Butterfly Avionics

Butterfly Connect

Integration Manual

Document name:	18.B104.2-1.0-EN
Document version:	0.4
Release date:	04/02/2013

Butterfly Avionics GmbH – Hauptstrasse 93/1 – D-69207 Sandhausen
Phone: +49 (0) 6224 82 83 87 0 – Fax: +49 (0) 6224 82 83 87 7 – Internet: www.butterfly.aero – eMail: support@butterfly.aero

Notes

General Information

IMPORTANT!

> ***Please read this manual carefully!***

> ***Please pay attention to the restrictions on use!***

Document identification / revision status

This manual supports the following product types:

- P/N B104-001 ”Butterfly Connect”

Actual version: **Connect Integration Manual • 18.B104.2-1.0-EN, Version 1.0**

Version history

Revision	Date	Status	Author	Changes, comments	Approved
0.1	2013/01/17	prerelease	M. Foerderer	Initial Version	-
0.2	2013/01/18	prerelease	M. Foerderer	Details Added	-
0.3	2013/01/18	prerelease	M. Foerderer	Minor Changes	-
0.4	2013/02/04	Release	M. Foerderer	Added SoftAP FW Infos	-
0.5	2013/02/04	Release	M. Foerderer	Added new version infos	-

Referenced documents

Reference	Title	Version	Date	Author
18.104.1-1.0-EN	Connect Pilot’s and Instal.. Manual	1.0	17.01.2012	Butterfly
RN-WIFLYCR	WiFly Command Reference	1.0r	21.09.2012	Roving Networks

Contents

1 General

Butterfly Connect is a compact, lightweight and rugged device wirelessly connecting Smartphones such as Apple® iOS®-devices or Android-devices to aircraft systems.
Butterfly Connect creates an IEEE 802.11 b/g wireless network (WiFi® soft Accesspoint) and transmits RS232 data via the TCP/IP protocol. Many Smartphone-Apps are capable of displaying and using such data. Examples for data are current GPS-position, pressure altitude, traffic or engine data.
It features one central Power/Data connector (D-SUB 9) and a standard SMA antenna connector including a small quarter-wave antenna. Butterfly Connect is supplied by a DC source in the range of 4.8V to 32V and typically consumes less than 100mA. It can be easily mounted in many places inside the aircraft, remote antenna options with cables are available.

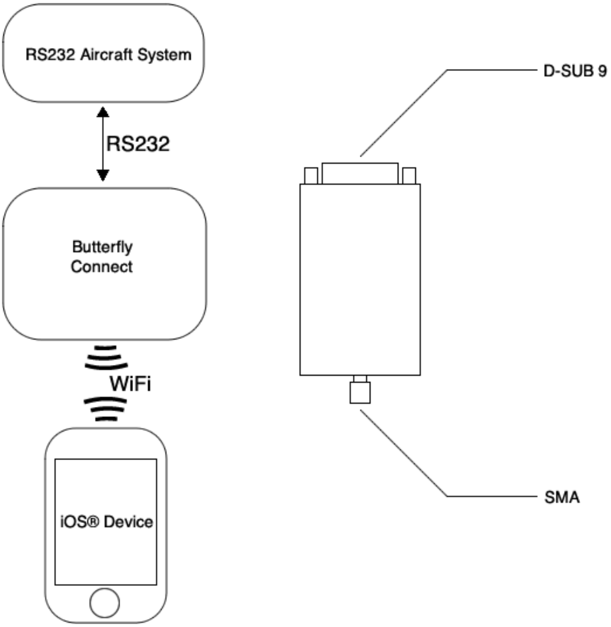

Figure 1: System Diagram

1.1 Safety instructions and restrictions on use

Installation and operation must be on the basis of non-interference with and no hazard to the existing suite of other equipment necessary for safe flying operation, or installed to comply with official requirements. Installation and operation must comply with official regulations and requirements.

> ***Never make safety critical decisions based on transferred information. Make sure to the end users of your App that displayed data are for situational awareness only.***

> ***Butterfly Connect does not have a JTSO or FAA-TSO airworthiness certification.***

1.2 Intellectual Property and Liability

Butterfly Avionics GmbH will not be liable for errors/changes/omissions in this document - specifications are subject to change without notice. Butterfly Avionics its associates, development team, suppliers, manufacturers and data suppliers accept no responsibility for any damage or claims that may arise from use of Butterfly Connect.

Trademarks referred to in this document are the property of their respective holders. Note that this document as well as the interface described with it are proprietary and copyright protected. Any non-licensed use, dissemination, copying, implementation, reverse engineering or decompilation is forbidden by law and will be prosecuted.

