armed in the logical trigger associated with the input
- During the glitch filter time, the triggered source input is supervised to detect state changes. If the active state is not hold for the glitch filter time, the flash request is aborded and *GLTICH_COUNT* is incremented.
- The system check that the device is ready to flash. If not, the flash is aborded, the *WAS_NOT_READY* flag is set and *WAS_NOT_READY_COUNT* is incremented
- The flash is triggered


After the flash:

- *FLASH_ENERGY* is computed and compared to the *TARGET_ENERGY* energy using *ENERGY_PASS_CRITERIAS*.
- If the *FLASH_ENERGY* is correct, *FLASH_COUNT* is incremented
- If the *FLASH_ENERGY* is not correct, *FLASH_ERROR* flag is set, *FLASH_ERROR_COUNT* and *HEALTH_INDICATOR* registers are incremented
- In energy regulation mode, the flash time associated with the triggered logical trigger is adjusted according to the *FLASH_ENERGY*. If regulation limits are reached *ENERGY_REG_ERROR* flag is set.

## 3.2   MODBUS ADDRESS SELECTION

Modbus address can be selected from 1 to 247 by *MODBUS_ADDRESS_PRESET* register setting.

Alternatively; by writing 0 in the *MODBUS_ADDRESS_PRESET* register; the Modbus address is the hardware selected address (*HW_SEL_ADDR)*, refer to §3.12.

**MODBUS_ADDRESS_PRESET (0x30) (R/W)**

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|
| - | | | | | Modbus address preset | | | | |

| Value | Description |
|-------|-------------|
| **0** | Modbus address is the hardware selected address, refer to §3.12 |
| **1 to 247** | Modbus address is set to the value |
| **>247** | Return exception. Modbus address is set to 1 |

*Design notes:*

*Factory default Modbus address is '1'*

## 3.3   ENERGY LEVEL SELECTION

For each of the four logical triggers, *ENERGY_LEVEL_PRESET_x* registers allow to select two energy levels from 0 to 15.

*PRIMARY_ENERGY_LEVEL_PRESET* is the default energy level preset assigned to the logical trigger.

*ALTERNATE_ENERGY_LEVEL_PRESET* is the alternate energy level preset assigned to the logical trigger.

If the active energy preset value is "16", the energy level is the hardware selected energy level (*HW_SEL_ENERGY)*, refer to §3.11.

**ENERGY_LEVEL_PRESET_1 (0x31) (R/W)**
**ENERGY_LEVEL_PRESET_2 (0x32) (R/W)**
**ENERGY_LEVEL_PRESET_3 (0x33) (R/W)**
**ENERGY_LEVEL_PRESET_4 (0x34) (R/W)**

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|
| Alternate energy level preset | | | | | Primary energy level preset | | | | |

**Bits 7-0: Primary energy level preset**

| Value | Description |
|-------|-------------|
| 0 to 15 | Primary energy level affected to logical trigger is the value |
| 16 | Primary energy level affected to logical trigger is the hardware selected energy level, refer to §3.11 |
| >16 | Return illegal value exception. |

**Bits 15-8: Alternate energy level preset**

| Value | Description |
|-------|-------------|
| 0 to 15 | Alternate energy level affected to logical trigger is the value |
| 16 | Alternate energy level affected to logical trigger is the hardware selected energy level, refer to §3.11 |
| >16 | Return illegal value exception. |

---

*Design notes:*

*Alternate energy levels allows to switch between energy levels (for example day/night) using a discrete signal.*

---

## 3.4   INPUTS POLARITY AND BUFFERS

Inputs polarity and buffer type can be configured using *INPUT_CONFIG* register.

**INPUT_CONFIG (0x3E) (R/W))**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|----|----|----|----|----|----|----|----|
| - | - | IN6_BSEL | IN5_BSEL | IN4_BSEL | IN3_BSEL | IN2_BSEL | IN1_BSEL |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|----|----|
|  |  | IN6_POL | IN5_POL | IN4_POL | IN3_POL | IN2_POL | IN1_POL |

**POL bits: Polarity selection**

| Value | Description |
|-------|-------------|
| 0 | The input is active low / trig on falling edges |
| 1 | The input is active high / trig on rising edges |

**IN1, IN2, IN3 and IN4 BSEL bits: Buffer selection**

| Value | Description |
|-------|-------------|
| 0 | Input buffer is optimized for 5V levels (levels are 1.0V and 4.0V with Schmitt trigger) |
| 1 | Input buffer is TTL 3.3V compatible (levels are 0.8V and 2.0V) [1] |

**IN5 and IN6 BSEL bits: Buffer selection**

| Value | Description |
|-------|-------------|
| 0 | Input buffer is optimized for 12V levels (levels are 2.0V and 8.0V with Schmitt trigger) |
| 1 | Input buffer is TTL 5V compatible (levels are 1.6V and 4.0V) [1] |

---

*Design notes:*

[1] *TTL buffers are not windowed, they have lower immunity to perturbations than Schmitt trigger buffers. When possible, Schmitt triggers shall be preferred.*

## 3.5 TRIGGERS INPUT SELECTION

For each of the four logical triggers, an input can be configured using *TRIGGERS_INPUT_SEL* register.

