

Product Document



Application Note

AN000670

TMF8X0X

Linux Device Driver ABI Descriptions

v3-00 • 2023-Jun-22

Content Guide

1	General Description	3	4.2	ToF Sensor App0 ABI List	9
1.1	ABI Details	3	4.3	Input Event Interface	17
1.2	ABI Data Format	5	4.4	ToF Sensor ABI Use Case Examples	18
2	ToF Sensor RAM Firmware Download	6	5	Summary	21
3	ToF Sensor Factory Calibration and Optical Configuration	7	6	Revision Information	22
4	ToF Sensor ABI	8	7	Legal Information	23
4.1	ToF Sensor Generic ABI List	8			

1 General Description

The following materials have been created to support the Linux device driver for the TMF8701, TMF8801 and TMF8805 Time-of-Flight (ToF) distance measurement device(s). The TMF8X0X device drivers have been tested on Android Nougat 7.0 running kernel 3.18.31, and on a Raspberry Pi running kernel 4.4.50.

1.1 ABI Details

The TMF8701/TMF8801/TMF8805 device driver interface is performed via 'sysfs' and an input device file. Since the final I²C directory designation (1-000C, 2-001F, etc.) are platform implementation dependent this document will present ToF related ABIs under './1-0041' (the I²C address for TMF8701/TMF8801/TMF8805 is 0x41). The TMF8801 and TMF8805 supports all of the sysfs interface files the TMF8701 supports, but includes extra attributes that cover features only available on the TMF8801 and TMF8805.

Figure 1:
Generic sysfs Entries

Output Format	Read/Write	Valid When PROGRAM ==	Filename	TMF8801/TMF8805 Only
string	RW	any	chip_enable	
string	RW	any	driver_debug	
string	RO	any	mod_alias	
string	RO	any	name	
string	RW	any	program	
string	RO	any	program_version	
string	RO	any	registers	
none	WO	any	registers_write	
none	WO	any	request_ram_patch	
string	RW	any	uevent	

Figure 2:
App0 sysfs Entries

Output Format	Read/Write	Valid When PROGRAM ==	Filename	TMF8801/TMF8805 Only
string	RW	0xC0	app0/alg_setting	
string	RW	0xC0	app0/app0_apply_config_calib	
string	RW	0xC0	app0/app0_apply_fac_calib	
string	RW	0xC0	app0/apply_state_data	
string	RW	0xC0	app0/app0_clk_iterations	
string	RW	0xC0	app0/app0_apply_config_calib	
string	RW	0xC0	app0/app0_apply_fac_calib	
string	RW	0xC0	app0/apply_state_data	
string	RW	0xC0	app0/app0_clk_iterations	
string	RW	0xC0	app0/app0_clk_trim_enable	
string	RW	0xC0	app0/app0_clk_trim_set	
string	RW	0xC0	app0/app0_command	
string	RO	0xC0	app0/app0_ctrl_reg	
string	RO	0xC0	app0/app0_diag_state_mask	
string	RO	0xC0	app0/app0_general_configuration	
string	RO	0xC0	app0/app0_get_fac_calib	
string	RO	0xC0	app0/app0_reflectivity_count	
string	RO	0xC0	app0/app0_temp	Y
bin	RO	0xC0	app0/app0_tof_output	
string	RW	0xC0	app0/capture	
string	RW	0xC0	app0/capture_delay	
none	WO	0xC0	app0/distance_high_thres	Y
none	WO	0xC0	app0/distance_low_thres	Y
none	WO	0xC0	app0/distance_pers	Y
string	RW	0xC0	app0/gpio_setting	
string	RO	0xC0	app0/int_compression_cfg	
string	RW	0xC0	app0/iterations	Y
string	RW	0xC0	app0/noise_threshold	
string	RW	0xC0	app0/period	

1.2 ABI Data Format

Unless specified otherwise, all input and output through the ABI files is ASCII formatted text strings. This allows these ABIs to be accessed easily via a command shell or script using the Linux echo and cat commands. For example, the chip_enable pin to the ToF device can be activated as:

```
"echo 1 > ../1-0041/chip_enable"
```

and to display the current control registers, you can 'cat' the "app0_ctrl_reg" sysfs file:

```
"cat ../1-0041/app0/app0_ctrl_reg"
```

Those ABI entries which perform binary I/O will cause unpredictable results when accessed directly from the command system line.

