PY32 microcontroller system memory boot mode

Introduction

The bootloader is stored in the internal boot ROM (system memory) of PY32 devices and is programmed by Puya during production. Its main task is to download the application program to the internal flash memory through one of the available serial peripherals (such as USART, CAN, USB, I2C, etc.). A communication protocol is defined for each serial interface, with a compatible command set and sequence. This document applies to the products listed in Table 1, referred to as PY32 throughout the document. It describes the supported peripherals and hardware requirements to consider when using the bootloader of PY32 devices.

Table 1.Applicable products

Type	Product Series
Microcontrollers	- PY32C611, PY32C610, PY32C670 - PY32F002A, PY32F003, PY32F030, PY32F031, PY32F040, PY32F071, PY32F072 - PY32F303, PY32F403 - PY32M030, PY32M070

The main functions of the bootloader are as follows:
● Using the embedded serial interface to download code according to a predefined communication protocol
● Can transfer and update Flash code, data, and vector table sections

This application note introduces the general concept of bootloader. It describes the peripherals that are supported and the hardware requirements that need to be considered when using the bootloader with the PY32 device listed in Table *1.* However, the low-level communication protocol specifications for each of the supported serial peripherals are described in a separate document. See AN3155 for the specification of the USART protocol used in the bootloader, AN3156 for the specification of the DFU (USB device) protocol used in the bootloader, and AN4221 for the specification of the I2C protocol used in the bootloader.

Contents

1   Related Documents

For information about each of the supported products (listed in Table 1), see the following documents available at https://www.puyasemi.com/:

● Datasheets or product briefs

● Reference manuals

2   General Information

2.1   Bootloader activation

The bootloader is automatically activated by configuring the BOOT0 and BOOT1 pins in a specific System Memory configuration (see Table 2.1-1) and then performing a reset. Depending on the pin configuration used, Flash, system memory or SRAM can be selected as bootloader space as shown in Table 2.1-1 below.

In some products, BOOT1 does not represent an I/O pin, but is a bit in the option byte area. This is the case for the PY32C6xx, PY32F0xx, and PY32M0x devices, where BOOT1 is configured via the nBoot1 bit in the option byte.

- When nBoot1 position 1, it is equivalent to BOOT1 reset to 0 in Table 2.1-1.

- When the nBoot1 bit is reset to 0, it is equivalent to BOOT1 set to 1 in Table 2.1-1.

Table 2.1-1. Bootloader Pin Configuration

Bootloader mode selection pin		bootloader model	Use of aliases
BOOT1	BOOT0		
X	0	User Flash	Selecting User Flash as Bootloader Space
0	1	system memory	Select system memory as bootloader space
1	1	Embedded SRAM	Selection of embedded SRAM as bootloader space

Table 2.1-1 shows that the PY32 microcontroller enters the system memory bootloader mode when the BOOT pin is configured as follows:

- BOOT0 = 1

- BOOT1 = 0

reset latches the value of the BOOT pin on the fourth rising edge of SYSCLK.

  NOTE: In some products, when the PY32 product provides the dual-memory bootloader feature (BOOT0 = 0 and BOOT1 = x), the bootloader can be entered.

  bootloader can be entered. See the Dual Memory Bootloader Feature section of the product section for more information.

  When the operating temperature is outside the ambient temperature range, the internal clock (HSI) varies with the temperature, causing the serial communication protocol clock to become corrupted, which may result in bootloader.

  clock is corrupted as the internal clock (HSI) changes with temperature, which may prevent the bootloader from functioning properly.

2.2   Exit system memory bootloader mode

In order to execute the application program, the system memory bootloader mode must first be exited. This is accomplished by performing a hardware reset. During reset, the BOOT pins/bits (BOOT0 and BOOT1) must be set to the appropriate level to select the desired bootloader mode (see Table 2.1-1). After reset, the CPU will start code execution from the bottom of the memory address space of the bootloader memory (start address 0x0000 0000).

