

FDX protocol

Generated by Doxygen 1.8.18

1 Data Structure Index	1
1.1 Data Structures	1
2 File Index	3
2.1 File List	3
3 Data Structure Documentation	5
3.1 Digital_Input_t Struct Reference	5
3.1.1 Detailed Description	5
3.2 Digital_Output_t Struct Reference	5
3.2.1 Detailed Description	6
3.3 FDX_DataExchange_t Struct Reference	6
3.3.1 Detailed Description	6
3.4 FDX_DataRequest_t Struct Reference	6
3.4.1 Detailed Description	7
3.5 FDX_Header_t Struct Reference	7
3.5.1 Detailed Description	7
3.6 FDX_Status_t Struct Reference	7
3.6.1 Detailed Description	8
3.7 FDX_Stop_t Struct Reference	8
3.7.1 Detailed Description	8
3.8 FXD_Start_t Struct Reference	8
3.8.1 Detailed Description	8
4 File Documentation	9
4.1 D:/Doxygen/test/FDX.c File Reference	9
4.1.1 Detailed Description	10
4.1.2 Function Documentation	10
4.1.2.1 FDX_CreateDataExchangeCmd()	10
4.1.2.2 FDX_CreateDataRequestCmd()	11
4.1.2.3 FDX_CreateFrameHeader()	11
4.1.2.4 FDX_CreateStartCmd()	11
4.1.2.5 FDX_CreateStatusCmd()	12
4.1.2.6 FDX_CreateStopCmd()	12
4.1.2.7 FDX_ParsingFrame()	13
4.2 D:/Doxygen/test/FDX.h File Reference	13
4.2.1 Detailed Description	15
4.2.2 Macro Definition Documentation	15
4.2.2.1 CMD_START_SIZE	16
4.2.2.2 DIR_GOLDENBOX_TO_APP	16
4.2.2.3 PERIPH_ID_DIGITAL_INPUT	16
4.2.2.4 SIGNATURE	16
4.2.3 Typedef Documentation	16

4.2.3.1 Std_ReturnType	16
4.2.4 Enumeration Type Documentation	16
4.2.4.1 anonymous enum	17
4.2.5 Function Documentation	17
4.2.5.1 FDX_CreateDataExchangeCmd()	17
4.2.5.2 FDX_CreateDataRequestCmd()	17
4.2.5.3 FDX_CreateFrameHeader()	18
4.2.5.4 FDX_CreateStartCmd()	18
4.2.5.5 FDX_CreateStatusCmd()	18
4.2.5.6 FDX_CreateStopCmd()	19
4.2.5.7 FDX_ParsingFrame()	19
4.2.6 Variable Documentation	20
4.2.6.1 FdxCommandE	20
Index	21

Chapter 1

Data Structure Index

1.1 Data Structures

Here are the data structures with brief descriptions:

Digital_Input_t	
Digital input channels structure	5
Digital_Output_t	
Digital output channels structure	5
FDX_DataExchange_t	
The data exchange structure	6
FDX_DataRequest_t	
The data request structure	6
FDX_Header_t	
The header structure	7
FDX_Status_t	
The status structure	7
FDX_Stop_t	
The stop structure	8
FXD_Start_t	
The start structure	8

Chapter 2

File Index

2.1 File List

Here is a list of all documented files with brief descriptions:

D:/Doxygen/test/ FDX.c	
This file is the implementation of FDX protocol	9
D:/Doxygen/test/ FDX.h	
This file is the Header file of FDX protocol	13

Chapter 3

Data Structure Documentation

3.1 Digital_Input_t Struct Reference

Digital input channels structure.

```
#include <FDX.h>
```

Data Fields

- uint8_t **CHANNEL_0**
- uint8_t **CHANNEL_1**
- uint8_t **CHANNEL_2**
- uint8_t **CHANNEL_3**
- uint8_t **CHANNEL_4**
- uint8_t **CHANNEL_5**
- uint8_t **CHANNEL_6**
- uint8_t **CHANNEL_7**

3.1.1 Detailed Description

Digital input channels structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.2 Digital_Output_t Struct Reference

Digital output channels structure.

```
#include <FDX.h>
```

Data Fields

- uint8_t **CHANNEL_0**
- uint8_t **CHANNEL_1**
- uint8_t **CHANNEL_2**
- uint8_t **CHANNEL_3**
- uint8_t **CHANNEL_4**
- uint8_t **CHANNEL_5**
- uint8_t **CHANNEL_6**
- uint8_t **CHANNEL_7**

