

# HartTools 7.5

## Advanced Details of FrameAlyst Software Documentation

Revision: 7.5.0  
Date: 10.2.2017



```
<?xml version="1.0"?>
<FrameAlystRecords>
  <Header>
    <FrameAlystVersion>7.4.2</FrameAlystVersion>
    <SessionInfo>FrameAlyst 7.4 for Hart[Data:Test-4.frax]</SessionInfo>
    <NumberOfFrames>6</NumberOfFrames>
    <TimeAndDate>23.07.2014 23:18:48</TimeAndDate>
  </Header>
  <Frames>
    <Frame Number="00000">
      <RawData>
        <Properties StartTime="48683431" EndTime="48683569" NumberOfBytes="15"
        <FrameBytes>255,255,255,255,255,130,189,253,1,2,3,0,0,194,255</FrameBytes>
      </RawData>
      <AddInfo>
        <HeadingComment>Script: CMD(0) / NO DATA</HeadingComment>
      </AddInfo>
    </Frame>
    <Frame Number="00001">
      <RawData>
        <Properties StartTime="48683581" EndTime="48683923" NumberOfBytes="38"
        <FrameBytes>255,255,255,255,255,134,189,253,1,2,3,0,24,0,41,254,253,253
      </RawData>
      <AddInfo />
    </Frame>
    <Frame Number="00002">
      <RawData>
        <Properties StartTime="48684025" EndTime="48684163" NumberOfBytes="15"
        <FrameBytes>255,255,255,255,255,130,189,253,1,2,3,1,0,195,255</FrameBytes>
      </RawData>
      <AddInfo>
        <HeadingComment>Script: CMD(1) / NO DATA</HeadingComment>
      </AddInfo>
    </Frame>
  </Frames>
</FrameAlystRecords>
```

Borst Automation  
Neue Reihe 33  
DE-27472 Cuxhaven  
GERMANY

Fon: +49 (0)4721 6985100  
Fax: +49 (0)6432 6985102

<http://borst-automation.com>

[info@borst-automation.de](mailto:info@borst-automation.de)



Copyright© 1998-2017 Borst Automation, Walter Borst, Cuxhaven, GERMANY

Hart® is a registered trademark of the Hart Communication Foundation  
Windows® is a registered trademark of Microsoft Corporation

# Contents

<b>Introduction .....</b>	<b>1</b>
<b>Advanced Details .....</b>	<b>1</b>
<b>File Menu .....</b>	<b>1</b>
Store in Xml and Html Format .....	2
<b>Service Menu .....</b>	<b>3</b>
Toggle Burst Mode .....	4
Set Poll Address .....	4
Search Device .....	4
Edit/Set Long Tag .....	4
Activate Hart 6/7 .....	5
Read Device Data.....	5
Set Tag/Dsc/Date.....	5
Set Range .....	6
Edit/Run Scripts.....	6
<b>General .....</b>	<b>7</b>
Decoding Data in a Frame .....	7
Copy to SendAnyFrame .....	8
Copy Bytes to the Clipboard .....	9
Editing Data Syntax.....	9
Displaying the Slave Emulation .....	10
Handling of Erroneous Frames .....	10
Setting Custom Colors .....	11
Frame Display Examples .....	11

## Introduction

The manual of Hart Tools 7.5 is only providing the most important information about FrameAlyst. There are some more features which are described in this document.

