4.7 FIFO

The device supports the following FIFO operating modes:

- Streaming mode: overwrites oldest data on FIFO full condition
- FIFO mode: discards newest data on FIFO full condition

The FIFO size is 2048[7] byte and supports the following interrupts:

- FIFO full interrupt
- FIFO watermark interrupt

FIFO is enabled for accelerometer data with FIFO_CONFIG_1.fifo_acc_en=0b1, for gyroscope data with FIFO_CONFIG_1.fifo_gyr_en=0b1, and auxiliary interface (e.g. magnetometer) data with FIFO_CONFIG_1.fifo_aux_en=0b1 (0b0=disabled).

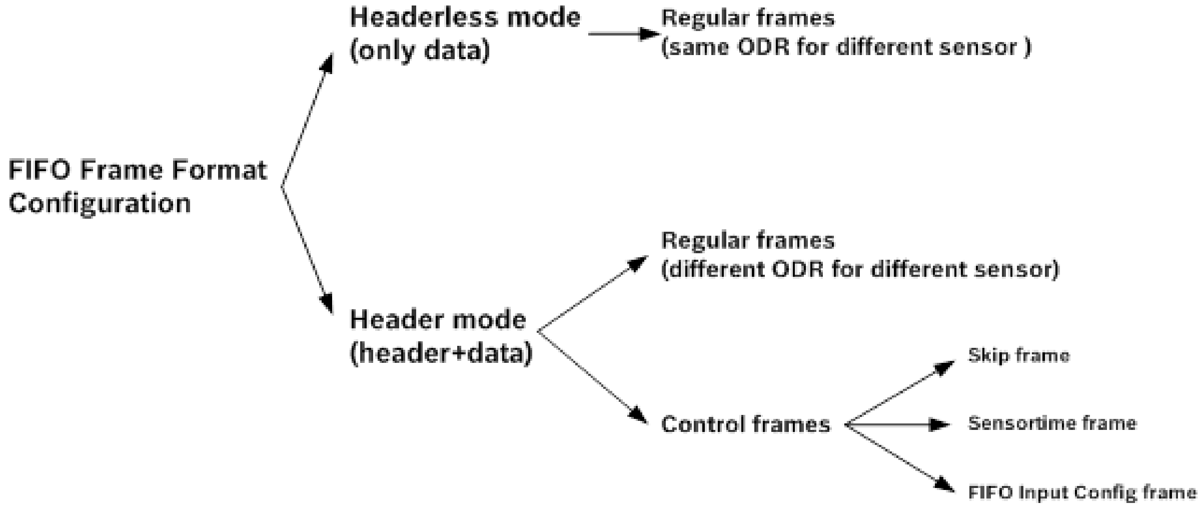
The FIFO may be used in all power modes of the device to record data. For readout conditions, see Subsection "FIFO in Low Power Mode".

4.7.1 Frames

The FIFO captures data in frames, which consist in header mode of a header and a payload data, in headerless mode only payload is stored.

In header mode (standard configuration) each regular frame consists of a one byte header describing properties of the frame (e.g. which sensors are included in this frame) and the payload data itself. Beside the regular frames, which contain the sensor data, there are control frames, which contain metadata (e.g. sensortime).

An overview of the possible frame types is show below



[7] See Application Note BST-BMI270-AN001 for 6KB FIFO

Header mode

The header has a fixed length of 8 bit and the following format:

Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Content	fh_mode<1:0>		fh_parm<3:0>				fh_ext<1:0>	

These *fh_mode* and *fh_parm* and *fh_ext* fields are defined below

fh_mode<1:0>	**Definition**	**fh_parm <3:0>**	**fh_ext<1:0>**
0b10	Regular frame	Enabled sensors	Tag of INT2 and INT1
0b01	Control frame	Control opcode	
0b00 and 0b11	Reserved	N/A	

fh_parm=0b0000 is invalid for regular mode, a header of 0x80 indicates an uninitialized frame, which is reported if the fifo read operations reads more data, than contained in the fifo. An uninitialized frame contains one byte of payload 0x00.