2.3   Bootloader identification

Depending on the PY32 device used, the bootloader can support one or more

embedded serial peripherals used to download code into the internal Flash. The bootloader identifier (ID) provides information about the supported serial peripherals. For a given PY32 device, the bootloader is identified by each of the following:

1. **Bootloader (Protocol) Version:** The version of the serial peripheral (USART, CAN, USB, etc.) communication protocol used in the bootloader. This version can be retrieved using the bootloader Get Version command.

2. Bootloader Identifier (ID): the version of the PY32 device bootloader, represented as a single-byte code of the form **0xXY**, where:
    - **X** specifies the embedded serial peripheral used by the device bootloader:
    X = 1: Use a USART
    X = 2: use two USARTs
    X = 3: Use of USART, CAN and DFU
    X = 4: Using USART and DFU
    X = 5: Use of USART and I2C
    X = 6: Use I2C
    X = 7: Using USART, CAN, DFU and I2C
    X = 8: using I2C and SPI
    X = 9: Using USART, CAN (or FDCAN), DFU, I2C and SPI
    X = 10: Using USART, DFU and I2C
    X = 11: Using USART, I2C and SPI
    X = 12: using USART and SPI
    X = 13: Using USART, DFU, I2C and SPI
    - **Y** Specifies the bootloader version of the device

    The following is an example of bootloader ID 0x10. This represents the first version of a device bootloader that uses only one USART.

    The bootloader ID is programmed in the space corresponding to the last byte address minus 1 of the device's system memory, and can be read by the bootloader "Read memory" command, or by directly accessing the system memory using JTAG/SWD.

The following table provides identification information about the embedded bootloader for the PY32 device.

Table 2.3-1. Embedded Bootloaders

PY32 range	component	Supported Serial Peripherals	Bootloader ID		bootloader (version
			ID	memory unit	
C	PY32C61xxx	USART1	V1.0	0x1FFF0000	USART(V1.0)
	PY32C67xxx	USART1	V1.0	0x1FFF0000	USART(V1.0)
F0	PY32F002Axx / PY32F003xx/ PY32F030xx	USART1	V1.0	0x1FFF0000	USART(V1.0)

	PY32F040xx/ PY32F071xx/ PY32F072xx	USART1/USART2/ USART3/USART4/ I2C1/DFU (USB Device FS)	V1.0	0x1FFF0000	USART(V1.0)/ I2C(V1.0)/ DFU (V1.0)
F3	PY32F303xx	USART1/USART2/ USART3/USART4/ I2C1/DFU (USB Device FS)	V1.0	0x1FFF0000	USART(V1.0)/ I2C(V1.0)/ DFU (V1.0)
F4	PY32F403xx	USART1/USART2/ USART3/USART4/ I2C1/DFU (USB Device FS)	V1.0	0x1FFF0000	USART(V1.0)/ I2C(V1.0)/ DFU (V1.0)
M0	PY32M030xx	USART1	V1.0	0x1FFF0000	USART(V1.0)
	PY32M070xx	USART1/USART2/ USART3/USART4/ I2C1/DFU (USB Device FS)	V1.0	0x1FFF0000	USART(V1.0)/ I2C(V1.0)/ DFU (V1.0)

3   Device-dependent bootloader parameters

For all PY32 devices, the bootloader protocol command set and sequence for each serial peripheral (USART, CAN, USB, and I2C) is the same.

 The bootloader protocol command set and sequence are the same for each serial peripheral (USART, CAN, USB and I2C). However, some parameters are device-specific. For some commands, some parameter values may depend on the device being used.

 device used. These parameters are listed below:

● PID (Product ID), which varies from device to device

● Valid memory addresses supported by the bootloader when the Read Memory, Go, and Write Memory commands are supported.

（● PID (product ID), which varies from device to device

● The size of the Flash sector used when executing the Write Protect command.

The following table shows the values of the above parameters for each PY32 device bootloader in production.

