19-Channel Fiberoptical Fluorometer

H. Hermes

November 20, 2002

Contents

1 Introduction

2 Electrical and Optical Connections

There are 4 electrical connections at the fluorometer:

- At the rear side is a 230 V 50 Hz AC power connector.
- The two upper SUB-D connectors at the front side are connectors for the CAN-Bus, a field-bus used for the transmission of the data (see first paragrpah of section 2.1.1 how to connect the devices).
- The lower connector is a RS-232 interface to be connected to a terminal for setting up the device parameters (see section 3). This connection is only needed to change the parameters. During normal operation it is not used.

2.1 The CAN-bus

2.1.1 General Information on the CAN-bus

The CAN-bus is a serial bus, to connect multiple devices. The devices are connected in a daisy-chain, i.e. all devices are connected to the others, forming a line. The two CAN-bus connectors on the front side of the fluorometer are equal, there is no difference between them. The order of the devices in the chain is also at random. Both ends of the chain should be terminated by an appropiate resistor.

The protokoll of the bus is message-orientated: The data is send in packages resp. messages. Each message contains an identifier and up to 8 data bytes. The identifier may vary between 0 and 2032 and is used to identify the message. The policy is, that all devices are listening to the bus. At arrival of a message, each device decides, if it ignores the message or reads the data for further treatment. On the other hand, each device sends data to the bus which could be interessant to other devices.

Three things must be obeyed, if a device should be connected to the bus:

1. All devices must use the same baud rate.
2. A device must know the identifier of the messages it wants to process.
3. If it sends data must know the identifier, with which the data is expected on the bus.

This are only the most fundamental informations you must know to use the fluorometer. There are books written on this theme.

2.1.2 The CAN-bus in the Fluorometer

The controller in the fluorometer can only handle up to 16 identifiers. So a artifice must be used to send the information of 20 fluorometer channels (19 + dark value).

The fluorometer uses 10 identifiers of which each identifies the information of 2 channels. The data is stored in the 8 byte long message. This can be interpreted as an array of 4 integers of 16 bit length. The first integer value is the gain of the first channel, the second is the signal of this channel. The other 2 integers contain the same information of the second channel.

For further information see section 3.

2.2 The RS-232 bus

This bus is designed to be connected to a serial port of a PC. You can perform all actions with any terminal program. There is no special program needed. The transmission parameters are 9600bd, 8 bits, 1 stop-bit, no parity, hardware handshake.

To test the transmission, connect the computer with the fluorometer, start the computer, start the terminal program and switch on the fluorometer. Now the version number of the fluorometer should appear on the terminal window and key-strokes should be echoed to the terminal.

If there are problems, check the terminal parameters like baud rate, parity, etc. and the cable.

This interface is used to set up or change some parameters of the fluorometer and can be disconnected during normal operation.

3 Setting up the Parameters

Before you use the fluorometer, you have to set up some parameters. Therfore you have to connect the serial port of your PC to the fluorometer as described in section 2.2. After setting up the parameters you may remove this connection. It is not needed for normal operation.

The most important parameters are:

1. The transmission speed of the CAN-bus.
2. The base of the CAN-bus identifier.
3. The controll-voltage of the Photomultiplier

There are also some Parameters of minor importantce.

3.1 Setting up the transmission speed

As mentiond in section 2.1.1 all devices at the CAN-bus must have the same transmission speed. You can set up this speed. To show the currently used speed, type in <D><S><Enter> (note: use upper case letters, <ENTER> stands for the enter-key). Unfortunatelly the speed is not shown directly, but by an index:

index	speed	index	speed	index	speed
0	1 MBd	3	125 kBd	6	20 kBd
1	500 kBd	4	100 kBd	7	10 kBd
2	250 kBd	5	50 kBd	8	5 kBd

To set the speed, type <S><S><Enter>. The current speed is displayed and you are asked to enter the new speed. Type the index and terminate with <Enter>.

3.2 Setting up the CAN-bus Identifiers

As described in sections 2.1.1 and 2.1.2 the identifiers of the CAN-bus messages must be defined. The fluorometer uses 10 subsequent identifiers. To show the first identifier which is currently used, type in <D><I><Enter>. To change this setting, type <S><I><Enter>. You will be asked for the new identifier. If you set this number to e.g. 10 the fluorometer will use the identifiers 10, 11, 12, . . . , 19.

3.3 Setting up the Controll-Voltage of the Photomultiplier

The fluorometer uses a Photomultiplier as detector. The sensitivity of this detector is highly dependend on the high-voltage applied to its electrodes. This high-voltage can be changed by changeing a controll-voltage. The currently used value you can display by typeing <D><V><Enter>. To change this value, type <S><V><Enter>. You may enter values between 0 and 100. the higher the value, the higher the voltage and the higher the sensitivity of the Photomultiplier. You can enter values bigger than 100, but this will not increase the voltage.

