AN12772

Emulating UART with the FlexIO on RT1010

Rev. 0 — March 10 2020

Application Note

Contents

1 FlexIO introduction

This application note illustrates how to use FlexIO module to emulate UART based on RT1010. Although RT1010 has four LPUART, FLEXIO simulation is also an option when the number of LPUART is not enough.

FlexIO is an on-chip peripheral available on NXP I.MXRT series. It is a highly configurable module which is capable of emulating a wide range of communication protocols, such as UART, I^{2}C,SPI, I^{2}S, and so on. Users can also use FlexIO to generate PWM and PFM waveform.

2 UART based on FLEXIO

2.1 UART overview

A universal asynchronous receiver/transmitter is a piece of computer hardware that translates data between parallel and serial forms. UARTs are commonly used in conjunction with communication standards such as EIA, RS-232, RS-422 or RS-485. The universal designation indicates that the data format and transmission speeds are configurable. The electric signaling levels and methods (such as differential signaling etc.) are handled by a driver circuit external to the UART. A UART is usually an individual (or part of an) integrated circuit used for serial communications over a computer or peripheral device serial port.

Transmitting and receiving UARTs must be set for the same bit speed, character length, parity and stop bits for proper operation. The receiving UART may detect some mismatched settings and set a **framing error** flag bit for the host system. Figure 1 shows a typical UART 8-bit data frame.

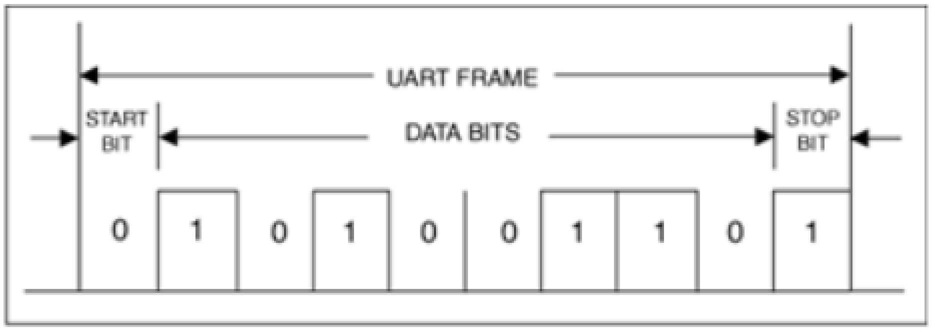


Figure 1. UART 8-bit data frame

2.2 UART with FlexIO

When FlexIO is used to simulate the UART, pay attention to the advantages and disadvantages in order to allocate resources reasonably.

Advantages:

- Increase the number of UARTs.

- High baud rate.
- Flexible configuration, can configure the number of bits of data in a frame.

Disadvantages:

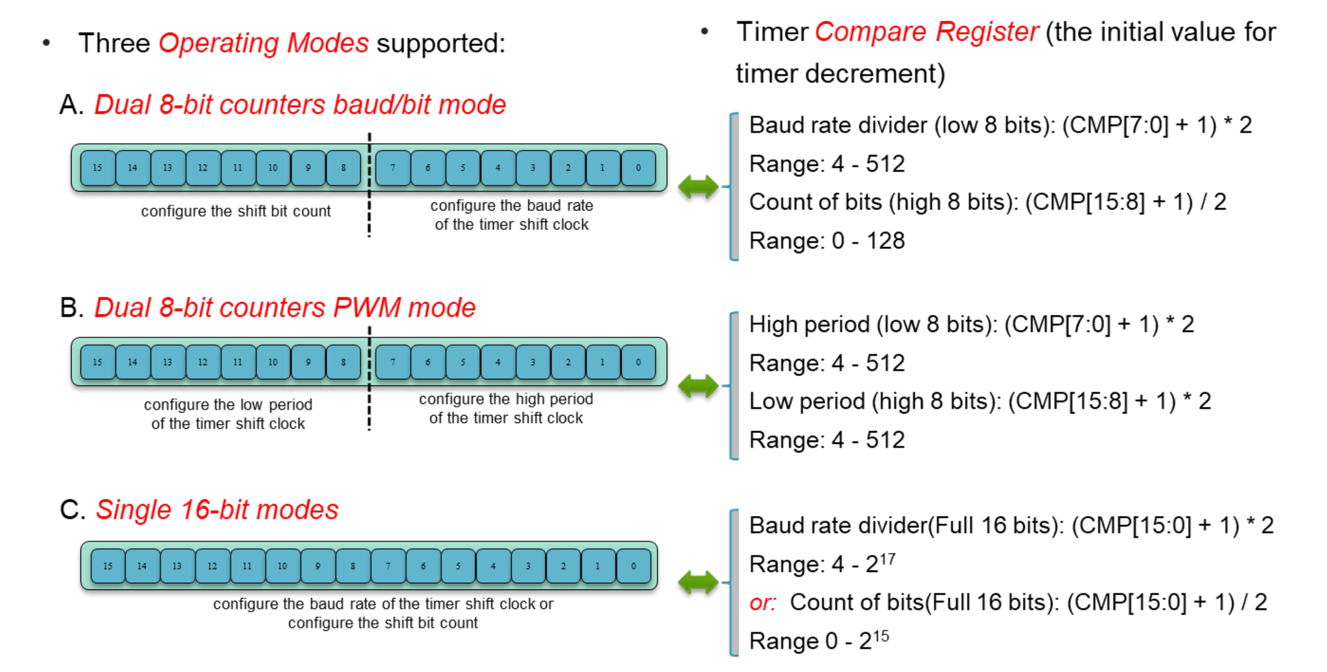
- Does not support parity.
- Does not support FIFO, recommended for high-speed processing with DMA.