2 ToF Sensor RAM Firmware Download

The ToF sensor does not contain a reprogrammable NVM module. For initial revisions of the sensor hardware that do not contain the latest firmware version, a download of the firmware to the sensor RAM must be performed.

The RAM firmware download (FWDL) utilizes the Linux firmware framework. The format of the firmware is an Intel hex record format. The driver expects the firmware filename to be **"tof8701_firmware.bin"** for the TMF8701 driver (**"tof8801_firmware.bin"** for the TMF8801 driver, **"tof8805_firmware.bin"** for the TMF8805 driver) and placed in the common Linux firmware directory, **/lib/firmware**.

After the firmware file is in place an application can trigger a firmware download by using the sysfs ABI, `request_ram_patch`. Please refer to the "ToF Sensor ABI" section for more information on this interface. The driver will download the new firmware and automatically reset the ToF sensor to the latest firmware. The speed of the download depends on several factors including, speed of the I²C bus, utilization the I²C bus, and the size of the firmware.

3 ToF Sensor Factory Calibration and Optical Configuration

The Factory Calibration and Optical Configuration Calibration data in the reference driver utilizes the Linux firmware framework. The format of the files are supported in binary format only. The driver expects the factory calibration filename to be **“tof8701_fac_calib.bin”** and the optical configuration calibration filename to be **“tof8701_config_calib.bin”** and both should be placed in the common Linux firmware directory, **/lib/firmware** (**tof8801_fac_calib.bin** and **tof8801_config_calib.bin** **tof8805_config_calib.bin** respectively for the TMF8801/TMF8805 driver). These files are read once during probe and then cached internally in the driver unless an explicit override is requested through the ABI's **“app0/app0_apply_fac_calib”** and **“app0/app0_apply_config_calib”** in which case the firmware files will be re-read. The calibration data is always applied at the start of a measurement.

4 ToF Sensor ABI

Refer to the device datasheet for pertinent information regarding the device interface.

4.1 ToF Sensor Generic ABI List

This section contains a list of the generic ABI sysfs files with their function.

4.1.1 **chip_enable**

- Boolean state ToF on/off
- Valid entries: 0-1 (0=off / 1=on)
- Returns: 0-1

4.1.2 **driver_debug (unused)**

Do not use.

4.1.3 **mod_alias (unused)**

Do not use.

4.1.4 **name**

Returns the sensor name.

- Valid entries: None
- Returns: e.g. "tof8801"

4.1.5 **program**

Reading this value returns the ID of currently running application, writing a value is a request to switch to the specified Application ID. If the driver fails to switch to the desired application ID, it will return to the Bootloader application, ID 0x80. Please note that 0xC1 is a valid entry but not currently supported.

- Valid entries: 0x80, 0xC0, 0xC1
- Returns: 0x80, 0xC0, 0xC1

4.1.6 program_version

Returns the currently running Application ID revision number.

- Valid entries: None
- Returns: major, minor, patch, build version as hex numbers

4.1.7 registers

Returns the sensor register list from I²C address 0x00 to 0xFF.

4.1.8 register_write

Sets a single I²C sensor register to a defined value.

Example – set register 0x10 to value 0x08:

```
echo '0x10:0x08' >register_write
```

4.1.9 request_ram_patch

Starts a RAM firmware download on the ToF sensor. Returns when the firmware download is complete.

- Valid entries: any

4.1.10 uevent (unused)

Do not use.

4.2 ToF Sensor App0 ABI List

This section contains a list of the App0 ABI sysfs files with their function.

4.2.1 app0/alg_setting

This value sets the algorithm operating mode in the ToF device. Driver default is `alg_setting == 0`.

Figure 3:
TMF8701 Algorithm Settings

Bits	Definition
0	(algProximityEnabled) ⁽¹⁾ when 1 proximity is enabled
1	(algDistanceEnabled) ⁽¹⁾ when 1 distance enabled
3:2	Reserved, set to 00b
4	algImmediateInterrupt – When 1 proximity and distance will immediately report to the host an interrupt of the capturing caused by a GPIO event; when 0, will only report to the host when proximity (and distance if enabled) was finished
5	When 1 combine the capture of the proximity and distance algorithms for maximum speed
6	Reserved, set to 0
7	When 1 do not go to standby between measurements (faster measurement times but higher current consumption)

(1) If both bits 0 and 1 are zero, the device will default to both algorithms enabled