**TRIGGERS_INPUT_SEL (0x43) (R/W)**

| 15 | - | - | 12 | 11 | - | - | 8 | 7 | - | - | 4 | 3 | - | - | 0 |
|----|---|---|----|----|---|---|---|---|---|---|---|---|---|---|---|
| TRIG4_SEL | | | | TRIG3_SEL | | | | TRIG2_SEL | | | | TRIG1_SEL | | | |

**TRIGx_SEL**

| Value | Description |
|-------|-------------|
| 0 | The logical trigger is not used |
| 1 to 6 | The logical trigger is associated with corresponding input (IN1 to IN6) |
| 7 | Not implemented (the logical trigger is not used) |

## ALTERNATE ENERGY LEVEL INPUT SELECTION

Alternate energy levels input can be configured using *MISC_INPUT_SEL* register.

**MISC_INPUT_SEL (0x44) (R/W)**

| 15 | - | - | 12 | 11 | - | - | 8 | 7 | - | - | 4 | 3 | - | - | 0 |
|----|---|---|----|----|---|---|---|---|---|---|---|---|---|---|---|
| - | | | | - | | | | - | | | | ALT_ENERGY_SEL | | | |

**ALT_ENERGY_SEL**

| Value | Description |
|-------|-------------|
| 0 | No input is associated to energy level selection. **Primary** energy levels are used. |
| 1 to 6 | Alternate energy level selection is associated to corresponding input (IN1 to IN6) <br> Primary energy levels are used if the associated input is unactive <br> Alternate energy levels are used if the associated input is active |
| 7 | No input is associated to energy level selection. **Alternate** energy levels are used. |

## 3.6 GLITCH FILTER AND DEBOUNCING

Glitch filtering and debouncing of trigger inputs sources is available by configuring the *GLITCH_FILTER* register.

**GLITCH_FILTER (0x22) (R/W)**

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|
| 0x00 | | | | | Glitch filter time | | | | |

| Value | Description |
|-------|-------------|
| 0 to 9 | Return illegal value exception |
| 10 to 255 | Set the glitch filter time to 10 to 255µs |
| >255 | Return illegal value exception |

> *Design notes:*
>
> *Factory default glitch filter time is 10µs that is the minimum value*

## 3.7 FLASH DELAY (0X36) (R/W)

The delay from the flash request to the occurrence of the flash can be configured from 10µs to 60ms by writing the *FLASH_DELAY* register.

| Value | Description |
|---|---|
| < 10µs | Return an illegal data exception |
| 10 to 60000 | Flash delay (µs) from the end of the deglitch time to the start of the flash |
| > 60000 | May return an illegal data exception |

## 3.8 SYNCHRONIZATION SIGNAL

The flash can generate a synchronization signal. By playing with the *FLASH_DELAY* and *SYNC_DELAY* settings, it is possible to shift the synchronization from 60ms before, to 60ms after the flash.

### 3.8.1 SYNC_DELAY (0X37) (R/W)

The delay from the flash request to sync signal activation can be configured from 0µs to 60ms by writing the *SYNC_DELAY* register.

| Value | Description |
|---|---|
| 0 to 60000 | Delay (µs) from the end of the deglitch time to the sync signal activation |
| > 60000 | Return an illegal data exception |

### 3.8.2 SYNC_PULSE_WIDTH (0X38) (R/W)

The synchronization signal pulse width is configurable from 0µs to 5ms by writing the *SYNC_WIDTH* register.

| Value | Description |
|---|---|
| 0 to 5000 | Synchronization signal pulse width (µs) |
| > 5000 | Return an illegal data exception |

## 3.9 RESET ON FAILURE (0X27)

Reset on failure function can be adjusted by writing *RESET_ON_FAILURE* register.

| Value | Description |
|---|---|
| 0 | Reset on failure is unactive |
| 1 to 65535 | Timeout from failure apparition to automatic device reset (seconds) |

## 3.10 OUTPUTS POLARITY AND FUNCTION SELECTION

Outputs polarity and function selection can be configured using *OUTPUT_CONFIG* register.