1.3 Developer Support

Please contact us before starting development on a project involving Butterfly Connect.
support@butterfly.aero or +49 (0) 6224 82 83 87 0

> *For detailed descriptions on the use of Butterfly Connect, please refer to the* Butterfly Connect Pilot's and installation Manual

2 Hardware

2.1 Connectors and Wiring

2.1.1 Main Power and Data Connector

Butterfly Connect features one central D-SUB 9 connector for power and data.

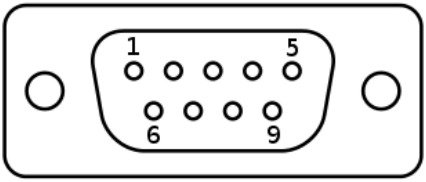

Figure 2: Main D-Sub 9 Power/Data connector

Pin Number D-SUB 9	Signal/Purpose
1	Do not connect!
2	RS232 RX - connect receives data
3	RS232 TX - connect sends data
4	Do not connect!
5	GND - main ground (minus)
6	Do not connect!
7	DC In - main power supply (plus)
8	Do not connect!
9	Do not connect!

2.1.2 RJ45 Cable

Included in delivery is a D-SUB 9 to RJ45 cable in IGC-Standard codeout. The cable allows for easy connection to many common aircraft systems such as FLARM®-compatible collision avoidance devices or many IGC-approved GPS Loggers.

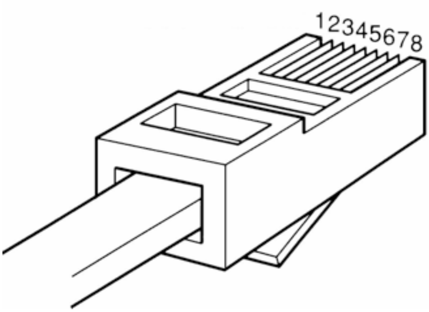

Figure 3: RJ45 Connector

Pin Number RJ45	Signal/Purpose
1	GND
2	Do not connect!
3	RS232 TX - connect sends data
4	RS232 RX - connect receives data
5	Do not connect!
6	Do not connect!
7	Do not connect!
8	DC IN - main power supply (plus)

> *Aditional Adapter Cables, Y-Splitters are available at Butterfly Store: www.butterflystore.aero*

2.1.3 Power Consumption and Requirements

Item	Value
Input Voltage Range	4.8V to 32V DC
Recommended Fusing	2.0A CB
Typical current	80mA at 12V DC
Peak current	350mA at 12V DC
Power Requirements	<1.5W at 12V DC

> ***Take care that the power supplied has the correct polarity, otherwise damage to the device may occur.***

3 Interfacing

3.1 RS232 Interface

Butterfly Connect is compatible to any aircraft system that delivers data via the TIA-232-F (”RS232”) standard. Butterfly Connect features bidirectional communication. Data received via RS232 is sent wirelessly to Smartphone-devices and data from Smartphone-devices is sent to RS232 aircraft systems.

3.1.1 RS232 Data-Rate

The RS232 interface features configurable data-rates. Butterfly Connect comes preconfigured with a data-rate of 19200 Baud. Other data-rates are possible. Please refer to the configuration part of this manual for details on data-rate configuration.

3.2 WiFi® Interface

3.2.1 Network Mode and Protocol

Butterfly Connect connects to other devices through a WiFi® connection. It then delivers received RS-232 data using the TCP/IP protocol on a fixed IP-Address and Port.

Item	Value
Network Type	Soft Accespoint
Network Security	None
IP-Adress	1.2.3.4
Port	2000
SSID	WiFly-GSX-XX

> *The SSID of the network always consists of* WiFly-GSX- *or* WiFly-EZX- *and a two digit hexadecimal identifier.*

3.2.2 Connection and Data transfer

Butterfly Connect delivers received RS-232 data using the TCP/IP protocol. The four-digit pin-code is required and requested after connection to the module has been established.

- Data transfer

  - Connect via TCP-IP to the defined IP-address using the defined port
  - The Module will now answer requesting the pin-code with the phrase *PASS?*
  - Authenticate sending the four-digit pin-code
  - After authentication RS232 data is transferred

3.2.3 Code Examples

A code example written in C can be found in the appendix of this document. Please contact us if you experience problems.

3.2.4 Joining the Network with iOS®- or Android Devices

In order to establish a connection to a mobile device (end user) the following process is recommended.

