

---

## CMT2300AW RF Parameter Configuration Guideline

---

### Overview

This document introduces the RF parameter configuration, OOK and FSK modulation parameter configuration for CMT2300AW RFPDK.

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

**Table 1. Product Models Covered in This Document**

Part No.	Frequency	Modem	Function	Configuration	Package
CMT2300AW	127 - 1020MHz	(G)FSK/OOK	Transceiver	Register	QFN16

Before reading this document, it is recommended to read **AN142-CMT2300AW Quick Start Guideline** to learn the basic usage of CMT2300AW.

---

**Table of Contents**

<b>1 Tx and Rx Parameter Configuration .....</b>	<b>3</b>
1.1 Basic Parameter Configuration.....	3
1.2 OOK Demodulation Configuration .....	4
1.3 FSK Demodulation Configuration .....	5
1.4 CDR Design Index and Mode Selection .....	6
1.5 Register Classification and Use in the Tx Bank.....	10
<b>2 Revise History .....</b>	<b>11</b>
<b>3 Contacts.....</b>	<b>12</b>

CMOSTEK Confidential

# 1 Tx and Rx Parameter Configuration

## 1.1 Basic Parameter Configuration

The screenshot shows a configuration window titled 'Chip Parameters' with a sub-section 'RF Settings'. It contains several input fields and dropdown menus:

- Frequency (126.334-1020):** 433.920 MHz
- Xtal Frequency:** 26.0000 MHz
- (De)Modulation:** FSK
- AGC:** On
- Data Rate (0.5-300):** 2.4 kbps
- Deviation (2-200):** 20.0 kHz
- Tx Xtal Tol. (0-50):** +/- 20 ppm
- Rx Xtal Tol. (0-50):** +/- 20 ppm
- TRx Matching Network Type:** 13 dBm
- Tx Power:** +13 dBm
- Gaussian BT:** NA
- PA Ramp:** On

Figure 1. RF Parameter Configuration Interface

The registers corresponding to these parameters are stored in the frequency bank, the symbol rate bank, and the Tx bank, It needs to be calculated by complex calculation. The user does not need to understand the calculation process and the specific meaning of the register.

RFPDK interface parameters are easy to understand. The following configurations need attention.

### How to set the Deviation:

Generally speaking, the following 3 points are recommended when configuring the Deviation:

1.  $4\text{kHz} < \text{Deviation} < 200\text{ kHz}$ , and  $\text{Symbol rate} * 0.5 + \text{Deviation} \leq 250\text{ kHz}$
2.  $\text{Symbol rate} * 0.25 \leq \text{Deviation}$ , that is, the modulation index must not be less than 0.5 (MSK).
3. In the case of meeting the above two conditions, if  $\text{Symbol rate} * 0.5 \leq \text{Deviation} \leq \text{Symbol rate} * 2$ , it can achieve the best sensitivity.

### How to set the crystal PPM

The user inputs the crystal tolerance values for Tx and Rx respectively in RFPDK. If inputting +/-20 ppm respectively, it means that the worst case between the transmitter and receiver is the crystal frequency difference of 40 ppm. The user should consider the worst case to set these two tolerance values, which affect every bandwidth setting within the receiver.