In a regular frame, fh_parm parameter defines which sensors are included in the data part of the frame. The format is

Name	**fh_parm<3:0>**			
Bit	3	2	1	0
Content	Reserved	FIFO_aux_data	FIFO_gyr_data	FIFO_acc_data

When FIFO_<sensor x>_data is 0b1 (0b0) data for sensor x is included (not included) in the data part of the frame.

The fh_ext<1:0> field are used for external tagging.

The order of the data in the FIFO data frame (see following table) differs from the order defined for the Registers DATA_0 to DATA_19.

A valid regular frame, contains data of at least one sensor (accelerometer, gyroscope, or auxiliary sensor). Only valid frames will be written into the FIFO. E.g. fh_parm=0b0111 in the header of a frame will result in the data layout shown below.

DATA[X]	**Acronym**	
X=0	AUX_0	copy of register Val(AUX_RD_ADDR) in auxiliary sensor register map
X=1	AUX_1	copy of register Val(AUX_RD_ADDR)+1 in auxiliary sensor register map
X=2	AUX_2	copy of register Val(AUX_RD_ADDR)+2 in auxiliary sensor register map
X=3	AUX_3	copy of register Val(AUX_RD_ADDR)+3 in auxiliary sensor register map
X=4	AUX_4	copy of register Val(AUX_RD_ADDR)+4 in auxiliary sensor register map
X=5	AUX_5	copy of register Val(AUX_RD_ADDR)+5 in auxiliary sensor register map
X=6	AUX_6	copy of register Val(AUX_RD_ADDR)+6 in auxiliary sensor register map
X=7	AUX_7	copy of register Val(AUX_RD_ADDR)+7 in auxiliary sensor register map
X=8	GYR_X<7:0> (LSB)	
X=9	GYR_X<15:8> (MSB)	
X=10	GYR_Y<7:0> (LSB)	
X=11	GYR_Y<15:8> (MSB)	
X=12	GYR_Z<7:0> (LSB)	
X=13	GYR_Z<15:8> (MSB)	
X=14	ACC_X<7:0> (LSB)	
X=15	ACC_X<15:8> (MSB)	
X=16	ACC_Y<7:0> (LSB)	
X=17	ACC_Y<15:8> (MSB)	
X=18	ACC_Z<7:0> (LSB)	
X=19	ACC_Z<15:8> (MSB)	

The length of the auxiliary sensor data block in a FIFO frame depends on the configured burst read length of the auxiliary interface in Register AUX_IF_CONF.aux_rd_burst:
If the read burst length for the auxiliary sensor is configured to less than 8 byte, the number of data bytes in the regular FIFO frame is reduced accordingly. I.e. in the above example, the gyro data would start before Byte 8.

Control frames

Control frames are only supported in header mode. There are a number of control frames defined through the fh_parm field. These are shown in below.
A skip frame indicates the number of skipped frames after a FIFO overrun occurred. A sensortime frame contains the sensortime when the last sampled frame stored in the FIFO is read. A FIFO input config frames indicates a change in sensor configuration which affects the sensor data.
The FIFO fill level is contained in registers FIFO_LENGTH_1.fifo_byte_counter_13_8 and FIFO_LENGTH_0.fifo_byte_counter_7_0. The fifo fill level includes the space needed for the regular and the control frames, with the exception of the sensortime frame.

fh_mode<3:0>	Definition	Number of
0x0	Skip Frame	1 byte payload
0x1	Sensortime Frame	3 bytes payload
0x2	Fifo_Input_Config Frame	4 bytes payload
0x3 – 0x7	Reserved	

Skip Frame (fh_parm=0x0)
In the case of FIFO overflows, a skip_frame is prepended to the FIFO content, when read out next time. The data for the frame consists of one byte and contains the number of skipped frames. When more than 0xFF frames have been skipped, 0xFF is returned. A skip frame is expected always as first frame in a FIFO read burst. A skip frame does not consume memory in the FIFO.

