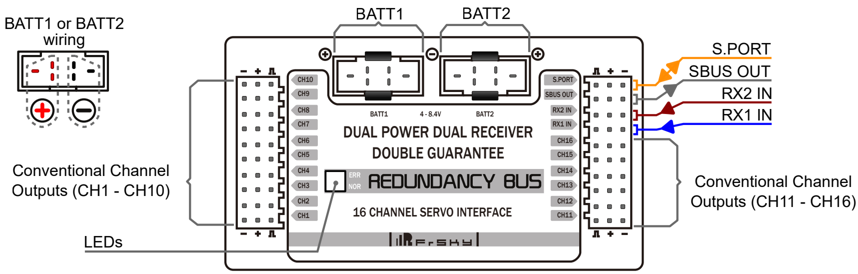
Instruction Manual for FrSky REDUNDANCY BUS

NOTICE : All instructions, warranties and other collateral documents are subject to change at the sole discretion of FrSky Electronic Co., Ltd. For Further information, please visit www.frsky-rc.com.

Overview



- **CH1~CH16** - Connect up to 16 servos (PWM).
- **S.PORT** - Connect to the S.Port of X series receiver and feedback integrated S.Port values (including voltage, current, capacity, overload indication, etc...).
- **SBUS OUT** - SBUS output, max overload current of 2.6A.
- **RX1 IN** - Connect to the SBUS port of X series receiver and supply power directly to the connected receiver, max overload current of 2.6A.
- **RX2 IN** - Connect to the SBUS port of X series receiver and supply power directly to the connected receiver, max overload current of 2.6A.
- **BATT1 and BATT2** - MPX connectors for batteries or BEC connection, supply power to the REDUNDANCY BUS and connected receivers

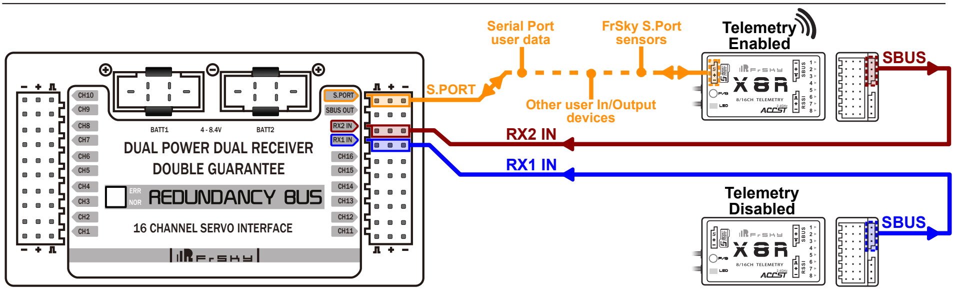
Features

- Overload protection on each channel
- Dual power dual receiver double guarantee (connect up to 2 batteries and 2 receivers)
- 50Hz mode of servo outputs (20ms period) or equal to the input of SBUS cycle
- Integrated S.Port telemetry feedbacks (including voltage, current, capacity, overload indication, etc)
- Servo signal output period settable
- High voltage servos supported
- Hot plug supported
- Compatible with both positive and negative SBUS signals (SBUS signal on FrSky receivers is negative)
- Compart size and firmware upgradable

Specifications

- Recommended input voltage range : 4V ~ 8.4V (1~2 LiPo cells or 4~6 NiMH cells)
- Number of servos : up to 16
- Operating temperature range : -20°C ~ +75°C
- Weight : 37g
- Dimension : 77 x 43 x 22mm

Connection



FrSky Electronic Co., Ltd
Website : www.frsky-rc.com Email : frsky@frsky-rc.com Technical Support : sales4tech@gmail.com

The REDUNDANCY BUS is a switchboard connected to the power supplies, receivers, servos and S/Port sensors.
The REDUNDANCY BUS does not contain circuitry to stabilize or regulate voltage to the servos. The level of the input voltage is equal to the level of (output) supply voltage to the servos. Be sure to match the proper type of servos with your selected power supply (for example : when using 2 LiPo cells without a regulator, it is necessary to use servos labeled "High Voltage").