Figure 4:
TMF8801 Algorithm Settings

Bits	Definition
0	Set to 1
1	Set to 1
2	VCSEL_clk_div2: If set, operates the VCSEL clock at half frequency and doubles the ranging active time where the VCSEL is enabled.
3	Reserved, set to 0
4	algImmediateInterrupt – When 1 target distance measurement will immediately report to the host an interrupt of the capturing caused by a GPIO event; when 0, will only report to the host when target distance measurement was finished
5	When 1 combine the capture of the short and long distance histogram for maximum speed
6	Reserved, set to 0
7	When 1 do not go to standby between measurements (faster measurement times but higher current consumption)



Information

If the ToF sensor is currently taking a measurement, the current measurement will be stopped and then restarted with the new parameter value. If the ToF sensor is not actively capturing a measurement, the parameter will be stored locally in the driver until the capture is triggered via the 'app0/capture' sysfs interface.

- Valid entry: 0 – 0xFF
- Returns: 0 – 0xFF

4.2.2 app0/app0_apply_config_calib

When written, this triggers the driver to re-read the optical configuration calibration file. When read, this dumps the currently cached optical configuration calibration data stored in the driver. The entries are newline separated.

- Valid entry: any
- Returns: <register>: <value>

4.2.3 app0/app0_apply_fac_calib

When written, this triggers the driver to re-read the factory calibration file. When read, this dumps the currently cached factory calibration data stored in the driver. The entries are newline separated.

- Valid entry: any
- Returns: <register>: <value>

4.2.4 app0/app0_apply_state_data (unused)

Do not use.

4.2.5 app0/app0_clk_iterations

This attribute is a Read/Write value to set the integration time of the clock trimming in number of measurement iterations. I.e. setting this value to '3' would mean the driver measures the clock difference of the ToF device over three measurement cycles to calculate the current clock value and trim the clock faster or slower as needed.

- Valid entry: any
- Returns: Current clock trim iteration setting

4.2.6 app0/app0_clk_trim_enable

This attribute is a Read/Write value to enable or disable the clock trimming of the ToF device.



Attention

Clock trimming is required by the TMF8801 part to achieve 2.5 meter distance.

- Valid entry: 0 to disable, 1 to enable
- Returns : 0 if clock trimming disabled, 1 if clock trimming enabled

4.2.7 app0/app0_clk_trim_set

This attribute is a Read/Write value to directly set the clock trim value on the ToF device. Warning: Setting the clock trim value may cause unexpected behavior.

- Valid entry: -256 to 255
- Returns: Current clock trim setting

4.2.8 app0/app0_command (unused)

Do not use.

4.2.9 app0/app0_ctrl_reg

This attribute returns the control register bank (0x0 – 0x21) values of the TMF8701/TMF8801 ToF device. The entries are newline separated. The register meanings are detailed in the device datasheet.

- Valid entry: None
- Returns: <register>: <value>

4.2.10 app0/app0_diag_state_mask (unused)

Do not use.

4.2.11 app0/app0_general_configuration

This attribute returns the general configuration settings register bank values of the TMF8701/TMF8801 ToF device. The entries are newline separated. The register meanings are detailed in the device datasheet.

- Valid entry: None
- Returns: <register>: <value>

4.2.12 app0/app0_get_fac_calib

This attribute is a read-only low level binary output file. This will return the factory calibration result data from the device so that it can be stored externally and then passed back into the device in the future for more accurate measurements.

For the reference driver, an example to trigger a factory calibration and store the result for later use:

```
cat app0/app0_get_fac_calib > /lib/firmware/tof8701_fac_calib.bin
```

- Valid entry: None
- Returns: Binary blob of factory calibration data

4.2.13 app0/app0_reflectivity_count

This attribute returns the last result summed reflectivity count of the object and the reference channel.

- Valid entry: None
- Returns: object hits: <value>
reference hits: <value>

4.2.14 app0/app0_temp (TMF8801 and TMF8805 only)

This attribute is a read-only value that returns the last known temperature of the ToF device (only supported on TMF8801). This number has units of degrees Celsius.

