CMOSTEK

AN198

CMT2300A/CMT2119B/CMT2219B state switching precautions

summary

This article introduces the operating matters that need to be paid attention to when switching the chip status of the three products CMT2300A / CMT2119B / CMT2219B to help Help users design and apply more conveniently.

The product models covered by this document are shown in the table below.

Table **1.** Product models covered by this document

Product number	Working frequency modulation mode Main function configuration mode	CMT2300A	encapsulation
126.33 - 1020MHz (G)FSK/OOK transceiver register	CMT2119B 126.33 - 1020MHz (G)FSK/OOK transmitter		QFN16
register CMT2219B 126.33 - 1020MHz (G)FSK/OOK receiver register			QFN16
			QFN16

Before reading this document, it is recommended to read "AN142-CMT2300A Quick Start Guide", "AN184-CMT2119B Quick Start Guide", and "AN161-CMT2219B Quick Start Guide" to understand the basic usage of the three products.

Table of contents

1. Precautions for status switching

1.1 State switching diagram and related registers

The following is the state switching diagram and related registers of CMT2300A. For CMT2119B (transmitter), users can ignore the RFS and RX status information; for CMT2219B (receiver), the user can ignore the information about TFS and TX status. besides,

The state switching operations and registers of the three products are exactly the same.

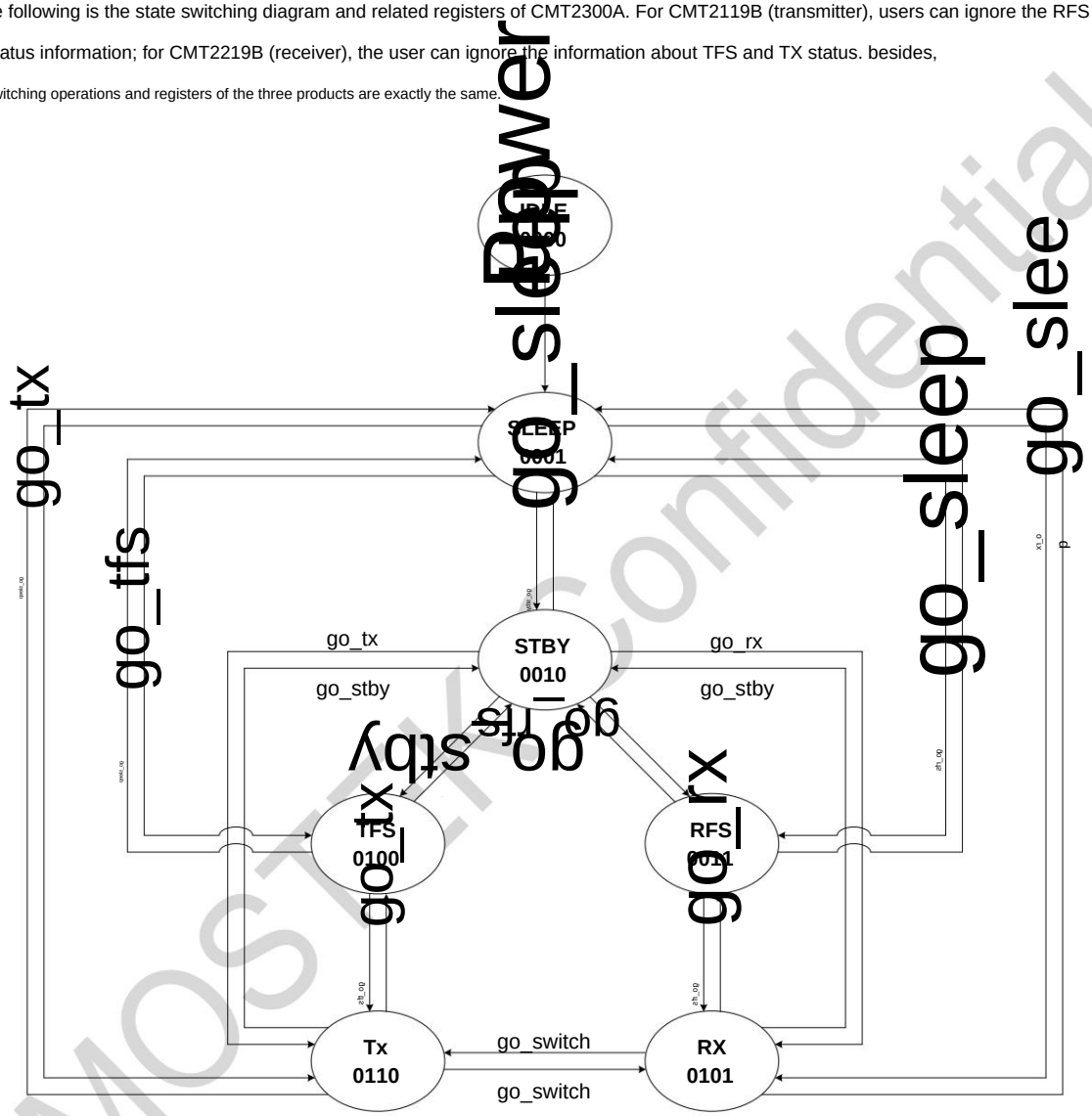


Figure **1. CMT2300A** state switching diagram

Table 2. Registers for switching states

Register name	Number of digits	R/W	Bit name	Function Description
CUS_MODE_CTL (0x60)	7:0	RW	CHIP_MODE_SWT<7:0>	State switching command: 00000010:go_stby 00000100:go_rfs 00001000: go_rx