**OUTPUT_CONFIG (0x3F) (R/W)**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|---|---|---|---|---|---|---|---|
| OUT8_FSEL | | | OUT8_POL | OUT7_FSEL | | | OUT7_POL |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---|---|---|---|---|---|---|---|

| OUT5_FSEL | OUT5_POL | OUT1_FSEL | OUT1_POL |
|---|---|---|---|

**POL bits: Polarity selection**

| Value | Description |
|---|---|
| 0 | The output is active low (closed) |
| 1 | The input is active high (released) |

**FSEL bits: Mode selection**

| Value | Description |
|---|---|
| 0 | The output is released (polarity bit don't care) |
| 1 | The output is low (polarity bit don't care) |
| 2 | The output is synchronization signal |
| 3 | The output is DONE signal |
| 4 | The output is READY signal |
| 5 | The output is FAULT signal |
| 6, 7 | Not implemented (output is released) |

## 3.11 ENERGY LEVEL HARDWARE SELECTION

*ENERGY_HARD_SELECT* registers set the combination of inputs that will generate the hardware selected energy level (*HW_SEL_ENERGY)*.

**ENERGY_HARD_SELEC (R/W)**

| 15 | - | - | 12 | 11 | - | - | 8 | 7 | - | - | 4 | 3 | - | - | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ENERGY_HSEL_3 | | | | ENERGY_HSEL_2 | | | | ENERGY_HSEL_1 | | | | ENERGY_HSEL_0 | | | |

**For each HSEL bit:**

| Value | Description |
|---|---|
| 0 | HSEL bit is read as 0 |
| 1 to 6 | HSEL bit is according to input 1 to 6 |
| 7 | Not implemented (HSEL bit is read à 0) |
| 8 to 11 | HSEL bit is according to selector bits 0 to 3 |
| 12 to 14 | Not implemented (HSEL bit is read à 0) |
| 15 | HSEL bit is read as 1 |

### 3.11.1 SELECTION EXEMPLES

**Energy level hardware selection is from the coder (default configuration)**

| Parameter | ENERGY_HSEL_3 | ENERGY_HSEL_2 | ENERGY_HSEL_1 | ENERGY_HSEL_0 |
|---|---|---|---|---|
| Value | 11 | 10 | 9 | 8 |
| Bit linked to | Coder bit 3 | Coder bit 2 | Coder bit 1 | Coder bit 0 |

**Energy level hardware selection is from inputs 1 to 4 state**

| Parameter | ENERGY_HSEL_3 | ENERGY_HSEL_2 | ENERGY_HSEL_1 | ENERGY_HSEL_0 |
|---|---|---|---|---|
| Value | 4 | 3 | 2 | 1 |
| Bit linked to | Input 4 | Input 3 | Input 2 | Input 1 |

**Energy level hardware selection is from inputs 1 to 3 state (only even values can be selected)**

| Parameter | ENERGY_HSEL_3 | ENERGY_HSEL_2 | ENERGY_HSEL_1 | ENERGY_HSEL_0 |
|---|---|---|---|---|
| Value | 3 | 2 | 1 | 0 |
| Bit linked to | Input 3 | Input 2 | Input 1 | 0 |

**Energy level hardware selection can be 4, 8, 12 and 16 selected from inputs 5 to 6.**

| Parameter | ENERGY_HSEL_3 | ENERGY_HSEL_2 | ENERGY_HSEL_1 | ENERGY_HSEL_0 |
|---|---|---|---|---|
| Value | 6 | 5 | 0 | 0 |
| Bit linked to | Input 6 | Input 5 | 0 | 0 |

## 3.12   MODBUS ADDRESS HARDWARE SELECTION

*ADDR_HARD_SELECT* registers set the combination of inputs that will generate the hardware selected Modbus address (*HW_SEL_ADDR*).

### ADDR_HARD_SELEC_0 (0x41) (R/W)

| 15 | - | - | 12 | 11 | - | - | 8 | 7 | - | - | 4 | 3 | - | - | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADDR_HSEL_3 | | | | ADDR_HSEL_2 | | | | ADDR_HSEL_1 | | | | ADDR_HSEL_0 | | | |

### ADDR_HARD_SELEC_1 (0x42) (R/W)

| 15 | - | - | 12 | 11 | - | - | 8 | 7 | - | - | 4 | 3 | - | - | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ADDR_HSEL_7 | | | | ADDR_HSEL_6 | | | | ADDR_HSEL_5 | | | | ADDR_HSEL_4 | | | |

**For each HSEL bit:**

| Value | Description |
|---|---|
| 0 | HSEL bit is read as 0 |
| 1 to 6 | HSEL bit is according to input 1 to 6 |
| 7 | Not implemented (HSEL bit is read à 0) |
| 8 to 11 | HSEL bit is according to selector bits 0 to 3 |
| 12 to 14 | Not implemented (HSEL bit is read à 0) |
| 15 | HSEL bit is read as 1 |