- Valid entry: None
- Returns: Current temperature of ToF die in degrees Celsius

4.2.15 app0/app0_tof_output (binary file)

This attribute is a read-only low level binary output file. Normal measurement result data should be read from the Linux input device file. To read an output frame from this file, the read buffer must be at least as large as the largest possible data frame, which is **515 Bytes**. Frames are read out one at a time, sequentially out of an internal ring buffer, each frame requires one read of the sysfs file. All non-configuration results will be output by this file which has the following framing format:

Identifier (1 Byte)	Data Size LSB (1 Byte)	Data Size MSB (1 Byte)	Data
------------------------	---------------------------	---------------------------	------

The Identifier bytes and corresponding Data format(s) are detailed in the ToF driver interface header file, tof8701.h / tof8801.h / tof8805.h

- Valid entry: None
- Returns: 0 or 1 Frame with format described above

4.2.16 app0/capture

Boolean value to start a measurement. If the period is non-zero, this setting will stay set until cleared. If period == 0, after the first result the capture value will reset to 0. Writing a '0' here is equivalent to sending the '0xFF' (STOP_CMD, reference in the TMF8701 datasheet) command to the ToF sensor resulting in an IDLE state.

- Valid entry: 0 – stop capture / non-zero – start capture
- Returns: measure command from TMF8701 datasheet

4.2.17 app0/capture_delay

Set the desired measurement delay. Values are specified in units of 100 microseconds. This value is used for synchronization purposes with other proximity devices. For the TMF8701 ToF device, this value is used for internal synchronization with the Proximity/ALS sensor.



Information

If the ToF sensor is currently taking a measurement, the current measurement will be stopped and then restarted with the new parameter value. If the ToF sensor is not actively capturing a measurement, the parameter will be stored locally in the driver until the capture is triggered via the '**app0/capture**' sysfs interface.

- Valid entry: 0 - 255
- Returns: 0 - 255

4.2.18 app0/distance_high_thres

This attribute is a write-only value to set the upper boundary in millimeters for the interrupt suppression feature. Please see datasheet chapter 8.9.11, command 0x8 for reference. Only available for TMF880x firmware version 3.0.22 or higher.

- Valid entry: 0 – 65535

4.2.19 app0/distance_low_thres

This attribute is a write-only value to set the lower boundary in millimeters for the interrupt suppression feature. Please see datasheet chapter 8.9.11, command 0x8 for reference. Only available for TMF880x firmware version 3.0.22 or higher.

- Valid entry: 0 – 65535

4.2.20 app0/distance_pers

This attribute is a write-only value to set the persistence for the interrupt suppression feature. Please see datasheet chapter 8.9.11, command 0x8 for reference. Only available for TMF880x firmware version 3.0.22 or higher.

- Valid entry: 0 – 255

Figure 5:
Sensor Behavior Based on the Persistence Value

Value	Behavior
0	Any result (detect and no-detect) will trigger an interrupt.
1 - 255	An interrupt will only be triggered if the detected distance is \geq distance_low_thres and distance \leq distance_high_thres and the detection happened at least <distance_pers> consecutive times.

4.2.21 app0/gpio_setting

This value controls the gpio configuration of the ToF sensor. Driver default is GPIO control for both GPIO0/GPIO1 is disabled (gpio_setting == 0).

Figure 6:
GPIO Settings

Bits	Definition
	GPIO0 settings
	0 - Input
	1 - Input: Active low disables collection, immediately abandoning current measurement. Returning to high restarts new measurement
	2 - Input: Active high disables collection, immediately abandoning current measurement. Returning to low restarts new measurement
3:0	3 - Output: VCSEL pulse output – see cmd_data4
	4 - Output low (default after startup)
	5 - Output high
	6:15 - Reserved, do not use

Bits	Definition
7:4	GPIO1 settings
	0 - Input
	1 - Input: Active low disables collection, immediately abandoning current measurement. Returning to high restarts new measurement
	2 - Input: Active high disables collection, immediately abandoning current measurement. Returning to low restarts new measurement
	3 - Output: VCSEL pulse output – see cmd_data4
	4 - Output low (default after startup)
	5 - Output high
6:15 - Reserved, do not use	



Information

If the ToF sensor is currently taking a measurement, the current measurement will be stopped and then restarted with the new parameter value. If the ToF sensor is not actively capturing a measurement, the parameter will be stored locally in the driver until the capture is triggered via the 'app0/capture' sysfs interface.

- Valid entry: 0 – 0xFF
- Returns: 0 – 0xFF

4.2.22 app0/int_compression_cfg (unused)

Do not use.

4.2.23 app0/iterations (TMF8801 and TMF8805 only)

This attribute is a read/write value that sets the number of measurement iterations the ToF device should perform (only supported on TMF8801). This number is truncated to the nearest thousandth.