3.4 Setting up the Thresholds for the Gain

The gain will automatically change, if the signal is too low or high. With <D><L><Enter> you can display the limits, where this changes take place. With <S><L><Enter> this limits can be changed.

3.5 Setting up the Integration Times

The amplifier of the fluorimeter integrates the signal over a givn time. This time is higher for the higher gain setting, so more of the signal will be accomodated.

You can choose a factor, which will determine the integration time. The factor may vary between 1 and 255. The integration time for the lower gain will be calculated by $factor(low) \times 5\,\mu\mathrm{s}$. The longer integration will endure $factor(high) \times 100\,\mu\mathrm{s}$.

The default values are for low gain 30 ($\times 5\mu\mathrm{s} = 150\,\mu\mathrm{s}$) and for high gain 150 ($\times 100\,\mu\mathrm{s} = 15\,\mathrm{ms}$).

You can display the factors of the integration time by typing <D><D><Enter> (delay). To specify the times type <S><D><Enter>.

You may wish to change this values, if you want higher or lower signal levels or if you want another relation between the gains than a factor 100. (In the measured data the used gain is indicated by 1 or 100 anyway.)

3.6 Choosing fixed or variable gain

Normally the gain is choosen automatically. With <T><V><Enter> you can toggle between automatic and fixed gain. In fixed gain mode you can toggle between high and low gain by typeing <T><G><Enter>.

3.7 Interchanging Left and Right Side LED's

The fluorimeter has two arrays of light sources. In the first series of devices, the order of measurement was first the right, then the left ones. This was a bit confusing. To be compatible to this devices, the newer ones allow to interchange the left and right light sources by software: Just type <T><O><Enter>.

3.8 Initializing the Parameters

By typing <I><Enter> the Parameters are initialized. this will destroy all your settings.

3.9 Other Parameters

There are some more commands for debugging. Since this are not neccessary for the normal user, they are not documented here. If you type some wrong key, abort with the <ESC>-key and try again.

4 Usage With the Program “Measure”

The program “Measure” is a quite universal tool for performing measurements. There exists an independent documentation on this program (unfortunately only in german language).

When declaring inputs in the configuration file of the program “Measure”, you can provide a device and a port and sub-port number. As device you select your CAN-bus interface. Insert the identifier as port-number. The sub-ports 4 and 5 are the gain and the signal of the first channel whereas the sub-ports 6 and 7 give the same information for the second channel. The format of the data is described in section 2.1.2. You obtain the integer values out of a can message

