

# **TPS32 Programmer User Guide**



Version: A0



Chapter 1 Introduction	2
1.1 Key Features	2
1.2 Software Requirements	2
1.3 Licensing	2
1.4 Documentation Conventions	2
Chapter 2 Main Window	4
2.1 Main Menu	4
2.1.1 File	5
2.1.2 Communication	5
2.1.3 Help	6
2.2 Tool Bar	7
2.2.1 Connection Mode	8
2.2.2 Connect Button	8
2.3 Log Panel	9
2.4 Clear Button	9
2.5 Status & Progress Bar	10
Chapter 3 Option Bytes	11
3.1 Main Page	11
3.2 Document Page	
Chapter 4 Program	13
4.1 Memory Erasing	
4.2 Image Programming	13
Chapter 5 Memory	15
5.1 Reading and Displaying Target Memory	
5.2 Editing Memory	
Chapter 6 Software Installation	17
6.1 Installation Procedures	17
Chapter 7 Start the TPS32 Programmer	
7.1 Setup Procedures	
IMPORTANT NOTICE AND DISCLAIMER	21



# **Chapter 1 Introduction**

TPS32 Programmer is a user-friendly graphical user interface (GUI) tool for programming the on-chip non-volatile memories of TPS32 devices. It supports both the debug port interface (DAPLink) and the bootloader interface (UART).

#### 1.1 Key Features

TPS32 Programmer is an all-in-one software tool for programming TPS32 devices. It offers the following features:

- Configure option bytes
- · Read and write image files
- Erase Flash memory
- Modify Flash memory

### **1.2 Software Requirements**

Operating Systems	Windows <sup>®</sup> 10 and 11, 64-bit (x64) or 32-bit (x86)
Screen Resolution	Minimum supported resolution is 1200 x 800 pixels.

# 1.3 Licensing

The TPS32 software is released under the 3PEAK license. See the license manifest in the installation directory for details.

## **1.4 Documentation Conventions**

Convention	Usage
Bold	Displays commands, menu paths, and icon names in procedures. For example: Click the <b>File</b> icon and then click <b>Open</b> .
File > New	Represents menu path. For instance: <b>File &gt; New &gt;</b> New Project
Courier New	Displays file locations, user entered text, and source code.



Convention	Usage
	<pre>For example:     <your_sdk_path>/example/tpsensor/     tpsensor_exp/source</your_sdk_path></pre>



# Chapter 2 Main Window

Upon initiating the software, the **Option Byte** configuration page will be displayed.

The main window includes the components described in the subsequent sections.

Communication	Help							
t 🗸 🖍								
otionByte Prog	gram Memory							
n Doc								
ihs_gate_dis	nrst_mode v	nboot0	nsw_boot0	nboot1	🗌 dbank		rdp	
swap_bank	wwdg_sw	iwdg_stdby	iwdg_stop	iwdg_sw	nrst_lvl	pcrop1_rdp	AA	
nrst_shdw	nrst_stdby	nrst_stop	boot_lock			and shot		
						pcrop I_start	pcropz_start	
rp1a_start	wrp1b_s	tart	wrp2a_start	wrp2t	b_start	1FFFF 🔤	1FFFF	÷
FF	€ FF	-	FF	÷ F	F 🗘	pcrop1_end	pcrop2_end	
rp1a_end	wrp1b_e	nd	wrp2a_end	wrp2t	b_end	0	0	-
0	÷ 0	*	0	÷ 0	)			and a second
		Read		Write				
								1

Figure 2-1 Main Window

### 2.1 Main Menu

Figure 2-2 shows the main menu.