3.2.1 Detailed Description

Digital output channels structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.3 FDX_DataExchange_t Struct Reference

The data exchange structure.

```
#include <FDX.h>
```

Data Fields

- uint16_t **CommandSize**
- uint16_t **CommandCode**
- uint16_t **GroupID**
- uint16_t **DataSize**
- uint8_t **DataBytes**

3.3.1 Detailed Description

The data exchange structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.4 FDX_DataRequest_t Struct Reference

The data request structure.

```
#include <FDX.h>
```

Data Fields

- uint16_t **CommandSize**
- uint16_t **CommandCode**
- uint16_t **GroupID**

3.4.1 Detailed Description

The data request structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.5 FDX_Header_t Struct Reference

The header structure.

```
#include <FDX.h>
```

Data Fields

- uint64_t **Signature**
- uint8_t **MajorVersion**
- uint8_t **MinorVersion**
- uint16_t **NumOfCMD**
- uint16_t **SeqNum**
- uint16_t **Reserved**

3.5.1 Detailed Description

The header structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.6 FDX_Status_t Struct Reference

The status structure.

```
#include <FDX.h>
```

Data Fields

- uint16_t **CommandSize**
- uint16_t **CommandCode**
- uint16_t **Status**
- uint16_t **Reserved**
- uint64_t **TimeStamp**

3.6.1 Detailed Description

The status structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.7 FDX_Stop_t Struct Reference

The stop structure.

```
#include <FDX.h>
```

Data Fields

- uint16_t **CommandSize**
- uint16_t **CommandCode**

3.7.1 Detailed Description

The stop structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

3.8 FXD_Start_t Struct Reference

The start structure.

```
#include <FDX.h>
```

Data Fields

- uint16_t **CommandSize**
- uint16_t **CommandCode**

3.8.1 Detailed Description

The start structure.

The documentation for this struct was generated from the following file:

- D:/Doxygen/test/[FDX.h](#)

Chapter 4

File Documentation

4.1 D:/Doxygen/test/FDX.c File Reference

This file is the implementation of FDX protocol.

```
#include "stdio.h"
#include "stdint.h"
#include "FDX.h"
```

Macros

- `#define CMD_SIZE_OFFSET (16u)`
- `#define CMD_CODE_OFFSET (18u)`
- `#define DATAEX_GROUPID_OFFSET (20u)`
- `#define DATAEX_SIZE_OFFSET (22u)`
- `#define DATAEX_BYTES_OFFSET (24u)`

Functions

- [Std_ReturnType FDX_ParsingFrame](#) (uint8_t *inbuffer, uint16_t *frametype, uint16_t *datasize, uint16_t *groupID)
This Function Specifies the Frame type.
- [Std_ReturnType FDX_CreateStartCmd](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the Start cmd of the frame.
- [Std_ReturnType FDX_CreateStopCmd](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the Stop cmd of the frame.
- [Std_ReturnType FDX_CreateDataExchangeCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t groupID, uint16_t dataSize, uint8_t *dataBytes)
This Function creates data exchange command.
- [Std_ReturnType FDX_CreateDataRequestCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t groupID)
This Function creates data request command.
- [Std_ReturnType FDX_CreateStatusCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t status, uint64_t timeStamp)
This Function creates Status command.
- [Std_ReturnType FDX_CreateFrameHeader](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the header of the frame.

4.1.1 Detailed Description

This file is the implementation of FDX protocol.

Author

Family

Version

0.1

Date

2020-06-04

Copyright

Copyright (c) 2020

4.1.2 Function Documentation

4.1.2.1 FDX_CreateDataExchangeCmd()

```
Std_ReturnType FDX_CreateDataExchangeCmd (
    uint8_t * buffer,
    uint16_t seqNum,
    uint16_t groupID,
    uint16_t dataSize,
    uint8_t * dataBytes )
```

This Function creates data exchange commad.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>groupID</i>	Holds the Frame direction and peripheral number
<i>dataSize</i>	Holds The number of bytes of the data
<i>dataBytes</i>	Holds the data

Returns

Std_ReturnType returns E_OK if there is a data exchange command and E_Not_OK if there is no data exchange command