### 3.12.1   SELECTION EXEMPLES

**Energy level hardware selection is 1 to 16 from the coder (default configuration)**

| Parameter | HSEL_7 | HSEL_6 | HSEL_5 | HSEL_4 | HSEL_3 | HSEL_2 | HSEL_1 | HSEL_0 |
|---|---|---|---|---|---|---|---|---|
| Value | 0 | 0 | 0 | 0 | 11 | 10 | 9 | 8 |
| Bit linked to | 0 | 0 | 0 | 0 | Coder b3 | Coder b2 | Coder b1 | Coder b0 |

**Energy level hardware selection is 32 x Input 2 + 16 x Input 1 + 1 to 16 from the coder**

| Parameter | HSEL_7 | HSEL_6 | HSEL_5 | HSEL_4 | HSEL_3 | HSEL_2 | HSEL_1 | HSEL_0 |
|---|---|---|---|---|---|---|---|---|
| Value | 0 | 0 | 2 | 1 | 11 | 10 | 9 | 8 |
| Bit linked to | 0 | 0 | Input 2 | Input 1 | Coder b3 | Coder b2 | Coder b1 | Coder b0 |

## 3.13   SAVING SETTINGS:

Registers' modifications change current setting but are not retained after a device reset.

In order to save current settings, the Modbus client shall issue a SAVE_SETTINGS command. Refer to §4.5.

# 4    SX-IP COMMANDS

Command requests are sent using Modbus write single coil instructions (refer to § 0 ).

## 4.1    RESET (0X01):

Request a software reset of the device.

## 4.2    INIT_DONE (0X02):

Set the INIT_DONE flag of the status register. INIT_DONE flag is cleared on reset.

This flag is intended to be set by the Modbus client after start-up then periodically read to check that the device has not been reset since last INIT_DONE command.

## 4.3    CLEAR (0X03):

Clear errors flags, failure flags and flash health value.

## 4.4    FLASH (0X04):

Software flash command.

## 4.5    SAVE_SETTINGS (0X05):

Save flash settings in a non-volatile memory. This operation can take up to 2s to complete. A suitable timeout shall be implemented on client side.

## 4.6    CLR_TUBE_COUNT (0X06):

Reset the flash tube counter. This command shall be sent after a flash tube replacement. This operation can take up to 200ms to complete. A suitable timeout shall be implemented on client side.

# 5    SX-IP READ ONLY REGISTERS

## 5.1    CURRENT REGISTERS:

### 5.1.1    IO STATE (0X106) (R):

Device inputs and outputs state is available from the *IO_STATE* register. This register is mainly for debug purpose, in order to test that the Sx-IP is correctly interfaced.

**IO_STATE (R)**

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 |
|------|------|------|------|------|------|------|------|
| OUT8 | OUT7 | OUT5 | OUT1 | SEL3 | SEL2 | SEL1 | SEL0 |

| 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|------|------|------|------|------|------|------|------|
| - | - | IN6 | IN5 | IN4 | IN3 | IN2 | IN1 |

**IN bits: Input state**

| Value | Description |
|-------|-------------|
| 0 | Input is unactive state (released or low according to the polarity bit) |
| 1 | Input is active state (low or released according to the polarity bit) |

**OUT bits: Output state**

| Value | Description |
|-------|-------------|
| 0 | Output is unactive (released or low according to the polarity bit) |
| 1 | Output is active (low or released according to the polarity bit) |

### 5.1.2    MODBUS ADDRESS (0X10E) (R):

Modbus address is available from the *MODBUS_ADDRESS* register.

| Value | Description |
|-------|-------------|
| **1 to 247** | Current Modbus address |

### 5.1.3    ENERGY LEVEL (0X10F & 0X110) (R):

For each of the four logical triggers, associated energy level is available from the *ENERGY_LEVEL_T* registers.

**0x10F (R/W)**

| 15 | - | - | - | - | - | - | 8 | 7 | - | - | - | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ENERGY_LEVEL_T2 | | | | | | | | ENERGY_LEVEL_T1 | | | | | | | |

**0x110 (R/W)**

| 15 | - | - | - | - | - | - | 8 | 7 | - | - | - | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| ENERGY_LEVEL_T4 | | | | | | | | ENERGY_LEVEL_T3 | | | | | | | |

**ENERGY_LEVEL**

| Value | Description |
|-------|-------------|
| **0 to 15** | Energy level associated with the logical trigger |

### 5.1.4    FLASH TIME ASSOCIATED WITH TRIGGERS (0X131 TO 0X134):

Flash time associated with logical triggers are available from the *FTIME_x* registers.

**0x131 - FTIME_1 (R)**

**0x132 - FTIME_2 (R)**

**0x133 - FTIME_3 (R)**

**0x134 - FTIME_4 (R)**

| Value | Description |
|-------|-------------|
| **0 to 65536** | Flash time associated with the logical trigger (µs) |

### 5.1.5    ENERGY ASSOCIATED WITH TRIGGERS (0X136 TO 0X139):

Flash time associated with logical triggers are available from the *FTIME_x* registers.