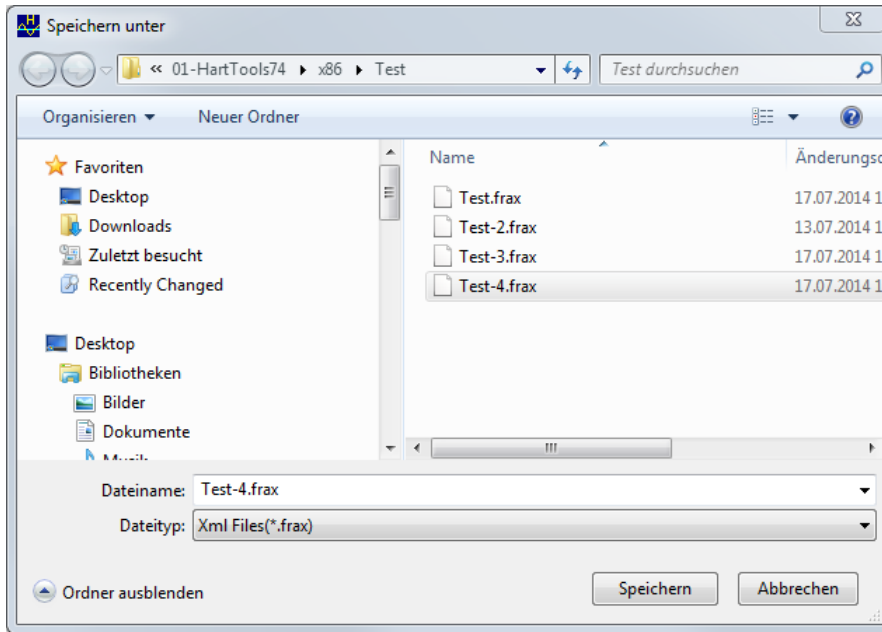
## Advanced Details

---

### File Menu

The frames are still stored in the format which was used in the past. However when saving the frame data you may also select an xml format or html format.

## Store in Xml and Html Format



If you select the file extension .frax, the frames will be stored in xml format.

Alternatively you may also choose an html format as a documentation of the debug session.

An example of an xml output is shown on the following page.

## Xml Format Example

```
<?xml version="1.0"?>
<FrameAlystRecords>
  <Header>
    <FrameAlystVersion>7.4.2</FrameAlystVersion>
    <SessionInfo>FrameAlyst 7.4 for Hart[Data:Test-4.frax]</SessionInfo>
    <NumberOfFrames>6</NumberOfFrames>
    <TimeAndDate>23.07.2014 23:18:48</TimeAndDate>
  </Header>
  <Frames>
    <Frame Number="00000">
      <RawData>
        <Properties StartTime="48683431" EndTime="48683569" NumberOfBytes="15" WasGapTimeOut="False" ClientTxFlag="True" IsValidFrame="True" />
        <FrameBytes>255,255,255,255,255,130,189,253,1,2,3,0,0,194,255</FrameBytes>
      </RawData>
      <AddInfo>
        <HeadingComment>Script: CMD(0) / NO DATA</HeadingComment>
      </AddInfo>
    </Frame>
    <Frame Number="00001">
      <RawData>
        <Properties StartTime="48683581" EndTime="48683923" NumberOfBytes="38" WasGapTimeOut="False" ClientTxFlag="False" IsValidFrame="True" />
        <FrameBytes>255,255,255,255,255,134,189,253,1,2,3,0,24,0,41,254,253,253,5,7,1,0,0,0,1,2,3,5,6,12,24,1,250,1,0,0,1,230</FrameBytes>
      </RawData>
      <AddInfo />
    </Frame>
    <Frame Number="00002">
      <RawData>
        <Properties StartTime="48684025" EndTime="48684163" NumberOfBytes="15" WasGapTimeOut="False" ClientTxFlag="True" IsValidFrame="True" />
        <FrameBytes>255,255,255,255,255,130,189,253,1,2,3,1,0,195,255</FrameBytes>
      </RawData>
      <AddInfo>
        <HeadingComment>Script: CMD(1) / NO DATA</HeadingComment>
      </AddInfo>
    </Frame>
```

Regarding Html format you may either store the records in an Html file or click 'html' in the print functions. The print function for 'html' is opening your standard browser directly to display the frames.

