



芯视界微电子
visionICs

用户手册

VI4302 A2 API

版本：V0.3
2024.08.6

南京芯视界微电子科技有限公司发布



前言

VI4302 是一款基于直接飞行时间(dToF)激光测距 SoC。该传感器为市场上的微型 ToF 传感器提供了一种紧凑的解决方案。利用自主研发的 SPAD (单光子雪崩二极管) 和独特的 ToF 采集与处理技术, VI4302 可实现最高 60 米的精确距离测量。测量数据及系统配置信息通过 SPI 接口进行传输。

该传感器模块由一个可选的集成激光源 (1 类)、一个 ToF 测距 SoC 以及定制的光学器件组成。该传感器可实现较强的环境光抑制, 因此可用于室外阳光环境下的距离测量。



目录

1	VI4302 概要	7
1.1	工作流程	7
1.2	测距数据格式	9
1.3	读取直方图数据格式	11
2	API 使用	12
2.1	添加 API	12
3	API 说明	13
3.1	VI4302_Reg_Init	13
3.2	VI4302_Stream_On	13
3.3	VI4302_Get_Frame_Cnt	13
3.4	VI4302_read_frame	14
3.5	VI4302_Set_Frame_Data_Format	14
3.6	VI4302_Set_Mp_Openonly	15
3.7	VI4302_Set_Mp_All	15
3.8	VI4302_SinglePixel_Output	15
3.9	VI4302_Set_Bvd	16
3.10	VI4302_BVD_Calculate	16
3.11	VI4302_Start_Ranging	16
3.12	VI4302_Stop_Ranging	17
3.13	VI4302_Update_Config	17
3.14	VI4302_Update_Success_Fail	17
3.15	VI4302_Update_firmware	18
3.16	VI4302_Read_His_Config	18
3.17	VI4302_Read_Histogram	18
3.18	VI4302_Temp_Bvd	19
3.19	VI4302_Frame_Rate_AutoCtrl	19
3.20	VI4302_Read_Reg	19



3.21 VI4302_Write_Reg..... 20

3.22 VI4302_Enable_DcDc..... 20

3.23 VI4302_Frame_Rate_Config..... 20

3.24 VI4302_Bvd_Cal 21

3.25 VI4302_TDC_Cal..... 21

3.26 VI4302_Read_OTP..... 21

3.27 A2_Configurable_Init..... 22

3.28 A2_Configurable_Parameters_Read 23

3.29 S_parameter_init..... 23

3.30 D_parameter_init..... 23

3.31 MP_cnt_init 23

3.32 Xtalk_Para_Init..... 24

3.33 xtalk_rang_cnt_init 24

3.34 Dust_detection_init..... 25

3.35 confidence_k_init..... 25

3.36 Search_logic_init..... 25

3.37 xtalk_cal 26

4 寄存器说明..... 26

4.1 常用寄存器配置说明 26

5 缩略语 28

6 版权与免责声明..... 29

7 修订记录..... 30



图目录

图 1：系统初始化流程图 7

图 2：测距模式流程图 8

图 3：单 PIXEL 模式流程图 8

图 4：MDK 包含头文件 12

表目录

表 1: VI4302 一帧单峰测距数据格式表	9
表 2: VI4302 一帧多峰测距数据格式表	10
表 3: VI4302 直方图数据格式	11
表 4: VI4302_REG_INIT	13
表 5: VI4302_STREAM_ON	13
表 6: VI4302_GET_FRAME_CNT	13
表 7: VI4302_READ_FRAME	14
表 8: VI4302_SET_FRAME_DATA_FORMAT	14
表 9: VI4302_SET_MP_OPENONLY	15
表 10: VI4302_SET_MP_ALL	15
表 11: VI4302_SINGLEPIXEL_OUTPUT	15
表 12: VI4302_SET_BVD	16
表 13: VI4302_BVD_CALCULATE	16
表 14: VI4302_START_RANGING	16
表 15: VI4302_STOP_RANGING	17
表 16: VI4302_UPDATE_CONFIG	17
表 17: VI4302_UPDATE_SUCCESS_FAIL	17
表 18: VI4302_UPDATE_FIRMWARE	18
表 19: VI4302_READ_HIS_CONFIG	18
表 20: VI4302_READ_HISTOGRAM	18
表 21: VI4302_TEMP_BVD	19
表 22: VI4302_FRAME_RATE_AUTOCTRL	19
表 23: VI4302_READ_REG	19
表 24: VI4302_WRITE_REG	20
表 25: VI4302_ENABLE_DCDC	20
表 26: VI4302_FRAME_RATE_CONFIG	20
表 27: VI4302_FRAME_RATE_CONFIG	21
表 28: VI4302_TDC_CAL	21
表 29: VI4302_READ_OTP	21
表 30: A2_CONFIGURABLE_INIT	22
表 31: A2_CONFIGURABLE_PARAMETERS_READ	23
表 32: S_PARAMETER_INIT	23
表 33: D_PARAMETER_INIT	23
表 34: MP_CNT_INIT	23
表 35: XTALK_PARA_INIT	24
表 36: XTALK_RANG_CNT_INIT	24
表 37: DUST_DETECTION_INIT	25
表 38: CONFIDENCE_K_INIT	25
表 39: SEARCH_LOGIC_INIT	25
表 40: XTALK_CAL	26
表 41: 常用寄存器配置说明	26