Do not connect more than one REDUNDANCY BUS to one servo.
Do not use Y harness to connect more than one servo to each servo output.

NOTE : The REDUNDANCY BUS could be daisy-chained with FrSky Smart Port sensor via S.PORT. Use it as the last item in the chain, or use Y lead if it is between the FrSky Smart Port sensors ans the receiver.

Power Supply :

Power supply of REDUNDANCY BUS could be provided from either one battery/BEC[1] (connect via BATT1 or BATT2), or two (connect to BATT1 and BATT2). When two power supplies are used, make sure both supply a 15A continuous and 90A peak current, otherwise the overload protection feature could not function efficiently.
If the voltages of two power supplies are the same, the power can be used from both supplies at the same time;
If the voltages of two power supplies are different, the power comes from the one with the higher voltage, and each supply is isolated from each other instead shared.
Use of different capacity, number of cells and chemistry batteries is allowed.

CAUTION : Do not connect power supplies to CH1 ~ CH16, S.PORT, SBUS OUT, RX 1 IN or RX 2 IN.

Overload Protection :

The REDUNDANCY BUS has an overload protection function by PPTC through a circuit inside on each servo ouput.
If overload current happens, the affected servo output will be disconnected from the power supply while the remaining servo ouputs are still powered.

Performance Specification :

Model	V max (V dc)	I max (A)	I hold @25°C (A)	I trip @25°C (A)	Pd Typ (W)	Maximum Time To Trip		Resistance	
						Current (A)	Time (Sec)	Ri min (Ω)	R1 max (Ω)
SMD1812R260SF	12.0	100	2.60	5.00	0.8	8	2.50	0.015	0.050

V max = Maximum operating voltage device can withstand without damage at rated current (I max).
I max = Maximum fault current device can withstand without damage at rated voltage (V max).
I hold = Hold Current. Maximum current device will not trip in 25°C still air.
I trip = Trip Current. Minimum current at which the device will always trip in 25°C still air.
Pd = Power dissipation when device is in the tripped state in 25°C still air environment at rated voltage.
Ri min/max = Minimum/Maximum device resistance prior to tripping at 25°C.
R1max = Maximum device resistance is measured one hour post reflow.
CAUTION : Operation beyond the specified ratings may result in damage and possible arcing and flame.

Ambient Temperature (°C)	-40°C	-20°C	0°C	25°C	40°C	50°C	60°C	70°C	85°C
Recommended Hold Current (A)	3.90	3.42	2.96	2.60	2.33	2.07	1.94	1.35	1.00

(1) - BEC means Battery Eliminator Circuit, in other words, it is a device that replaces the receiver battery. It is a linear voltage regulator allowing, from the battery of propulsion, to produce a lower tension adapted to the supply of the receiver.
(2) - PPTC means Polymer Positive Temperature Coefficient. It is an electronic component that cuts off the current due to overconsumption that causes a rise in temperature of this component. This component returns to a conductive state once the current has dropped, acting as a circuit breaker, thus allowing the circuit to operate again without opening the case or replacing the component.

How to configure OpenTx to access telemetry data

On a FrSky radio with OPEN TX firmware version 2.2 and higher:
1- Turn on the transmitter and select the "TELEMETRY" menu of your current model;
2- In the "TELEMETRY" page, go down and select "Discover new sensors";
3- The illustration below shows the newly discovered sensors from the REDUNDANCY BUS.