Composition of FLEXIO Shifter and Timer. Shift Registers with transmit, receive and data match modes. For UART case, shifter works under Transmit or Received mode. Besides this, shifter supports Start/Stop bit generation. Timer supports variety of internal/external trigger and enable/disable/reset/decrement conditions.

Timer has three operating modes:

- Dual 8-bit baud mode
- Dual 8-bit PWM mode
- Single 16 bit mode

For the UART case, the timer works in the Dual 8 bit baud mode.



- Three *Operating Modes* supported:

A. *Dual 8-bit counters baud/bit mode*

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
configure the shift bit count
configure the baud rate of the timer shift clock

B. *Dual 8-bit counters PWM mode*

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
configure the low period of the timer shift clock
configure the high period of the timer shift clock

C. *Single 16-bit modes*

15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0
configure the baud rate of the timer shift clock or configure the shift bit count

- Timer *Compare Register* (the initial value for timer decrement)

Baud rate divider (low 8 bits): (CMP[7:0] + 1) * 2
Range: 4 - 512
Count of bits (high 8 bits): (CMP[15:8] + 1) / 2
Range: 0 - 128

High period (low 8 bits): (CMP[7:0] + 1) * 2
Range: 4 - 512
Low period (high 8 bits): (CMP[15:8] + 1) * 2
Range: 4 - 512

Baud rate divider(Full 16 bits): (CMP[15:0] + 1) * 2
Range: 4 - 2^{17}
or: Count of bits(Full 16 bits): (CMP[15:0] + 1) / 2
Range 0 - 2^{15}

Figure 2. Timer operating modes

Figure 3 shows a simple UART data transmitting and receiving process.

The process for the transmitting process includes:

1. The data are stored in the SHIFTBUFi.
2. The data are loaded to the SHIFTERi.
3. The data are sent to output pin triggered by timer and clock.

The process for the receiving process include:

1. The input data get from FLEXIO input pin.
2. The input data trigger the timer and are stored to the SHIFTERi.
3. The inpit data are output to the SHIFTBUFi.

For more detailed steps, please find the attachment file of this Application Note.