表 42：缩略语表.....	28
表 43：修订记录.....	30

For

博达传感

Only



1 VI4302 概要

1.1 工作流程

VI4302 工作流程见下图。

图 1：系统初始化流程图

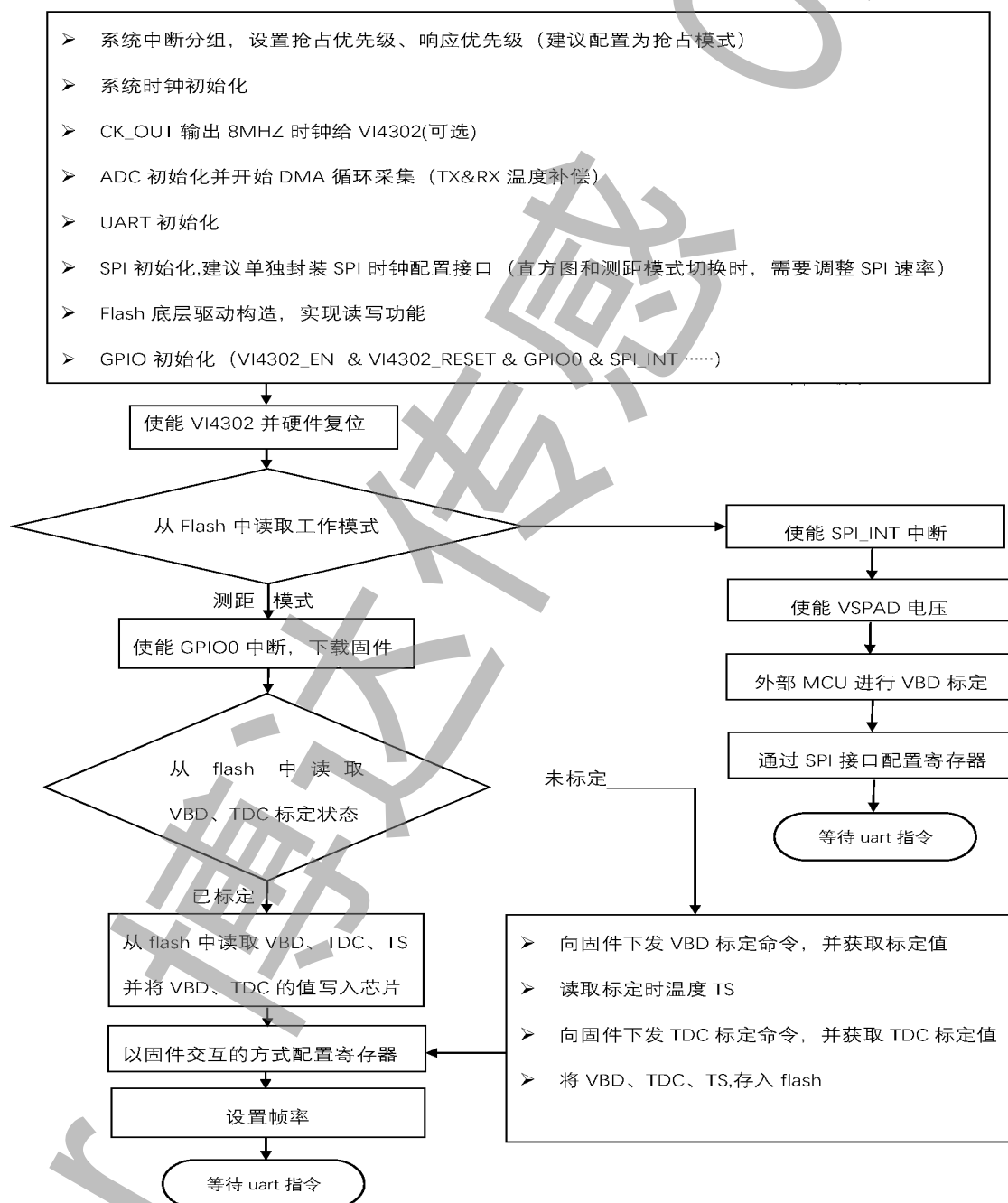


图 2：测距模式流程图

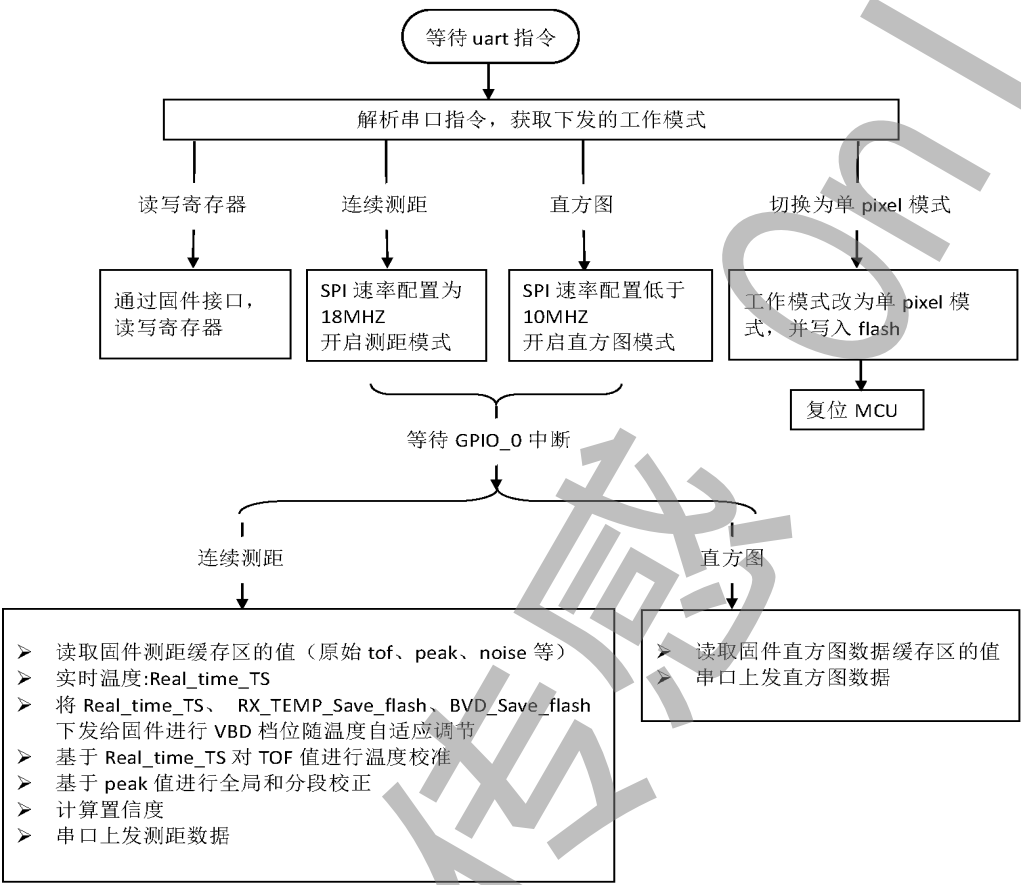
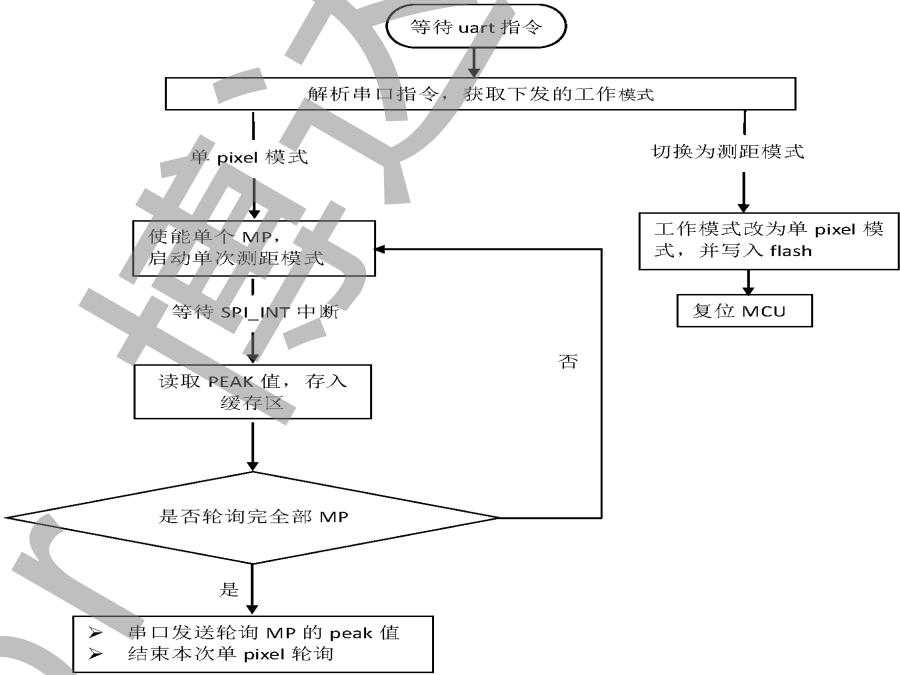


图 3：单 pixel 模式流程图



1.2 测距数据格式

VI4302 A2 单峰模式一帧数据是 16 个字节，如下表所示：

表 1：VI4302 一帧单峰测距数据格式表

域	值	说明
REG_30	0-255	普通测距 TOF 值(低 8 位)
REG_31	0-255	普通测距 TOF 值(高 8 位)
REG_32	0-255	普通 PEAK 值(低 8 位)
REG_33	0-255	普通 PEAK 值(高 8 位)
REG_34	0-255	普通噪声值(低 8 位)
REG_35	0-255	普通噪声值(中 8 位)
REG_36	0-255	普通噪声值(高 8 位)
REG_37	0-255	TS/积分次数值(低 8 位)
REG_38	0-255	TS/积分次数值(高 8 位)
REG_39	0-255	衰减 PEAK 值(低 8 位)
REG_3a	0-255	衰减 PEAK 值(高 8 位)
REG_3b	0-255	衰减噪声值(低 8 位)
REG_3c	0-255	衰减噪声值(中 8 位)
REG_3d	0-255	Flag
REG_3e	0-255	参考 TOF 值(低 8 位)
REG_3f	0-255	参考 TOF 值(高 8 位)