Note : • The number assigned to each sensor may differ depending on the equipment of your model.
• The illustration only shows the sensors from the REDUNDANCY BUS.
• Sensors from the receiver(s) or other equipment are not voluntarily shown in the illustration.

```
TELEMETRY                                   13/14
Sensors                  Value           ID
1: RBCS        *          OK             26
2: RBS         *                         26
3: RB1C        *          15mAh          26
4: RB2C        *          0mAh           26
5: RB1V        *          5.89V          26
6: RB1A        *          0.18A          26
7: RB2V        *          5.77V          26
8: RB2A        *          0.00A          26
```

4- Select "Stop discovery",
5- You can now configure what telemetry displays 1-4 should display.

Description of native telemetry data

	Short name	Full name	Definition of the data	Unit
1	RBCS	**R**edundancy **B**us **C**hannel **S**tatus	This raw data contains several pieces of information that can be extracted using a LUA script described in the section "*Generation of virtual sensors*". The display "RBCS OK" means that there is no over-current on channels 1 to 16 of the REDUNDANCY BUS. Any other display highlights one of the information contained in this composite sensor (see "*Description of Composite Sensors*").	None
2	RBS	**R**edundancy **B**us **S**tatus	This raw data contains several pieces of information that can be extracted using a LUA script described in the section "*Generation of virtual sensors*". When there is no information in front of "RBS" this means there is no particular problem between the receivers and the REDUNDANCY BUS. Any other display highlights one of the information contained in this composite sensor (see "*Description of Composite Sensors*").	None
3	RB1C	**R**edundancy **B**us 1 **C**onsumption	Total energy used on the battery n°1	mAh
4	RB2C	**R**edundancy **B**us 2 **C**onsumption	Total energy used on the battery n°2	mAh
5	RB1V	**R**edundancy **B**us 1 **V**oltage	Current value of battery voltage n°1	V
6	RB1A	**R**edundancy **B**us 1 **A**mpere	Current value of the current from the battery n°1	A
7	RB2V	**R**edundancy **B**us 2 **V**oltage	Current value of battery voltage n°2	V
8	RB2A	**R**edundancy **B**us 2 **A**mpere	Current value of the current from the battery n°2	A

Description of the RBCS and RBS composite sensors

A FrSky radio with OPEN TX firmware can handle up to 32 sensors. If the information contained in RBCS and RBS had been directly available as native sensors, the REDUNDANCY BUS alone would use 33 sensors!
However, it remains possible to obtain one or more information from RBCS and RBS through the use of an LUA script as described in "*Generation of Virtual Sensors*".

Principle of operation of a composite sensor :

Imagine that the internal value of RBS is 1227 (this value is recoverable only with an LUA script).
This decimal value 1227 can also be written in binary by a succession of 1 and 0 called "Bit".
So we have :

$$\text{RBS} = 0b\overbrace{10011001011}^{\textbf{Binary value}} = \overbrace{1227}^{\textbf{Decimal value}}$$

Bit n°11 Bit n°10 Bit n°9 Bit n°8 Bit n°7 Bit n°6 Bit n°5 Bit n°4 Bit n°3 Bit n°2 Bit n°1

Each of the "bits" that make up the value of RBS represents an information that can only take two states:
- Active : Bit n°# = 1
- Inactive : Bit n°# = 0

RBS - Detail of composite sensor "Redundancy Bus Status"