Sensortime Frame (fh_parm=0x1)
The data for the sensortime frame is a copy of the Register SENSORTIME_0 to SENSORTIME_2 when the last byte of the last sample frame was read. One sensortime frame is always expected as last frame in the FIFO. A sensortime frame is only sent if the FIFO becomes empty during the burst read. A sensortime frame does not consume memory in the FIFO. Sensortime frames are enabled (disabled) by setting FIFO_CONFIG_0.fifo_time_en to 0b1 (0b0).

Fifo_Input_Config Frame (fh_parm=0x2)
Whenever the filter configuration of the FIFO input data sources changes, a FIFO input config frame is inserted into the FIFO, before the configuration change becomes active. E.g. when the bandwidth for the accelerometer filter is changed in Register ACC_CONF, a FIFO input config frame is inserted before the first frame with accelerometer data with the new bandwidth configuration. The FIFO input config frame contains four byte of data with the format

Bit	7	6	5	4	3	2	1	0
Byte 0	reserved	reserved	aux_ if_ch	aux_ conf_ch	gyr_ range_ch	gyr_ conf_ch	acc_ range_ch	acc_ conf_ch
Byte 1	Sensortime _0 for next frame (may be drop frame)							
Byte 2	Sensortime_1 for next frame (may be drop frame)							
Byte 3	Sensortime _2 for next frame (may be drop frame)							

aux_if_ch	A write to Register AUX_IF_CONF, AUX_RD_ADDR, or AUX_WR_ADDR becomes active.
aux_conf_ch	A write to Register AUX_CONF becomes active.

gyr_range_ch	A write to Register GYR_RANGE becomes active.
gyr_conf_ch	A write to Register GYR_CONF or gyr_FIFO_filt_data or gyr_FIFO_downsampling in Register FIFO_DOWNS becomes active.
acc_range_ch	A write to Register ACC_RANGE becomes active.
acc_conf_ch	A write to Register ACC_CONF or acc_FIFO_filt_data or acc_FIFO_downsampling in Register FIFO_DOWNS becomes active.

If Byte 0 is 0x00, this indicates that this Fifo_Input_Config Frame is written, because the fifo or sensor was enabled.

Headerless mode

When the data rates of all enabled sensor elements are identical, the FIFO header may be disabled in FIFO_CONFIG_1.fifo_header_en.

The headerless mode supports only regular frames. To be able to distinguish frames from each other, all frames must have the same size. For this reason, any change in configuration that have an impact to frame size or order of data within a frame will cause an instant flush of FIFO, restarting capturing of data with the new settings.

If the auxiliary sensor interface is enabled, the number of auxiliary sensor bytes in a FIFO frame is always AUX_IF_CONF.aux_rd_burst bytes (see section 4.10). If the burst length is less than 4, the device will pad the values read form the auxiliary sensor to 4 bytes. E.g. if AUX_IF_CONF.aux_rd_burst=0b01 (2 Bytes), a frame with auxiliary sensor, accelerometer, and gyroscope data will look like

DATA[X]	Acronym	
X=0	AUX_0	copy of register Val(AUX_RD_ADDR.read_addr) in auxiliary sensor register map
X=1	AUX_1	copy of register Val(AUX_RD_ADDR.read_addr+1) in auxiliary sensor register map
X=2	Padding byte	Undefined value
X=3	Padding byte	Undefined value
X=4	GYR_X<7:0> (LSB)	
X=5	GYR_X<15:8> (MSB)	
X=6	GYR_Y<7:0> (LSB)	
X=7	GYR_Y<15:8> (MSB)	
X=8	GYR_Z<7:0> (LSB)	
X=9	GYR_Z<15:8> (MSB)	
X=10	ACC_X<7:0> (LSB)	
X=11	ACC_X<15:8> (MSB)	
X=12	ACC_Y<7:0> (LSB)	
X=13	ACC_Y<15:8> (MSB)	
X=14	ACC_Z<7:0> (LSB)	
X=15	ACC_Z<15:8> (MSB)	

4.7.2 Conditions and Details

FIFO frame reads

If a frame is fully read through the Register FIFO_DATA, it gets deleted from the FIFO of the device. If a frame is only partially read it will be repeated completely with the next access both in headerless and in header mode. In headermode, this includes the header. In the case of a FIFO overflow between the first partial read and the second read attempt, the frame is kept only if FIFO_CONFIG_0.fifo_stop_on_full =0b1.