## Html Output Example

```

HartTools Version: 7.4.2.0
Date: 23.07.2014 / Time: 23:31

Script: CMD(0) / NO DATA
1: --->LSTXP|FF FF FF FF|82|BD FD 01 02 03|Cmd 0| 0
   No Data
   138>|C2|?FF

Script: CMD(1) / NO DATA
2: 12>LACKP|FF FF FF FF|86|BD FD 01 02 03|Cmd 0|24| 0|00101001
   Data: FE FD FD 05 07 01 00 00 00 01 02 03 05 06 0C 18 01 FA 01 00 00 01
   342>|E6| |Result: Cold Start, Prim Var Limits

Script: CMD(1) / NO DATA
3: 102>LSTXP|FF FF FF FF|82|BD FD 01 02 03|Cmd 1| 0
   No Data
   138>|C3|?FF

Script: CMD(1) / NO DATA
4: 62>LACKP|FF FF FF FF|86|BD FD 01 02 03|Cmd 1| 7| 0|00101001
   Data: 20 43 13 E5 1D
   189>|61| |Result: Cold Start, Prim Var Limits

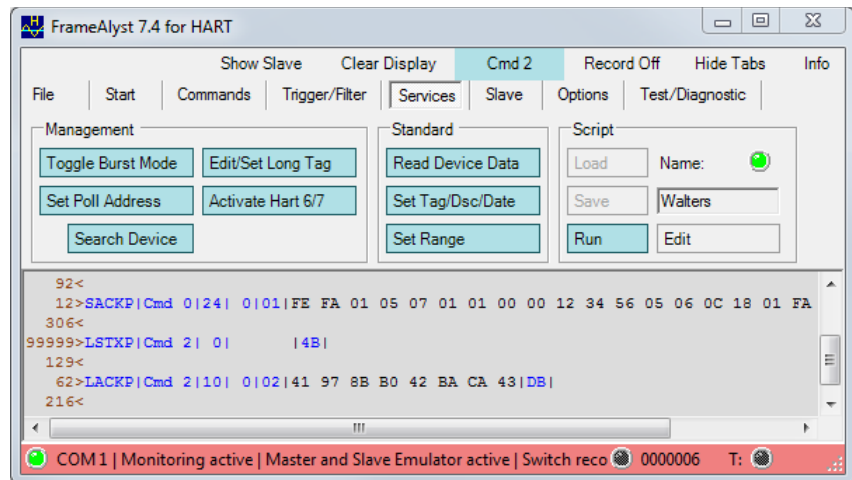
Script: CMD(18) / DATA(Pasc6;TIC120 ;Pasc12;DESCRIPTOR ;17;7;114)
5: 99>LSTXP|FF FF FF FF|82|BD FD 01 02 03|Cmd18|21
   Data: 50 90 F1 CB 08 20 10 54 C3 48 94 14 3D 28 20 82 08 20 11 07 72
   331>|A3|?FF

Script: CMD(18) / DATA(Pasc6;TIC120 ;Pasc12;DESCRIPTOR ;17;7;114)
6: 61>LACKP|FF FF FF FF|86|BD FD 01 02 03|Cmd18|23| 0|01101011
   Data: 50 90 F1 CB 08 20 10 54 C3 48 94 14 3D 28 20 82 08 20 11 07 72
   333>|CE| |Result: Cold Start, Var Limits, Prim Var Limits

End of Records Output
    
```

## Service Menu

Services are some more complex functions as only sending a command.



The services are only working if the FrameAlyst is using the master emulation.

## Toggle Burst Mode

This service is handling command 109.

## Set Poll Address

Set slave poll address is handling command 6. Note: Hart5 is only supporting addresses 0..15 while Hart 7 has a range of 0..63.

## Search Device

This service searches for slaves in a range of poll addresses from 0 to 63.

## Edit/Set Long Tag

The long tag is an iso latin-1 string of a length of a maximum of 32 characters. If it contains less than 32 characters it is terminated by 0x00.

## Activate Hart 6/7

There is no form provided which is used to realize this mean. The service is using commands 7 and 6 to signal the slave device that a Hart 6/7 host is connected.

## Read Device Data

This service is reading the main information from a device.

## Set Tag/Dsc/Date

This application is setting the short tag, the descriptor and the date.

## Set Range

Write Primary Variable Range

Upper Value (20 mA): 200.0 °C  
Lower Value (4 mA): -20.0 °C

Write Response  
064  
Command not Implemented

Write

Cancel Read Exit

Reading Completed!

The service is trying to write the upper and the lower range value of the primary variable of a device.

## Edit/Run Scripts

Running scripts is simply sending a series of commands with or without request data.