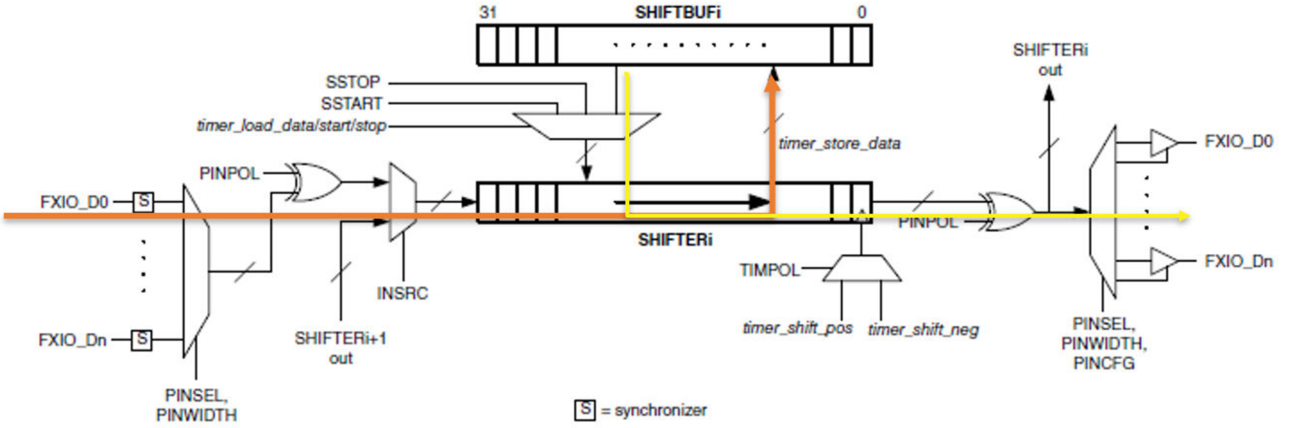


Figure 3. A simple UART data transmitting and receiving process

2.3 FlexIO UART transmitter

When FlexIO is configured in Transmitter mode, the following resources are needed:

- 1 Timer: Configured as Dual 8-bit baud counter mode to control the data shift.
- 1 Shifter: Controlled by Timer to shift data from SHIFTBUF.
- 1 Pin: Connected to the Shifter to output data.

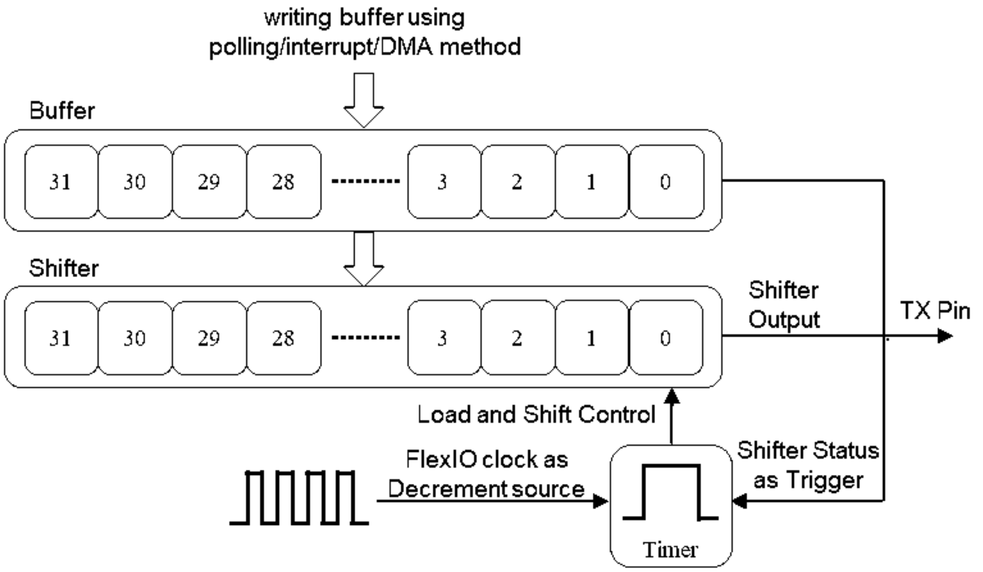


Figure 4. FlexIO UART transmitter

Table 1. Shifter configurations of FlexIO UART transmitter

Shifter configurations	
SMOD	Transmit

Table continues on the next page...

Table 1. Shifter configurations of FlexIO UART transmitter (continued)

Shifter configurations	
TIM POL	On Posedge
PIN SEL	TX Pin
PIN CFG	Output
PIN POL	Active High
IN SRC	Form Pin
SSTART	Low
SSTOP	High

Table 2. Timer configurations of FlexIO UART Transmitter

Timer configurations	
TIMOD	Dual 8-bit baud/bit
TRG SEL	TX Shifter Status
TRG POL	Active Low
TRG SRC	Internal
PIN SEL	TX pin
PIN CFG	Output Disabled
PIN POL	Active High
TIM OUTPUT	1 Not Affect by RST
TIM DEC	DEC on FlexIO clock, Shift on timer output
TIM ENA	On Trigger High
TIM DIS	On Timer Compare
TIM RST	Never
TSTART	Enabled
TSTOP	On Timer Compare

2.4 FlexIO UART receiver

When FlexIO is configured in Transmitter mode, the following resources are needed:

- 1 Timer: Configured as Dual 8-bit baud counter mode to control the data shift.
- 1 Shifter: Controlled by Timer to shift data into SHIFTBUF.
- 1 Pin: Connected to the Shifter to input data.