## 1.2 OOK Demodulation Configuration

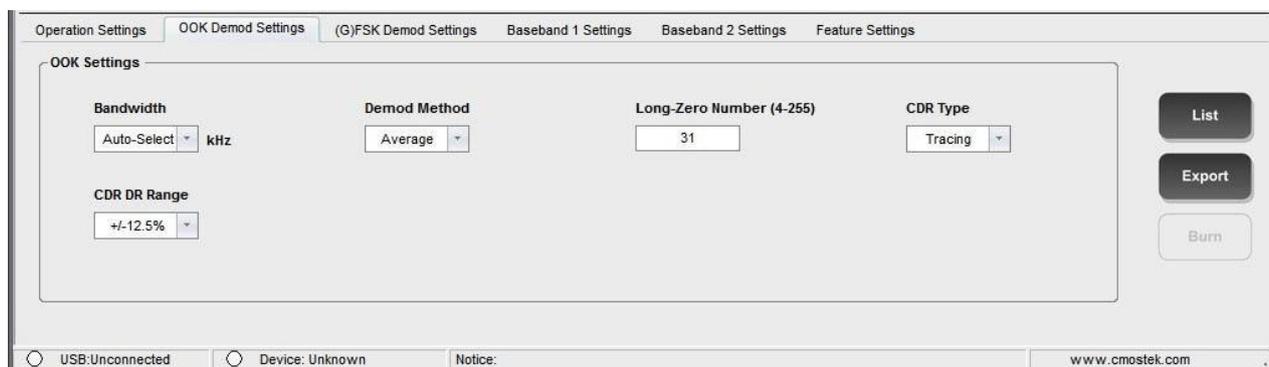


Figure 2. OOK Configuration Parameter

Here is the configuration suggestion:

Table 2. OOK Parameter Configuration

RFPDK Input Parameter	Configuration Suggestion
Bandwidth	It is recommended to select Auto-Select
Demod Method	<p>Middle Demodulation response speed is fast. The advantage is that the correct demodulation can be made immediately after each communication is started. The disadvantage is that if there is a spike mutation in the packet (a sudden abrupt increase in amplitude), it will affect the subsequent period of demodulation.</p> <p>Average Demodulation response speed is slow. The disadvantage is that it takes time for each communication to demodulate correctly (usually 10-20 symbols). The advantage is that the demodulation effect will not be affected if there is a spike mutation in the packet (a sudden abrupt increase in amplitude).</p> <p>The experience is that the spike mutation only exists in some special applications, so it is generally recommended that the demodulation method of Middle can speed up the demodulation and reduce the receiving time.</p>
Long-Zero Number	The user needs to input this value to tell the receiver how many consecutive 0 there are at most in the packet. The receiver will modulate the relevant parameters to ensure the best demodulation results.
CDR Type	In the packet mode, the user needs to detect Preamble or Sync. The symbol rate clock recovery function has to be turned on, and the later chapter will describe how to use CDR in detail.
CDR Range	How to input Parameters when CDR = Tracing. The later chapter will describe how to use CDR in detail.

## 1.3 FSK Demodulation Configuration

The screenshot shows the (G)FSK Demod Settings configuration window. The parameters are as follows:

Parameter	Value
Bandwidth	Auto-Select
AFC	On
AFC Method	Auto-Select
CDR Type	Tracing
CDR DR Range	+/-12.5%
Data Representation	0:F-low 1:F-high
AFC_OVF_TH	10

Buttons: List, Export, Burn

Figure 3. FSK Configuration Parameter

Here is the configuration suggestion.

Table 3. FSK Parameter Configuration

RFPDK Input Parameter	Configuration Suggestion
Bandwidth	Recommend selecting the Auto-Select.
AFC	AFC switch. It is usually recommended to open.
AFC Method	AFC method. Recommend selecting the Auto-Select.
CDR Type	In the packet mode, the user needs to detect Preamble or Sync. The symbol rate clock recovery function has to be turned on, and the later chapter will describe how to use CDR in detail.
CDR Range	How to input parameters when CDR = Tracing. The later chapter will describe how to use CDR in detail.
Data Representation	Select the two frequencies of the (G)FSK, which represents 0 and which represents 1. The default configuration is usually selected.

## 1.4 CDR Design Index and Mode Selection

The corresponding RFPDK interface and parameters are presented:



**Figure 4. CDR Configuration Parameter**

Here is the configuration suggestion:

**Table 4. CDR Parameter Configuration**

Register Bit RFPDK Parameter	Register Bit
CDR Type	CDR_MODE<1:0>
CDR DR Range	CDR_RANGE_SEL<1:0>
Automatically select in conjunction with other configurations in RFPDK. Users do not have to select.	CDR_DET_SEL
Automatically select in conjunction with other configurations in RFPDK. Users do not have to select.	CDR_AVG_SEL<2:0>
Automatically select in conjunction with other configurations in RFPDK. Users do not have to select.	CDR_3RD_EN
Automatically select in conjunction with other configurations in RFPDK. Users do not have to select.	CDR_4TH_EN
Automatically select in conjunction with other configurations in RFPDK. Users do not have to select.	CDR_BR_TH<15:8>

The contents and explanations of the register can be seen in the following table.