0x30-0x31: 表示测距距离值，需要转换为 mm 为单位的数 据 $TOF(mm) = (reg(0x30)|reg(0x31)<<8)*498>>11;$

0x32-0x33:表示测距信号强度， $Peak1 = reg(0x32)|reg(0x33)<<8;$

0x34-0x36:表示测距背景噪声， $Noise1 = reg(0x34)|reg(0x35)<<8|reg(0x36)<<16;$



0x37-0x38:根据测距的选择, 此部分表示芯片温度或者设置的积分次数;

0x39-0x3a:表示衰减通道信号强度, $\text{Peak2} = \text{reg}(0x39) | \text{reg}(0x3a) < 8$;

0x3b-0x3c:表示衰减通道背景噪声, $\text{Noise2} = \text{reg}(0x3b) | \text{reg}(0x3c) < 8$;

0x3d:表示灰尘检测的标志位, $\text{flag}=0$: 没有灰尘 $\text{flag}=1$ 存在积灰建议清理 $\text{flag}=2$: 存在较严重的积灰, 或者贴着罩子的位置存在真是目标物;

0x3e-0x3f:表示反馈信号距离值, 不接外部信号, 可以不使用。

VI4302 A2 多峰模式一帧数据是 16 个字节, 如下表 1-2-2 所示:

表 2: VI4302 一帧多峰测距数据格式表

域	值	说明
REG_30	0-255	普通测距 TOF0 值(低 8 位)
REG_31	0-255	普通测距 TOF0 值(高 8 位)
REG_32	0-255	普通 PEAK0 值(低 8 位)
REG_33	0-255	普通 PEAK0 值(高 8 位)
REG_34	0-255	普通测距 TOF1 值(低 8 位)
REG_35	0-255	普通测距 TOF1 值(高 8 位)
REG_36	0-255	普通 PEAK1 值(低 8 位)
REG_37	0-255	普通 PEAK1 值(高 8 位)
REG_38	0-255	普通测距 TOF2 值(低 8 位)
REG_39	0-255	普通测距 TOF2 值(高 8 位)
REG_3a	0-255	普通 PEAK2 值(低 8 位)
REG_3b	0-255	普通 PEAK2 值(高 8 位)
REG_3c	0-255	衰减噪声值(低 8 位)
REG_3d	0-255	衰减噪声值(高 8 位)
REG_3e	0-255	衰减 PEAK 值(低 8 位)
REG_3f	0-255	衰减噪声值(高 8 位)

0x30-0x31: 表示测距距离值，需要转换为 mm 为单位的数据 $TOF0(mm) = (reg(0x30)|reg(0x31) << 8) * 498 >> 11;$

0x32-0x33:表示测距信号强度， $Peak0 = reg(0x32)|reg(0x33) << 8;$

0x34-0x35:表示测距距离值，需要转换为 mm 为单位的数据 $TOF1(mm) = (reg(0x34)|reg(0x35) << 8) * 498 >> 11;$

0x36-0x37:表示测距信号强度， $Peak1 = reg(0x36)|reg(0x37) << 8;$

0x38-0x39:表示测距距离值，需要转换为 mm 为单位的数据 $TOF2(mm) = (reg(0x38)|reg(0x39) << 8) * 498 >> 11;$

0x3a-0x3b:表示测距信号强度， $Peak2 = reg(0x3a)|reg(0x3b) << 8;$

0x3c-0x3d:表普通通道背景噪声， $Noise = reg(0x3c)|reg(0x3d) << 8;$

0x3e:表示衰减通道信号强度， $A_Peak = reg(0x3e);$

0x3f:表示衰减通道背景噪声， $A_Noise = reg(0x3f);$

1.3 读取直方图数据格式

VI4302 直方图数据 2048 个字节，共 1024 个 BIN 值，如下表所示。

表 3：VI4302 直方图数据格式

域	值	说明
第 1 个字节	0-255	第 1 个 BIN 值(低 8 位)
第 2 个字节	0-255	第 1 个 BIN 值(高 8 位)
第 3 个字节	0-255	第 2 个 BIN 值(低 8 位)
第 4 个字节	0-255	第 2 个 BIN 值(高 8 位)
第 5 个字节	0-255	第 3 个 BIN 值(低 8 位)
第 6 个字节	0-255	第 3 个 BIN 值(高 8 位)
...	0-255	...
...	0-255	...
...	0-255	...
...	0-255	...



