SFP out - SFP output mailbox

SD021 | Posted on August 3, 2021 | Updated on June 30, 2025

Jessy ANÇAY
Sales & Project Engineer
imperix

Table of Contents

- Simulink Block
  - Signal specification
  - Parameters
- PLECS block
  - Signal specification
  - Parameters
- C++ functions

The **SFP output mailbox** block allows sending data float (single) signals via the SFP optic cables. To recieve data float (single), the SFP input mailbox block should be used.

The SFP output mailbox block takes data from the input port of the block and sends it via SFP. It also features a second input that triggers when to write new data.

To see how to use SFP blocks in simulation and code generation please refer to the multi-master product note.

The SFP mailboxes are implemented in hardware. Therefore, they do not load the CPU and are synchronous with the control task. Given that the control tasks of the source (SFP out) and the target (SFP in) are running at the same frequency, the writing can be executed at every control task execution without losing any data.
The SFP communication is a one-to-one type of data transmission. The mailboxes therefore only work in pairs; one input and one output.

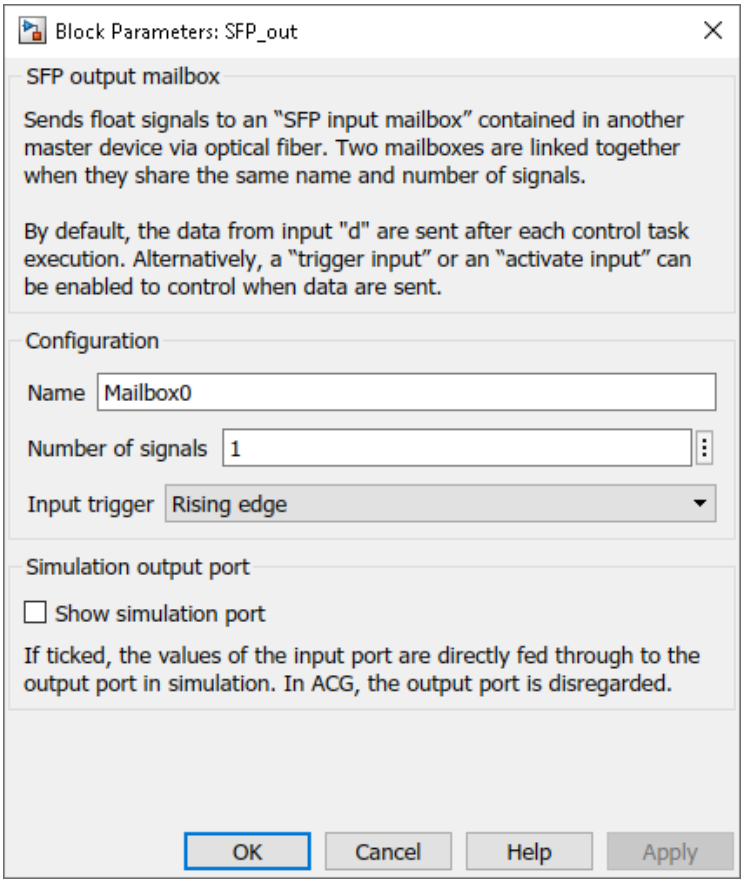
Simulink Block

Signal specification

- The data input signal “d” supports a vector of signals of type single. The vector length can be configured with the `Number of signals` parameter.
- The second signal is the `input trigger` signal. It can be used to initiate data transmission. Two modes are supported: rising edge and activate.

Parameters

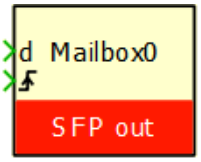
- `Name:` defines the output mailbox name. Data will go to a SFP input mailbox block with the same name.
- `Number of signals:` sets the vector length of the input data.
- `Input trigger:` defines the condition when the data is to be sent. When set to none, data is sent at each interrupt. When set to `Rising edge`, data is sent on the rising edge of the `input trigger` signal. Finally, when set to `Activate`, data is sent when the `input trigger` signal is high.
- `Show simulation input port` defines if the simulation input port is displayed or not.