Table 3.3-1. Device-Related Bootloader Parameters

PY32 range	component	offerings (device) ID	SRAM memory (unit)	Flash	Flash Sector Size	Option Byte Area	system memory
C	PY32C61xxx	0x0440	0x20000200-0x20001FFF	0x08000000-0x0800FFFF	4KB (32 pages, 2.5MB) (128B per page)	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F
	PY32C67xxx	0x0444	0x20000200-0x20001FFF	0x08000000-0x0800FFFF	4KB (32 pages, 2.5MB) (128B per page)	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F
F0	PY32F002Axx	0x0440	0x20000200-0x20000BFF	0x08000000-0x08004FFF	4KB (32 pages, 2.5MB) (128B	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F

				per page)		
PY32F003xx	0x0440	0x20000200-0x20001FFF	0x08000000-0x0800FFFF	4KB (32 pages, 2.5MB) (128B per page)	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F
PY32F030xx	0x0440	0x20000200-0x20001FFF	0x08000000-0x0800FFFF	4KB (32 pages, 2.5MB) (128B per page)	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F
PY32F040xx	0x0448	0x20000800-0x20003FFF	0x08000000-0x0801FFFF	8KB (32 pages, 2.5 pages, 1.5 pages, 2.5 pages) (256B per page)	0x1FFF3100-0x1FFF311F	0x1FFF0000-0x1FFF2EFF
PY32F071xx	0x0448	0x20000800-0x20003FFF	0x08000000-0x0801FFFF	8KB (32 pages, 2.5 pages, 1.5 pages, 2.5 pages) (256B per page)	0x1FFF3100-0x1FFF311F	0x1FFF0000-0x1FFF2EFF
PY32F072xx	0x0448	0x20000800-	0x08000000-	8KB (32	0x1FFF3100-	0x1FFF0000-

			0x20003FFF	0x0801FFFF	pages, 2.5 pages, 1.5 pages, 2.5 pages) (256B per page)	0x1FFF311F	0x1FFF2EFF
F3	PY32F303xx	0x0413	0x20000800-0x20007FFF	0x08000000-0x0803FFFF	2KB (8 pages, 2KB) (256B per page)	0x1FFF5000-0x1FFF5000F	0x1FFF0000-0x1FFF4FFF
F4	PY32F403xx	0x0413	0x20000800-0x2000FFFF	0x08000000-0x0805FFFF	2KB (8 pages, 2KB) (256B per page)	0x1FFF5000-0x1FFF5000F	0x1FFF0000-0x1FFF4FFF
M0	PY32M030xx	0x0440	0x20000200-0x20001FFF	0x08000000-0x0800FFFF	4KB (32 pages, 2.5MB) (128B per page)	0x1FFF0E80-0x1FFF0E8F	0x1FFF0000-0x1FFF0D7F
	PY32M070xx	0x0448	0x20000800-0x20003FFF	0x08000000-0x0801FFFF	8KB (32 pages, 2.5 pages, 1.5 pages, 2.5 pages) (256B per page)	0x1FFF3100-0x1FFF311F	0x1FFF0000-0x1FFF2EFF

4   PY32C61x, PY32F002A, PY32F003, PY32F030, PY32M030 device bootloader

4.1   Bootloader configuration

The PY32C61x, PY32F002A, PY32F003, PY32F030, and PY32M030 device embedded bootloader support only the USART1 interface.

The following table describes the hardware resources that the bootloader needs to use in the system memory bootloader mode.

Table 4.1-1. PY32 Device Configuration in System Memory Bootloader Mode

Bootloader	Functions/Peripherals	state of affairs	note
USART1 bootloader	clock source	HSI Enable	Uses HSI as the system clock at 24MHz
	RAM	-	A space of 512 bytes from address 0x20000000 is available for the bootloader.
	system memory	-	The 3.25KB space from address 0x1FFF0000 contains the bootloader firmware.
	IWDG	-	The Independent Watchdog (IWDG) prescaler is configured to a maximum value and is periodically refreshed to prevent the watchdog from resetting (if the user enables the hardware IWDG option beforehand).
	USART1	enable	After initialization, USART1 is configured with:8 bits, even parity bits and 1 stop bit.
	USART1 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
Group 1 USART1	USART1_RX pin	importation	PA3 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA2 pin: USART1 is used for transmitting.
	The USART1_RX(PA10), USART1_RX(PA15) pins must be held high or low during the detection phase.		
Group 2 USART1	USART1_RX pin	importation	PA10 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA9 pin: USART1 is used for transmitting.
	The USART1_RX(PA3), USART1_RX(PA15) pins must be held high or low during the detection phase.		
Group 3 USART1	USART1_RX pin	importation	PA15 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA14 pin: USART1 is used for transmitting.
	The USART1_RX(PA3), USART1_RX(PA10) pins must be held high or low during the		

	detection phase.