Edit Command Sequence Script

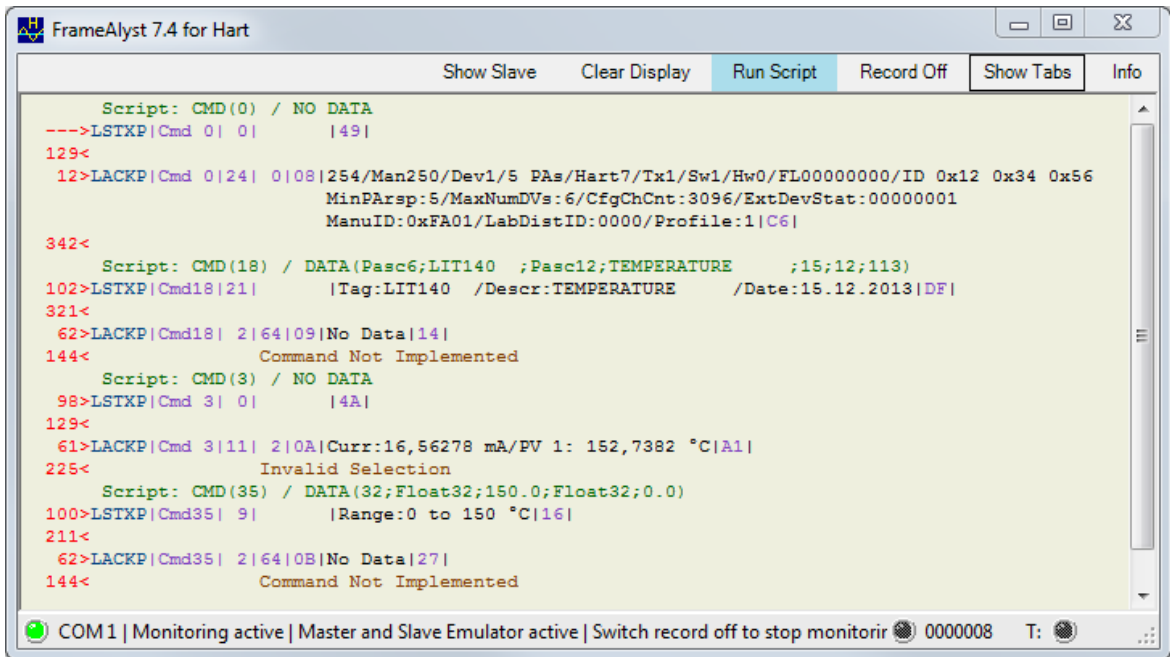
Name: Walters Run

No	CMD	Data	Syntax
1	0		
2	18	Pasc6;LIT140	;Pasc12;TEMPERATURE ;15;12;113
3	3		
4	35	32;Float32;150.0;Float32;0.0	

Clear All  
Delete  
Insert  
Append  
Edit Snbx  
Cancel  
OK

Use data syntax editor.

The example above is sending the commands 0, 18, 3 and 35.



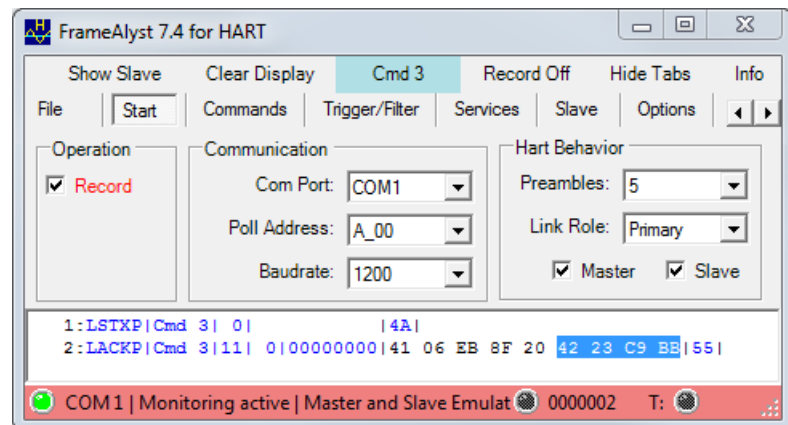
The script may be stored in a file and be loaded from a file. The active script is always stored in the settings of the software and automatically reloaded after the start of FrameAlyst.