PLECS block

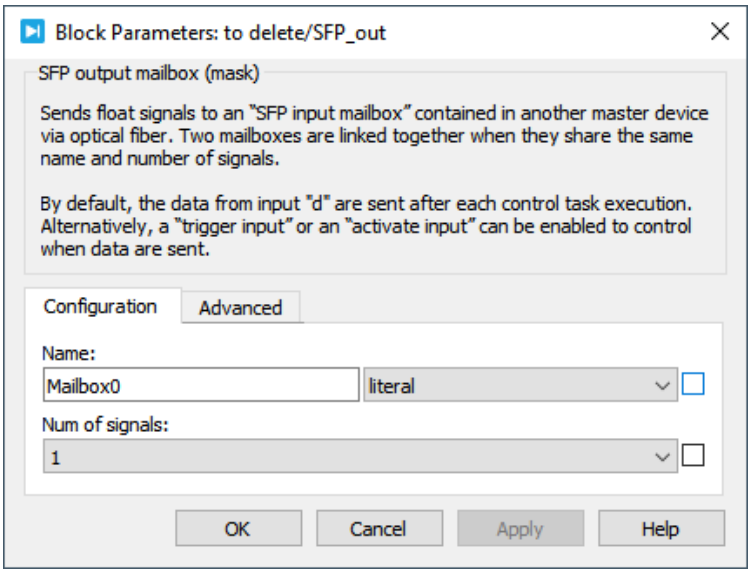
Signal specification

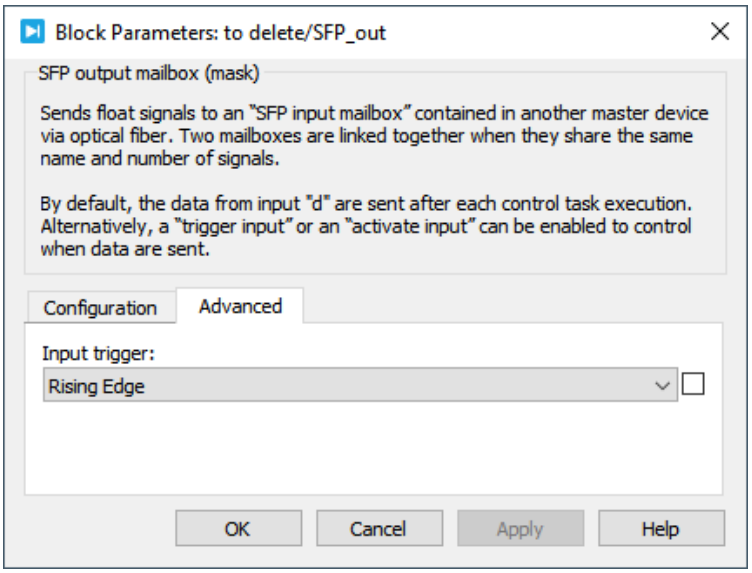
- The data input signal “d” supports a vector of signals of type *float*. The vector length can be configured with the `Number of signals` parameter.
- The second signal is the `input trigger` signal. It can be used to initiate data transmission. Two modes are supported: rising edge and activate.



Parameters

- `Name:` defines the output mailbox name. Data will go to a SFP input mailbox block with the same name.
- `Number of signals:` sets the vector length of the input data.
- `Input trigger:` defines the condition when the data is to be sent. When set to none, data is sent at each interrupt. When set to `Rising edge`, data is sent on the rising edge of the `input trigger` signal. Finally, when set to `Activate`, data is sent when the `input trigger` signal is high.





C++ functions

`Sfp_ConfigureOutput` — Configure an SFP output mailbox

```
int Sfp_ConfigureOutput(int uid, const char* name, int size_bytes)Code language: C++ (cpp)
```

Configures an SFP output mailbox.

It has to be called in `UserInit()`.

Parameters

- `uid:` a unique ID used to distiguish mailboxes.
- `name:` sets the name of the mailbox. SFP input and ouput mailboxes with the same name will be linked together.
- `size_bytes:` sets the size in bytes of the data to be read.

Return value

- `int:` returns false if too many input mailboxes were created or if two input mailboxes have the same name.

`Sfp_Write — Write`

```
void Sfp_Write(int uid, void* data)
```
Code language: C++ (cpp)

This function is used to write the data.

It has to be called during the control interrupt

Parameters

- `uid:` a unique ID used to distiguish mailboxes.
- `data:` data to be written to the input mailbox.