The system clock is provided by the embedded internal high-speed RC, and the bootloader code does not require an external quartz clock.

After downloading the application binary code, if you choose to execute the Go command, the peripheral registers used by the bootloader (shown in the table above) are not initialized to their default reset values before jumping to the user application. If these registers are to be used, they should be reconfigured in the user application. Therefore, if the application is using IWDG, the IWDG prescaler value must be adjusted to meet the requirements of the application (since the bootloader has set the prescaler to its maximum value).

4.2   Hardware connection requirements

The hardware required to place the PY32 in system memory bootloader mode consists of any associated circuitry, switches, or jumpers that hold the BOOT0 pin high and the BOOT1 pin low during a reset.

To connect the PY32 in system memory bootloader mode, the RS232 serial interface must be connected directly to the USART1_RX (PA10) and USART1_TX (PA9) pins.

Note: The USART1_CK, USART1_CTS, and USART1_RTS pins are not used, so the user can use these pins for other peripherals or GPIOs.

4.3   Bootloader selection

Immediately after entering the system memory bootloader mode and the microcontroller has been configured as described above, the bootloader code begins scanning the USART1_RX pin and waits to receive the 0x7F data frame: a start bit, the 0x7F data bit, an even parity bit, and a stop bit.

The start bit of this data frame is used to determine which set of USART1s to use, and the data bits of this data frame are used for automatic baud rate detection on the serial port.

The code will then initialize the serial interface accordingly. With the calculated baud rate, an acknowledgement byte (0x79) is returned to the host indicating that the PY32 is ready to receive user commands.

4.4   Bootloader version

The following table lists the bootloader versions for the PY32C61x, PY32F002A, PY32F003, PY32F030, and PY32M030 devices.

Table 4.4-1. Bootloader Versions

Version number	clarification
V1.0	Initial bootloader version.

5 PY32C67x, PY32F031 device bootloader

5.1 Bootloader configuration

The PY32C67x and PY32F031 device embedded bootloader supports only one interface, USART1.

The following table describes the hardware resources that the bootloader needs to use in the system memory bootloader mode.

Table 5.1-1. PY32 Device Configuration in System Memory Bootloader Mode

Bootloader	Functions/Peripherals	state of affairs	note
USART1 bootloader	clock source	HSI Enable	Uses HSI as the system clock at 24MHz
	RAM	-	A space of 512 bytes from address 0x20000000 is available for the bootloader.
	system memory	-	The 3.25KB space from address 0x1FFF0000 contains bootloader firmware.
	IWDG	-	The Independent Watchdog (IWDG) prescaler is configured to a maximum value and is periodically refreshed to prevent the watchdog from resetting (if the user enables the hardware IWDG option beforehand).
	USART1	enable	After initialization, USART1 is configured with:8 bits, even parity bits and 1 stop bit.
	USART1 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
Group 1 USART1	USART1_RX pin	importation	PA3 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA2 pin: USART1 is used for transmitting.
	The USART1_RX(PA10), USART1_RX(PA15) pins must be held high or low during the detection phase.		
Group 2 USART1	USART1_RX pin	importation	PA10 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA9 pin: USART1 is used for transmitting.
	The USART1_RX(PA3), USART1_RX(PA15) pins must be held high or low during the detection phase.		
Group 3 USART1	USART1_RX pin	importation	PA15 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA14 pin: USART1 is used for transmitting.
	The USART1_RX(PA3), USART1_RX(PA10) pins must be held high or low during the detection phase.		

The system clock is provided by the embedded internal high-speed RC, and the bootloader code does not require an external quartz clock.

After downloading the application binary code, if you choose to execute the Go command, the peripheral registers used by the bootloader (shown in the table above) are not initialized to their default reset values before jumping to the user application. If these registers are to be used, they should be reconfigured in the user application. Therefore, if the application is using IWDG, the IWDG prescaler value must be adjusted to meet the requirements of the application (since the bootloader has set the prescaler to its maximum value).