**0x136 - ENERGY_1 (R)**

**0x137 - ENERGY_2 (R)**

**0x138 - ENERGY_3 (R)**

**0x139 - ENERGY_4 (R)**

| Value | Description |
|-------|-------------|
| **0 to 65536** | Energy associated with the logical trigger (hundredths of Joules) |

### 5.1.6     INPUT VOLTAGE (0X111) (R):

Input voltage is available from the *INPUT_VOLTAGE* register.

| Value | Description |
|-------|-------------|
| **0 to 3500** | Input voltage in hundredths of volts |

### 5.1.7     TEMPERATURE (0X112) (R)

Internal temperature is available from the *TEMPERATURE* register.

| Value | Description |
|-------|-------------|
| **-50 to 120** | Temperature in °C |

### 5.1.8     HT VOLTAGE (0X114 (R):

Flash voltage is available from the *HT_VOLTAGE* register.

| Value | Description |
|-------|-------------|
| **0 to 3100** | Input voltage in tenths of volts |

### 5.1.9     FLASH ENERGY (0X117) (R):

Last flash energy is available from the *FLASH_ENERGY* register.

| Value | Description |
|-------|-------------|
| **0 to 50000** | Last flash energy in hundredths of Joules |

### 5.1.10    AVERAGE POWER (0X118) (R):

Average output power calculated on an 8s sliding period is available from the *AVERAGE_POWER* register.

| Value | Description |
|-------|-------------|
| **0 to 100** | Average power in Watts |

### 5.1.11    FLASH ERROR COUNT (0X11A) (R):

Count of missed flashes since last reset is available from the *FLASH_ERROR_COUNT* register.

| Value | Description |
|-------|-------------|
| **0 to 65536** | Number of missed flashes |

*FLASH_ERROR_COUNT* register is cleared on reset.

### 5.1.12    NOT READY COUNT (0X119) (R):

Count of flashes requests that occurs while the flash was not ready is available from the *NOT_READY_COUNT* register.

| Value | Description |
|-------|-------------|
| **0 to 65536** | Count of flash requests that were not fulfilled because the flash was not ready |

*NOT_READY_COUNT* register is cleared on reset.

### 5.1.13    GLITCH COUNT (0X130) (R):

Count of glitches detected on trigger source inputs since last reset is available from the *GLITCH_COUNT* register.

| Value | Description |
|---|---|
| **0 to 65536** | Count of glitches detected on trigger source inputs |

*GLITCH_COUNT* register is cleared on reset.

### 5.1.14    FLASH HEALTH (0X11B) (R):

*HEALTH_INDICATOR* register is a flash fault rate indicator. It is mainly designed for flash tube health supervision.

This indicator is increased by *MISSED_FLASH_WEIGTH* on a missed flash, decreased by 1 on a successful flash.

A flash failure occurs if the value reach *HEALTH_TRESHOLD*.

| Value | Description |
|---|---|
| **0 to 65536** | Flash fault rate |

*HEALTH_INDICATOR* register is cleared on reset.

### 5.1.15    MINI INPUT VOLTAGE (0X11E) (R):

*MIN_INPUT_VOLTAGE* register records the minimum input voltage seen since last reset.

| Value | Description |
|---|---|
| **0 to 45055** | Minimum input voltage in mV |

### 5.1.16    TARGET ENERGY (0X11F) (R):

*TARGET_ENERGY* register indicates the last flash target energy.

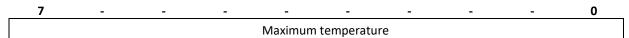| Value | Description |
|---|---|
| **0 to 65536** | Last flash target flash energy in in hundredths of Joules |

## 5.1.17    MIN & MAX TEMPERATURE (0X64) (R):

Sx-IP records minimal and maximal temperatures. These values are available from the *MIN_MAX_TEMP* register.

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|----|----|----|----|----|----|----|----|----|----|
| Minimum temperature | | | | | Maximum temperature | | | | |

**MAX_TEMP (0x64 LSB) (R)**

| 7 | - | - | - | - | - | - | - | - | 0 |
|----|----|----|----|----|----|----|----|----|----|
| Maximum temperature | | | | | | | | | |

| Value | Description |
|-------|-------------|
| -50 to 120 | Maximum temperature in °C |

**MIN_TEMP (0x64 MSB) (R)**

| 7 | - | - | - | - | - | - | - | - | 0 |
|----|----|----|----|----|----|----|----|----|----|
| Minimum temperature | | | | | | | | | |

| Value | Description |
|-------|-------------|
| -50 to 120 | Minimum temperature in °C |