Itele       Communication       Help         Uart <ul> <li></li></ul>	4 TPS32_Programme	r						– 🗆 X
Uart <pre>             Program Memory Main Doc Main Doc</pre>	File Communication	n Help						
Opponging       Program       Memory         Main Doc       Image: Doc       Image: Doc       Image: Doc         Image: Base dis       Inst_stade       Image: Doc       Image: Doc         Image: Base dis       Imst_stdby       Image: Doc       Image: Doc       Image: Doc         Image: Base dis       Image: Doc       Image: Doc       Image: Doc       Image: Doc       Image: Doc         Image: Base dis       Image: Doc       Image: Doc </th <th>Uart V</th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th> <th></th>	Uart V							
Ihs_gate_dis       mst_mode       Image: hooot0       inswither inst_state       Image: hooot1	Main Doc	gram Memory						
swap_bank       wwdg_sw       iwdg_stdby       iwdg_stdp       iwdg_sw       nrst_M         nrst_stdw       nrst_stdby       nrst_stop       boot_lock       pcrop1_start       pcrop1_start         wrp1a_start       wrp1a_start       wrp2a_start       wrp2a_start       ifff ©       ifff ©         iff       iff       iff       iff       ifff ©       ifff ©         0       0       0       0       0       0         Read       Write       Write       iffe       iffe	ihs_gate_dis	nrst_mode	nboot0	nsw_boot0	nboot1	dbank		rdp
Inst_stady       Inst_stady <td>swap_bank</td> <td>wwdg_sw</td> <td>iwdg_stdby</td> <td>iwdg_stop</td> <td>iwdg_sw</td> <td>nrst_lvl</td> <td>pcrop1_rdp</td> <td>AA</td>	swap_bank	wwdg_sw	iwdg_stdby	iwdg_stop	iwdg_sw	nrst_lvl	pcrop1_rdp	AA
wrpla_start     wrp2a_start     wrp2a_start     wrp2a_start       FF     FF     FF     pcrop1_end       0     0     0     0	nrst_shdw	nrst_stdby	nrst_stop	boot_lock			pcrop1 start	pcrop2 start
FF     FF     FF     FF     FF     Prop1end     prop2end       0     0     0     0     0     0     0	wrp1a_start	wrp1b_st	art	wrp2a_start	wrp2	2b_start	1FFFF 🗘	1FFFF 🗘
wrp1a_end     wrp2a_end     wrp2a_end       0     0     0       0     0       Read     Write	FF	÷ FF	-	FF	*	FF 🔹	pcrop1 end	pcrop2 end
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	wrp1a_end	wrp1b_er	nd	wrp2a_end	wrp2	2b_end	0	0
Read Write	0	÷ 0	\$	0	*	0	•	•
			Read		Write			
								- V





#### 2.1.1 File

Figure 2-3 shows the **File** menu.

ile Communication Exit	n Help						
OptionByte Pro ain Doc	gram Memory						
☐ ihs_gate_dis ☐ swap_bank ☐ nrst_shdw	nrst_mode vwdg_sw	<ul> <li>nboot0</li> <li>iwdg_stdby</li> <li>nrst_stop</li> </ul>	<ul> <li>nsw_boot0</li> <li>iwdg_stop</li> <li>boot_lock</li> </ul>	nboot1	dbank	pcrop1_rdp	rdp AA pcrop2_start
wrp1a_start FF wrp1a_end 0	wrp1b_ FF wrp1b_ wrp1b_	end 🔹	wrp2a_start FF + wrp2a_end	wrp2b Ff wrp2b 0	start end	1FFFF ÷ pcrop1_end 0 ÷	1FFFF ÷ pcrop2_end
		Read		Write			

Figure 2-3 File Menu

To exit the application, click **File > Exit**.

#### 2.1.2 Communication

Figure 2-4 shows the **Communication** menu.

Communication	Help						-	
n Settings	Memory							
ihs_gate_dis ) swap_bank ) nrst_shdw	nrst_mode vwwdg_sw	<ul> <li>nboot0</li> <li>iwdg_stdby</li> <li>nrst_stop</li> </ul>	<pre>nsw_boot0 iwdg_stop boot_lock</pre>	iwdg_sw	☐ dbank ☐ nrst_lvl	pcrop1_rdp	rdp AA pcrop2 start	
rp1a_start	wrp1b_st	tart 🗘	wrp2a_start	wrp2b_start	\$	1FFFF	1FFFF pcrop2 end	÷
rp1a_end	wrp1b_e	nd	wrp2a_end	wrp2b_end	\$	0	0	×
		Read		Write				
		Read		Write				

Figure 2-4 Communication Menu

This menu has three submenus:

- **Connect**: Used to establish connection with the target board. This option will be grayed out when target board has been already connected.
- **Disconnect**: Used to disconnect from the target board. This option will be grayed out when target board has been already disconnected.
- **Settings**: Used to configure UART parameters.