5.2   Hardware connection requirements

The hardware required to place the PY32 in system memory bootloader mode consists of any associated circuitry, switches, or jumpers that hold the BOOT0 pin high and the BOOT1 pin low during a reset.

To connect the PY32 in system memory bootloader mode, the RS232 serial interface must be connected directly to the USART1_RX (PA10) and USART1_TX (PA9) pins.

Note: The USART1_CK, USART1_CTS, and USART1_RTS pins are not used, so the user can use these pins for other peripherals or GPIOs.

5.3   Bootloader selection

Immediately after entering the system memory bootloader mode and the microcontroller has been configured as described above, the bootloader code begins scanning the USART1_RX pin and waits to receive the 0x7F data frame: a start bit, the 0x7F data bit, an even parity bit, and a stop bit.

The start bit of this data frame is used to determine which set of USART1s to use, and the data bits of this data frame are used for automatic baud rate detection on the serial port.

The code will then initialize the serial interface accordingly. With the calculated baud rate, an acknowledgement byte (0x79) is returned to the host indicating that the PY32 is ready to receive user commands.

5.4   Bootloader version

The following table lists the bootloader versions for the PY32C67x, PY32F031 devices.

Table 5.4-1. Bootloader Versions

Version number	clarification
V1.0	Initial bootloader version.

6   PY32F040 device bootloader

6.1   Bootloader configuration

The PY32F040 device embedded bootloader supports three serial peripherals: USART1, USART2, USART3, USART4, and I2C.

The following table describes the hardware resources that need to be used by the bootloader for the PY32F040 device in system memory bootloader mode.

Table 6.1-1. PY32 Device Configuration in System Memory Bootloader Mode

bootloader	Functions/Peripherals	state of affairs	note
Common to all bootloaders	clock source	HSI Enable	Uses PLL as system clock at 48MHz
	RAM	-	The 2048 bytes of space from address 0x20000000 are available for use by the bootloader.
	system memory	-	The 12KB space from address 0x1FFF0000 contains the bootloader firmware.
	IWDG	-	The Independent Watchdog (IWDG) prescaler is configured to a maximum value and is periodically refreshed to prevent the watchdog from resetting (if the user enables the hardware IWDG option beforehand).
USART1 bootloader	USART1	enable	After initialization, USART1 is configured with:8 bits, even parity bits and 1 stop bit.
	USART1_RX pin	importation	PA10 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA9 pin: USART1 is used for transmitting.
	USART1 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART2_RX(PA15), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART2 bootloader	USART2	enable	After initialization, USART2 is configured with:8 bits, even parity bits and 1 stop bit.
	USART2_RX pin	importation	PA15 pin: USART2 is used for reception.
	USART2_TX pin	push-pull output	Pin PA14: USART2 is used for transmitting.
	USART2 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.

	The USART1_RX(PA10), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART3 bootloader	USART3	enable	After initialization, USART3 is configured with:8 bits, even parity bits and 1 stop bit.
	USART3_RX pin	importation	PB11 pin: USART3 is used for reception.
	USART3_TX pin	push-pull output	PB10 pin: USART3 is used for transmitting.
	USART3 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PA15), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART4 bootloader	USART4	enable	After initialization, USART4 is configured with:8 bits, even parity bits and 1 stop bit.
	USART4_RX pin	importation	PC11 pin: USART4 is used for reception.
	USART4_TX pin	push-pull output	PC10 pin: USART4 is used for transmitting.
	USART4 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PA15), USART3_RX(PB11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
I2C1 bootloader	I2C1	enable	The PLL is used as the I2C clock source. After initialization, I2C1 is configured with a 7-bit address, slave mode, slave address: 0x76.
	I2C1_SCL pin	Input/Output	Pin PB6: I2C1 clock line is used in open-drain mode.
	I2C1_SDA pin	Input/Output	Pin PB7: I2C1 clock line is used in open-drain mode.
	The USART1_RX(PA10), USART2_RX(PA15), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		

The system clock is provided by the embedded internal high-speed RC, and the bootloader code does not require an external quartz clock.