- Joining for mobile-devices

  - Power up Butterfly Connect
  - Enter main Settings App.
  - go to *Wi-Fi* and make sure Wi-Fi is activated
  - in the *Choose a Network* list the SSID of your Butterfly Connect device should appear. The SSID is always named *WiFly-GSX* followed by a two digit code.
  - tap on the *WiFly-GSX*-Network to create a connection

> *There is no network security. Any device is able to Join the Network. Currently only one device may join at a time.*

> *Make sure your device remains connected, in some cases open WiFi-Networks are joined automatically which causes Butterfly Connect to loose connection. Activate* Ask to Join Networks *in iOS® to avoid this.*

4 Configuration

4.1 General Information

Butterfly Connect may be configured using a computer with a terminal software or any other device that is connected to Butterfly Connect via WiFi® or RS232.

Platform	Recommended Software
MAC OS X®	Terminal
iOS®	Terminal App
Microsoft Windows®	Hyperterminal

> *After Configuration, Butterfly Connect has to be rebooted in order for the new configuration to take effect.*

4.2 Connecting with a PC

4.2.1 Connection via WiFi®

- Connecting

- Power up Butterfly Connect, data connection to RS232 devices is not required.

- connect Butterfly Connect to your computer using WiFi®.

- open your terminal software, type the following command and execute it with typing *enter*: *telnet 1.2.3.4 2000*

4.2.2 Connection via RS232

- Connecting

- Power up Butterfly Connect, connect it using an RS232 cable, data connection via WiFi is not required.

- Start a serial connection at 19200Baud, 8N1, no flow control.

- You should now see some sentences at startup.

4.3 Configuration Mode

- Entering configuration mode

- Power up Butterfly Connect and connect via desired method

- Butterfly Connect will now answer requesting the 4-digit PIN-code printed to the top of the enclosure.

- Enter the PIN-code and execute with *enter*.

- Enter *$$$* and execute with *enter*.

- Commands in configuration mode

  - To save configurations, type *save* and execute with *enter*.

  - To exit configuration mode, type *exit* and execute with *enter*.

  - To reboot the module, type *reboot* and execute with *enter*.

4.3.1 Configuration of Data-Rate

To configure the RS232 data-rate, first enter configuration mode and then type *set u b 19200* and execute with *enter*.
In this case *19200* may be interchanged with any desired baud-rate. Common data rates are *4800* for standard NMEA or *9600* for GARMIN® TIS. One Example is shown below.

4.3.2 Configuration of PIN-Code

> ***Carefully write down and remember your set pin-code. There is no way to reset the module if your pin-code went missing. Modules with unknown pin-code may be rendered unusable***

To configure the four-digit pin-code (first configuration printed on top of Butterfly Connect), first enter configuration mode and then type *set o p 1111* and execute with *enter*.
In this case *1111* may be interchanged with any desired pin-code.

4.3.3 Further Configuration

please consult the *Roving Networks RN171 usermanual* (see referenced documents table) for further configuration options.

4.3.4 Configuration Example: Set RS232 Data-Rate

In this example the RS232 data-rate is set to the value 4800Baud.

- Enter configuration mode

  - Power up Butterfly Connect, data connection to RS232 devices is not required.

  - connect Butterfly Connect to your computer using WiFi®.

  - open your terminal software, type the following command and execute it with typing *enter*: *telnet 1.2.3.4 2000*

  - Butterfly Connect will now answer requesting the 4-digit PIN-code printed to the top of the enclosure.

  - Enter the PIN-code and execute with *enter*.
  - Enter *$$$* and execute with *enter*.

- Set data-rate to 4800Baud

  - type *set u b 4800* and execute with *enter*

• Save Settings and exit	- type *save* and execute with *enter* - type *exit* and execute with *enter*.

> *After Configuration, Butterfly Connect has to be rebooted in order for the new configuration to take effect.*

4.4 Module Firmware Updates

Butterfly Connect features field-upgradeable firmware

> *Security for the connection will be available in future releases of the module firmware.*

• Preparing the Update	- Connect the module via RS232 and enter configuration mode
• Connecting module to a WiFi® network with internet access	- Enter *set wlan ssid x*, enter the networks SSID instead of ”x” and execute with *enter*. - Enter *set wlan pass x*, enter the networks password instead of ”x” and execute with *enter* - Enter *save* and execute with *enter*. - Enter *join* and execute with *enter*.
• Downloading and installing firmware	- Enter *set ftp address 0* and execute with *enter*. - Enter *set dns name rn.microchip.com* and execute with *enter*. - Enter *ftp Update wifly-EZX-AP.img* and execute with *enter*. - Enter *reboot* and execute with *enter*.