...	0-255	...
...	0-255	...
第 2047 个字节	0-255	第 1024 个 BIN 值(低 8 位)
第 2048 个字节	0-255	第 1024 个 BIN 值(高 8 位)

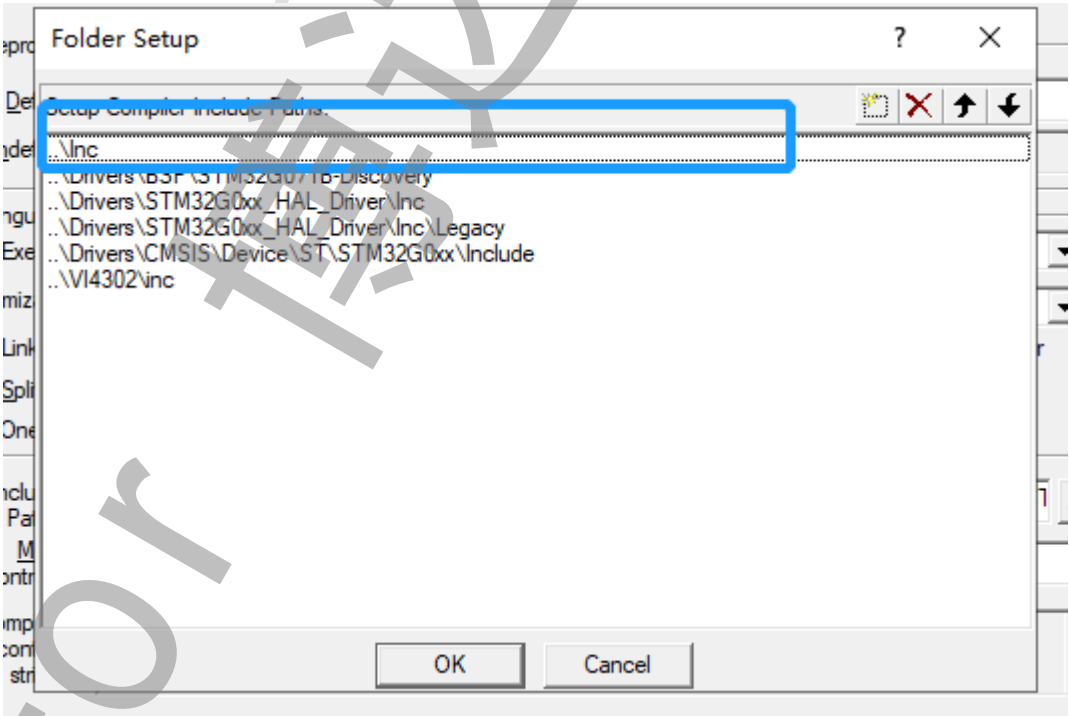
2 API 使用

2.1 添加 API

以 MDK 开发为例，新建 MDK 项目工程（MCU 可选）后，执行如下操作：

- 1、将 VI4302 文件夹添加到当前新建工程项目中，并在项目中添加引用 VI4302_Config.c,VI4302_System.c,User_Driver.c。
- 2、VI4302_Config.c:里面放置了 VI4302 正常工作的寄存器配置，以二维数据的方式存储。
VI4302_System.c:里面放置了 VI4302 正常工作的接口函数。
User_Driver.c:里面放置了 VI4302 引脚和寄存器初始化函数。
- 3、选择 Include Paths，选择项目路径下的选择 VI4302 inc 文件夹，将.c 所引用的头文件包含进来，点击 OK，如下图所示：

图 4：MDK 包含头文件



3 API 说明

3.1 VI4302_Reg_Init

表 4: VI4302_Reg_Init

函数名	VI4302_Reg_Init
类型	uint8_t
输入参数	None
输出参数	None
返回值	0 success !0 error
函数功能	初始化 VI4302 工作需要的寄存器。其中寄存器放置在 Vi4302_Config.c, 用户调用该函数前, 需要实例化 Spi_Write_Reg 函数指针。

3.2 VI4302_Stream_On

表 5: VI4302_Stream_On

函数名	VI4302_Stream_On	
类型	uint8_t	
输入参数	uint8_t mode	STREAM_SINGLE: 单峰测距模式
		STREAM_MULT: 多峰测距模式
	uint8_t STATUS_E en	enable:使能测距
		disable: 失能测距
输出参数	None	
返回值	0 success !0 error	
函数功能	设置 VI4302 单帧或多帧模式, 开始测距 (仅 SINGLE_PIXEL_MODE 有效)	

3.3 VI4302_Get_Frame_Cnt

表 6: VI4302_Get_Frame_Cnt

函数名	VI4302_Get_Frame_Cnt
类型	uint8_t



输入参数	None
输出参数	None
返回值	frame count
函数功能	获取当前数据帧 (仅 SINGLE_PIXEL_MODE 有效)

3.4 VI4302_read_frame

表 7: VI4302_read_frame

函数名	VI4302_read_frame	
类型	uint8_t	
输入参数	uint8_t num	读取的帧数[1-8]
	uint16_t frame_data_start	FIFO 起始地址, Reg[0x00fd]
	uint16_t frame_data_end	FIFO 末尾地址, Reg[0x012a]
输出参数	uint8_t* recv_frame_buf	读取的数据帧存储的缓存区
返回值	0 success !0 error	
函数功能	读取数据帧 (仅 SINGLE_PIXEL_MODE 有效)	

3.5 VI4302_Set_Frame_Data_Format

表 8: VI4302_Set_Frame_Data_Format

函数名	VI4302_Set_Frame_Data_Format	
类型	uint8_t	
输入参数	uint8_t mp_type	MP_TYPE_NORMAL
		MP_TYPE_ATTENUATION
		MP_TYPE_REFERENCE
	uint8_t tof_mask	BIT[3:0] normal tof output mask BIT[7:4]attenl tof output mask
		BIT[0] reference tof output mask
	uint8_t peak_mask	BIT[3:0]normal peak output mask BIT[7:4]attenl peak output mask

