

XIN_SR.TDD

This driver reads byte(s) from the XPort I/O extension system with a status bit, which controls the flow of the data.

Installation of the driver

INSTALL DEVICE #D, "XIN_SR_xx.TDD" [, P1, ..., P27]

D is a constant, variable or an expression of data type BYTE, WORD, LONG in range of 0...63 and is the device number of the driver.

xx in the file name of the device driver represents the driver's input buffer size (R1 = 256 bytes buffer, K1 = 1024 bytes buffer).

P1...P5 are more parameters, which changes the settings of the XIN_SR_xx.TDD driver.

| | Default | Description of parameters |
|----|---------|----------------------------|
| P1 | - | Number of samples per 1 ms |
| P2 | - | Busy time |
| P3 | - | XPort address status bit |
| P4 | - | XPort address data byte |
| P5 | - | Bit number of status bit |

User Function Codes

User-Function-Codes of XIN_SR_xx.TDD for requesting parameters (Instruction GET, secondary address 0):

| No. | Symbol Prefix UFCI_ | Description |
|-----|------------------------|---|
| 01H | UFCI_IBU_FILL | No. of bytes in input buffer (Byte) |
| 02H | UFCI_IBU_FREE | Free space in input buffer (Byte) |
| 03H | UFCI_IBU_VOL | Size of input buffer (Byte) |
| 93H | XIN_XPADR_STATUS | XPort address of status bit |
| 94H | XIN_XPADR_DATA | XPort address of data byte |
| 95H | XIN_BITNO_STATUS | Bit number of status bit |
| 96H | XIN_ACTIVE | Active Flag 0: driver is active ↔0: driver inactive |
| 97H | XIN_BYTES_PER_MS | Maximum number of bytes read per ms |
| 98H | XIN_BUSY | Maximum number of samples of status bit |

User-Function-Codes of XIN_SR_xx.TDD for setting of parameters (Instruction PUT, secondary address 0):

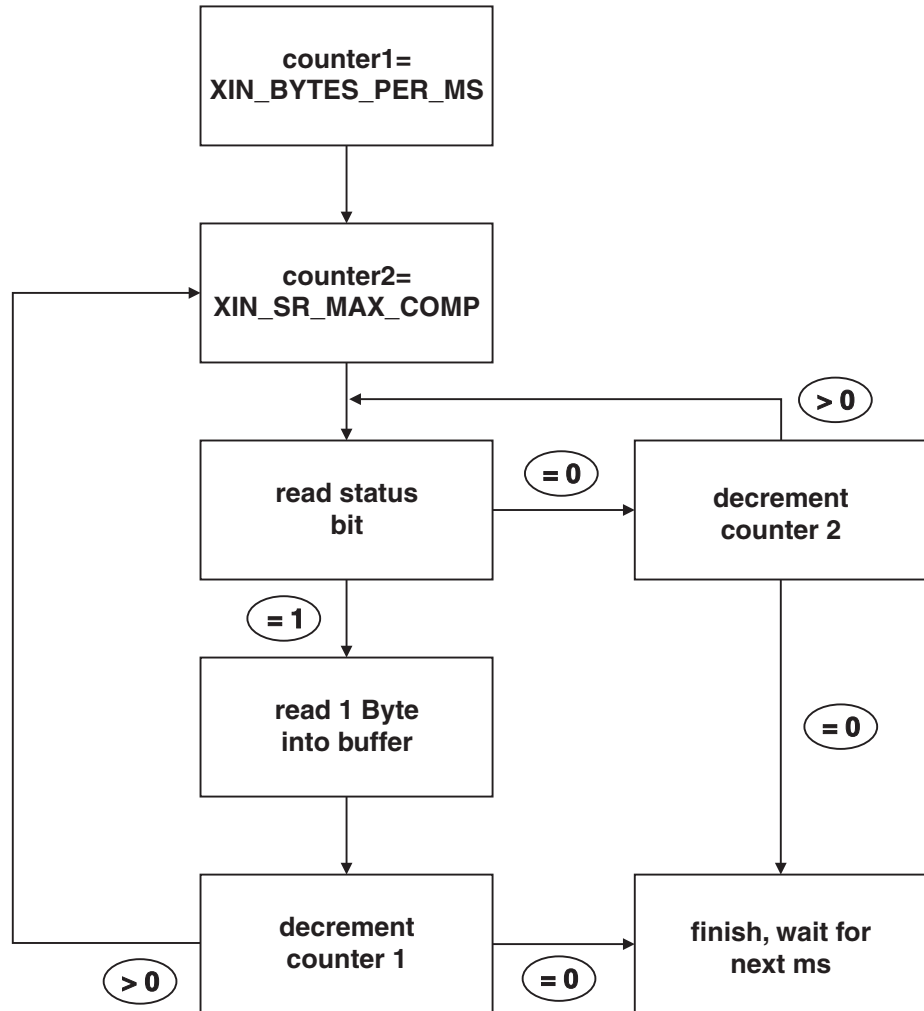
| No. | Symbol Prefix: UFCO_ | Description |
|-----|-------------------------|---|
| 01H | UFCO_IBU_ERASE | Delete input buffer |
| 93H | XIN_XPADR_STATUS | XPort address of status bit |
| 94H | XIN_XPADR_DATA | XPort address of data byte |
| 95H | XIN_BITNO_STATUS | Bit number of status bit |
| 96H | XIN_ACTIVE | Active Flag 0: driver is active ↔0: driver inactive |
| 97H | XIN_BYTES_PER_MS | Maximum number of bytes read per ms |
| 98H | XIN_SR_MAX_COMP | Maximum number of status bit comparisons (retries) |