The special

For those, who do not want to read full documentation of the program “Measure”, a sample configuration file is included:

```
# This is a sample configuration file for one 19-channel
# fiberoptical fluorometer. This file contains comments, so
# it can be used as a short documentation to the program "Measure".
# "Measure" is capable of a lot more features, but this will do
# for the moment.
#
# To start a measurement type in "Measure llf.cnf" at the command
# line of a terminal.

# A CAN-dongle is used to read the data
#       +------------------------------ Name of the device, choose any
#       |    +------------------------- Indicates the Type of the device
#       |    |           +------------- name of the special file in the
#       |    |           |              Linux file-system
#       |    |           |           +- baud rate
#       v    v           v           v
device: CAN, CAN-dongle, /dev/pcan24, 250k

# Use no calibration
#           +--------- Name of the calibration, choose as you want
#           |     +--- Type of the calibration
#           v     v
calibration: none, none

# Define the base address of the fluorimeter e.g. 64
define: BASE, 64

# Define the inputs. The device uses the CAN-bus
# identifiers 0, 1, ... , 9
#
#     +---------------------------- Name of the input, choose at your own
#     |                             free will
```

```
#      |   +------------------------ Device (defined above)
#      |   |    +------------------- Port
#      |   |    |        +---------- sub-port
#      |   |    |        |  +------- Calibration (use the data as they come)
#      |   |    |        |  |     +- in this format the data is stored (C-syntax)
#      v   v    v        v  v     v
input: g0, CAN, BASE + 0.4, none, "%3.0f"      # channel 0
input: s0, CAN, BASE + 0.5, none, "%5.0f"

input: g1, CAN, BASE + 0.6, none, "%3.0f"      # channel 1
input: s1, CAN, BASE + 0.7, none, "%5.0f"

input: g2, CAN, BASE + 1.4, none, "%3.0f"      # channel 2
input: s2, CAN, BASE + 1.5, none, "%5.0f"

input: g3, CAN, BASE + 1.6, none, "%3.0f"      # channel 3
input: s3, CAN, BASE + 1.7, none, "%5.0f"

input: g4, CAN, BASE + 2.4, none, "%3.0f"      # channel 4
input: s4, CAN, BASE + 2.5, none, "%5.0f"

input: g5, CAN, BASE + 2.6, none, "%3.0f"      # channel 5
input: s5, CAN, BASE + 2.7, none, "%5.0f"

input: g6, CAN, BASE + 3.4, none, "%3.0f"      # channel 6
input: s6, CAN, BASE + 3.5, none, "%5.0f"

input: g7, CAN, BASE + 3.6, none, "%3.0f"      # channel 7
input: s7, CAN, BASE + 3.7, none, "%5.0f"

input: g8, CAN, BASE + 4.4, none, "%3.0f"      # channel 8
input: s8, CAN, BASE + 4.5, none, "%5.0f"

input: g9, CAN, BASE + 4.6, none, "%3.0f"      # channel 9
input: s9, CAN, BASE + 4.7, none, "%5.0f"

input: g10, CAN, BASE + 5.4, none, "%3.0f"     # channel 10
input: s10, CAN, BASE + 5.5, none, "%5.0f"

input: g11, CAN, BASE + 5.6, none, "%3.0f"     # channel 11
input: s11, CAN, BASE + 5.7, none, "%5.0f"

input: g12, CAN, BASE + 6.4, none, "%3.0f"     # channel 12
input: s12, CAN, BASE + 6.5, none, "%5.0f"

input: g13, CAN, BASE + 6.6, none, "%3.0f"     # channel 13
input: s13, CAN, BASE + 6.7, none, "%5.0f"

input: g14, CAN, BASE + 7.4, none, "%3.0f"     # channel 14
input: s14, CAN, BASE + 7.5, none, "%5.0f"
```

```
input: g15, CAN, BASE + 7.6, none, "%3.0f"     # channel 15
input: s15, CAN, BASE + 7.7, none, "%5.0f"

input: g16, CAN, BASE + 8.4, none, "%3.0f"     # channel 16
input: s16, CAN, BASE + 8.5, none, "%5.0f"

input: g17, CAN, BASE + 8.6, none, "%3.0f"     # channel 17
input: s17, CAN, BASE + 8.7, none, "%5.0f"

input: g18, CAN, BASE + 9.4, none, "%3.0f"     # channel 18
input: s18, CAN, BASE + 9.5, none, "%5.0f"

input: g19, CAN, BASE + 9.6, none, "%3.0f"     # channel 19 (dark value)
input: s19, CAN, BASE + 9.7, none, "%5.0f"


# Define the format, in which the data is displayed (C-syntax)
display.default.format = "%5.0f"

# defien where to display the data
#        |   +------- The data is displayed in this row...
#        |   |  +---- ... and this column
#        v   v  v
display: g0, 0, 0
display: s0, 0, 1

display: g1, 0, 2
display: s1, 0, 3

display: g2, 0, 4
display: s2, 0, 5

display: g3, 0, 6
display: s3, 0, 7

display: g4, 0, 8
display: s4, 0, 9

display: g5, 1, 0
display: s5, 1, 1

display: g6, 1, 2
display: s6, 1, 3

display: g7, 1, 4
display: s7, 1, 5

display: g8, 1, 6
display: s8, 1, 7

display: g9, 1, 8
```

```
display: s9, 1, 9

display: g10, 2, 0
display: s10, 2, 1

display: g11, 2, 2
display: s11, 2, 3

display: g12, 2, 4
display: s12, 2, 5

display: g13, 2, 6
display: s13, 2, 7

display: g14, 2, 8
display: s14, 2, 9

display: g15, 3, 0
display: s15, 3, 1

display: g16, 3, 2
display: s16, 3, 3

display: g17, 3, 4
display: s17, 3, 5

display: g18, 3, 6
display: s18, 3, 7

display: g19, 3, 8
display: s19, 3, 9

# At last define what is to be done, and when it is to be done.
# Everything between the lines "repeat: ..." and "repeat: end"
# is repeated as defined.
#       +----------- ignore this
#       |  +-------- measure all 2 seconds
#       |  |  +----- collect 5 values, then store the average
#       |  |  |  +-- store the data to this file
#       v  v  v  v
repeat: 0, 2, 5, llf_01.dat
  read: g0
  read: s0

  read: g1
  read: s1

  read: g2
  read: s2

  read: g3
```

```
  read: s3

  read: g4
  read: s4

  read: g5
  read: s5

  read: g6
  read: s6

  read: g7
  read: s7

  read: g8
  read: s8

  read: g9
  read: s9

  read: g10
  read: s10

  read: g11
  read: s11

  read: g12
  read: s12

  read: g13
  read: s13

  read: g14
  read: s14

  read: g15
  read: s15

  read: g16
  read: s16

  read: g17
  read: s17

  read: g18
  read: s18

  read: g19
  read: s19
repeat: end
```