4.1.2.2 FDX_CreateDataRequestCmd()

```
Std_ReturnType FDX_CreateDataRequestCmd (
    uint8_t * buffer,
    uint16_t seqNum,
    uint16_t groupID )
```

This Function creates data request commad.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>groupID</i>	Holds the Frame direction and peripheral number

Returns

Std_ReturnType returns E_OK if there is a data request command and E_Not_OK if there is no data request command

4.1.2.3 FDX_CreateFrameHeader()

```
Std_ReturnType FDX_CreateFrameHeader (
    uint8_t * buffer,
    uint16_t seqNum )
```

This Function creates the header of the frame.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there a frame header and E_Not_OK if there is no a frame header

4.1.2.4 FDX_CreateStartCmd()

```
Std_ReturnType FDX_CreateStartCmd (
    uint8_t * buffer,
    uint16_t seqNum )
```

This Function creates the Start cmd of the frame.

Parameters

<i>buffer</i>	Holds the input Frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there is a start command and E_Not_OK if there is no start command

4.1.2.5 FDX_CreateStatusCmd()

```
Std_ReturnType FDX_CreateStatusCmd (
    uint8_t * buffer,
    uint16_t seqNum,
    uint16_t status,
    uint64_t timeStamp )
```

This Function creates Status command.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>status</i>	Holds the frame state
<i>timeStamp</i>	Holds the estimated action time

Returns

Std_ReturnType returns E_OK if there is a data request command and E_Not_OK if there is no data request command

4.1.2.6 FDX_CreateStopCmd()

```
Std_ReturnType FDX_CreateStopCmd (
    uint8_t * buffer,
    uint16_t seqNum )
```

This Function creates the Stop cmd of the frame.

Parameters

<i>buffer</i>	Holds the input Frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there is a stop command and E_Not_OK if there is no stop command

4.1.2.7 FDX_ParsingFrame()

```
Std_ReturnType FDX_ParsingFrame (
    uint8_t * inbuffer,
    uint16_t * frametype,
    uint16_t * datasize,
    uint16_t * groupId )
```

This Function Specifies the Frame type.

Parameters

<i>inbuffer</i>	Holds The input Frame
<i>frametype</i>	Holds The Command Code
<i>datasize</i>	Holds The number of bytes of the data
<i>groupId</i>	Holds the Frame direction and peripheral number

Returns

Std_ReturnType returns E_OK if there is a valid header and E_Not_OK if there is invalid header

4.2 D:/Doxygen/test/FDX.h File Reference

This file is the Header file of FDX protocol.

Data Structures

- struct [FDX_Header_t](#)
The header structure.
- struct [FXD_Start_t](#)
The start structure.
- struct [FDX_Stop_t](#)
The stop structure.
- struct [FDX_DataExchange_t](#)
The data exchange structure.
- struct [FDX_DataRequest_t](#)
The data request structure.
- struct [FDX_Status_t](#)
The status structure.
- struct [Digital_Input_t](#)
Digital input channels structure.
- struct [Digital_Output_t](#)
Digital output channels structure.

Macros

- `#define E_OK 0x00u`
- `#define E_NOT_OK 0x01u`
- `#define STD_HIGH 0x01u`
- `#define STD_LOW 0x00u`
- `#define STD_ACTIVE 0x01u`
- `#define STD_IDLE 0x00u`
- `#define STD_ON 0x01u`
- `#define STD_OFF 0x00u`
- `#define SIGNATURE 0x4341474e414d5350`

The Header info, the signature stands for CAGNAMSP.

- `#define MAJOR_VERSION 1`
- `#define MINOR_VERSION 0`
- `#define NUM_OF_CMD 1`
- `#define RESERVED_VALUE 0`
- `#define CMD_START_SIZE (4u)`

The commands sizes.

- `#define CMD_STOP_SIZE (4u)`
- `#define CMD_DATA_EXCHANGE_SIZE (10u)`
- `#define CMD_DATA_REG_SIZE (6u)`
- `#define CMD_STATUS_SIZE (16u)`
- `#define DIR_GOLDENBOX_TO_APP 0`

the frame direction

- `#define DIR_APP_TO_GOLDENBOX 1`
- `#define PERIPH_ID_DIGITAL_INPUT 0`

the Peripheral ID

- `#define PERIPH_ID_DIGITAL_OUTPUT 1`

Typedefs

- `typedef uint8_t Std_ReturnType`

The standerd types.