Table 5. CDR Related Register Instructions

8.5	Bit	R/W	Bit Name	Function Description
CUS_CDR1 (0x2B)	6:4	RW	CDR_AVG_SEL<2:0>	In Tracing mode, the average filter order of the CDR output: 0: 6/8 1: 1/2 2: 6/16 3: 1/4 4: 11/64 5: 1/8 6: 3/32 7: 1/16
	3:2	RW	CDR_RANGE_SEL<1:0>	In Tracing mode, the maximum deviation range of the symbol length for the CDR: 0: +/-6.3% (Symbol rate range -5.9% to +6.7%) 1: +/-9.4% (Symbol rate range -8.6% to +10.4%) 2: +/-12.5% (Symbol rate range -11.1% to +14.3%) 3: +/-15.6% (Symbol rate range -13.5% to +18.5%)
	1:0	RW	CDR_MODE<1:0>	CDR working mode selection: 0: Tracing 1: Counting 2: Manchester 3: No CDR
	7	RW	CDR_DET_SEL	In tracing mode, select the method to detect the symbol rate: 0: Method 0 1: Method 1
	6	RW	CDR_3RD_EN	In tracing mode, enable the symbol rate detection when the consecutive three symbols do not skip: 0: Disable

CUS_CDR2 (0x2C)	5	RW	CDR_4TH_EN	In tracing mode, enable the symbol rate detection when the consecutive four symbols do not skip: 0: Disable
CUS_CDR3 (0x2D)	7:0	RW	CDR_BR_TH<7:0>	In tracing mode, it is a threshold that CDR needs to use. The calculation method is: TH = crystal frequency / symbol rate. For example, the crystal frequency is 26MHz when the symbol rate is 15 kHz, and the result is 1733 after rounding.
CUS_CDR4 (0x2E)	7:0	RW	CDR_BR_TH<15:8>	

CMT2300AW clock recovery related parameters are stored in the symbol rate bank, automatically configured through the RFPDK. If users do not have special requirements, they do not need to understand the contents of the register described above. Here we introduce the design index and mode selection method of clock recovery. Hereinafter the clock recovery will be abbreviated as CDR.

The basic task of a CDR system is to recover the clock signal that is synchronized with the symbol rate, while receiving the data. Not only for decoding inside the chip, but also for outputting to GPIO for users to sample the data. So CDR's task is simple and important. If the recovered clock frequency is in error with the actual symbol rate, it will cause data acquisition errors, error codes, and decoding errors at the time of reception.

In fact, the application environment is not ideal. Generally speaking, there are two situations when the system works.

1. The symbol rates for TX and RX are accurate, or the error is a few hundred thousandths or a few ten thousandths. This is often because the TX and RX configuration and the mechanism of computing the symbol rate are more advanced, such as using a variety of high-end TRX products.
2. The symbol rates of TX and RX are greatly deviation, such as a few thousandths to a few percent.

At the same time, CDR has three important indicators:

1. How long data can be correctly collected in the case of no data flipping (that is always receiving 0 or receiving 1.)?
2. What is the symbol rate deviation between TX and RX?
3. Can the sensitivity be improved? Can the error code be reduced?

Overall considering the two applications and the three indicators, CMT2300AW has designed three sets of CDR systems, as follows:

