RTL9607C
SOC

PRBS
Application Note
(CONFIDENTIAL: Development Partners Only)

Rev. 1.0.0
28th March 2017

Realtek Semiconductor Corp.
No. 2, Innovation Road II, Hsinchu Science Park, Hsinchu 300, Taiwan
Tel.: +886-3-578-0211 Fax: +886-3-577-6047
www.realtek.com

COPYRIGHT

TRADEMARKS

Realtek is a trademark of Realtek Semiconductor Corporation. Other names mentioned in this document are trademarks/registered trademarks of their respective owners.

DISCLAIMER

Realtek provides this document "as is", without warranty of any kind, neither expressed nor implied, including, but not limited to, the particular purpose. Realtek may make improvements and/or changes in this document or in the product described in this document at any time. This document could include technical inaccuracies or typographical errors.

USING THIS DOCUMENT

Though every effort has been made to assure that this document is current and accurate, more information may have become available subsequent to the production of this guide. In that event, please contact your Realtek representative for additional information that may help in the development process.

CONFIDENTIALITY

This document is confidential and should not be provided to a third-party without the permission of Realtek Semiconductor Corporation.

REVISION HISTORY

Revision	Release Date	Summary
1.0.0	2016/03/18	First Release

Table of Contents

List of Tables

1. Overview

Pseudo-Random Bit Sequence (PRBS) is a repeating pattern that has properties similar to random sequences and useful to be a verification tool. The SerDes in RTL9607C offer BIST functions in the form of a PRBS pattern generator in the Serializer and a corresponding pattern checker in the De-serializer. In the BIST mode, instead of the normal data inputs, the Serializer takes the parallel data from the BIST pattern generator to generate the serial outputs. The PRBS pattern checker in the De-serializer checks the received data against the expected pattern to determine if there is any bit error. A SerDes can test itself by looping back the PRBS pattern from the Serializer to the De-serializer.

2. GPON Mode

2.1. PRBS Configuration

Since PRBS is a debug and verification tool only, we don't provide the specific APIs for user to configure it. For starting PRBS functionality, it can be separated to two parts: PRBS selection and PRBS enable. About selection, RTL9607C provide a separated setting of patterns for generator and de-serializer.

Below show the list of supported patterns:

Table 1. GPON PRBS pattern selection

Value	Type	Length	Freq.		
f	prbs31	16 bits	2.488G		
e	prbs23	16 bits	2.488G		
d	prbs15	16 bits	2.488G		
c	prbs07	16 bits	2.488G		
b	prbs03	16 bits	2.488G		
a	prbs31	8 bits	1.244G		
9	prbs23	8 bits	1.244G		
8	prbs15	8 bits	1.244G		
7	prbs07	8 bits	1.244G		
6	prbs03	8 bits	1.244G		
5	txdis test1	16 bits	rx not check error		ben 0x5555 / dis 0xaaaa
4	txdis test2	16 bits	rx not check error		ben 0x3333 / dis 0xcccc
3	0x5555	16 bits	rx not check error		
2	0x3333	16 bits	rx not check error		
1	0x1111	16 bits	rx not check error		
0	0x0000	16 bits	rx not check error		

For selected pattern or other PRBS operation, chip provide below options:

Table 2. GPON PRBS selection

Register	Field	Description	Bits
CFG_PRBS_TYPE_SEL	clear prbs error counter	1: clear prbs errors	<15>
	latch rx receive data	1: latch rx_data	<14>
	select even bits for 8bits rx receive data	0: select bit[1, 3, …, 15] of rx_data to check 1: select bit[0, 2, …, 14] of rx_data to check	<13>
	select txdis test period	1 : ben 400T / dis 400T ; 0 : ben 510 / dis 511	<12>
	select txdis test delay	0 : no delay ; 1 : delay 1T ; 2 : delay 2T ; 3 : delay 3T	<11:10>
	select txdis test position	00: F000FF ; 01: F100EF ; 10: F300CF ; 11: F7008F	<9:8>
	tx test pattern select	Reference Table 1	<7:4>
	rx test pattern select	Reference Table 1	<3:0>

After selected which TX/RX mode that you want, you can enable the PRBS for starting to test.