Enumerations

- `enum {`
Cmd_Start = 0x0001, **Cmd_Stop** = 0x0002, **Cmd_Key** = 0x0003, **Cmd_DataRequest** = 0x0006,
Cmd_DataExchange = 0x0005, **Cmd_DataError** = 0x0007, **Cmd_FreeRunning** = 0x0008, **Cmd_Free↵**
RunningCancel = 0x0009,
Cmd_Status = 0x0004, **Cmd_StatusRequest** = 0x000A, **Cmd_SeqNumError** = 0x000B }
`}`

This enum Holds the Types of command code.

Functions

- [Std_ReturnType FDX_CreateStartCmd](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the Start cmd of the frame.
- [Std_ReturnType FDX_CreateStopCmd](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the Stop cmd of the frame.
- [Std_ReturnType FDX_CreateDataExchangeCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t GroupID, uint16_t DataSize, uint8_t *DataBytes)
This Function creates data exchange commad.
- [Std_ReturnType FDX_CreateDataRequestCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t GroupID)
This Function creates data request commad.
- [Std_ReturnType FDX_CreateStatusCmd](#) (uint8_t *buffer, uint16_t seqNum, uint16_t status, uint64_t timeStamp)
This Function creates Status command.
- [Std_ReturnType FDX_ParsingFrame](#) (uint8_t *inbuffer, uint16_t *frametype, uint16_t *datasize, uint16_t *groupID)
This Function Specifies the Frame type.
- [Std_ReturnType FDX_CreateFrameHeader](#) (uint8_t *buffer, uint16_t seqNum)
This Function creates the header of the frame.

Variables

- enum { ... } [FdxCommandE](#)
This enum Holds the Types of command code.

4.2.1 Detailed Description

This file is the Header file of FDX protocol.

Author

Family

Version

0.1

Date

2020-06-04

Copyright

Copyright (c) 2020

4.2.2 Macro Definition Documentation

4.2.2.1 CMD_START_SIZE

```
#define CMD_START_SIZE (4u)
```

The commands sizes.

4.2.2.2 DIR_GOLDENBOX_TO_APP

```
#define DIR_GOLDENBOX_TO_APP 0
```

the frame direction

4.2.2.3 PERIPH_ID_DIGITAL_INPUT

```
#define PERIPH_ID_DIGITAL_INPUT 0
```

the Peripheral ID

4.2.2.4 SIGNATURE

```
#define SIGNATURE 0x4341474e414d5350
```

The Header info, the signature stands for CAGNAMSP.

4.2.3 Typedef Documentation

4.2.3.1 Std_ReturnType

```
typedef uint8_t Std_ReturnType
```

The standerd types.

4.2.4 Enumeration Type Documentation

4.2.4.1 anonymous enum

anonymous enum

This enum Holds the Types of command code.

4.2.5 Function Documentation

4.2.5.1 FDX_CreateDataExchangeCmd()

```
Std_ReturnType FDX_CreateDataExchangeCmd (
    uint8_t * buffer,
    uint16_t seqNum,
    uint16_t groupID,
    uint16_t dataSize,
    uint8_t * dataBytes )
```

This Function creates data exchange commad.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>groupID</i>	Holds the Frame direction and peripheral number
<i>dataSize</i>	Holds The number of bytes of the data
<i>dataBytes</i>	Holds the data

Returns

Std_ReturnType returns E_OK if there is a data exchange command and E_Not_OK if there is no data exchange command

4.2.5.2 FDX_CreateDataRequestCmd()

```
Std_ReturnType FDX_CreateDataRequestCmd (
    uint8_t * buffer,
    uint16_t seqNum,
    uint16_t groupID )
```

This Function creates data request commad.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>groupID</i>	Holds the Frame direction and peripheral number

Returns

Std_ReturnType returns E_OK if there is a data request command and E_Not_OK if there is no data request command

4.2.5.3 FDX_CreateFrameHeader()

```
Std_ReturnType FDX_CreateFrameHeader (
    uint8_t * buffer,
    uint16_t seqNum )
```

This Function creates the header of the frame.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there a frame header and E_Not_OK if there is no a frame header

4.2.5.4 FDX_CreateStartCmd()

```
Std_ReturnType FDX_CreateStartCmd (
    uint8_t * buffer,
    uint16_t seqNum )
```

This Function creates the Start cmd of the frame.