- Valid entry: 0 - UINT_MAX
- Returns: Current number of iterations configured

4.2.24 app0/noise_threshold

This value sets the noise threshold in units of standard deviations away from the mean. To use the device default, the value should be 255 (0xFF).



Information

If the ToF sensor is currently taking a measurement, the current measurement will be stopped and then restarted with the new parameter value. If the ToF sensor is not actively capturing a measurement, the parameter will be stored locally in the driver until the capture is triggered via the **'app0/capture'** sysfs interface.

- Valid entry: 0 - 255
- Returns: 0 - 255

4.2.25 app0/period

Set the desired result reporting period of measurement captures. Values are specified in units of milliseconds. The period is set to 0, then only a single measurement will be taken once the 'capture' attribute is set. If $0 < \text{period} < \text{min_reporting_period}$, then the measurement reporting rate will be the max frequency as specified in the device datasheet.



Information

If the ToF sensor is currently taking a measurement, the current measurement will be stopped and then restarted with the new parameter value. If the ToF sensor is not actively capturing a measurement, the parameter will be stored locally in the driver until the capture is triggered via the **'app0/capture'** sysfs interface.

- Valid entry: 0 - 255
- Returns: 0 - 255

4.3 Input Event Interface

The TMF8701/TMF8801/TMF8805 Driver also supports an input event interface for reporting distance information to user space. To find the input event node created for the device, either iterate over the event nodes and compare the IOCTL on EVIOCGNAME to look for the TMF8701/TMF8801/TMF8805, or you can also find which input event node is created by referring to the sysfs `./input/` path, see below.

```
$ ls input/input3/

capabilities/ event0/      modalias      phys          properties    uevent

device/      id/          name          power/       subsystem/    uniq

$ cat input/input3/name

tof8701
```

```
$ cat input/input3/event0/device/name
tof8701
```

The Linux `input_event` struct is defined (in `linux/input.h`) as:

```
struct input_event {
    struct timeval time;

    __u16 type; // Set to EV_ABS (as defined in linux/input-event-codes.h)
    __u16 code; // Set to ABS_DISTANCE (as defined in linux/input-event-codes.h)
    __s32 value; // Format for result data in table below
};
```

The Distance value is packed into the data integer as:

__s32 value:

Field	31:24	23:22	21:16	15:0
Meaning	Incrementing Result counter (rolls-over after 255)	GPIO Interruption flags (see TMF8701 datasheet)	Confidence (0 – 63)	Distance in mm



Information

In the reference driver, when there is no object detected, an input event with value [15:0] == 0 is reported, but events will continue to be published because of the auto-incrementing result counter.

4.4 ToF Sensor ABI Use Case Examples

This section describes some use cases for configuring and retrieving the result data from the ToF sensor. This assumes you are reading and writing to the sysfs files through the shell prompt.

The most useful/important ABI sysfs entries are listed below. These files control the setup and triggering mechanism for the ToF sensor:

- **app0/alg_setting** – Algorithm operating mode
- **app0/capture** – Trigger to start/stop a measurement
- **app0/gpio_setting** – GPIO synchronization configuration
- **app0/noise_threshold** – Noise threshold configuration
- **app0/period** – Measurement period

The capture settings can be configured in any sequence, the trigger sysfs file **app0/capture** should be enabled last. If the ToF sensor is in continuous operating mode at the time of changing one of the parameters, the current measurement is stopped, then restarted with the new parameters. If the ToF sensor is in IDLE state, then the settings are stored locally in the driver and committed to the sensor when the capture trigger is enabled. If the ToF sensor capture is stopped via the **app0/capture** sysfs interface, the last used settings are retained in the ToF sensor and there is no need to reconfigure the capture settings.

Figure 7:
TMF8701 Example Use Cases (change alg_setting for use with TMF8801 or TMF8805)