## 5.1.18    START COUNTER (0X69, 0X6A) (R):

Sx-IP records the number of device power-up events (software reset are not accounted) This 32bits counter available from the *START_COUNT_L* and *START_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|----|----|----|----|----|----|----|----|----|
| Start count H (0x6A) | | | | | Start count L (0x69) | | | | |
| Start count | | | | | | | | | |

| Value | Description |
|-------|-------------|
| 0 to $2^{32}$ | Number of device power-up (over device life) |

## 5.1.19    TIME COUNTER (0X6B, 0X6C) (R):

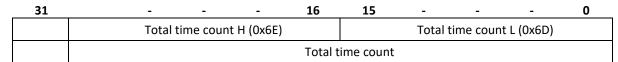The power-up time (since last power-up) is available from *TIME_COUNT_L* and *TIME_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|----|----|----|----|----|----|----|----|----|
| Time count H (0x6C) | | | | | Time count L (0x6B) | | | | |
| Time count | | | | | | | | | |

| Value | Description |
|-------|-------------|
| 0 to $2^{32}$ | Power-up time in minutes since last power-up |

### 5.1.20    TOTAL TIME COUNTER (0X6D, 0X6E) (R):

Sx-IP records the total power-up time (over device life). This 32bits counter is available from *TOTAL_TIME_COUNT_L* and *TOTAL_TIME_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|---|---|---|----|----|---|---|---|---|
| | Total time count H (0x6E) | | | | Total time count L (0x6D) | | | | |
| | Total time count | | | | | | | | |

| Value | Description |
|-------|-------------|
| **0 to $2^{32}$** | Total power-up time in minutes since last (over device life) |

### 5.1.21    FLASH COUNTER (0X6F, 0X70) (R):

Flash count (since last power-up) is available from *FLASH_COUNT_L* and *FLASH_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|---|---|---|----|----|---|---|---|---|
| | Flash count H (0x70) | | | | Flash count L (0x6F) | | | | |
| | Flashes count | | | | | | | | |

| Value | Description |
|-------|-------------|
| **0 to $2^{32}$** | Flash count last (since last power-up) |

### 5.1.22    TOTAL FLASH COUNTER (0X71, 0X72) (R):

Sx-IP records the total flash count (over device life). This 32bits counter is available from *TOTAL_FLASH_COUNT_L* and *TOTAL_FLASH_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|---|---|---|----|----|---|---|---|---|
| | Total flash count H (0x72) | | | | Total flash count L (0x71) | | | | |
| | Flashes count | | | | | | | | |

| Value | Description |
|-------|-------------|
| **0 to $2^{32}$** | Total flash count last (over device life) |

### 5.1.23    FLASH TUBE COUNTER (0X75, 0X76) (R):

Flash tube count record the total flash count of the flash tube. This 32bits counter is available from *FT_COUNT_L* and *FT_COUNT_H* registers.

| 31 | - | - | - | 16 | 15 | - | - | - | 0 |
|----|---|---|---|----|----|---|---|---|---|
| | Flash tube count H (0x76) | | | | Flash tube count L (0x75) | | | | |
| | Flash tube count | | | | | | | | |

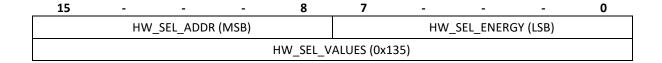| Value | Description |
|-------|-------------|
| **0 to $2^{32}$** | Flash tube count (since last Flash tube count clear) |

Flash tube count can be cleared after flash tube replacement by sending a CLR_FT_COUNT command (refer to §4.6).

### 5.1.24   HARDWARE SELECTED VALUES (0X135) (R):

Computed values for energy level and address hardware selection.

Values are resulting from hardware inputs multiplexed according to *ENERGY_HW_SEL* and *ADDRESS_HW_SEL* registers' rules.

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|----|---|---|---|---|---|---|---|---|---|
| HW_SEL_ADDR (MSB) | | | | | HW_SEL_ENERGY (LSB) | | | | |
| HW_SEL_VALUES (0x135) | | | | | | | | | |

**HW_SEL_ENERGY**

| Value | Description |
|-------|-------------|
| **0 to 15** | Hardware selected energy level |

**HW_SEL_ADDR**

| Value | Description |
|-------|-------------|
| **1 to 247** | Hardware selected Modbus address |

## 5.2   STATUS REGISTER:

*STATUS* register provides most important device information concerning the device.

**STATUS (0x100) (R)**

| Bit | Flag name | Description |
|---|---|---|
| 1 | INITIALIZED | Set using INIT_DONE command |
| 2 | READY | Sx-IP is ready to flash (internal 100ms refresh rate) |
| 3 | COM_ERROR | An error occurs on the Modbus link (details available from *COM_ERRORS* register) |
| 4 | ERROR | Global error flag |
| 5 | FAILURE | Global failure flag |
| 8 | ALT_ENERGY_LEVEL | Alternate energy level is selected (night mode) |

## 5.3   ERRORS AND FAILURES FLAGS REGISTERS:

*ERRORS*, *FAILURES*, *INT_ERRORS* and *COM_ERRORS* register provides detailed error information.