> *wifly-EZX-AP.img is the AP-mode Firmware ID, wifly-EZX.img an AdHoc mode firmware*

• Resetting module for changes to take effect	- Connect the module via RS232 and enter configuration mode. - Enter *factory R* and execute with *enter*. - Enter *reboot* and execute with *enter*.

4.5 More information about connected devices

4.5.1 Showing devices connected to the module via WiFi®

• Showing connected devices	- Connect the module via RS232 and enter configuration mode. - Enter *show lease* and execute with *enter*. - Enter *exit* and execute with *enter* if you want to leave the dialog.

5 Version Information

5.1 Available Software versions

There are two different software versions available in the filed. All units delivered before May of 2013 contain a different software version than new modules. The following table shows the main differences.

	Old Version	**New Version**
Module SSID	WIFLY-GSX	WIFLY-EZX
IP-Address	169.254.1.1	1.2.3.4
Network Mode	Ad-Hoc	Soft AccessPoint
Serial Number	lower than B104-1100	Starting with B104-1101

> *The four-digit PIN-CODE is alway calculated by adding the number* 3642 *to the devices serial number. Therefore individual device serial numbers may be calculated from the pin-code entered by the user.*

6 Appendix

Code example - Establishing a TCP/IP Connection to the Module

The following example shows the establishing of a TCP/IP connection to the module in C

```
/*
 *
 * Example code for the Butterfly Connect WiFi module
 *
 * 2012 by Butterfly Avionics GmbH, Tobias Fetzer
 *
 * For any questions send an email to support@butterfly.aero
 *
 * TODO:
 *
 * - Replace the PASSWORD define below with the password, which
 *   was provided with your Butterfly Connect module.
 *
 * - Connect your pc to the modules' AdHoc network (Wifl-GSX-XX)
 *
 */

#include <stdio.h>
#include <stdlib.h>
#include <sys/socket.h>
#include <netinet/in.h>
#include <netdb.h>
#include <errno.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/time.h>
#include <string.h>

#define PORT        2000
#define HOSTNAME    "1.2.3.4"
#define PASSWORD    "1234"

int sock;
int connectionEstablished = 0;

void init_sockaddr (struct sockaddr_in *name,
                    const char *hostname,
                    unsigned short int port);

int initConnection();
int sendData(char *data, int len);
int main(int argc, char *argv[])
{
  if (initConnection())
  {
    int quit = 0;
    char buffer[512];
    int nBytes;

    while (!quit)
    {
      nBytes = read(sock, buffer, 512);

      // if data read..
      if (nBytes > 0)
      {
        buffer[nBytes] = ’\0’;
        printf(buffer);
        // if the module requests a password..
        if (strstr(buffer, ”PASS?”) != NULL)
        {
          printf(”\nSending␣password:␣%s\n”, PASSWORD);
          sendData(PASSWORD, strlen(PASSWORD));
        }
      }
    }
  }
}

void init_sockaddr (struct sockaddr_in *name,
                    const char *hostname,
                    unsigned short int port)
{
  struct hostent *hostinfo;

  name->sin_family = AF_INET;
  name->sin_port = htons (port);
  hostinfo = gethostbyname (hostname);

  if (hostinfo == NULL)
  {
    fprintf (stderr, ”Unknown␣host␣%s.\n”, hostname);
    exit (EXIT_FAILURE);
  }
  name->sin_addr = *(struct in_addr *) hostinfo->h_addr;
}
```

```
int initConnection()
{
  // init the connection
  struct sockaddr_in servername;

  connectionEstablished = 0;

  // Create the socket.
  sock = socket(PF_INET, SOCK_STREAM, 0);
  if (sock < 0)
  {
    perror("initConnection:␣create␣socket␣failed");
    return 0;
  }

  // set SO_REUSEADDR on a socket to true (1):
  struct timeval optval;
  optval.tv_sec = 0;
  optval.tv_usec = 500;

  setsockopt(sock, SOL_SOCKET, SO_RCVTIMEO, &optval, sizeof(optval));

  // Connect to the server.
  init_sockaddr (&servername,HOSTNAME, PORT);
  if (connect(sock, (struct sockaddr *) &servername, sizeof (servername)) < 0)
  {
    perror("connect␣failed");
    return 0;
  }

  connectionEstablished = 1;

  return 1;
}

int sendData(char *data, int len)
{
  if (connectionEstablished)
  {
    int nbytes = write (sock, data, len);
    if (nbytes < 0)
    {
      perror("send␣data␣failed.");
      return 0;
    }
```

```
        return 1;
    }
    return 0;
}
```