Desired Configuration	Driver ABI Procedure
<ul style="list-style-type: none"> • Snapshot capture • Distance + proximity • No gpio setting 	<ul style="list-style-type: none"> • echo '1' > ./app0/capture • input event is generated on /dev/input/event0 with result
<ul style="list-style-type: none"> • Snapshot capture • Proximity only • No gpio setting 	<ul style="list-style-type: none"> • echo '1' > ./app0/alg_setting • echo '1' > ./app0/capture • input event is generated on /dev/input/event0 with result
<ul style="list-style-type: none"> • Continuous capture, 60 Hz • Keep charge pump and TDC RAM powered between measurements • Distance only • GPIO1 active high disables measurement 	<ul style="list-style-type: none"> • echo '1' > ./app0/period • echo '0x82' > ./app0/alg_setting • echo '0x20' > ./app0/gpio_setting • echo '1' > ./app0/capture • input events are continuously generated on /dev/input/event0 with results • echo '0' > ./app0/capture (stop measuring)
<ul style="list-style-type: none"> • Continuous capture, 4 Hz • Proximity only • GPIO0 Active high disables measurement • GPIO1 Active low disables measurement 	<ul style="list-style-type: none"> • echo '250' > ./app0/period • echo '1' > ./app0/alg_setting • echo '0x12' > ./app0/gpio_setting • echo '1' > ./app0/capture • input events are continuously generated on /dev/input/event0 with results • echo '0' > ./app0/capture (stop measuring)

Desired Configuration	Driver ABI Procedure
<ul style="list-style-type: none"> ● TMF880x only: Configure interrupt suppression. ● Range from 200 mm to 500 m. ● Needs at least two consecutive detects to trigger an interrupt. 	<ul style="list-style-type: none"> ● echo '0' > ./app0/capture ● echo '200' > ./app0/distance_low_thres ● echo '500' > ./app0/distance_high_thres ● echo '2' > ./app0/distance_pers ● echo '1' > ./app0/capture

5 Summary



For further information, please refer to the following documents:

- TMF8701/TMF8801/TMF8805 Datasheet(s)
-

6 Revision Information

Changes from previous version to current revision v3-00	Page
Added description for missing and unused ABI files	8
Split sensor ABI description in two sections	8
Added interrupt suppression feature configuration description	17

- Page and figure numbers for the previous version may differ from page and figure numbers in the current revision.
- Correction of typographical errors is not explicitly mentioned.

7 Legal Information

Copyrights & Disclaimer

Copyright ams-OSRAM AG, Tobelbader Strasse 30, 8141 Premstaetten, Austria-Europe. Trademarks Registered. All rights reserved. The material herein may not be reproduced, adapted, merged, translated, stored, or used without the prior written consent of the copyright owner.

Information in this document is believed to be accurate and reliable. However, ams-OSRAM AG does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and shall have no liability for the consequences of use of such information.

Applications that are described herein are for illustrative purposes only. ams-OSRAM AG makes no representation or warranty that such applications will be appropriate for the specified use without further testing or modification. ams-OSRAM AG takes no responsibility for the design, operation and testing of the applications and end-products as well as assistance with the applications or end-product designs when using ams-OSRAM AG products. ams-OSRAM AG is not liable for the suitability and fit of ams-OSRAM AG products in applications and end-products planned.

ams-OSRAM AG shall not be liable to recipient or any third party for any damages, including but not limited to personal injury, property damage, loss of profits, loss of use, interruption of business or indirect, special, incidental or consequential damages, of any kind, in connection with or arising out of the furnishing, performance or use of the technical data or applications described herein. No obligation or liability to recipient or any third party shall arise or flow out of ams-OSRAM AG rendering of technical or other services.

ams-OSRAM AG reserves the right to change information in this document at any time and without notice.

RoHS Compliant & ams Green Statement

RoHS Compliant: The term RoHS compliant means that ams-OSRAM AG products fully comply with current RoHS directives. Our semiconductor products do not contain any chemicals for all 6 substance categories plus additional 4 substance categories (per amendment EU 2015/863), including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, RoHS compliant products are suitable for use in specified lead-free processes.

ams Green (RoHS compliant and no Sb/Br/Cl): ams Green defines that in addition to RoHS compliance, our products are free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material) and do not contain Chlorine (Cl not exceed 0.1% by weight in homogeneous material).

Important Information: The information provided in this statement represents ams-OSRAM AG knowledge and belief as of the date that it is provided. ams-OSRAM AG bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. ams-OSRAM AG has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. ams-OSRAM AG and ams-OSRAM AG suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

Headquarters

ams-OSRAM AG
Tobelbader Strasse 30
8141 Premstaetten
Austria, Europe
Tel: +43 (0) 3136 500 0

Please visit our website at www.ams.com

Buy our products or get free samples online at www.ams.com/Products

Technical Support is available at www.ams.com/Technical-Support

Provide feedback about this document at www.ams.com/Document-Feedback

For sales offices, distributors and representatives go to www.ams.com/Contact

For further information and requests, e-mail us at ams_sales@ams.com