				00010000: go_sleep 00100000: go_tfs 01000000: go_tx 10000000: go_switch Remaining values: Not
CUS_MODE_STA (0x61)	3:0 RW	CHIP_MODE_STA<3:0>		allowed to send. Chip status: 0000: IDLE 0001: SLEEP 0010ÿSTBY 0011:RFS 0100ÿTFS 0101ÿRX 0110ÿTX 1000ÿLOCKING 1001:CAL Remaining values: Invalid LOCKING state refers to the state in which the PLL is locking. When the locking is completed, it will continue to enter the TX/RX state CAL status is the calibration status. The chip will not stay in the calibration status for a long time, so users usually cannot query the CAL status.

Neither the LOCKING nor the CAL status appear in the flow chart because under normal circumstances, they are a short-lived process that varies from customer to customer. It can definitely be obtained by querying through the serial port (it may also be obtained by querying, depending on the query speed of the serial port).

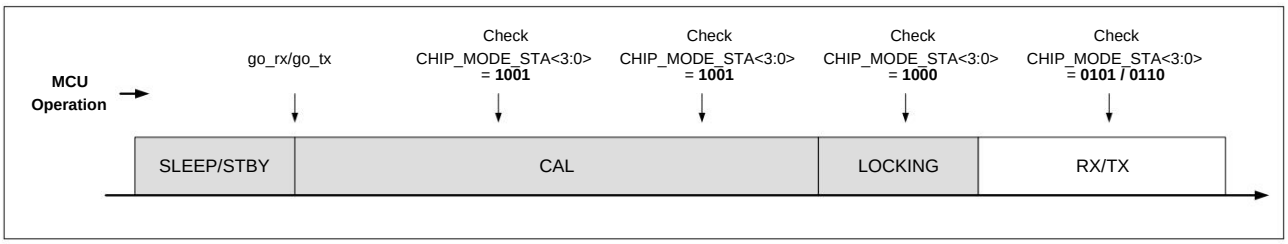
The user must force the register LOCKING_EN in the control area to 1 to make the LOCKING state valid. If LOCKING_EN is set to It is 0, which means that for each LOCKING state, the system waits for 100 us by default before entering the TX/RX state. However, the PLL may not be locked, resulting in transmission or reception errors. Therefore LOCKING_EN must be set to 1 under normal circumstances.

1.2 Operation of status query

Normally, after the user sends a state switching command by setting CHIOP_MODE_SWT<7:0>, the user needs to The CHIP_MODE_STA<3:0> register is queried to confirm that the command has been executed and the status has been successfully switched. In a variety of ever-changing application scenarios , there are the following possible situations.