Table 3. GPON PRBS enable

Register	Description	Bits
CFG_PRBS_EN	1: enable or 0: disable PRBS	<0>

Note 1: The field of latch RX and TX/RX test pattern select must be filled before you enable the PRBS test.

Note 2： The Tx power of transceiver should be enabled before prbs test. In our sdk, we use GPIO pin 13 to control tx power of transceiver. If you follow our examples in chapter 5 and 6, the tx power is default enabled.

2.2. PRBS Verification

RTL9607C provide two read only registers as Table 4 for checking the result of PRBS.

Table 4. GPON PRBS status counters

Register	Description	Bits
CFG_PRBS_STATUS	rx receive data	<15:0>

Table 5. GPON PRBS error counters

Register	Field	Description	Bits
CFG_PRBS_ERRORS	error counter	prbs error counter	<15:0>

The receive data should always increase after enabling the PRBS and latch RX receive data. When an error occurred, the counter of PRBS error will be increased. If you want to clear error counters, you can write 1 to bit 15 of ***CFG_PRBS_TYPE_SEL*** and restart testing.

2.3. Registers

Below show the address of the PRBS related register and address:

Table 6. GPON PRBS register

Register	Address
CFG_PRBS_SEL	0x40094
CFG_PRBS_EN	0x40098
CFG_PRBS_ERRORS	0x4009C
CFG_PRBS_STATUS	0x400A0

3. EPON/Fiber Mode

3.1. PRBS Configuration

Since PRBS is a debug and verification tool only, we don't provide the specific APIs for user to configure it. For starting PRBS functionality, it can be separated to two parts: PRBS selection and PRBS enable. About selection, RTL9607C provide a separated setting of patterns for generator and de-serializer.

Below show the list of supported patterns:

Table 7. EPON/Fiber PRBS pattern selection

Value	Type	Length	Freq.
3	prbs15	8 bits	1.25G
2	prbs11	8 bits	1.25G
1	prbs9	8 bits	1.25G
0	prbs7	8 bits	1.25G

For selected pattern or other PRBS operation, chip provide below options:

Table 8. EPON/Fiber PRBS selection

Register	Field	Description	Bits
SEP_CFG_PRBS_SEL	test pattern select	Reference Table 7.	<9:8>

And reverse the polarity of the PRBS code to communicate with the device:

Table 9. EPON/Fiber PRBS reverse

Register	Description	Bits
SEP_CFG_PRBS_REVERSE	Use to reverse the polarity of the PRBS code to communicate with the device.	<13>

After selected which TX/RX mode that you want, you can enable the PRBS for starting to test.

Table 10. EPON/Fiber PRBS enable

Register	Description	Bits
CFG_PRBS_EN	1: enable or 0: disable PRBS	<11>

Note 1： The Tx power of transceiver should be enabled before prbs test. In our sdk, we use GPIO pin 13 to control tx power of transceiver. If you follow our examples in chapter 5 and 6, the tx power is default enabled.

3.2. PRBS Verification

RTL9607C provide two registers as Table 4 for checking the result of PRBS error counter.

Table 11. EPON/Fiber PRBS error counters

Register	Field	Description	Bits
CFG_PRBS_ERRORS [31:16]	error counter	prbs error counter	<15:0>
CFG_PRBS_ERRORS [15:0]	error counter	prbs error counter	<15:0>

The receive data should always increase after enabling the PRBS. When an error occurred, the counter of PRBS error will be increased. If you want to clear error counters, you can **write all 0** to both of ***CFG_PRBS_ERRORS*** and restart testing.