After downloading the application binary code, if you choose to execute the Go command, the peripheral registers used by the bootloader (shown in the table above) are not initialized to their default reset values before jumping to the user application. If these registers are to be used, they should be reconfigured in the user application. Therefore, if the application is using IWDG, the IWDG prescaler value must be adjusted to meet the requirements of the application (since the bootloader has set the prescaler to its maximum value).

6.2   Hardware connection requirements

The hardware required to place the PY32 in system memory bootloader mode consists of any associated circuitry, switches, or jumpers that hold the BOOT0 pin high and the BOOT1 pin low during a reset.

To connect the PY32 in system memory bootloader mode, the RS232 serial interface must be connected directly to the USART1_RX (PA10) and USART1_TX (PA9) pins.

Note: The USART1_CK, USART1_CTS, and USART1_RTS pins are not used, so the user can use these pins for other peripherals or GPIOs.

6.3   Bootloader selection

Immediately after entering the system memory bootloader mode and the microcontroller has been configured as described above, the bootloader code begins scanning the USART1_RX pin and waits to receive the 0x7F data frame: a start bit, the 0x7F data bit, an even parity bit, and a stop bit.

The start bit of this data frame is used to determine which set of USART1s to use, and the data bits of this data frame are used for automatic baud rate detection on the serial port.

The code will then initialize the serial interface accordingly. With the calculated baud rate, an acknowledgement byte (0x79) is returned to the host indicating that the PY32 is ready to receive user commands.

6.4   Bootloader version

The following table lists the bootloader versions for the PY32F040 device.

Table 6.4-1. Bootloader Versions

Version number	clarification
V1.0	Initial bootloader version.

7   PY32F07x, PY32M07x device bootloader

7.1   Bootloader configuration

The PY32F07x, PY32M07x device embedded bootloader supports four serial peripherals: USART1, USART2, USART3, USART4, I2C and DFU (USB).

The following table describes the hardware resources that need to be used by the bootloader for the PY32F07x and PY32M07x devices in the system memory bootloader mode.

Table 7.1-1. PY32 Device Configuration in System Memory Bootloader Mode

Bootloader	Functions/Peripherals	state of affairs	note
Common to all bootloaders	clock source	HSI Enable	Uses PLL as system clock at 48MHz
	RAM	-	The 2048 bytes of space from address 0x20000000 are available for use by the bootloader.
	system memory	-	The 12KB space from address 0x1FFF0000 contains the bootloader firmware.
	IWDG	-	The Independent Watchdog (IWDG) prescaler is configured to a maximum value and is periodically refreshed to prevent the watchdog from resetting (if the user enables the hardware IWDG option beforehand).
USART1 bootloader	USART1	enable	After initialization, USART1 is configured with:8 bits, even parity bits and 1 stop bit.
	USART1_RX pin	importation	PA10 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA9 pin: USART1 is used for transmitting.
	USART1 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART2_RX(PA15), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART2 bootloader	USART2	enable	After initialization, USART2 is configured with:8 bits, even parity bits and 1 stop bit.
	USART2_RX pin	importation	PA15 pin: USART2 is used for reception.
	USART2_TX pin	push-pull	Pin PA14: USART2 is used for

		output	transmitting.
	USART2 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART3 bootloader	USART3	enable	After initialization, USART3 is configured with:8 bits, even parity bits and 1 stop bit.
	USART3_RX pin	importation	PB11 pin: USART3 is used for reception.
	USART3_TX pin	push-pull output	PB10 pin: USART3 is used for transmitting.
	USART3 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PA15), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART4 bootloader	USART4	enable	After initialization, USART4 is configured with:8 bits, even parity bits and 1 stop bit.
	USART4_RX pin	importation	PC11 pin: USART4 is used for reception.
	USART4_TX pin	push-pull output	PC10 pin: USART4 is used for transmitting.
	USART4 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PA15), USART3_RX(PB11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
I2C1 bootloader	I2C1	enable	The PLL is used as the I2C clock source. After initialization, I2C1 is configured with a 7-bit address, slave mode, slave address: 0x76.
	I2C1_SCL pin	Input/Output	Pin PB6: I2C1 clock line is used in open-drain mode.
	I2C1_SDA pin	Input/Output	Pin PB7: I2C1 clock line is used in open-drain mode.
	The USART1_RX(PA10), USART2_RX(PA15), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
DFU bootloader	USB_FS	enable (sb. to do sth)	USBD
	USB_DM	reuse	PA11: USB send-receive data cable.