Figure 2-5 shows the Communication Window for UART.

Communication Config		×
Communication Way	Uart ~	Config
(Com1,115200,Even,One	,8,None)	
		OK

Figure 2-5 Configuration Window for UART

Figure 2-6 shows the configuration parameters for UART.

Uart			
PortName	COM35	~	Ω
BaudRate	115200	~	
Parity	Even	~	
DataBits	8	~	
StopBits	One	~	
FlowCtl	None	~	
	OK		

Figure 2-6 Configuration Parameters for UART

#### 2.1.3 Help

Figure 2-7 shows the **Help** menu.



👍 TPS32_Programme	r						-	×
File Communication	Help							
Uart 🗸 🖌	About							
OptionByte Pro	gram Memory							
Main Doc								
ihs_gate_dis	nrst_mode v	nboot0	nsw_boot0	nboot1	dbank		rdp	
swap_bank	wwdg_sw	iwdg_stdby	iwdg_stop	iwdg_sw	nrst_lvl	pcrop1_rdp	AA	
nrst_shdw	nrst_stdby	nrst_stop	boot_lock					
						pcrop I_start	pcrop2_start	-
wrp1a_start	wrp1b_s	tart	wrp2a_start	wrp2	b_start	1FFFF 🗘	1FFFF	÷
FF	€ FF	×.	FF	•	FF 🗘	pcrop1_end	pcrop2_end	
wrp1a_end	wrp1b_e	nd	wrp2a_end	wrp2	b_end	0	0	\$
0	÷ 0	÷	0	÷	0 🗘			1000
		Read		Write				
								4 ×

Figure 2-7 Help Menu

This menu displays the software version and copyright information.

About	? ×	
	TPS32 Programmer	
	Version 1.0.0	
	Version 1.0.0	
	3PEAK INCORPORATED	
	All rights reserved.	
	Ok	

## 2.2 Tool Bar

Figure 2-8 shows the tool bar. The tool bar has two items, a combo box is used to select the communication way, and a button is used to connect/disconnect the target board.



TPS32_Programmer	r						-	
ile Communication	Help							
lart 🗸 🌌								
OptionByte Pro lain Doc	gram Memory							
Dibe ante die	ant made	C abaat0	and heat	- about	- dheadh		rdp	
Ins_gate_dis	nrst_mode ~	hude station	hsw_booto	- hoodi		pcrop1_rdp	AA	
swap_bank	wwag_sw	iwag_staby	iwdg_stop	iwdg_sw	nrst_ivi			
nrst_sndw	nrst_staby	nrst_stop	DOOT_IOCK			pcrop1_start	pcrop2_start	
wrp1a_start	wrp1b_s	tart	wrp2a_start	wrp2b	o_start	1FFFF 🗘	1FFFF	*
FF	÷ FF	\$	FF 🗘	F	F 🗘	pcrop1 end	pcrop2 end	
wrp1a_end	wrp1b_e	nd	wrp2a_end	wrp2b	_end	0	0	
0	• 0	-	0	0	÷	•	0	•
		Read		Write				
								÷ (
								v

Figure 2-8 Tool Bar

#### 2.2.1 Connection Mode

Figure 2-9 shows the connection mode.