If command 255 is specified in the script, the data will be sent as is not formatted as a Hart frame.

## General

### Decoding Data in a Frame

For decoding data in a frame the data to be decoded has to be marked.





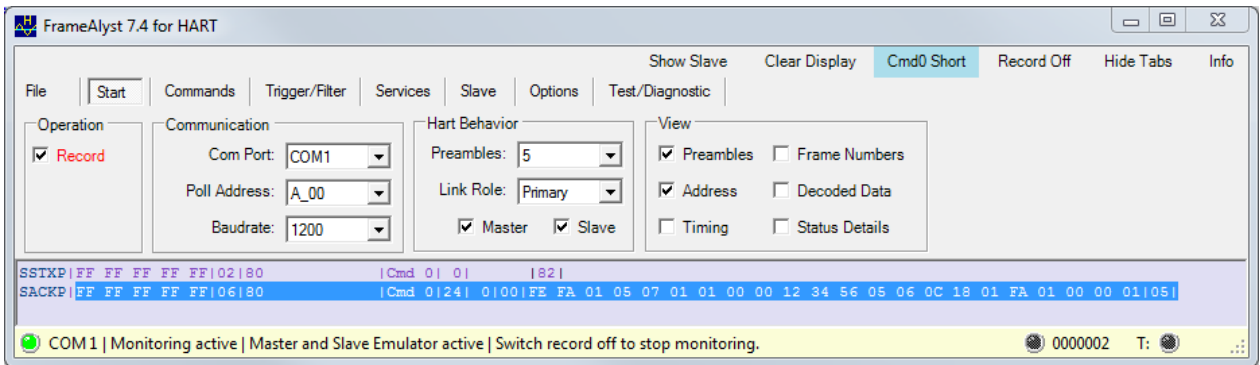
By using the right mouse button a context menu will be displayed.

- Integer
- Float
- HartUnit
- PackedASCII
- Text
- Binary
- 
- Copy to AnyFrame

Select the decoding of your choice and the value will be displayed in a tool tip.

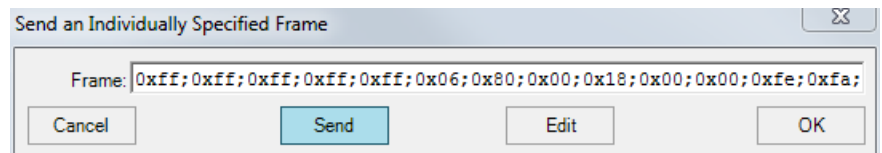
### Copy to SendAnyFrame

Sometimes it could be helpful to copy a frame to the send any frame function to modify and send it.

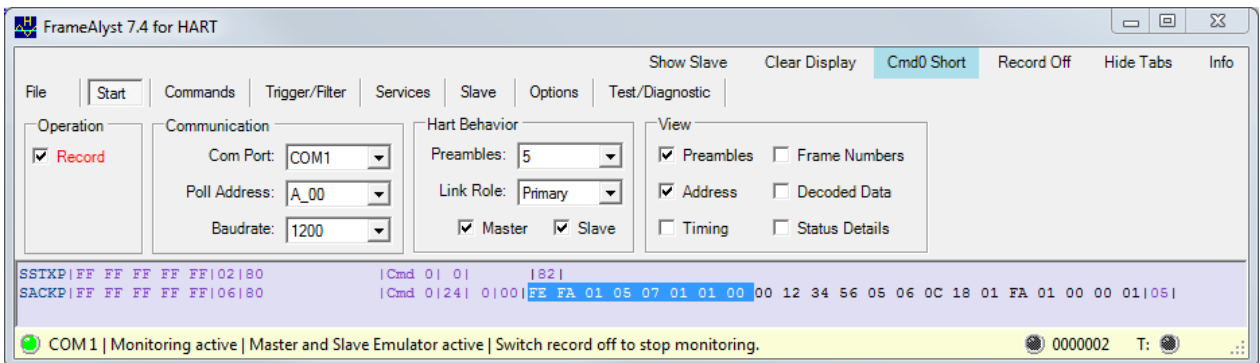


Select the whole frame, click the right mouse button and click 'Copy to AnyFrame' in the context menu.