### 5.3.1   ERRORS (0X101) (R)

| Bit | Flag name | Description |
|---|---|---|
| 0 | VIN_OUT_OF_RANGE | Input voltage is not in the required range |
| 2 | WAS_NOT_READY | A flash request occurs while the device was not ready (cleared on next flash) |
| 4 | ENERGY_REG_ERROR | The energy regulation algorithm reaches one of its flash time limits |
| 5 | FLASH_ERROR | Last flash was missed or out of expected energy range |
| 6 | FAN_ERROR | Internal fan is not spinning at expected speed |
| 7 | OVERTEMP | Internal temperature exceed 80°C (2°C hysteresis) |
| 8 | INTERNAL_ERROR | An internal error occurs (details available from *INT_ERRORS* register) |
| 9 | CONFIG_ERROR | Some value in settings is not supported |
| 10 | OVERLOAD | The average output power overpass the limit |

### 5.3.2   FAILURES (0X102) (R)

| Bit | Flag name | Description |
|---|---|---|
| 5 | FLASH_FAILURE | Flash error occurrences exceed defined health parameters |
| 6 | FAN_FAILURE | Internal fan was in error for 10s |
| 7 | DCDC_ERROR | An over-voltage condition occurs on the internal high voltage supply |
| 8 | SIMMER_ERROR | Unexpected ignition of the flash tube |

### 5.3.3 COM_ERRORS AND INTERNAL ERRORS (0X103) (R)

| 15 | - | - | - | 8 | 7 | - | - | - | 0 |
|---|---|---|---|---|---|---|---|---|---|
| INTERNAL_ERRORS (MSB) | | | | | COM_ERRORS (LSB) | | | | |

**COM_ERRORS (0x103 LSB) (R)**

| Bit | Flag name | Description |
|---|---|---|
| 0 | RS485_FRAMING_ERROR | A received frame was not correct (uart frame level) |
| 1 | RS485_PARITY_ERROR | A received frame comes with bad parity |
| 2 | RS485_FIFO_OVERFLOW | More characters were received that the FIFO can hold |
| 3 | RS485_BUFFER_OVERFLOW | A character was received before last frame was processed |
| 5 | MODBUS_FRAME_ERROR | A frame was received with unexpected size |
| 6 | MODBUS_CRC_ERROR | A frame was received with a bad CRC. |

**INTERNAL_ERRORS (0x103 MSB) (R)**

| Bit | Flag name | Description |
|---|---|---|
| 0 | INTERNAL_VCC_ERROR | Internal VCC supply out of range error |
| 1 | MEMORY_ERROR | Memory is corrupted or contains unexpected value(s) |
| 2 | INT_COM_ERROR | Internal communication error |
| 3 | EXECUTION_ERROR | Something unexpected occurs |
| 4 | INTERNAL_12V_ERROR | Internal 12V supply out of range error |

### 5.3.4 CONFIG_ERRORS (0X104) (R)

| Bit | Flag name | Description |
|---|---|---|
| 0 | IDENTITY_ERROR | Device's identity registers' content is not valid |
| 1 | HARD_PARAM_ERROR | Hardware parameters registers' content is not valid |
| 2 | OP_PARAM_ERRORS | Operational parameters are not valid |
| 3 | SETTINGS_ERROR | Settings are not valid |
| 4 | IO_SETTINGS_ERROR | Inputs / outputs settings are not valid |
| 5 | FLASH_SEQUENCE_ERROR | Parameters do not allow to build a flash sequence |
| 6 | SYNC_SEQUENCE_ERROR | Parameters do not allow to build a synchronization signal sequence |
| 7 | FTIME_TABLE_ERROR | Ftime table content is not valid |
| 8 | ENERGY_TABLE_ERROR | Energy table content is not valid |
| 9 | CONFIG_CRC_ERROR | Configuration registers' check sum is not valid |

# 6 ADVANCED FUNCTIONNALITIES

## 6.1 HEALTH MONITORING

Sx-IP device has a build in configurable health monitoring function.

It allows to detect when a device regularly miss flashes, that is relevant of a flash tube end of life.