👍 TPS32_Programmer	r						- 0	×
File Communication	n Help							
Uart 🗸 🖍								
Daplink Pro	gram Memory							
Doc								
ihs_gate_dis	nrst_mode ~	nboot0	nsw_boot0	nboot1	🗌 dbank		rdp	
swap_bank	wwdg_sw	iwdg_stdby	iwdg_stop	iwdg_sw	nrst_lvl	pcrop1_rdp	AA	
nrst_shdw	nrst_stdby	nrst_stop	boot_lock			and shot	3 -tt	
						pcrop I_start	pcrop2_start	
wrp1a_start	wrp1b_s	tart	wrp2a_start	wrpi	2b_start	1FFFF	1FFFF 🗘	1
FF	÷ FF	-	FF	-	FF 🗘	pcrop1_end	pcrop2_end	
wrp1a_end	wrp1b_e	nd	wrp2a_end	wrpi	2b_end	0	0	1
0	• 0	\$	0	•	0			
		Read		Write				
								* E
COM2E 11E200 Even O	(no 9 Mana)							*

Figure 2-9 Connection Mode

- **UART**: Connect the target board using the serial port.
  - **NOTE:** When the RDP level is not equal to 0 (0xAA), all operations via the UART connection are prohibited.
- **DAPLink**: Connect the target board using the CMSIS DAPLink with OpenOCD.

#### 2.2.2 Connect Button

Figure 2-10 shows the connect button.



4 TPS32_Programme	ir.						- 0	X
File Communication	n Help							
Uart 🗸 🛃								
OptionByte Pro Main Doc	gram Memory							
				<b>O</b> 1 11	<b>—</b> • •		rde	
ihs_gate_dis	nrst_mode ~	nboot0	nsw_boot0	nboot1	dbank	pcrop1_rdp	AA	
swap_bank	wwdg_sw	iwdg_stdby	iwdg_stop	iwdg_sw	nrst_lvl			
nrst_shdw	nrst_stdby	nrst_stop	boot_lock			pcrop1_start	pcrop2_start	
wrp1a_start	wrp1b_s	tart	wrp2a_start	wrp2l	b_start	1FFFF 🖨	1FFFF	÷
FF			FF	•	F 🗣	pcrop1 end	pcrop2 end	
wrp1a_end	wrp1b_e	end	wrp2a_end	wrp2l	b_end	perop i_end	perope_ene	
0	•	-	0	•	)	•	U	•
		Read		Write				
								~ ¥
								v

Figure 2-10 Connect Button

To connect or disconnect with the target board, toggle the button icon. The button icon will change to reflect the current connection status.

When a device is connected, its ID is sent to the TPS32 Programmer. The tool then configures and displays the memory map based on the connected device type.

### 2.3 Log Panel

Figure 2-11 shows the log panel.

n Doc						
) ihs_gate_dis	nrst_mode 1 ~	onboot0	nsw_boot0	onboot1 🗌 dbank	pcrop1 rdp	rdp AA
swap_bank	wwdg_sw	viwdg_stdby	viwdg_stop	iwdg_sw 🗌 nrst_lvl	0,000	
nrst_shdw	nrst_stdby	nrst_stop	boot_lock		pcrop1_start	pcrop2_start
rp1a start	wrp1b st	art	wrp2a start	wrp2b start	1FFFF 🗘	1FFFF
FF	E FE		FF 🕀	FF 🗳	ncron1 end	pcrop2 epd
p1a_end	wrp1b_er	d	wrp2a_end	wrp2b_end	peroprient	peropr_end
0	0	\$	0	0 🗘	• •	0
				Write		
		Read				

Figure 2-11 Log Panel

The panel displays errors, warnings, and informational events related to the operations executed by the tool.

#### 2.4 Clear Button

Figure 2-12 shows the clear button.



5.0729> Option Register [0x48002030] = 0xft, 5.0729> Option Register: [0x48002044] = 0x1ffff,	2
507239 Option Register: [04800248] = 000, 507239 Option Register: [04800244] = 00ft, 507330 Option Register: [048002050] = 0oft,	1
5:07:30> Option Register; [0:48002070] = 0:00, 15:07:30> Read Option Bytes succeeded. 5:07:31> Nead Hain memory success.	1

Figure 2-12 Clear Button

Click on the button to clear the log.

## 2.5 Status & Progress Bar

Figure 2-13 shows the status bar.