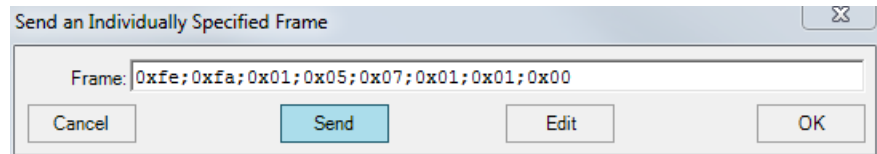
The data will be copied to this function and the edit any frame window will open.



It is also possible to copy only a part of the data.



It will appear as is in the any frame editing function.



## Copy Bytes to the Clipboard

The same functionality as shown allows also to copy data bytes to the Windows clipboard by selecting 'Bytes to ClipBoard' in the context menu.

## Editing Data Syntax

Data syntax allows to easily specify a stream of bytes to be send.

Prefix	Type	Example	Comment
None	Decimal or Hexadecimal	24; 0x18	The software will determine the required length
dec8, dec16, dec24, dec32	Decimal number	dec16; 1011	
bin8, bin16, bin24, bin32	Binary number	bin8; 10001101	
hex8, hex 16, hex24, hex32	Hexadecimal number	hex16; fa13	
float32	Single precision	float32; 1.34	
float64	Double precision	float64; 1.11e+48	
pca6, pca12, pca24	Packed ascii	pca6;LITT1400	pca6 = 8 characters pca12 = 16 characters pca24 = 32 characters
str8, str16, str32	Fixed length string	str32;my-device	Resulting byte array will be filled by 0s

All items the prefix and the data element are separated by a colon ';'.  
;

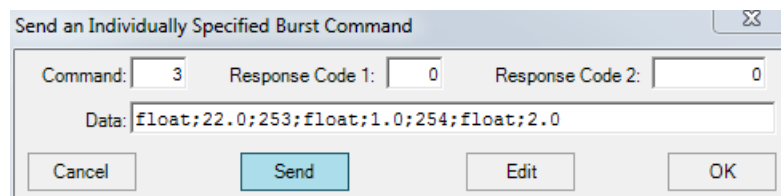
```

Data Syntax
Pasc6;LIT140;Pasc12;TEMPERATURE;15;12;113
32;Float32;150.0;Float32;0.0
str32;32 characters iso latin 1
    
```

A few examples are shown above

However, it could be much easier to do this by the data syntax editor.

When editing a command that requires data to be specified

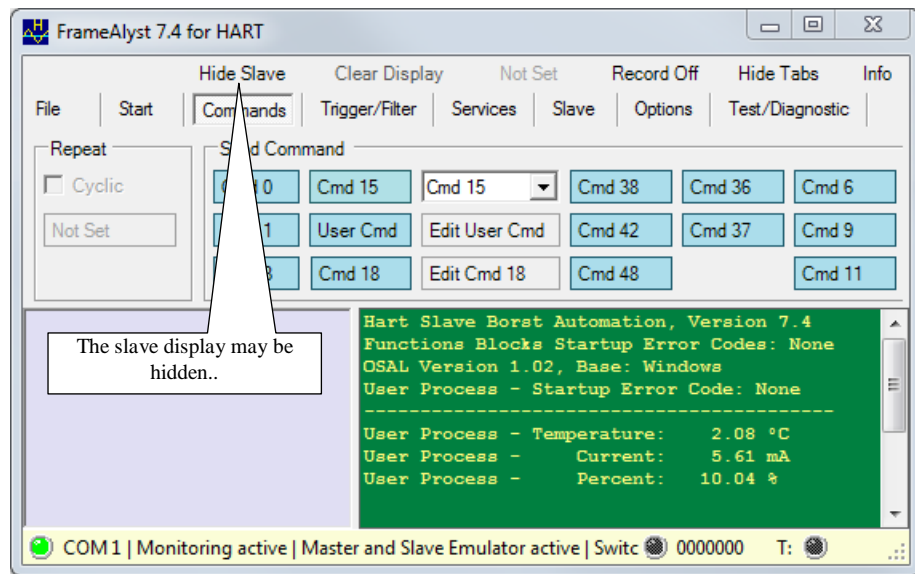


the data syntax editor will open on a click of the edit button.