3.3. Registers

Below show the address of the PRBS related register and address:

Table 12. EPON/Fiber PRBS register

Register	Address
CFG_PRBS_SEL	0x40a30
CFG_PRBS_EN	0x40a2C
CFG_PRBS_ERRORS [31:16]	0x40a14
CFG_PRBS_ERRORS [15:0]	0x40a10

4. HSG0/SG0/FIB1G Mode

4.1. PRBS Configuration

Since PRBS is a debug and verification tool only, we don't provide the specific APIs for user to configure it. For starting PRBS functionality, it can be separated to two parts: PRBS selection and PRBS enable. About selection, RTL9607C provide a separated setting of patterns for generator and de-serializer.

Below show the list of supported patterns:

Table 13. HSG0/SG0/FIB1G pattern selection

Value	Type	Length	Freq.
3	prbs15	8 bits	1.25G
2	prbs11	8 bits	1.25G
1	prbs9	8 bits	1.25G
0	prbs7	8 bits	1.25G

For selected pattern or other PRBS operation, chip provide below options:

Table 14. HSG0/SG0/FIB1G PRBS selection

Register	Field	Description	Bits
SEP_CFG_PRBS_SEL	test pattern select	Reference Table 7.	<9:8>

And reverse the polarity of the PRBS code to communicate with the device:

Table 15. HSG0/SG0/FIB1G PRBS reverse

Register	Description	Bits
SEP_CFG_PRBS_REVERSE	Use to reverse the polarity of the PRBS code to communicate with the device.	<13>

After selected which TX/RX mode that you want, you can enable the PRBS for starting to test.

Table 16. HSG0/SG0/FIB1G PRBS enable

Register	Description	Bits
CFG_PRBS_EN	1: enable or 0: disable PRBS	<11>

Note 1： The Tx power of transceiver should be enabled before prbs test. In our sdk, we use GPIO pin 13 to control tx power of transceiver. If you follow our examples in chapter 5 and 6, the tx power is default enabled.

4.2. PRBS Verification

RTL9607C provide two registers as Table 4 for checking the result of PRBS error counter.

Table 17. HSG0/SG0/FIB1G PRBS error counters

Register	Field	Description	Bits
CFG_PRBS_ERRORS [31:16]	error counter	prbs error counter	<15:0>
CFG_PRBS_ERRORS [15:0]	error counter	prbs error counter	<15:0>

The receive data should always increase after enabling the PRBS. When an error occurred, the counter of PRBS error will be increased. If you want to clear error counters, you can **write all 0** to both of ***CFG_PRBS_ERRORS*** and restart testing.

4.3. Registers

Below show the address of the PRBS related register and address:

Table 18. HSG0/SG0/FIB1G PRBS register

Register	Address
CFG_PRBS_SEL	0x41a30
CFG_PRBS_EN	0x41a2C
CFG_PRBS_ERRORS [31:16]	0x41a14
CFG_PRBS_ERRORS [15:0]	0x41a10

5. HSG1/SG1 Mode

5.1. PRBS Configuration

Since PRBS is a debug and verification tool only, we don't provide the specific APIs for user to configure it. For starting PRBS functionality, it can be separated to two parts: PRBS selection and PRBS enable. About selection, RTL9607C provide a separated setting of patterns for generator and de-serializer.

Below show the list of supported patterns:

Table 19. HSG1/SG1/FIB1G PRBS pattern selection

Value	Type	Length	Freq.
3	prbs15	8 bits	1.25G
2	prbs11	8 bits	1.25G
1	prbs9	8 bits	1.25G
0	prbs7	8 bits	1.25G

For selected pattern or other PRBS operation, chip provide below options:

Table 20. HSG1/SG1/FIB1G PRBS selection

Register	Field	Description	Bits
SEP_CFG_PRBS_SEL	test pattern select	Reference Table 7.	<9:8>

And reverse the polarity of the PRBS code to communicate with the device:

Table 21. HSG1/SG1/FIB1G PRBS reverse

Register	Description	Bits
SEP_CFG_PRBS_REVERSE	Use to reverse the polarity of the PRBS code to communicate with the device.	<13>

After selected which TX/RX mode that you want, you can enable the PRBS for starting to test.