FIFO overreads

When more data are read from the FIFO than it contains valid data, 0x8000 is returned in headerless mode. In header mode 0x80 indicates an invalid frame.

Frame rates

The frame sampling rate of the FIFO is defined by the maximum output data rate of the sensors enabled for FIFO sampling. The FIFO sampling configuration is set in register FIFO_CONFIG_0 to FIFO_CONFIG_1. It is possible to select filtered or pre-filtered data as an input to the FIFO. If pre-filtered data is selected in register FIFO_DOWNS.acc_fifo_filt_data for the accelerometer, the sample rate is 1600 Hz. If pre-filtered data is selected in register FIFO_DOWNS.gyr_fifo_filt_data for the gyroscope, the sample rate is 6400 Hz. The range of the pre-filtered gyroscope data is defined by GYR_RANGE.ois_range and independent of the range configured for the data register and filtered data in the fifo defined by GYR_RANGE.gyr_range. The input data rate to the FIFO can be reduced by selecting a down-sampling factor 2^k in registers FIFO_DOWNS.acc_fifo_downs or FIFO_DOWNS.gyr_fifo_downs where k={0..7}.

FIFO overflow

In the case of an overflow the FIFO can either stop recording data or overwrite the oldest data. The behavior is controlled by Register FIFO_CONFIG_0.fifo_stop_on_full. If FIFO_CONFIG_0.fifo_stop_on_full *=0b0,* the FIFO logic may delete the oldest frames. If header mode is enabled and the free FIFO space falls below the maximum size frame, the skip frame is the prepended at the next FIFO readout.

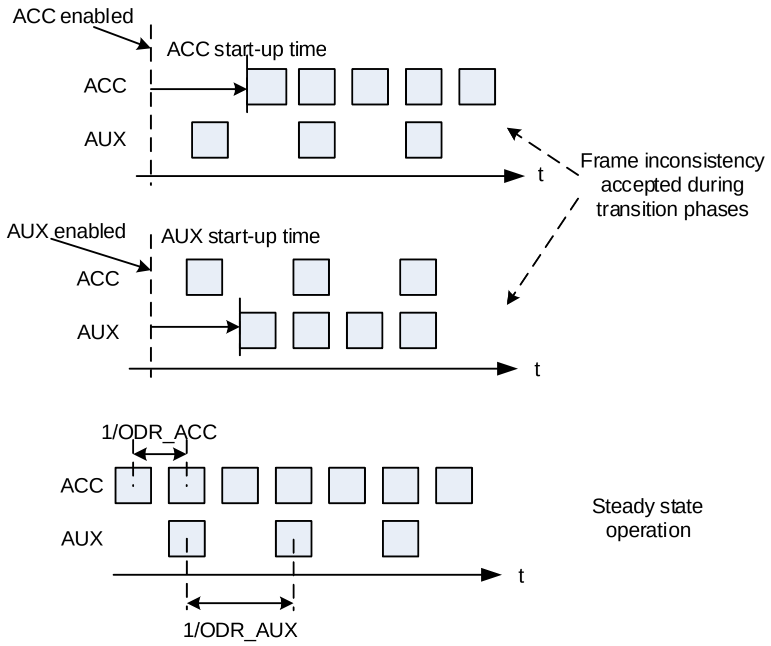
If FIFO_CONFIG_0.fifo_stop_on_full =0b1, the newest frame may be discarded, if the free FIFO space falls below the maximum size frame. If header mode is enabled, a skip frame is prepended at the next FIFO readout (which is **not** the position where the frame(s) have been discarded).