A TPS32_Programmer							- 0	×
File Communication	Help							
Daplink 🗠 🖋								
OptionByte Proc	gram Memory							
Main Doc								
ihs_gate_dis	nrst_mode 1 ~	🛃 nboot0	sw_boot0	🛃 nboot1	dbank		rdp	
swap_bank	wwdg_sw	🛃 iwdg_stdby	🛃 iwdg_stop	🛃 iwdg_sw	nrst_lvl	pcrop1_rdp	AA	
nrst_shdw	nrst_stdby	nrst_stop	boot_lock			pcrop1 start	pcrop2 start	
wrp1a_start	wrp1b_st	art	wrp2a_start	wrp2	2b_start	IIIII 💌	IIIII 💌	
FF	¢ FF	-	FF	\$	FF 🗘	pcrop1_end	pcrop2_end	
wrp1a_end	wrp1b_e	nd	wrp2a_end	wrp2	2b_end	0	0	
0	÷ 0	-	0	÷	0		-	
		Read		Write				
15:07:29> Option Regis 15:07:29> Option Regis 15:07:29> Option Regis 15:07:29> Option Regis 15:07:30> Option Regis 15:07:30> Option Regis 15:07:30> Read Option 15:07:31> Read flash m	tter: [0x48002030] = 0xff, tter: [0x48002044] = 0x1ff tter: [0x48002048] = 0x0, tter: [0x48002048] = 0xff, tter: [0x48002050] = 0xff, tter: [0x48002070] = 0x0, Bytes succeeded. emory success.							1
Daplink								

Figure 2-13 Status Bar

The status bar displays some connection information and progress information.



# Chapter 3 Option Bytes

The **Option Bytes** panel is a feature that allows users to view and modify target option bytes. Option bytes are typically used in microcontrollers to configure various hardware options and features.

With this panel, users can see the option bytes organized into categories, making it easier to navigate and understand their purpose. The values of these option bytes can be modified by updating the corresponding value fields. After making the desired changes, users can click the **Write** button to program and verify that the modified option bytes are correctly programmed.

Additionally, users can click the **Read** button at any time to refresh and display the current values of the option bytes.

#### 3.1 Main Page

This page provides the functionality to read and write option bytes. When the mouse hovers over a specified label, a help prompt is displayed, as shown in Figure 3-1.

] ihs_gate_dis ] swap_bank ] nrst_shdw	nrst_mode 1 v wwdg_sw nrst_stdby	v nboot0 iwdg_stdby	✓ nsw_boot0 ✓ iwdg_stop □ boot_lock	🗹 nboot1 🗹 iwdg_sw	dbank nrst_lvi User optio	pcrop1_rdp	rdp AA
vrp1a_start FF vrp1a_end	wrp1b	start end	wrp2a_start FF wrp2a_end	wrp2	2b_start FF 🔹 2b_end	1FFFF \$ pcrop1_end 0 \$	1FFFF \$
			U	Write	orotection optic	n Read prot	ection optio
		Read		Write			

Figure 3-1 Option Bytes Page

All the option bytes configurations can be classified into three categories:

- User option: Users can configure boot mode, watchdog mode and reset behavior under lowpower mode, and more.
- Write protection option: Users can set up to four write protection areas to prevent accidental code erasure.
  - **NOTE:** The minimum granularity of write protection is one sector (4KB in dual bank mode and 8KB in single bank mode).
- Read protection option: Users can configure two proprietary code read out protection areas or global RDP level to I to prevent code from being read out.



**NOTE:** Proprietary code read out protection (PCROP area) can be deactivated through global RDP degrading. When pcrop1\_rdp bit is set to 1 and RDP level degrades from 1 (0xBB) to 0 (0xAA), the PCROP area will be reset to unprotected initial value and all Flash area has been erased.

If RDP has been set to 0xCC (level 2), the device internal memory can't be accessed any more.

The minimum granularity of read protection are double words in dual bank mode and four words in single bank mode.

### 3.2 Document Page

Figure 3-2 shows the **Option Bytes** document page.