Data & control pins used for XBus

| Pins | Description |
|--------|----------------------|
| Port 6 | Data bus |
| L33 | ACLK (Address Clock) |
| L34 | DCLK (Data Clock) |
| L35 | INE (Input Enable) |

Read data from XPort

The following flowchart shows the functionality of XIN_SR_xx.TDD. The driver tests the status bit; if the result is 1, one byte is imported into the input buffer. This procedure is repeated every 1 ms. Maximum number of status bit comparisons and maximum number of bytes per ms can be set in the install device or with a User Function Code.



Read out imported data byte(s)

With secondary address 0 you can read out the input buffer of XIN_SR_xx.TDD.

GET #D, #0, Number, Variable

D is a constant, a variable or expression of the data type BYTE, WORD, LONG in the range from 0...63 and determines the device number of the driver.

Number is a constant, a variable or expression of the data type BYTE, WORD, LONG and specifies the length of output.

Variable is a variable of the data type BYTE, WORD, LONG or STRING which contains the data of the input buffer.

Start and stop the device driver

With User Function Code XIN_ACTIVE you can start and stop the activity of XIN_SR_xx.TDD.

PUT #D, #0, #XIN_ACTIVE, Variable

D is a constant, a variable or expression of the data type BYTE, WORD, LONG in the range from 0...63 and determines the device number of the driver.

Variable is a variable of the data type BYTE, WORD, LONG or STRING.
0: start device driver
↔0: stop device driver

Program example:

```
#include define_a.inc
#include ufunc3.inc
#define XIN_XPADR_STATUS      093H
#define XIN_XPADR_DATA       094H
#define XIN_BITNO_STATUS     095H
#define XIN_ACTIVE           096H
#define XIN_BYTES_PER_MS    097H
#define XIN_BUSY             098H

task main

    long ibu_fill
    string input_data$

    install_device #0, "XIN_SR_R1.TDD", 1, 1, 07H, 06H, 0
    '
    '                               ! ! ! ! !-- Bitno statusbit
    '                               ! ! ! ! !----- XP addr databyte
    '                               ! ! ! ! !----- XP addr statusbit
    '                               ! ! ! ! !----- SR retries
    '                               ! ! ! ! !----- Max bytes per lms

    PUT    #0, #0, #XIN_ACTIVE, YES      ' start the device driver
    '                                     ' YES: start device driver
    '                                     ' NO: stop device driver

    while 1=1

        ibu_fill = 0                    ' init variable
        while ibu_fill = 0              ' wait for data <===== LOOP =====>
            GET    #0, #0, #UFCI_IBU_FILL, 0, ibu_fill ' get fill of buffer
            endwhile                      ' wait for data <===== LOOP =====>

            GET #0, 0, input_data$        ' read out data
        endwhile

    end
```