1. COUNTING system—The system is designed for the symbol rates to be more accurate. If the symbol rate is 100% aligned, the unlimited length of 0 can be received continuously without error. But if there is a deviation, the deviation will have a cumulative effect when the data is not flipped. For example, a symbol deviation of 5%, then 10 consecutive symbols without flipping, the deviation will accumulate over 50%. This can lead to a sampling error. The reason is the same as the traditional UART transmission. Once the signal is flipped, the deviation will be cleared and a new accumulation will be began. So, when the deviation is very small, such as 1/5000, then you can continuously receive 2500 symbols without flipping data without error. The performance is very good.
2. TRACING system –The system is designed for the symbol rate deviation. It has the tracking function. It can automatically detect the symbol rate transmitted by TX, and adjust quickly the local symbol rate of RX at the same time, so as to minimize

the error between them. The system can withstand up to 15.6%. Other similar products in the industry cannot reach this level. On the other hand, the effect of this system depends on the accuracy of the local symbol rate adjustment. Because the crystal frequency is fixed at 26MHz, the higher the symbol rate, the lower the accuracy of the adjustment, Generally speaking, the accuracy will be higher if the symbol rate is below 50 kHz. It can receive more continuous no flipping data without error. That is, the lower the symbol rate, the more the correct sampling of data, the actual symbol depends on the actual measurement. If the symbol rate is above 50 kHz, the amount of no flipping data that can be sampled correctly will become less and less. If the symbol rate is 250 kHz, it is possible to only correctly sample 2-3 bytes without flipping data because of the low accuracy.

3. MANCHESTER system –This system is changed by the COUNTER system. The basic feature is the same. The only difference is that the system is specially designed for Manchester codec, Special processing can be done when the TX symbol rate has unexpected changes. A common mutation is that TX data sometimes has two consecutive 0 or 1 patterns after being encoded by Manchester, If one of the 0 or 1 symbol length suddenly grows longer than 50%, then the RX may sample three 0 or 1. If the symbol length suddenly becomes shorter more than 50%, RX will only sample one 0 or 1. In order to answer such a special situation, in this clock recovery system, the sampling will stop if 3 or more symbols without flipping are detected in the received data. This avoids the error when the symbol suddenly grows. What happens if the symbol length suddenly becomes shorter? Because this situation is often common in the current application, users can uniformly speed up the symbol rate of RX. The amount of symbol can be correctly sampled when the symbol length is shorter than 50%.

#### How to select the suitable CDR system:

RFPDK will select the system according to the configuration of the symbol rate. When the symbol rate is less than or equal to 50 kHz, the TRACING system is selected. The system can not only realize the tracking, but also correctly sample the long data without flipping. When the symbol rate is greater than 50 kHz, less than 200 kHz, it can be set according to the actual need of users. If users tend to accept the large symbol rate error, users can use the TRACING system. If users tend to accept the long data without flipping, users can use the COUNTING system. When the symbol rate is greater than 200 kHz, TRACING performs better than COUNTING. When the entire packet conforms to the Manchester coding rule (i.e., no more than 2 consecutive symbols without flipping), and finds the symbol rate mutation in the application, MANCHESTER system can be considered.

In addition, no matter which type of CDR system is adopted, it will improve the sensitivity of the 1-2dB than receiving the RAW data after the system is turned on.

#### How to debug the TRACING parameter:

Tracing's tracking function is done every time when the data is flipped. So the ideal situation is that each symbol will make an adjustment when the packet has preamble. This will speed up the tracking speed.

The following introduces the debugging methods for several parameters.

- **CDR\_RANGE\_SEL<1:0>**

This parameter selects the symbol rate deviation that can be tracked. The percentage given in the register content is not the percentage of the symbol rate, but it's the percentage of the symbol length, For example, the symbol rate is 10 kHz, a symbol length is 100 ns, then +/-15.6% is refers to the TX symbol length is in the range of 84.4 ns to 115.6 ns, RX can detect the symbol and adjust its length to close to TX, so as to eliminate the deviation of the symbol rate. If it is converted to a percentage of the symbol rate, then +15.6% length percentage can be converted to -13.5% symbol rate percentage, and -15.6% can be converted to +18.5%.

The user might ask why not directly select the maximum range of tracking. This is because the larger the tracking range, the harder it is to track, and the more time it takes. Therefore, it is recommended that the user select the most appropriate range according to the actual application requirement.