	,				
ptionByte	Program	Memory			
in Doc					
	Field	Field Name	Field Description	Header Description	
			0: gate IHS clock when trim this (default)	1; enable	
	[31]	IHSGATEDIS	1: not gate IHS when trim this	0: disable	
	[30]	(Reserved)			
			00: Reserved		
		01: Reset Input only; a low level on the NRST pin generates a system reset, internal RESET is not propag			
	[29:28]	NRSTMODE	10: Reserved		
			11: Bidirectional reset; NRST pin configured in reset input/output mode (legacy mode)		
			NBOOT0 option bit	1: enable	
	[27]	NBOOT0	0: NBOOT0 = 0	0: disable	
			1: NBOOT0 = 1		
			Software BOOT0	13	
	[26]	NSWBOOT0	0: BOOT0 taken from the option bit NBOOT0		
			1: BOOT0 taken from PB8/BOOT0 pin		
	[25:24]	(Reserved)			
	[23]	NROOT1	Boot configuration		
	20	NBOOTT	Together with the BOOT0 pin, this bit selects boot mode from the Flash main memory, SRAM1 or the System memory.		

#### Figure 3-2 Option Bytes Document

This page shows registers documentation for option bytes. For more information, refer to the TPS325M Technical Reference Manual.



# **Chapter 4 Program**

#### Figure 4-1 shows the **Program** page.

A TPS32_Programmer			-	×
File Communication Help				
Uart V				
OptionByte Program Memory				
Download	Frace flack mem	001		
Elle Dette	Erase nash mem	ory		
rile Path v open	Select_All	Erase Selected Sectors Full Eras	se	
Start Address (HEX) 08000000	Ealast Index	Start Address	Cine	_
	Select Index	Start Address	Size	- 11
Skip flash erase before programming		0x08001000	46	-
Verify programming		0x08002000	4K	 -111
enveny programming	3	0x08003000	4K	- 1
Run after programming	4	0x08004000	4K	- 1
	5	0x08005000	4K	-
	6	0x08006000	4K	
	7	0x08007000	4K	_
Start Program	8	0x08008000	4K	- 11
	9	0x08009000	4K	- 11
	10	0x0800A000	4K	- 11
	0 11	0x0800B000	4K	 - 1
	12	0x0800C000	4K	- 1
	13	0x0800D000	48	- 1
	14	0x0800E000	45	- 1
	15	0x0801000	4K AK	 -
	17	0x08011000	AK	- 1
	18	0x08012000	AK	- 1
		0.00043000	***	
07-37-12> Internal version: 0x00000007. 07-37-13 > Read Hash memory Read 256 bytes form address 0x08000000 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x080000100 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x08000200 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x080000300 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x080000300 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x080000300 compi 07-37-13 > Read Hash memory. Read 256 bytes form address 0x080000300 compi 07-37-13 > Read Hash memory.	leted. leted. leted. leted.			8
(COM35,115200,Even,One,8,None)				

Figure 4-1 Program Page

#### 4.1 Memory Erasing

When connected to a target device, the memory sectors are shown in the right panel, displaying the starting address and size of each sector.

#### **Partial Erase**

- **1.** Locate the memory map list on the right panel, and choose the row you want to erase.
- 2. Click Erase Selected Sectors.

#### **Full Erase**

- 1. In the Erase Flash Memory area, click Select All.
- 2. Click Full Erase to complete the process.

NOTE: The Full Erase button will erase the whole memory, so exercise caution while using it.

#### 4.2 Image Programming

To program a memory, follow these steps:

1. Click the **Open** button and select the file that you want to program.

Supported file formats are binary (.bin) and Intel hex (.hex).



• For .bin files, assigning the correct Start Address is necessary.

NOTE: When programming a binary file, ensure the Start Address is correctly set.

- For .hex files, configuring the Start Address is not required as the programming address is embedded within the .hex file.
- **2.** Select the programming options:
  - Skip flash erase before programming: if checked, the memory will not be erased before programming. This option must be checked only when you are sure that the target memory is already erased.
  - **Verify programming**: this option reads back the programmed memory and compares it byte by byte with the file.
  - Run after programming: this option starts the application immediately after programming.

NOTE: DAPLink doesn't support Run after programming function.