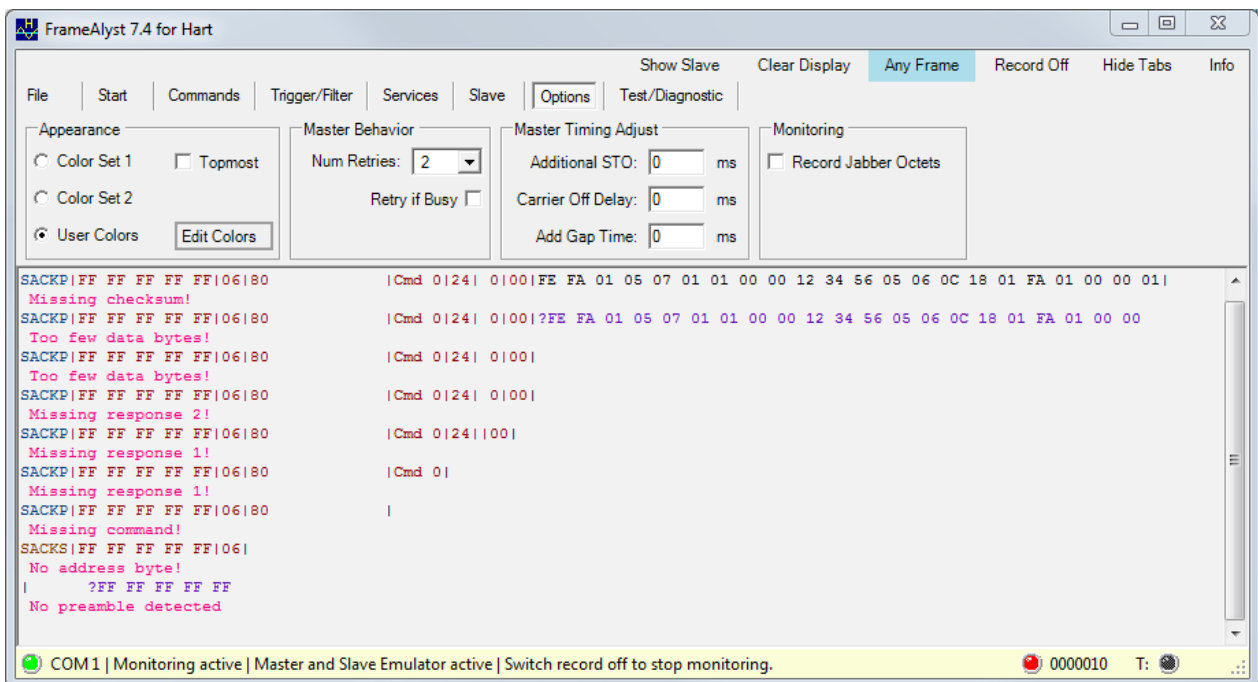
No	Item	Data
1	Float32	22.0
2	Int8	253
3	Float32	1.0
4	Int8	254
5	Float32	2.0

## Displaying the Slave Emulation

If the slave emulation is active, FrameAlyst provides a callback to the slave simulation which is used by this software for printing text with the printf function in the C libraries.

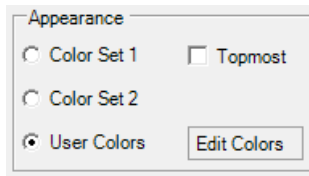


## Handling of Erroneous Frames



FrameAlyst is displaying the results while trying to read frames.