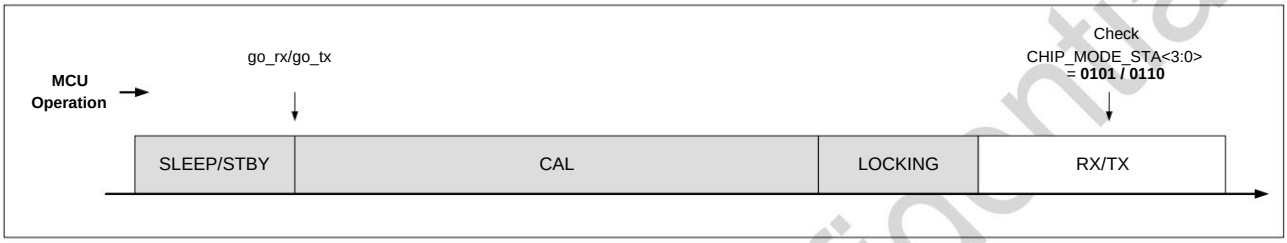
1.2.1 Query after sending **go_rx/go_tx** in **SLEEP/STBY** state

Normally, it takes 350 us to switch from SLEEP/STBY to RX/TX (excluding the crystal oscillator start-up and stabilization time, which depends on the crystal characteristics. The start-up time is about 0.5-1 ms, and the stabilization time is determined by the user. Configuration register XTAL_STB_TIME <2:0> set). If the MCU operates the serial port quickly, the CAL and LOCKING processes will be queried during this process, as shown in the following figure:



If the MCU operates the serial port slowly, the process of CAL and LOCKING may not be queried. The chip has already been queried during the first query.

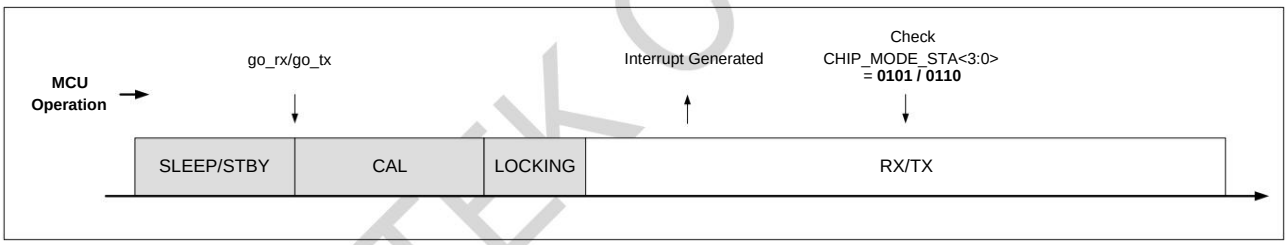
Entered RX/TX state, as shown in the figure below:



In the above situation, another situation may occur if the user enables certain interrupts in the RX/TX state, such as sync detection

Interrupts and FIFO interrupts, etc. These interrupts may be generated before the first query. At this time, the MCU may execute the interrupt program and delay the status.

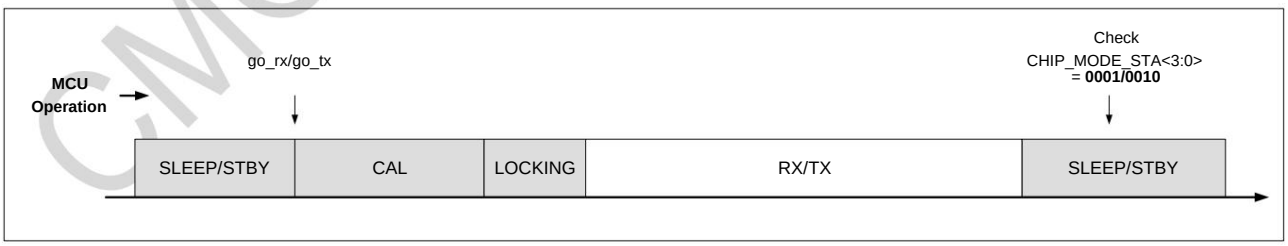
Inquire:



If the MCU operates the serial port very slowly, or the user sets the RX TIMER to allow the chip to automatically exit the RX state, and the RX

The time window is very short; or the TX transmission time is very short, and the chip automatically returns to the SLEEP/STBY state after completion, then the following may occur.