3. Click the Start Programming button to begin the programming process.

The progress bar at the bottom of the window will show the progress of the erase and programming operations.



# **Chapter 5 Memory**

### 5.1 Reading and Displaying Target Memory

This panel is used to display Flash data read back from the target MCU.

File Communication Help				V
File Communication Help				= _ ×
Daplink 🗸 🚿				
OptionByte Program Memory				
Davies Memory O. 51				
Device Memory Open File				
Address(hex) 08000000 1 Size(hex) 4	100 <b>2</b> Width	32-bit <b>3</b>		4 Write Read
Address 0	4	8	C	Ascii
0x08000000 20002A18	08000311	08000E69	0800033D	o* co cio c=o c
0x08000010 08000E65	08000DE5	08001005	0000000	eo o• coo o
0x08000020 0000000	0000000	0000000	08000E71	qo o
0x08000030 08000E11	0000000	08000E6D	08000E75	co o mo cuo o
0x08000040 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000050 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000060 0800034D	0800034D	0800034D	00000000 5	Ma aMa aMa a
0x08000070 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000080 0800034D	08000DE9	08000DFD	0800034D	Ma at at a Ma a
0x08000090 0800034D	0000000	0000000	0800034D	Maa Maa
0x080000 00000000	0000000	0000000	0800034D	Moo
0x080000B0 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x080000C0 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x080000 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x080000E0 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x080000F0 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000100 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000110 0800034D	0800034D	0800034D	0800034D	Ma aMa aMa aMa a
0x08000120 0000000	0000000	00000000	0800034D	Moo
0x08000130_0800034D	08000E55	08000E45	0000000	Ma alla afa a
15.07.29> Option Register: [0x48002030] = 0xft, 15.07.29> Option Register: [0x48002044] = 0x1ftft, 15.07.29> Option Register: [0x49002046] = 0x0, 15.07.29> Option Register: [0x49002046] = 0x0, 15.07.30> Option Register: [0x49002050] = 0xft, 15.07.30> Cption Register: [0x49002070] = 0x0, 15.07.30> Red Option Bytes succeeded. 15.07.31> Read flash memory success.				¥ ا

Figure 5-1 Device Memory

The table below provides descriptions of the various areas depicted in Figure 5-1.

Area No.	Description
1	Indicate the start address.
2	Indicate the size of memory to read.
3	Indicate the display width.
4	Indicate the Write and Read buttons.
5	Display memory data.

### 5.2 Editing Memory

To edit a memory, follow these steps:

- **1.** Double-click the cell to enter edit mode.
- 2. Modify the data.
- 3. Click outside the cell to remove focus.



After modifying the cell, it will be displayed with a green background color, indicating that it is marked to be written.

**4.** Click the Write button to write the changed data.

The progress bar at the bottom of the window will show the progress of the erase and programming operations.

5. Once the data is successfully written, the interface will be refreshed.

plink 🗸 🎽				
ionByte Program Memory				
ice Memory Open File				
ddress(hex) 08000000 ~ Size(h	ex) 400 Width 3	2-bit ~		Write Read
idress 0	4	8	C	Ascii
8000000 12345678	FFFFFFF	FFFFFFF	FFFFFFF	***************
08000010 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
8000020 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
8000030 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
8000040 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
8000050 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
08000060 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	**************
08000070 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	*****************
08000080 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	****************
08000090 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	****************
080000 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
080000B0 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
080000C0 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	****************
080000 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
D80000E0 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
D80000F0 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	****************
08000100 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	****************
08000110 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
08000120 FFFFFFF	FFFFFFF	FFFFFFF	FFFFFFF	***************
08000130 FEFEFFF	FFFFFFF	FFFFFFF	FFFFFFF	***********



NOTE: Memory erase must be executed before memory program.

When using UART, 64-bit write in dual bank mode and 128-bit write in single bank mode are supported; When using DAPLink, only 64-bit write in dual bank is supported.