## Setting Custom Colors

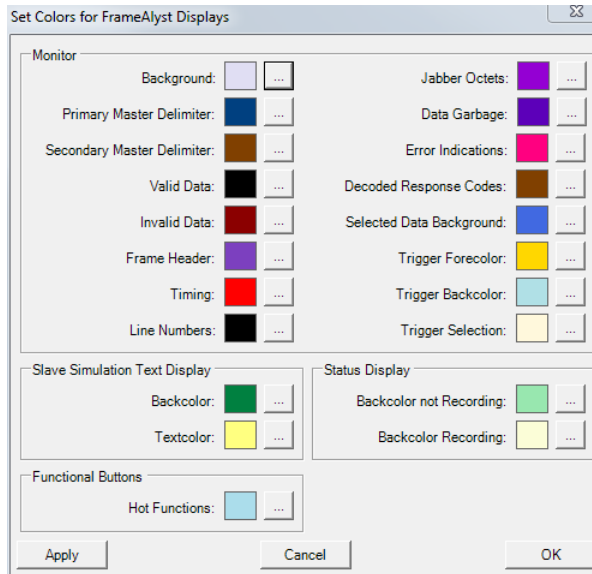


The tab Options is providing User Colors.

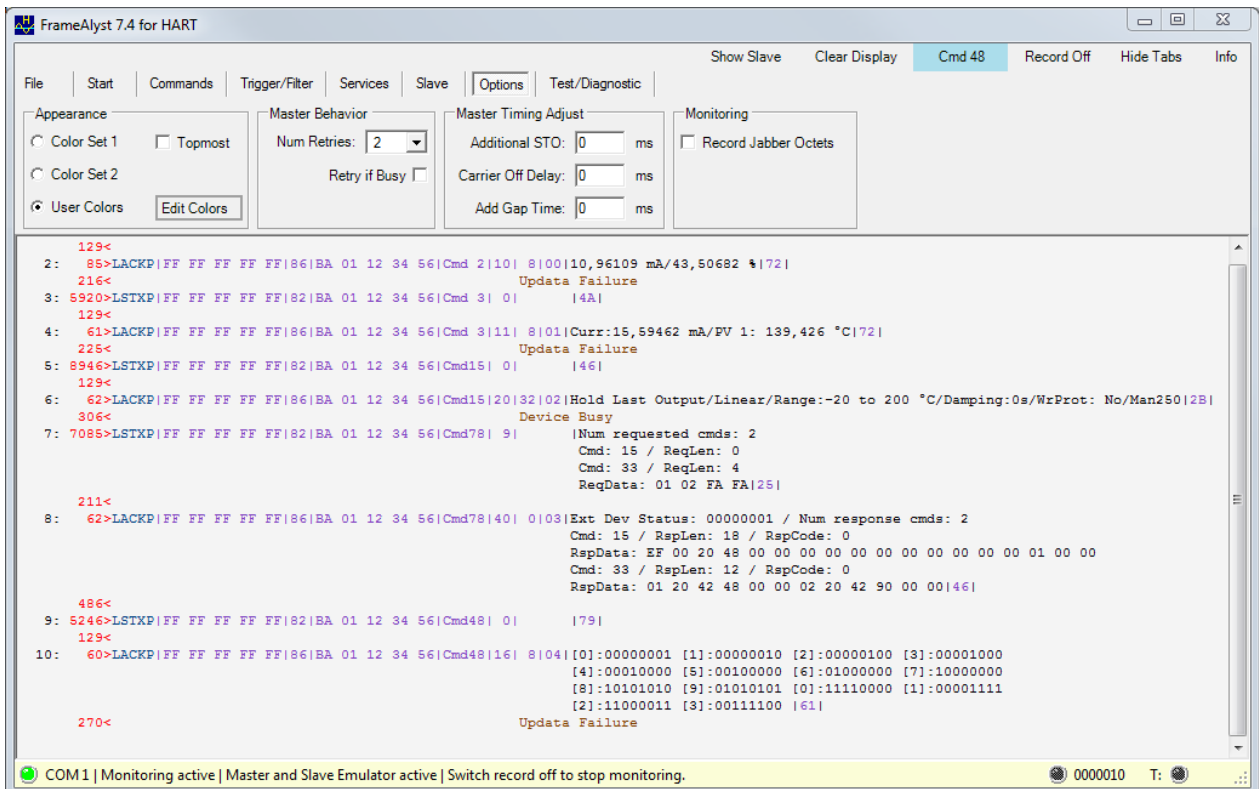
The user colors can be edited by clicking the button 'Edit Colors'.

The color editing form is shown in the

following.



## Frame Display Examples



FrameAlyst is decoding data of some commands.