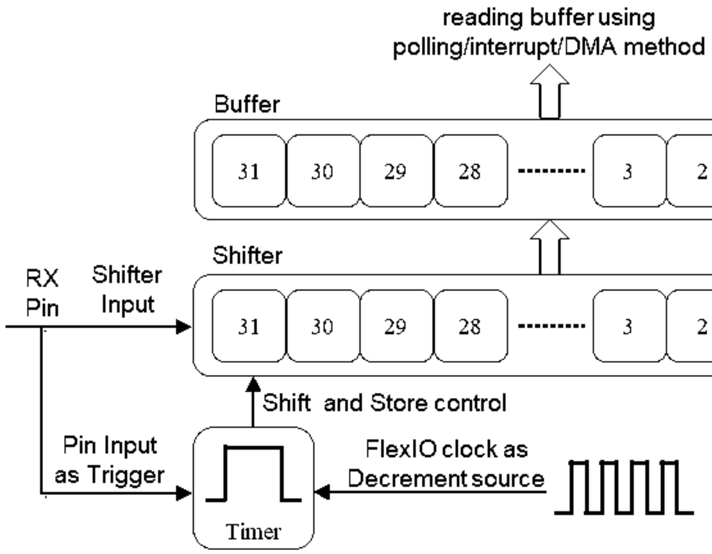


Figure 5. FlexIO UART receiver

Table 3. Shifter configurations of FlexIO UART receiver

Shifter configurations	
SMOD	Receive
TIM POL	On Negedge
PIN SEL	RX Pin
PIN CFG	Output Disable
PIN POL	Active High
IN SRC	Form Pin
SSTART	Low
SSTOP	High

Table 4. Timer configurations of FlexIO UART Transmitter

Timer configurations	
TIMOD	Dual 8-bit baud/bit
TRG SEL	RX Pin
TRG POL	Active High
TRG SRC	External
PIN SEL	RX pin
PIN CFG	Output Disabled
PIN POL	Active Low
TIM OUTPUT	1 Not Affect by RST

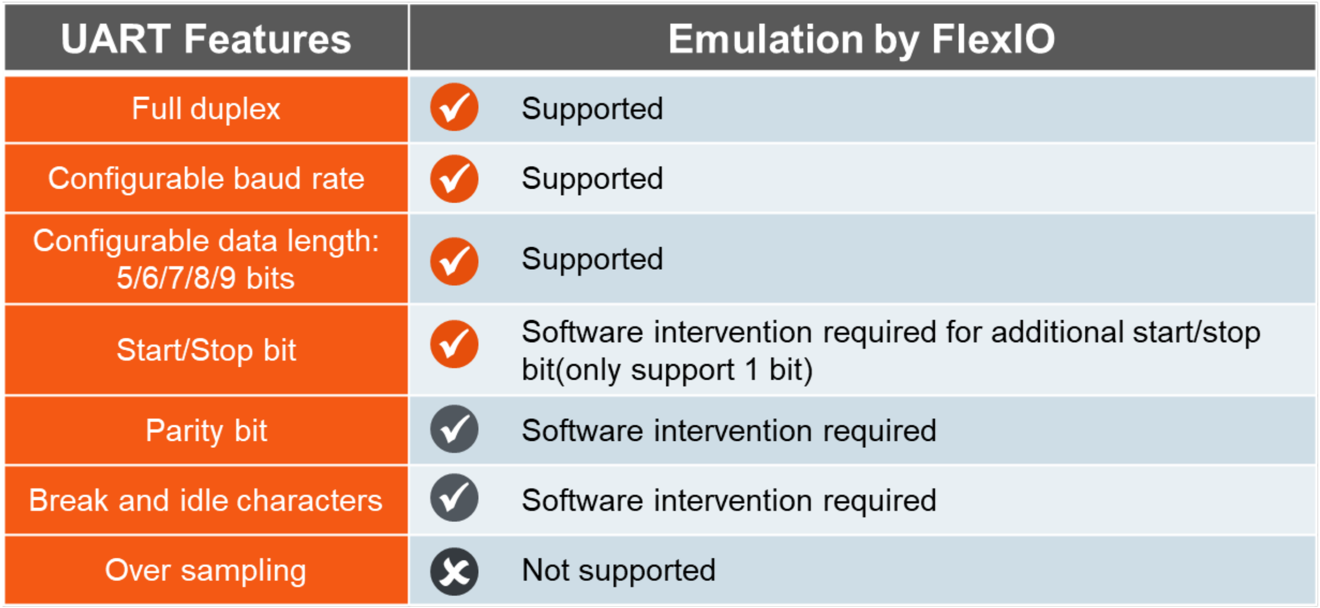
Table continues on the next page...

Table 4. Timer configurations of FlexIO UART Transmitter (continued)

Timer configurations	
TIM DEC	DEC on FlexIO clock, Shift on timer output
TIM ENA	On Pin Rising Edge
TIM DIS	On Timer Compare
TIM RST	Never
TSTART	Enabled
TSTOP	On Timer Disabled

3 Summary

Figure 6 compares the functions of UART and FLEXIO emulated UART. Almost all functions can be implemented except for oversampling.