During a FIFO read operation of the host, no data at the FIFO tail may be dropped. If the host reads the FIFO with a slower rate than it is filled, it may happen that the sensor needs to drop new data, even when FIFO_CONFIG_0.fifo_stop_on_full =0b0. These events are recorded in the Register ERR_REG.fifo_err.

4.7.3 FIFO data synchronization

All sensor data are sampled with respect to a common ODR time grid. Even if a different ODR is selected for the acceleration and the auxiliary sensor the data remains synchronized:
If a frame contains a sample from a sensor element with ODR x, then it must contain also samples of all sensor elements with an ODR y>=x. This applies for steady state operation. In transition phases, it is more important not to lose data, therefore exceptions are possible if the sensor elements with ODR y>=x do not have data, e.g. due to a sensor configuration change.

FIFO Data Synchronization Scheme in the following figure illustrates the steady state and transient operating conditions.



4.7.4 FIFO synchronization with external events

External events at the INT<x> pin may be synchronized into the FIFO data. For this operation mode the FIFO_CONFIG_1.fifo_tag_int<x>1_en and INT<x>_IO_CTRL.input_en need to be enabled. The `fh_ext` field in FIFO header of a regular frame will then be set according to an event at the INT<x> pin. If FIFO_CONFIG_1.fifo_tag_int<x>_en is configured to `int_level`, the value of the INT<x> pin at the time when the FIFO regular frame is written is copied into the `fh_ext` field. If FIFO_CONFIG_1.fifo_tag_int<x>_en is configured to `int_edge`, the corresponding bit in the `fh_ext` field of a regular frame will be set, if a positive edge of a pulse of minimum length 10ns on the INT<x> pin occured in the sampling interval before this frame is written into the FIFO. E.g. if the ODR is set to 100 Hz and the `fh_ext` field in FIFO header is set, then in the 10 ms before the regular frame was written into FIFO an positive edge occurred at the INT<x> pins.

4.7.5 FIFO Interrupts

The FIFO supports two interrupts, a FIFO full interrupt and a watermark interrupt:

- The FIFO full interrupt is issued when the FIFO fill level is above the full threshold. The full threshold is reached just before the last two frames are stored in the FIFO.
- The FIFO watermark is issued when the FIFO fill level is equal or above a watermark defined in Register FIFO_WTM_1.fifo_water_mark_12_8.

In order to enable/use the FIFO full or watermark interrupts, map them on the desired interrupt pin via INT_MAP_DATA.

Latched FIFO interrupts will only be cleared, if the status register gets read and the fill level is below the corresponding FIFO interrupt (full or watermark).

4.7.6 FIFO Reset

The user can trigger a FIFO reset by writing the command fifo_flush (0xB0) in CMD.
Automatic resets are only performed in the following cases:

- A sensor is enabled or disabled in headerless mode
- A transition between headerless and headermode or vice versa has occurred.
- Size of auxiliary sensor data in a frame changed in header or headerless mode

4.7.7 FIFO in Low Power Mode

In the low power mode the device supports FIFO usage. The data storage into the FIFO is identical to the normal and performance mode, for the readout the description below applies:

- If PWR_CONF.fifo_self_wakeup=0b0 the advanced power save configuration needs to be disabled (PWR_CONF.adv_power_save=0b0) before reading out FIFO data.
- If PWR_CONF.fifo_self_wakeup=0b1 and the FIFO watermark or FIFO full interrupt is triggered, the restriction for PWR_CONF.adv_power_save=0b1 (see Section 4.5) do not apply as long as a single burst read on Register FIFO_DATA completes. This may be used to read the complete FIFO with one single burst read without leaving low power mode. Without a FIFO watermark interrupt or full interrupt, the advanced power save configuration needs to be disabled (PWR_CONF.adv_power_save=0b0) before reading out FIFO data.