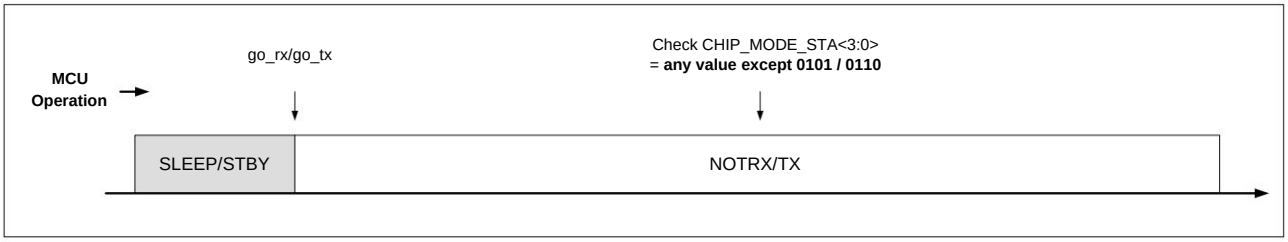
In this case, the chip has completed the first query and exited the RX/TX state:



There is also a very rare situation where the chip suffers from strong electrostatic interference and malfunctions, which is manifested as being unable to enter the RX/TX state. Suppose

At this time, the MCU does not malfunction, but continues to query the chip status, then the correct status may never be queried. This reminds the user that in the MCU

If there is a loop that keeps querying in the program, a timeout mechanism should be added to prevent the system from crashing when this happens.



Of course, in addition to the situations listed above, other situations may occur, but for a specific application, as long as the user clearly analyzes the serial port speed, packet format, data rate, control mechanism, RX and TX time used, With the use of interrupts, you can perform appropriate query operations and formulate a timeout mechanism in the MCU program to avoid the situation where the query is always unavailable or the query is incorrect.

1.2.2 Query after sending **go_rx** in **RFS** state or **go_tx** in **TFS** state

There is no need to go through the CAL and LOCKING processes when switching from the RFS state to the RX state, or from the TFS state to the TX state. However, there will also be several situations mentioned above after entering the RX/TX state, and users also need to pay attention.

1.3 Use of **go_switch** command

As mentioned earlier, the go_switch command is used to switch directly to RX in the TX state, or directly switch to TX in the RX state. It is forbidden to use this command in the status, otherwise it will cause system errors.

1.3.1 Switch from **TX** to **RX**

Sending go_switch from the TX state is conditional, and the condition is that the Data Mode must be set to Direct mode. This is because in Packet mode, the TX state automatically exits after the data packet is sent, and cannot be exited by sending the go_* command through the MCU, otherwise the system may become unstable.

1.3.2 Switch from **RX** to **TX**

In the RX state, if the RX TIMER is enabled to automatically exit the RX state, do not use the go_switch command. The command can only be used when the chip returns the switching authority to the MCU. For the various automatic and manual control modes that may exist in RX, please refer to "AN146-CMT2300A Low Power Mode User Guide" and "AN164-CMT2219B Low Power Mode User Guide".

2. Document change record

Table 3. Document change record table

version number	chapter	Change description	date
0.8	All	initial releases	2017-10-28

3.Contact information

Wuxi Zetai Microelectronics Co., Ltd. Shenzhen Branch

Room 203, Hon Hai Building, Qianhai Road, Nanshan District, Shenzhen City, Guangdong Province, China

post code: 518000

Telephone: +86-755-83235017

fax: +86-755-82761326

Sale: sales@cmostek.com

Technical support: support@cmostek.com

URL: www.cmostek.com