### 6.1.1 HEALTH MONITORING ALGORITHM

The algorithm is:

- Each missed flash *HEALTH_INDICATOR* register is incremented by *MISS_FLASH_WEIGHT*.
- Each succeeded flash *HEALTH_INDICATOR* register is decremented by 1 (if not zero).
- If *HEALTH_INDICATOR* reach *FLASH_HEALTH_TH*, a flash failure is triggered

> *Design notes:*
>
> *Setting* FLASH_HEALTH_TH *to zero inhibits triggering a flash failure.*
>
> *Flash health evolution laws and failure thresholds are factory configured. It can be adapted to client needs.*
>
> *Default values are 1000 for* MISSED_FLASH_WEIGHT *and 1000 for* HEALTH_THRESHOLD. *It allows 10 missed flashes every 10 000 flashes.*

### 6.1.2 HEALTH MONITORING SET-UP

Health monitoring algorithm allows to adjust the health monitoring based on two parameters:

- The maximum allowed consecutive missed flashes; subsequently called **CMF**
- The maximum allowed missed flash rate; subsequently called **MFR**

Health monitoring registers' value can be calculated using following equations:

*MISS_FLASH_WEIGHT* = 1 / MFR

*FLASH_HEALTH_TH* = (CMF + 1) * *MISS_FLASH_WEIGHT*

**Calculation example:**

Let's assume that we want the flash failure to trigger if the missed flash rate rise above 1% or if more than two consecutive flashes are missed:

- Expected CMF = 2 consecutive flashes
- Expected MFR = 1%

*MISS_FLASH_WEIGHT* = 1 / 1% = 100

*FLASH_HEALTH_TH* = (2+ 1) * 100 = 300

# 7 TROUBLESHOOTING

## 7.1 FAILURES MANAGEMENT

Failures make Sx-IP unable to operate.

In case of failure, Sx-IP will try to resume by a software reset every 5s (configurable refer to §3.9).

## 7.2 ERRORS MANAGEMENT

Errors are usually not critical.

Most errors (fan error, internal errors, communication errors) are for information. These errors do not cause degradation of the device performances.

Configuration errors make the device unable to flash. Most of time, configuration errors result of inappropriate settings values that need to be corrected.

Over-temp error makes the device unable to flash until the temperature drop two degrees below the over-temp threshold.

Overload error makes the device unable to flash for a few seconds, until compute average power drop below 90% of the maximum allowed average power.

Fan error will cause a failure if it lasts for 10s.

Vin out of range error is maintained for 10s every time the input voltage is detected out of range.

If the input voltage remains out of range for more than several milliseconds, it may cause multiple errors and failures due to internal supplies fall.

## 7.3   ERRORS AND FAILURES WORKAROUND

| Error flag | Causes | Workaround |
|---|---|---|
| **Vin out of range** | Input voltage has been detected out of range during the last 10s. | Check power supply volage, current capacity and voltage drop in cables<br>Supervise the *MIN_VOLTAGE* register |
| **HT out of reg** | The device fails in reaching the energy storage regulation point | Contact Phoxene |
| **Was not ready** | A flash request occurs while the device was not ready. | Check that the delay between two flash requests is compatible with the flash configuration |
| **Energy reg error** | The energy regulation algorithm reaches a limit | Check / replace the lamp<br>Contact Phoxene |
| **Flash error** | Last flash was missed or out of expected energy range | Nothing to do. |
| **Fan error** | Internal fan has been detected low | Nothing to do. |
| **Over-temp** | Internal temperature exceeds over-temperature threshold | Reduce average power (flash energy and/or cadence)<br>Reduce ambient temperature.<br>Protect the flash housing from direct sunlight.<br>Increase air-flow around the flash housing |
| **Config error** | Some settings are not supported | Fix settings<br>Reload factory settings<br>Contact Phoxene |
| **Overload** | The average output power overpass the limit. | Reduce average power (flash energy and/or cadence) |
| **COM error** | An error occurs on the Modbus link | Detailed error flags are available in *COM_ERROR* register |
| **RS485 Framing error** | Received frame is not correct | Check serial parameters (baudrate, nb bits, stop bits)<br>Check RS485 link quality (common mode voltage…) |
| **RS485 parity error** | Received frame parity is not correct | Check serial parameters (baudrate, nb bits, stop bits)<br>Check RS485 link quality (common mode voltage…) |
| **RS485 FIFO overflow** | More characters were received that the FIFO can hold | Check frame length<br>Check frame cadence (was the last frame answered ?) |
| **RS485 buffer overflow** | A character was received before last frame was processed | Check frame cadence (was the last frame answered ?) |
| **Modbus Frame error** | Frame length is not as expected | Check Modbus client frame construction |
| **Modbus CRC error** | The frame CRC is not valid | Check Modbus client CRC algorithm<br>Check Modbus client frame construction |
| **Internal error** | An internal error occurs | Contact Phoxene |
| **Flash failure** | Device fails to flash and/or flash energy is not in expected range. | Check / replace the flash tube<br>Contact Phoxene |
| **Fan failure** | Internal fan failure | Contact Phoxene |
| **DC/DC error** | An over-voltage condition occurs on the internal high voltage supply | Contact Phoxene |
| **Simmer error** | Unexpected ignition of the flash tube | Contact Phoxene |