Table 22. HSG1/SG1/FIB1G PRBS enable

Register	Description	Bits
CFG_PRBS_EN	1: enable or 0: disable PRBS	<11>

Note 1： The Tx power of transceiver should be enabled before prbs test. In our sdk, we use GPIO pin 13 to control tx power of transceiver. If you follow our examples in chapter 5 and 6, the tx power is default enabled.

5.2. PRBS Verification

RTL9607C provide two registers as Table 4 for checking the result of PRBS error counter.

Table 23. HSG1/SG1/FIB1G PRBS error counters

Register	Field	Description	Bits
CFG_PRBS_ERRORS [31:16]	error counter	prbs error counter	<15:0>
CFG_PRBS_ERRORS [15:0]	error counter	prbs error counter	<15:0>

The receive data should always increase after enabling the PRBS. When an error occurred, the counter of PRBS error will be increased. If you want to clear error counters, you can **write all 0** to both of ***CFG_PRBS_ERRORS*** and restart testing.

5.3. Registers

Below show the address of the PRBS related register and address:

Table 24. HSG1/SG1/FIB1G PRBS register

Register	Address
CFG_PRBS_SEL	0x42a30
CFG_PRBS_EN	0x42a2C
CFG_PRBS_ERRORS [31:16]	0x42a14
CFG_PRBS_ERRORS [15:0]	0x42a10

6. GPON Mode Example

Since PRBS on RTL9607C is a debug and verification tool only, we only provide the example with RTK shell. The shell procedure including: initialization, configuration, verification, clear counters and SerDes loopback.

6.1. Initialization

Before testing, you should set to GPON mode, and then reboot. After rebooting, you should go into diagshell.

Example:

```
# flash set PON_MODE 1  /* Set PON Mode to GPON mode */
# reboot -f             /* Reboot ONT */
# diag                  /* Entering diagshell */
RTK.0>
```


6.2. Configuration

Below show an example of PRBS configuration for TX: 1.244G, RX: 2.488G with PRBS 31 patterns:

Example:

```
/*
  TX: PRBS 31 1.244G
  RX: PRBS 31 2.488G
*/

RTK.0> register set 0x40094 0x40af  /*PRBS selection*/
RTK.0> register set 0x40098 0x1     /*enable PRBS*/
```

6.3. Verification

After testing, you can read the counter for checking result:

Example:

```
/*
  PRBS Status and Error Counter
```

```
*/
RTK.0> register get 0x4009c      /*Get PRBS error counter*/
Register 0x4003C : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
RTK.0> register get 0x400A0      /*Get PRBS status counter*/
Register 0x40040 : 0x000001e7    /*Status counter 0x1e7*/
```

6.4. Clear Counters

If you want to retest, you can clear counter with below command:

Example:

```
/*
  PRBS Status and Error Counter Clear
*/

RTK.0> register set 0x40094 0x80af /*PRBS selection for clear counter*/
RTK.0> register set 0x40094 0x40af /*PRBS selection for latch counter*/
```

7. EPON/Fiber Mode Example

Since PRBS on RTL9607C is a debug and verification tool only, we only provide the example with RTK shell. The shell procedure including: initialization, configuration, verification, clear counters and SerDes loopback.

7.1. Initialization

Before testing, you should set to EPON/Fiber mode, and then reboot. After rebooting, you should go into diagshell and initialize SerDes for EPON mode prbs test.

Example:

```
# flash set PON_MODE 2  /* Set PON Mode to EPON mode */
                        /* 2:EPON 3:FIB1G */
# reboot -f             /* Reboot ONT */
# diag                  /* Entering diagshell */
```


7.2. Configuration

Below show an example of PRBS configuration for TX: 1.25G, RX: 1.25G with PRBS7 patterns:

Example:

```
/*
  TX: PRBS 7 1.25G
  RX: PRBS 7 1.25G
*/

RTK.0> register set 0x040a30 0x803  /*enable PRBS*/
RTK.0> register set 0x040a2C 0x0000 /*PRBS selection*, *PRBS reverse*/
```

7.3. Verification

After testing, you can read the counter for checking result:

Example:

```
/*
  PRBS Status and Error Counter
*/
RTK.0> register get 0x040a14      /*Get PRBS error counter [31:16]*/
Register 0x40a14 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
RTK.0> register get 0x040a10      /*Get PRBS error counter [15:00]*/
Register 0x40a10 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
```

7.4. Clear Counters

If you want to retest, you can clear counter with below command.