UART Features	Emulation by FlexIO
Full duplex	✓ Supported
Configurable baud rate	✓ Supported
Configurable data length: 5/6/7/8/9 bits	✓ Supported
Start/Stop bit	✓ Software intervention required for additional start/stop bit(only support 1 bit)
Parity bit	✓ Software intervention required
Break and idle characters	✓ Software intervention required
Over sampling	✗ Not supported

Figure 6. Functions of UART and FLEXIO emulated UART

There are some demo examples in RT1010 SDK. The path is:

```
boards\evkmimxrt1010\driver_examples\flexio\uart
```

The examples include: `edma_transfer`, `int_rb_transfer`, `interrupt_transfer`, and `polling_transfer`.

Based on the FlexIO UART function, users can further develop IRDA functions based on FlexIO. For details, refer to *Emulating IRDA with the FlexIO on RT1010* (document AN12721).

How To Reach Us

Home Page:

nxp.com

Web Support:

nxp.com/support

Information in this document is provided solely to enable system and software implementers to use NXP products. There are no express or implied copyright licenses granted hereunder to design or fabricate any integrated circuits based on the information in this document. NXP reserves the right to make changes without further notice to any products herein.

NXP makes no warranty, representation, or guarantee regarding the suitability of its products for any particular purpose, nor does NXP assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages. “Typical” parameters that may be provided in NXP data sheets and/or specifications can and do vary in different applications, and actual performance may vary over time. All operating parameters, including “typicals,” must be validated for each customer application by customer's technical experts. NXP does not convey any license under its patent rights nor the rights of others. NXP sells products pursuant to standard terms and conditions of sale, which can be found at the following address: nxp.com/SalesTermsandConditions.

While NXP has implemented advanced security features, all products may be subject to unidentified vulnerabilities. Customers are responsible for the design and operation of their applications and products to reduce the effect of these vulnerabilities on customer’s applications and products, and NXP accepts no liability for any vulnerability that is discovered. Customers should implement appropriate design and operating safeguards to minimize the risks associated with their applications and products.

NXP, the NXP logo, NXP SECURE CONNECTIONS FOR A SMARTER WORLD, COOLFLUX, EMBRACE, GREENCHIP, HITAG, I2C BUS, ICODE, JCOP, LIFE VIBES, MIFARE, MIFARE CLASSIC, MIFARE DESFire, MIFARE PLUS, MIFARE FLEX, MANTIS, MIFARE ULTRALIGHT, MIFARE4MOBILE, MIGLO, NTAG, ROADLINK, SMARTLX, SMARTMX, STARPLUG, TOPFET, TRENCHMOS, UCODE, Freescale, the Freescale logo, AltiVec, C-5, CodeTEST, CodeWarrior, ColdFire, ColdFire+, C-Ware, the Energy Efficient Solutions logo, Kinetis, Layerscape, MagniV, mobileGT, PEG, PowerQUICC, Processor Expert, QorIQ, QorIQ Qonverge, Ready Play, SafeAssure, the SafeAssure logo, StarCore, Symphony, VortiQa, Vybrid, Airfast, BeeKit, BeeStack, CoreNet, Flexis, MXC, Platform in a Package, QUICC Engine, SMARTMOS, Tower, TurboLink, UMEMS, EdgeScale, EdgeLock, eIQ, and Immersive3D are trademarks of NXP B.V. All other product or service names are the property of their respective owners. AMBA, Arm, Arm7, Arm7TDMI, Arm9, Arm11, Artisan, big.LITTLE, Cordio, CoreLink, CoreSight, Cortex, DesignStart, DynamIQ, Jazelle, Keil, Mali, Mbed, Mbed Enabled, NEON, POP, RealView, SecurCore, Socrates, Thumb, TrustZone, ULINK, ULINK2, ULINK-ME, ULINK-PLUS, ULINKpro, μVision, Versatile are trademarks or registered trademarks of Arm Limited (or its subsidiaries) in the US and/or elsewhere. The related technology may be protected by any or all of patents, copyrights, designs and trade secrets. All rights reserved. Oracle and Java are registered trademarks of Oracle and/or its affiliates. The Power Architecture and Power.org word marks and the Power and Power.org logos and related marks are trademarks and service marks licensed by Power.org.

For more information, please visit: http://www.nxp.com
For sales office addresses, please send an email to: salesaddresses@nxp.com

Date of release: March 10 2020
Document identifier: AN12772