- **CDR\_3RD\_EN, CDR\_4TH\_EN**

The two parameters are tied to the choice of CDR\_RANGE\_SEL<1:0>. When +/-15.6% is selected, none of them can be enabled. When +/-12.5% is selected, CDR\_4TH\_EN cannot be enabled. The rest can be enabled. It's not explained in detail here

- **CDR\_AVG\_SEL<2:0>**

The smaller the filter coefficient is, the narrower the bandwidth is, the slower the symbol rate tracking speed is, and the smoother the output is. The larger the filter coefficient is, the wider the bandwidth is, the faster the symbol rate tracking speed is, and the easier the output jitter, that's similar to how PLL works. Usually we hope the tracking speed as fast as possible (for example, completed during several symbols in front of preamble), also hope to output stability, otherwise it is easy to cause the error later, and especially there are more long continuous 0 or 1. Therefore, if the user wants to adjust this parameter himself, it is necessary to debug the effect according to the actual situation.

In addition, generally speaking, the higher the symbol rate, the more jitter will be. The smaller the filter coefficient is, the higher the receiving accuracy of the continuous long 0 and 1. At the same time, there is an acceptable tradeoff and choice whether to slow down the tracking speed.

## 1.5 Register Classification and Use in the Tx Bank

Usually, the register of the Tx bank is entirely generated by the RFPDK, It is OK that the user configures once in the chip initialization phase. Then, considering that some users will individually modify the Tx power, symbol rate and other parameters in the application process, the Tx bank can be further classified and divided so that users can configure the target accordingly.

The following table shows different bands in different color:

Addr	R/W	Name	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0x55	RW	CUS_TX1								
0x56	RW	CUS_TX2								
0x57	RW	CUS_TX3								
0x58	RW	CUS_TX4								
0x59	RW	CUS_TX5								
0x5A	RW	CUS_TX6								
0x5B	RW	CUS_TX7								
0x5C	RW	CUS_TX8								
0x5D	RW	CUS_TX9								
0x5E	RW	CUS_TX10								
0x5F	RW	CUS_LBD								

Figure 5. Tx Related Register Classification

Table 6. Tx Related Register Classification

Color	Function Description
White	Set switches or options for various Tx functions.
Gray	Set the Deviation in FSK mode.
Green	Set the Ramp parameter. It is related to the symbol rate. That is, each time users change the TX symbol rate, users must configure the bank again.
Blue	Set Tx power and LBD power compensation, where the contents of LBD_COMP_OFFSET<2:0> cannot be fully relied on RFPDK, because it is closely related to the application environment, which will be described in detail below.
Yellow	LBD threshold usage, please refer to the related content of "Low Battery Detection (LBD)" in the next chapter.

The user does not need to know the contents of the register, and can configure the target accordingly, minimizing the number of registers to write.

## 2 Revise History

Table 7. Revise History Records

Version No.	Chapter	Description	Date
0.8	All	Initial release	2017-03-24
0.9	All	Add the advice to read AN142; Modify individual typos	2017-07-12
1.0	All	Update RFPDK screenshot to keep consistent with the latest software	2020-05-22

CMOSTEK Confidential

## 3 Contacts

CMOSTEK Microelectronics Co., Ltd. Shenzhen Branch

Address: 2/F Building 3, Pingshan Private Enterprise S.T. Park, Xili, Nanshan District, Shenzhen, Guangdong, China

**Tel:** +86-755-83231427

**Post Code:** 518071

**Sales:** [sales@cmostek.com](mailto:sales@cmostek.com)

**Supports:** [support@cmostek.com](mailto:support@cmostek.com)

**Website:** [www.cmostek.com](http://www.cmostek.com)

**Copyright. CMOSTEK Microelectronics Co., Ltd. All rights are reserved.**

The information furnished by CMOSTEK is believed to be accurate and reliable. However, no responsibility is assumed for inaccuracies and specifications within this document are subject to change without notice. The material contained herein is the exclusive property of CMOSTEK and shall not be distributed, reproduced, or disclosed in whole or in part without prior written permission of CMOSTEK. CMOSTEK products are not authorized for use as critical components in life support devices or systems without express written approval of CMOSTEK. The CMOSTEK logo is a registered trademark of CMOSTEK Microelectronics Co., Ltd. All other names are the property of their respective owners.