		BIT[0] reference peak output mask
	uint8_t noise_multishot_mask	BIT[0] norm_muti_shot output mask
		BIT[1] attenl_muti_shot output mask
输出参数	None	
返回值	0 success !0 error	
函数功能	设置输出的帧数据格式。默认配置可在初始寄存器中配置（仅在 SINGLE_PIXEL_MODE 有效）	

3.6 VI4302_Set_Mp_Openonly

表 9: VI4302_Set_Mp_Openonly

函数名	VI4302_Set_Mp_Openonly	
类型	uint8_t	
输入参数	uint8_t mp_index	设置开启哪一个 MP
输出参数	None	
返回值	0 success !0 error	
函数功能	设置只开启某个 MP（仅在 SINGLE_PIXEL_MODE 有效）	

3.7 VI4302_Set_Mp_All

表 10: VI4302_Set_Mp_All

函数名	VI4302_Set_Mp_All	
类型	uint8_t	
输入参数	STATUS_E en	enable: 开启所有 MP
		disable: 不开启所有 MP
输出参数	None	
返回值	0 success !0 error	
函数功能	设置开启所有的 MP（仅在 SINGLE_PIXEL_MODE 有效）	

3.8 VI4302_SinglePixel_Output

表 11: VI4302_SinglePixel_Output

函数名	VI4302_SinglePixel_Output	
类型	void	
输入参数	uint16_t* peak_data	获取 peak 数据的指针
输出参数	None	
返回值	None	
函数功能	输出 25 个 pixel 对应的 peak 值 (仅在 SINGLE_PIXEL_MODE 有效)	

3.9 VI4302_Set_Bvd

表 12: VI4302_Set_Bvd

函数名	VI4302_Set_Bvd	
类型	Void	
输入参数	uint8_t base	BVD 粗调档位
	uint8_t step	BVD 细调档位
输出参数	None	
返回值	None	
函数功能	设置 BVD	

3.10 VI4302_BVD_Calculate

表 13: VI4302_BVD_Calculate

函数名	VI4302_BVD_Calculate	
类型	uint8_t	
输入参数	None	
输出参数	None	
返回值	获得校正档位, 0x0A 错误值	
函数功能	Vspad 档位调节	

3.11 VI4302_Start_Ranging

表 14: VI4302_Start_Ranging

函数名	VI4302_Start_Ranging	
-----	----------------------	--

类型	uint8_t
输入参数	None
输出参数	None
返回值	0 timeout 0x11 success 0x12 fail
函数功能	开始测距(仅 RANG_MODE 有效, SINGLE_PIXEL_MODE 无效)

3.12 VI4302_Stop_Ranging

表 15: VI4302_Stop_Ranging

函数名	VI4302_Stop_Ranging
类型	uint8_t
输入参数	None
输出参数	None
返回值	0 timeout 0x11 success 0x12 fail
函数功能	停止测距(仅 RANG_MODE 有效, SINGLE_PIXEL_MODE 无效)

3.13 VI4302_Update_Config

表 16: VI4302_Update_Config

函数名	VI4302_Update_Config
类型	void
输入参数	uint8_t Start_End 1 开始更新固件前配置 VI4302 0 更新固件后配置 VI4302
输出参数	None
返回值	None
函数功能	下载固件前后配置 VI4302(仅 RANG_MODE 有效)

3.14 VI4302_Update_Success_Fail

表 17: VI4302_Update_Success_Fail

函数名	VI4302_Update_Success_Fail
类型	uint8_t



输入参数	None
输出参数	None
返回值	0 timeout 0x11 success 0x12 fail
函数功能	显示固件下载失败或成功(仅 RANG_MODE 有效)

3.15 VI4302_Update_firmware

表 18: VI4302_Update_firmware

函数名	VI4302_Update_firmware	
类型	void	
输入参数	uint8_t* p	指向下载的固件的数据块
	uint16_t len	固件长度
输出参数	None	
返回值	None	
函数功能	下载固件 (仅 SINGLE_PIXEL_MODE 有效)	

3.16 VI4302_Read_His_Config

表 19: VI4302_Read_His_Config

函数名	VI4302_Read_His_Config	
类型	uint8_t	
输入参数	HIS_MODE HIS_mode	NORM_HIS
		ATTEN_HIS
		REF_HIS
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	直方图的配置	

3.17 VI4302_Read_Histogram

表 20: VI4302_Read_Histogram

函数名	VI4302_Read_Histogram	
类型	uint8_t	
输入参数	HIS_MODE HIS_mode	获取当前哪个直方图
输出参数	uint8_t* HisData	获取直方图数据
返回值	0 success !0 error	直方图数据获取的成功与否
函数功能	获取直方图数据(仅 RANG_MODE 有效)	

3.18 VI4302_Temp_Bvd

表 21: VI4302_Temp_Bvd

函数名	VI4302_Temp_Bvd	
类型	void	
输入参数	BVD_CAL* Temp_Bvd	Cur_Temp:当前的温度, OTP_Temp:获得 bvd 档位的温度,OTP_BVD:BVD 值
输出参数	None	
返回值	None	
函数功能	bvd 温度校准	