# **Chapter 6 Software Installation**

### 6.1 Installation Procedures

**1.** Double-click on the installation file.



2. Read and accept the license agreement, then click Next to continue.

						7
ense Agreement						)
Please read the follow	ing important informatio	on before continuing.				(
Please read the follow with the installation.	ing License Agreement.	You must accept the te	rms of this agreem	ent before	continu	uin
SOFTWARE LICENSE	AGREEMENT					
By downloading, insta	lling, copying or otherw	rise using the Licensed I	Materials (as define	d below),		
regardless of whether	you will use the License	d Materials for your ow	n benefit or on bel	half of your	r	
company (you and yo	ur company are collectiv	vely referred to as "You"	'), You agree to be	bound and	d to	
abide by the terms of	this Software License Ag	greement ("Agreement"	).			
1. DEFINITIONS						
"3PEAK" means 3PEA	K INCORPORATED and	its affiliates.				
"3PEAK Devices" mea	ns semiconductor device	es designed, manufactur	ed by or for 3PEA	К.		
<ul> <li>I accept the agreer</li> </ul>	nent					
I do not accept the	agreement					

3. Choose the desired installation directory.



占 Setup - TPS32 Programmer version 1.0		_		×
Select Destination Location Where should TPS32 Programmer be installed?				
Setup will install TPS32 Programmer into the following fold	ler.			
To continue, click Next. If you would like to select a different folde	er, dick Browse.			
C:\User Data\Local\Programs\TPS32 Programm	ner	В	rowse	
At least 354.1 MB of free disk space is required.				
	Back	Nevt	Can	cel
	DOLK	NEAL	Call	Cel

**4.** Opt to create a desktop shortcut for easy access if desired.

📥 Setup - T	PS32 Programmer version 1.0			_		×
Select Add Which ad	itional Tasks ditional tasks should be performed?					
Select the Next.	additional tasks you would like Setup to	o perform while i	nstalling TPS32	Programmer, th	en dick	
Addition	al shortcuts:					
🔽 Creat	e a desktop shortcut					
			Back	Next	Ca	ancel
			Back	Next	Ca	anc

**5.** Complete the installation process. Select to launch TPS32 Programmer automatically when finished if preferred.



📥 Setup - TPS32 Programmer v	ersion 1.0 — 🗆 🗙
	Completing the TPS32 Programmer Setup Wizard
	Setup has finished installing TPS32 Programmer on your computer. The application may be launched by selecting the installed shortcuts.
HOW	Launch TPS32 Programmer
	Finish



# Chapter 7 Start the TPS32 Programmer

## 7.1 Setup Procedures

1. Connect target board with DAPLink (CMSIS-DAP) debugger or UART0 PA1 and PA2 pins.



2. Launch the TPS32 Programmer by double-clicking its desktop shortcut.



3. Select the desired connection type and click the connect button.

#### **IMPORTANT NOTICE AND DISCLAIMER**

Copyright<sup>©</sup> 3PEAK 2012-2023. All rights reserved.

**Trademarks**. Any of the 思瑞浦 or 3PEAK trade names, trademarks, graphic marks, and domain names contained in this document /material are the property of 3PEAK. You may NOT reproduce, modify, publish, transmit or distribute any Trademark without the prior written consent of 3PEAK.

**Performance Information**. Performance tests or performance range contained in this document/material are either results of design simulation or actual tests conducted under designated testing environment. Any variation in testing environment or simulation environment, including but not limited to testing method, testing process or testing temperature, may affect actual performance of the product.

**Disclaimer**. 3PEAK provides technical and reliability data (including data sheets), design resources (including reference designs), application or other design recommendations, networking tools, security information and other resources "As Is". 3PEAK makes no warranty as to the absence of defects, and makes no warranties of any kind, express or implied, including without limitation, implied warranties as to merchantability, fitness for a particular purpose or non-infringement of any third-party's intellectual property rights. Unless otherwise specified in writing, products supplied by 3PEAK are not designed to be used in any life-threatening scenarios, including critical medical applications, automotive safety-critical systems, aviation, aerospace, or any situations where failure could result in bodily harm, loss of life, or significant property damage. 3PEAK disclaims all liability for any such unauthorized use.