USB_DP	reuse	PA12: USB send-receive data cable.
disruptions	enable	Enable the USB interrupt vector for USB DFU communication.
The USART1_RX(PA10), USART2_RX(PA15), USART3_RX(PB11) , and USART4_RX(PC11) pins must be held high or low during the detection phase.		

The system clock is provided by the embedded internal high-speed RC, and the bootloader code does not require an external quartz clock.

After downloading the application binary code, if you choose to execute the Go command, the peripheral registers used by the bootloader (shown in the table above) are not initialized to their default reset values before jumping to the user application. If these registers are to be used, they should be reconfigured in the user application. Therefore, if the application is using IWDG, the IWDG prescaler value must be adjusted to meet the requirements of the application (since the bootloader has set the prescaler to its maximum value).

7.2   Hardware connection requirements

The hardware required to place the PY32 in system memory bootloader mode consists of any associated circuitry, switches, or jumpers that hold the BOOT0 pin high and the BOOT1 pin low during a reset.

To connect the PY32 in system memory bootloader mode, the RS232 serial interface must be connected directly to the USART1_RX (PA10) and USART1_TX (PA9) pins.

Note: The USART1_CK, USART1_CTS, and USART1_RTS pins are not used, so the user can use these pins for other peripherals or GPIOs.

7.3   Bootloader selection

Immediately after entering the system memory bootloader mode and the microcontroller has been configured as described above, the bootloader code begins scanning the USART1_RX pin and waits to receive the 0x7F data frame: a start bit, the 0x7F data bit, an even parity bit, and a stop bit.

The start bit of this data frame is used to determine which set of USART1s to use, and the data bits of this data frame are used for automatic baud rate detection on the serial port.

The code will then initialize the serial interface accordingly. With the calculated baud rate, an acknowledgement byte (0x79) is returned to the host indicating that the PY32 is ready to receive user commands.

7.4   Bootloader version

The following table lists the bootloader versions for the PY32F07x, PY32M07x devices.

Table 7.4-1. Bootloader Versions

Version number	clarification
V1.0	Initial bootloader version.

8   PY32F303, PY32F403 device bootloader

8.1   Bootloader configuration

The PY32F303, PY32F403 device embedded bootloader supports four serial peripherals: USART1, USART2, USART3, USART4, I2C and DFU (USB).

The following table describes the hardware resources that need to be used by the bootloader for the PY32F303 and PY32F403 devices in the system memory bootloader mode.

Table 8.1-1. PY32 Device Configuration in System Memory Bootloader Mode

bootloader	Functions/Peripherals	state of affairs	note
Common to all bootloaders	clock source	HSI Enable	Uses PLL as system clock at 48MHz
	RAM	-	The 2048 bytes of space from address 0x20000000 are available for use by the bootloader.
	system memory	-	The 20KB space from address 0x1FFF0000 contains the bootloader firmware.
	IWDG	-	The Independent Watchdog (IWDG) prescaler is configured to a maximum value and is periodically refreshed to prevent the watchdog from resetting (if the user enables the hardware IWDG option beforehand).
USART1 bootloader	USART1	enable	After initialization, USART1 is configured with:8 bits, even parity bits and 1 stop bit.
	USART1_RX pin	importation	PA10 pin: USART1 is used for reception.
	USART1_TX pin	push-pull output	PA9 pin: USART1 is used for transmitting.
	USART1 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART2_RX(PD6), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART2 bootloader	USART2	enable	After initialization, USART2 is configured with:8 bits, even parity bits and 1 stop bit.
	USART2_RX pin	importation	PD6 pin: USART2 is used for reception.
	USART2_TX pin	push-pull output	PD5 pin: USART2 is used for transmitting.
	USART2 automatic wave	enable	Used to automatically detect the baud