3.19 VI4302_Frame_Rate_AutoCtrl

表 22: VI4302_Frame_Rate_AutoCtrl

函数名	VI4302_Frame_Rate_AutoCtrl	
类型	uint8_t	
输入参数	uint16_t Set_Fps	设置的帧率
输出参数	uint16_t *Real_Fps	返回的帧率
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	设置帧率(仅 42.00.00.01.r03 以上版本可用)	

3.20 VI4302_Read_Reg

表 23: VI4302_Read_Reg

函数名	VI4302_Read_Reg	
-----	-----------------	--

类型	uint8_t	
输入参数	uint16_t Reg_Addr,	读取的寄存器地址
	uint8_t *Val	读取的寄存器值
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	通过 VI4302 内部 MCU 读取寄存器值	

3.21 VI4302_Write_Reg

表 24: VI4302_Write_Reg

函数名	VI4302_Write_Reg	
类型	uint8_t	
输入参数	uint16_t Reg_Addr,	写的寄存器地址
	uint8_t Val	写的寄存器值
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	通过 VI4302 内部 MCU 读取寄存器的值	

3.22 VI4302_Enable_DcDc

表 25: VI4302_Enable_DcDc

函数名	VI4302_Enable_DcDc	
类型	uint8_t	
输入参数	None	
输出参数	None	
返回值	None	
函数功能	开启 vspad 电压	

3.23 VI4302_Frame_Rate_Config

表 26: VI4302_Frame_Rate_Config

函数名	VI4302_Frame_Rate_Config	
类型	uint8_t	
输入参数	uint16_t us	设置测距之间的延时, 单位 us
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	设置测距之间的延时, 用于调整帧率; 与直接设置帧率的接口 VI4302_Frame_Rate_AutoCtrl 互斥, 最后设置的生效	

3.24 VI4302_Bvd_Cal

表 27: VI4302_Frame_Rate_Config

函数名	VI4302_Bvd_Cal	
类型	uint8_t	
输入参数	None	
输出参数	uint8_t *Bvd_Val	标定出的 bvd 档位值
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	通过 VI4302 内部 MCU 进行 BVD 档位标定	

3.25 VI4302_TDC_Cal

表 28: VI4302_TDC_Cal

函数名	VI4302_TDC_Cal	
类型	uint8_t	
输入参数	None	
输出参数	uint8_t *TDC_Val	标定出的 TDC 值
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	通过 VI4302 内部 MCU 进行 TDC 标定	

3.26 VI4302_Read_OTP

表 29: VI4302_Read_OTP

函数名	VI4302_Read_OTP	
类型	uint8_t	
输入参数	uint8_t addr	otp 数据地址
输出参数	None	
返回值	otp 地址对应的数据	
函数功能	读取 VI4302 内部 OTP 数据	

3.27 A2_Configurable_Init

表 30 A2_Configurable_Init

函数名	A2_Configurable_Init	
类型	uint8_t	
输入参数	A2_Configurable_Parameters *Conf_Para_struct	parameter_types:配置参数选择
		para_S: S 参数 (默认: 开)
		para_D: D 参数 (关)
		para_MP: xtalk 标定时 MP 数量 (21)
		xtalk_bin: xtalk 相对 reftof bin 的位置
		xtalk_peak: xtalk 峰的 peak 值
		xtalk_rang_cnt: xtalk 标定的次数 (64)
		flag_1_gap: 灰尘检测 flag=1 的 gap (5~15)
		flag_1_peak_ratio: 灰尘检测 peak 系数 (0~10)
		flag_2_gap: 灰尘检测 flag=2 的 gap (2~13)
		confidence_k: 置信度 upper 系数 (0~64)
		confidence_mode: 输出峰的逻辑 (0 or 1)
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	A2 通用固件可配参数写入	



3.28 A2_Configurable_Parameters_Read

表 31: A2_Configurable_Parameters_Read

函数名	A2_Configurable_Parameters_Read	
类型	uint8_t	
输入参数	None	
输出参数	uint8_t *Rx_Data	A2 可配置参数
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	读取 A2 可配置参数	

3.29 S_parameter_init

表 32: S_parameter_init

函数名	S_parameter_init	
类型	uint8_t	
输入参数	uint8_t s_para	0: S 参数关闭 1: S 参数功能开启
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	单独配置 S 参数功能是否开启	

3.30 D_parameter_init

表 33: D_parameter_init

函数名	D_parameter_init	
类型	uint8_t	
输入参数	uint8_t D_para	0: D 参数关闭 1: D 参数功能开启
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	单独配置 D 参数功能是否开启	

3.31 MP_cnt_init

表 34: MP_cnt_init

函数名	MP_cnt_init	
类型	uint8_t	
输入参数	uint8_t MP_cnt	Xtalk 标定时开启的 MP 数量，默认为 21
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	单独配置 Xtalk 标定时开启的 MP 的数量	

3.32 Xtalk_Para_Init

表 35: Xtalk_Para_Init

函数名	Xtalk_Para_Init	
类型	uint8_t	
输入参数	uint16_t xtalk_bin,	Xtalk 峰相对 reftof 峰的位置
	uint16_t xtalk_peak	Xtalk 峰 peak 值 (MA 后的, 不是直方图的峰值)
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	配置 Xtalk 的 bin 和 peak, 可以下发经验值, 也可以在标定时将返回的值写入 flash, 再次上电时, 读取 flash 中的值调用该接口写入	