Example:

```
/*
  PRBS Error Counter Clear
*/

RTK.0> register set 0x040a14 0x0000 /*clear PRBS error counter [31:16]*/
RTK.0> register set 0x040a10 0x0000 /*clear PRBS error counter [15:00]*/
```

8. HSG0/SG0/FIB1G Mode Example

Since PRBS on RTL9607C is a debug and verification tool only, we only provide the example with RTK shell. The shell procedure including: initialization, configuration, verification, clear counters and SerDes loopback.

8.1. Configuration

Below show an example of PRBS configuration for TX: 1.25G, RX: 1.25G with PRBS7 patterns:

Example:

```
/*
  TX: PRBS 7 1.25G
  RX: PRBS 7 1.25G
*/

RTK.0> register set 0x041a30 0x803  /*enable PRBS*/
RTK.0> register set 0x041a2C 0x0000 /*PRBS selection*, *PRBS reverse*/
```


8.2. Verification

After testing, you can read the counter for checking result:

Example:

```
/*
  PRBS Status and Error Counter
*/
RTK.0> register get 0x041a14      /*Get PRBS error counter [31:16]*/
Register 0x40a14 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
RTK.0> register get 0x041a10      /*Get PRBS error counter [15:00]*/
Register 0x40a10 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
```

8.3. Clear Counters

If you want to retest, you can clear counter with below command:

Example:

```
/*
  PRBS Error Counter Clear
*/

RTK.0> register set 0x041a14 0x0000 /*clear PRBS error counter [31:16]*/
RTK.0> register set 0x041a10 0x0000 /*clear PRBS error counter [15:00]*/
```

9. HSG1/SG1/FIB1G Mode Example

Since PRBS on RTL9607C is a debug and verification tool only, we only provide the example with RTK shell. The shell procedure including: initialization, configuration, verification, clear counters and SerDes loopback.

9.1. Configuration

Below show an example of PRBS configuration for TX: 1.25G, RX: 1.25G with PRBS7 patterns:

Example:

```
/*
  TX: PRBS 7 1.25G
  RX: PRBS 7 1.25G
*/

RTK.0> register set 0x042a30 0x803  /*enable PRBS*/
RTK.0> register set 0x042a2C 0x0000 /*PRBS selection*, *PRBS reverse*/
```


9.2. Verification

After testing, you can read the counter for checking result:

Example:

```
/*
  PRBS Status and Error Counter
*/
RTK.0> register get 0x042a14      /*Get PRBS error counter [31:16]*/
Register 0x40a14 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
RTK.0> register get 0x042a10      /*Get PRBS error counter [15:00]*/
Register 0x40a10 : 0x0000ffff    /*Error occurred, Error counter 0xffff*/
```

9.3. Clear Counters

If you want to retest, you can clear counter with below command:

Example:

```
/*
  PRBS Error Counter Clear
*/

RTK.0> register set 0x042a14 0x0000 /*clear PRBS error counter [31:16]*/
RTK.0> register set 0x042a10 0x0000 /*clear PRBS error counter [15:00]*/
```

全文阅读已结束，下载本文需要使用

200 积分

下载此文档

阅读了该文档的用户还阅读了这些文档

RTL9607C_LED_Apr

RTL9607C_RTK_Por

RTL9607C_L2_Table

RTL9607C_Storm_Fi

EngNote_RTL9607C
note_V01(2019111

Application_Nc

发表评论

验证码： 换一张

匿名评论

提交