	rate detection		rate of the host's serial port.
	The USART1_RX(PA10), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART3 bootloader	USART3	enable	After initialization, USART3 is configured with:8 bits, even parity bits and 1 stop bit.
	USART3_RX pin	importation	PB11 pin: USART3 is used for reception.
	USART3_TX pin	push-pull output	PB10 pin: USART3 is used for transmitting.
	USART3 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PD6), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
USART4 bootloader	USART4	enable	After initialization, USART4 is configured with:8 bits, even parity bits and 1 stop bit.
	USART4_RX pin	importation	PC11 pin: USART4 is used for reception.
	USART4_TX pin	push-pull output	PC10 pin: USART4 is used for transmitting.
	USART4 automatic wave rate detection	enable	Used to automatically detect the baud rate of the host's serial port.
	The USART1_RX(PA10), USART2_RX(PD6), USART3_RX(PB11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
I2C1 bootloader	I2C1	enable	The PLL is used as the I2C clock source. After initialization, I2C1 is configured with a 7-bit address, slave mode, slave address: 0x7E.
	I2C1_SCL pin	Input/Output	Pin PB6: I2C1 clock line is used in open-drain mode.
	I2C1_SDA pin	Input/Output	Pin PB7: I2C1 clock line is used in open-drain mode.
	The USART1_RX(PA10), USART2_RX(PD6), USART3_RX(PB11), USART4_RX(PC11), USB_DM(PA11), and USB_DP(PA12) pins must be held either high or low during the detection phase.		
DFU bootloader	USB_FS	enable	USBD
	USB_DM	reuse	PA11: USB send-receive data cable.
	USB_DP	reuse	PA12: USB send-receive data cable.
	disruptions	enable	Enable the USB interrupt vector for USB DFU communication.
	The USART1_RX(PA10), USART2_RX(PD6), USART3_RX(PB11) , and USART4_RX(PC11) pins must be held high or low during the detection phase.		

The system clock is provided by the embedded internal high-speed RC, and the

bootloader code does not require an external quartz clock.

After downloading the application binary code, if you choose to execute the Go command, the peripheral registers used by the bootloader (shown in the table above) are not initialized to their default reset values before jumping to the user application. If these registers are to be used, they should be reconfigured in the user application. Therefore, if the application is using IWDG, the IWDG prescaler value must be adjusted to meet the requirements of the application (since the bootloader has set the prescaler to its maximum value).

8.2   Hardware connection requirements

The hardware required to place the PY32 in system memory bootloader mode consists of any associated circuitry, switches, or jumpers that hold the BOOT0 pin high and the BOOT1 pin low during a reset.

To connect the PY32 in system memory bootloader mode, the RS232 serial interface must be connected directly to the USART1_RX (PA10) and USART1_TX (PA9) pins.

Note: The USART1_CK, USART1_CTS, and USART1_RTS pins are not used, so the user can use these pins for other peripherals or GPIOs.

8.3   Bootloader selection

Immediately after entering the system memory bootloader mode and the microcontroller has been configured as described above, the bootloader code begins scanning the USART1_RX pin and waits to receive the 0x7F data frame: a start bit, the 0x7F data bit, an even parity bit, and a stop bit.

The start bit of this data frame is used to determine which set of USART1s to use, and the data bits of this data frame are used for automatic baud rate detection on the serial port.

The code will then initialize the serial interface accordingly. With the calculated baud rate, an acknowledgement byte (0x79) is returned to the host indicating that the PY32 is ready to receive user commands.

8.4   Bootloader version

The following table lists the bootloader versions for the PY32F303, PY32F403 devices.

Table 8.4-1. Bootloader Versions

Version number	clarification
V1.0	Initial bootloader version.

9 Version History

releases	dates	update records
V1.0	2023.7.7	first edition

Puya Semiconductor Co.

IMPORTANT NOTICE
Puya Semiconductor reserves the right to make changes without further notice to any products or specifications herein. Puya Semiconductor does not assume any responsibility for use of any its products for any particular purpose, nor does Puya Semiconductor assume any liability arising out of the application or use of any its products or circuits. Puya Semiconductor does not convey any license under its patent rights or other rights nor the rights of others.