3.33 xtalk_rang_cnt_init

表 36: xtalk_rang_cnt_init

函数名	xtalk_rang_cnt_init	
类型	uint8_t	
输入参数	uint8_t rang_cnt	Xtalk 标定时重复检测的次数, 默认 64 次
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	配置 xtalk 标定时重复测距的次数	

3.34 Dust_detection_init

表 37: Dust_detection_init

函数名	Dust_detection_init	
类型	uint8_t	
输入参数	uint8_t gap1	灰尘检测得出 flag=1, gap (5~15)
	uint8_t gap2	灰尘检测得出 flag=2, gap (2~13)
	uint8_t raito	灰尘检测得出 flag=1, peak 系数 (0~10) , 默认 5
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	配置灰尘检测的相关参数	

3.35 confidence_k_init

表 38: confidence_k_init

函数名	confidence_k_init	
类型	uint8_t	
输入参数	uint8_t conf_k	0<conf_k<=64 (默认)
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	配置置信度上限的计算系数, 设置小一些有助于测到玻璃和大角度高反	

3.36 Search_logic_init

表 39: Search_logic_init

函数名	Search_logic_init	
类型	uint8_t	
输入参数	uint8_t logic	0: 输出 tof 小的峰 1: 输出 peak 高的峰
输出参数	None	
返回值	0 timeout 0x11 success 0x12 fail	

函数功能	配置出峰逻辑
------	--------

3.37 xtalk_cal

表 40: xtalk_cal

函数名	xtalk_cal	
类型	uint8_t	
输入参数	uint8_t bin_num	直方图截断的范围
	uint32_t retry_cnt	下发命令后等待 GPIO_0 中断的循环次数
输出参数	uint16_t *X_bin	标定得出的 Xtalk 峰相对 reftof 峰的位置
	uint16_t *X_peak	标定得出的 Xtalk 峰的 peak 值
返回值	0 timeout 0x11 success 0x12 fail	
函数功能	Xtalk 标定命令，由于标定中途要监测 GPIO_0 的中断，所以在标定前要将标志位清 0，并停止测距	

4 寄存器说明

4.1 常用寄存器配置说明

表 41: 常用寄存器配置说明

功能	地址	值	读写	备注
TX TRIGGER	0x0209	0x20	W	粗调
	0x0243	0x00		精调
校准项	0x024F	0xEF		设置 Vspad 档位
	0x0242	0x22		标定 TDC
宏像素选择 (MP)	0x0231	0xFF	W	MP[7:0]
	0x0232	0xFF		MP[15:8]
	0x0233	0xFF		MP[23:16]
	0x0234	0x01		MP[24]
测距参数	0x0080	0x00	W	配置激光重频为 1M Hz



	0x0081	0x96		配置积分为 150
	0x0086	0x00		
	0x0087	0xC8		
QCH & CD	0x0230	0xF0	W	配置 QCH 和 CD
	0x0235	0x42		
	读测距帧			
读一帧数据	0x0030		R	NORM_TOF0
	0x0031			
	0x0032			
	0x0033			NORM_PEAK0
	0x0034			NORM_TOF1
	0x0035			
	0x0036			
	0x0037			NORM_PEAK1
	0x0038			NORM_TOF2
	0x0039			
	0x003a			
	0x003b			NORM_PEAK2
	0x003c			
	0x003d			NORM_NOISE
	0x003e			ATT_PEAK
	0x003f			ATT_NOISE

5 缩略语

表 42：缩略语表

缩写	英文全称	含义
BVD	Breakdown voltage detection	击穿电压检测
SPAD	Single photon avalanche diode	单光子雪崩二极管
TDC	Time to Digital Converter	时间数字转换器
ToF	Time of Flight	飞行时间
dToF	Direct Time of Flight	直接飞行时间

6 版权与免责声明

本文档中的信息为南京芯视界微电子科技有限公司（“芯视界”）的专有技术信息。芯视界有权在任何时候对其产品和/或本文档进行更改、修正、优化、和改进而不做另行通知。购买者在订购前应先获取有关芯视界产品的最新资料。芯视界产品的销售条款和条件以订单确认时为准。

购买者须对选择和使用芯视界产品全权负责，芯视界对购买者产品的应用协助或设计不承担任何责任。

芯视界在此未授予任何知识产权的明示或默示许可。

若基于与本文档所述信息不一致的条款转售芯视界产品，则芯视界对该产品授予的任何保证都将无效。

芯视界的名称及其标志均为芯视界公司的注册商标。本文提及的所有其他产品或服务名称均为其各自所有者的财产。

本文档当前版本中的信息应取代所有以前任何版本中提供的信息。对于与本文档中技术数据的提供、性能或使用有关的或因该等技术数据而产生的任何损失或损害，芯视界不向接收方或任何第三方承担任何责任。

版权所有 ©南京芯视界微电子科技有限公司 2021。保留所有权利。



7 修订记录

表 43: 修订记录

版本	日期	更新说明
V0.1	2024.04.1	初稿继承 A0 版本
V0.2	2024.06.27	增加 Xtalk 标定
V0.3	2024.08.6	增加 A2 通用固件可配置参数