Bit n°	Short name	Full name	Definition of the data (when Bit n°# = 1)
1	Rx1 Ovl	Rx1 **Over**l**oad**	Receiver n°1 connected to "Rx1 In" of the REDUNDANCY BUS causes over-consumption of current.
2	Rx2 Ovl	Rx2 **Over**l**oad**	Receiver n°2 connected to "Rx2 In" of the REDUNDANCY BUS causes over-consumption of current.
3	SBUS Ovl	SBUS **Over**l**oad**	The device connected to "SBUS OUT" of the REDUNDANCY BUS causes over-consumption of current.
4	Rx1 FS	Rx1 **F**ail**S**afe	Receiver n°1 has switched to FailSafe mode.
5	Rx1 LF	Rx1 **L**ost **F**rame	The receiver n°1, not being able to decode the received information, returns the last valid packet to the REDUNDANCY BUS and informs it of this state. If this situation lasts, then receiver n°1 will go into FailSafe mode.
6	Rx2 FS	Rx2 **F**ail**S**afe	Receiver n°2 has switched to FailSafe mode.
7	Rx2 LF	Rx2 **L**ost **F**rame	The receiver n°2, not being able to decode the received information, returns the last valid packet to the REDUNDANCY BUS and informs it of this state. If this situation lasts, then receiver n°2 will go into FailSafe mode.
8	Rx1 Lost	Rx1 Lost	The REDUNDANCY BUS no longer detects a physical connection to the receiver n°1.
9	Rx2 Lost	Rx2 Lost	The REDUNDANCY BUS no longer detects a physical connection to the receiver n°2.
10	Rx1 NS	Rx1 **N**o **S**ignal	The REDUNDANCY BUS detects a physical connection to the receiver n°1, but it does not receive a SBUS signal from it.
11	Rx2 NS	Rx2 **N**o **S**ignal	The REDUNDANCY BUS detects a physical connection to the receiver n°2, but it does not receive a SBUS signal from it.

RBCS - Detail of composite sensor "Redundancy Bus Channel Status"

Bit n°	Short name	Full name	Definition of the data (when Bit n°# = 1)
1	CS01	**C**hannel **S**tatus 01	Channel 1 of the REDUNDANCY BUS tripped (over-current).
↓	↓	↓	↓ ↓ ↓
16	CS16	**C**hannel **S**tatus 16	Channel 16 of the REDUNDANCY BUS tripped (over-current).

Generation of virtual sensors

Prerequisites :

- "*LUA*" compilation option checked in the "Settings > Settings..." menu of the *Companion* software version 2.2 or higher available at http://www.open-tx.org/
- FrSky Radio with OPEN TX firmware version 2.2 or higher with the option listed above.
- Micro SD card in the radio with, at its root, the tree "*SCRIPTS/MIXES/*" in which will be placed the file "***rb_dec.lua***" described below.

For more information on updating your FrSky radio with OPEN TX,
visit http://www.open-tx.org/ and https://www.rcgroups.com/forums/index.php.

LUA script named "*rb_dec.lua*" :

Using a text editor, enter the program below and save it as "***rb_dec.lua***".

```lua
--
-- Copyright (C) OpenTX
--
-- License GPLv2: http://www.gnu.org/licenses/gpl-2.0.html
--
-- This program is free software; you can redistribute it and/or modify
-- it under the terms of the GNU General Public License version 2 as
-- published by the Free Software Foundation.
--
-- This program is distributed in the hope that it will be useful,
-- but WITHOUT ANY WARRANTY; without even the implied warranty of
-- MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.  See the
-- GNU General Public License for more details.
--
-- LUA Mix Script
local outputs = { "RFS" }
local function bit(p)
  return 2 ^ (p - 1) -- 1-based indexing
end
-- Typical call:  if hasbit(x, bit(3)) then ...
local function hasbit(x, p)
  return ((x % (p + p) >= p) and 1 or 0)
end
--  RBS :"Rx1 Ovl", "Rx2 Ovl", "SBUS Ovl", "Rx1 FS", "Rx1 LF", "Rx2 FS",
--       "Rx2 LF", "Rx1 Lost", "Rx2 Lost", "Rx1 NS", "Rx2 NS"
local function run()
  local status = getValue("RBS")
  print("RBS VALUE "..status)
  setTelemetryValue(0xb20, 1, 2, hasbit(status, bit(4)), 0, 0, "R1FS")
  setTelemetryValue(0xb20, 2, 2, hasbit(status, bit(6)), 0, 0, "R2FS")
  setTelemetryValue(0xb20, 3, 2, hasbit(status, bit(5)), 0, 0, "R1FL")
  setTelemetryValue(0xb20, 4, 2, hasbit(status, bit(7)), 0, 0, "R2FL")
  if status ~= 0 then
    return 10
  else
    return 0
  end
end
return { run=run, output=outputs }
```

Retrieving the raw value of the native "RBS" sensor

Creation of a virtual sensor "R1FS" corresponding to **Bit n°4** of "RBS".

according to the table on page 4, "R1FS" corresponds to "Rx1 FailSafe".