Parameters

<i>buffer</i>	Holds the input Frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there is a start command and E_Not_OK if there is no start command

4.2.5.5 FDX_CreateStatusCmd()

```
Std_ReturnType FDX_CreateStatusCmd (
    uint8_t * buffer,
```

```
uint16_t seqNum,  
uint16_t status,  
uint64_t timeStamp )
```

This Function creates Status command.

Parameters

<i>buffer</i>	Holds the input frame
<i>seqNum</i>	Holds the frame order
<i>status</i>	Holds the frame state
<i>timeStamp</i>	Holds the estimated action time

Returns

Std_ReturnType returns E_OK if there is a data request command and E_Not_OK if there is no data request command

4.2.5.6 FDX_CreateStopCmd()

```
Std_ReturnType FDX_CreateStopCmd (  
    uint8_t * buffer,  
    uint16_t seqNum )
```

This Function creates the Stop cmd of the frame.

Parameters

<i>buffer</i>	Holds the input Frame
<i>seqNum</i>	Holds the frame order

Returns

Std_ReturnType returns E_OK if there is a stop command and E_Not_OK if there is no stop command

4.2.5.7 FDX_ParsingFrame()

```
Std_ReturnType FDX_ParsingFrame (  
    uint8_t * inbuffer,  
    uint16_t * frametype,  
    uint16_t * datasize,  
    uint16_t * groupId )
```

This Function Specifies the Frame type.

Parameters

<i>inbuffer</i>	Holds The input Frame
<i>frametype</i>	Holds The Command Code
<i>datasize</i>	Holds The number of bytes of the data
<i>groupId</i>	Holds the Frame direction and peripheral number

Returns

Std_ReturnType returns E_OK if there is a valid header and E_Not_OK if there is invalid header

4.2.6 Variable Documentation

4.2.6.1 FdxCommandE

```
enum { ... } FdxCommandE
```

This enum Holds the Types of command code.

Index

CMD_START_SIZE

FDX.h, [15](#)

D:/Doxygen/test/FDX.c, [9](#)

D:/Doxygen/test/FDX.h, [13](#)

Digital_Input_t, [5](#)

Digital_Output_t, [5](#)

DIR_GOLDENBOX_TO_APP

FDX.h, [16](#)

FDX.c

FDX_CreateDataExchangeCmd, [10](#)

FDX_CreateDataRequestCmd, [10](#)

FDX_CreateFrameHeader, [11](#)

FDX_CreateStartCmd, [11](#)

FDX_CreateStatusCmd, [12](#)

FDX_CreateStopCmd, [12](#)

FDX_ParsingFrame, [13](#)

FDX.h

CMD_START_SIZE, [15](#)

DIR_GOLDENBOX_TO_APP, [16](#)

FDX_CreateDataExchangeCmd, [17](#)

FDX_CreateDataRequestCmd, [17](#)

FDX_CreateFrameHeader, [18](#)

FDX_CreateStartCmd, [18](#)

FDX_CreateStatusCmd, [18](#)

FDX_CreateStopCmd, [19](#)

FDX_ParsingFrame, [19](#)

FdxCommandE, [20](#)

PERIPH_ID_DIGITAL_INPUT, [16](#)

SIGNATURE, [16](#)

Std_ReturnType, [16](#)

FDX_CreateDataExchangeCmd

FDX.c, [10](#)

FDX.h, [17](#)

FDX_CreateDataRequestCmd

FDX.c, [10](#)

FDX.h, [17](#)

FDX_CreateFrameHeader

FDX.c, [11](#)

FDX.h, [18](#)

FDX_CreateStartCmd

FDX.c, [11](#)

FDX.h, [18](#)

FDX_CreateStatusCmd

FDX.c, [12](#)

FDX.h, [18](#)

FDX_CreateStopCmd

FDX.c, [12](#)

FDX.h, [19](#)

FDX_DataExchange_t, [6](#)

FDX_DataRequest_t, [6](#)

FDX_Header_t, [7](#)

FDX_ParsingFrame

FDX.c, [13](#)

FDX.h, [19](#)

FDX_Status_t, [7](#)

FDX_Stop_t, [8](#)

FdxCommandE

FDX.h, [20](#)

FXD_Start_t, [8](#)

PERIPH_ID_DIGITAL_INPUT

FDX.h, [16](#)

SIGNATURE

FDX.h, [16](#)

Std_ReturnType

FDX.h, [16](#)