Using the script "rb_dec.lua" for the generation of virtual sensors

1- Turn on the transmitter and select the "CUSTOM SCRIPTS" menu of your current model,
2- Select "LUA1" and in the "Script" field, select "rb_dec". You will see on the left the "1RFS".
   From this step, the "rb_dec.lua" script now runs periodically in the background.
3- Repeat the sensor discovery operation according to the instructions in the "How to configure OpenTx to access telemetry data" section on page 3 of this manual.
4- You now have four new additional virtual sensors :
   - "R1FS" corresponding to "Rx1 FailSafe"
   - "R2FS" corresponding to "Rx2 FailSafe"
   - "R1FL" corresponding to "Rx1 Lost Frame"
   - "R2FL" corresponding to "Rx2 Lost Frame"

   These four virtual sensors come from informations periodically extracted from the native RBS sensor.

By analogy, by modifying the "*rb_dec.lua*" script, you can thus obtain any information described on the previous page.

Setting for Servo Signal Output Period

The default period for the signal output is 20ms, and it could be set to receiver synchronized.
Analog servos are not recommended to set to receiver synchronized.

Follow steps below to set the signal output period :
Step 1 - Connect signal pins of CH1 and CH2 by a jumper;
Step 2 - Connect the power supply to BATT1 or BATT2;
Step 3 - The Green LED flashes quickly, indicating the setting process of setting the signal output period from default 20ms to receiver synchronized is completed;
Step 4 - Disconnect the jumper from CH1 and CH2, disconnect the power supply.

How to distinguish between 20ms and receiver synchronized :
Connect a receiver to RX1 IN or RX2 IN, connect power supply to BATT1 or BATT2.
If Green LED flashes quickly, the period for the signal output is receiver synchronized.
If Green LED stays on, the period for the signal output is 20ms.

When working in the synchronous mode, the period of the PWM output is the same to that of the SBUS input.

For example, if the period of the SBUS input is 9ms, the period of the PWM output is 9ms as well. If the SBUS input period of RX1 IN is different with that of RX2 IN but both receivers are powered on at the same time, the period of the PWM output is the same to the one receiver with longer period. But if both receivers are not powered at the same time, the period of the PWM output is the same to the one receiver that is powered first.

For example, the SBUS input period of RX1 IN is 9ms, while that of RX2 IN is 18ms. If both receivers are powered on at the same time, the period of the PWM output is 18ms. If RX1 is powered on before RX2, the period of the PWM output is 9ms.

In synchronous mode, the signal delay is 3.05ms.

Change the SBUS signal from negative to positive for RX1 IN and RX2 IN

The SBUS signal from FrSky REDUNDANCY BUS is negative. Follow the steps below to change if from negative to positive.
For the example, the manipulation is described for RX1 IN :

Step 1 - Connect the signal pins of CH11 and CH12 by a jumper;
Step 2 - Connect the power supply to BATT1 or BATT2, Green LED will be on;
Step 3 - Remove the jumper and disconnect the power supply.

Note :
Connect the signal pins of CH13 and CH14 by a jumper, and follow step 2 and 3 to change the SBUS signal from negative to positive for RX2 IN.
Connect the signal pins of CH11 and CH12, as well as CH13 and CH14 by jumpers, and follow step 2 and 3 to change the SBUS signal from negative to positive for RX1 IN and RX2 IN at the same time.
Follow step 1 to step 3 to switch back the SBUS signal from positive to negative.