ANSILIC

AS32A601 Datasheet

Contents

1 Chip Introduction

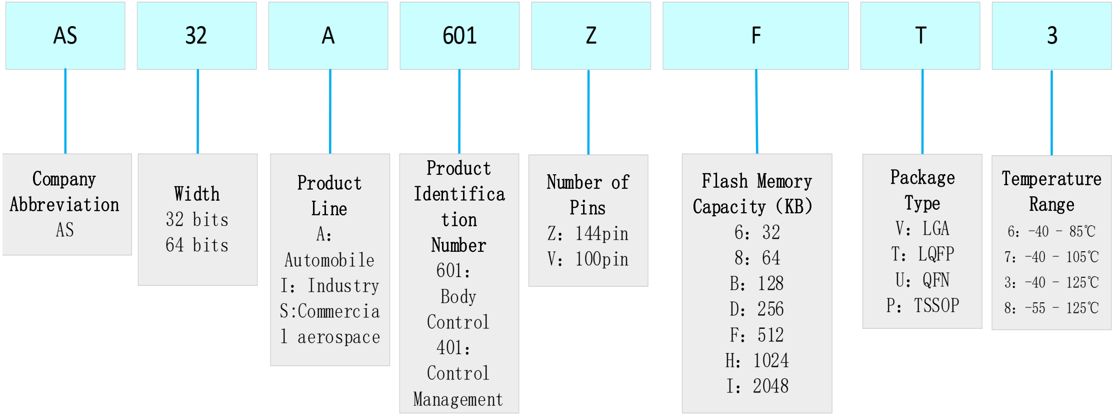
AS32X601 is a MCU product developed by ANSILIC based on the 32-bit RISC-V instruction. The product has a rich Flash capacity, supports functional safety ISO26262 at ASIL-B level, and features high security, low failure, multiple IO, and low cost. There are three product levels: AS32I601、AS32A601 and AS32S601.

- Operating frequency up to 180MHz
- Operating input voltage : 2.7V ~ 5.5V
- Dormancy current : ≤ 200uA (wake-up)
- Typical operating current : ≤ 50mA
- Complies with AEC-Q100 grade1 certification standards (Automotive grade)
- SEU: ≥75Mev.cm^2/mg or 10^{-5} times/device.day (Commercial aerospace grade)
- SEL: ≥75Mev.cm^2/mg (Commercial aerospace grade)
- Packaging : LQFP144

The chip model is as follows :

Chip type	Chip model	Grade	Package	Remarks
MCU	AS32I601ZIT6	Industrial	LQFP144	
MCU	AS32A601ZIT3	Automobile	LQFP144	
MCU	AS32S601ZIT8	Commercial aerospace	LQFP144	

Chip Naming Rules


AS
32
A
601
Z
F
T
3
Company Abbreviation AS
Width 32 bits 64 bits
Product Line A: Automobile I: Industry S:Commercial aerospace
Product Identification Number 601: Body Control 401: Control Management
Number of Pins Z: 144pin V: 100pin
Flash Memory Capacity (KB) 6: 32 8: 64 B: 128 D: 256 F: 512 H: 1024 I: 2048
Package Type V: LGA T: LQFP U: QFN P: TSSOP
Temperature Range 6: -40 - 85℃ 7: -40 - 105℃ 3: -40 - 125℃ 8: -55 - 125℃

1.1 Features

Module	Instruction
Kernel	The self-developed E7 kernel, with FPU and L1 Cache : 16KB data cache, 16KB instruction cache, allows zero-wait access to

	embedded Flash and external memory, with a maximum frequency of 180MHz。804DIMPS/2.68DIMPS/MHz
Clock	● External Crystal Oscillator (OSC) : 8MHz ~ 40MHz ● Internal high frequency oscillator (FIRC) : 16MHz ● Internal low frequency oscillator (FIRC) : 32KHz ● System phase-locked loop (PLL) : maximum support for 500MHz output
Storage	● 512KB internal SRAM (with ECC) ● 16KB ICache and 16KB DCache (with ECC) ● 512KB D-Flash (with ECC) ● 2MB P-Flash (with ECC)
System	● 2 16-channel DMA modules ● 5 memory protection modules (MPU) ● 4 clock monitoring modules (CMU) ● 1 error control module (FCU) ● 1 Power Management Unit (PMU) ● 1 System Control Module (SMU) ● 1 Real-Time Counter Module (RTC) ● 1 hardware encryption module (DSU) supporting AES, SM2 / 3 / 4 and TRNG ● 1 core interrupt control module (CLINT) ● 1 external interrupt control module (PLIC) ● 1 CRC check module
Power Management	● 4 power management modes : RUN, SRUN, SLEEP, DEEPSLEEP ● Low voltage detection and reset function (LVD / LVR) ● High Voltage Detection (HVD)
Analog Interface	● 3 12-bit analog-to-digital converters (ADC) with up to 48 analog channels

	● 2 Analog Comparators (ACMP) ● 2 8-bit digital-to-analog converters (DACs) ● 1 temperature sensor
Timer	● 4 32-bit advanced timers ● 4 16-bit universal timers
Communication Interface	● 6-channel SPI, supporting standard SPI protocol in master-slave mode, with a maximum rate of 30MHz ● 4-channel CAN, all support CANFD ● 4-channel USART module, supporting LIN mode and synchronous serial port mode ● 1 Ethernet (MAC) module supporting 10 / 100M mode, full / half duplex mode ● 4-channel I2C, supporting standard IIC protocol in master-slave mode
Debugging Interface	A debugger with JTAG interface that meets the RISC-V Debug Spec 0.3.2 standard

1.2 Target Application

AS32I601ZIT6

- Industrial field: robot control (joint control, communication management control), industrial general control system, automation control system (platform lift control)

AS32A601ZIT3

- Automotive field: BCM body control system (interior and exterior lighting control, central lock control, window control, horn control, rearview mirror control, etc.), motor drive system (engine cooling fan, water pump, high and low pressure compressor, etc.)

AS32S601ZIT8

- Commercial aerospace: motion control, signal system

2 Chip Overview

2.1 Kernel

The E7 core with hardware FPU is the latest generation of embedded processor developed by ourselves. Its design goal is low cost, low power consumption, high real-time performance, high security, on the basis of this while providing excellent computing performance.

E7 processor is an efficient high-performance processor :

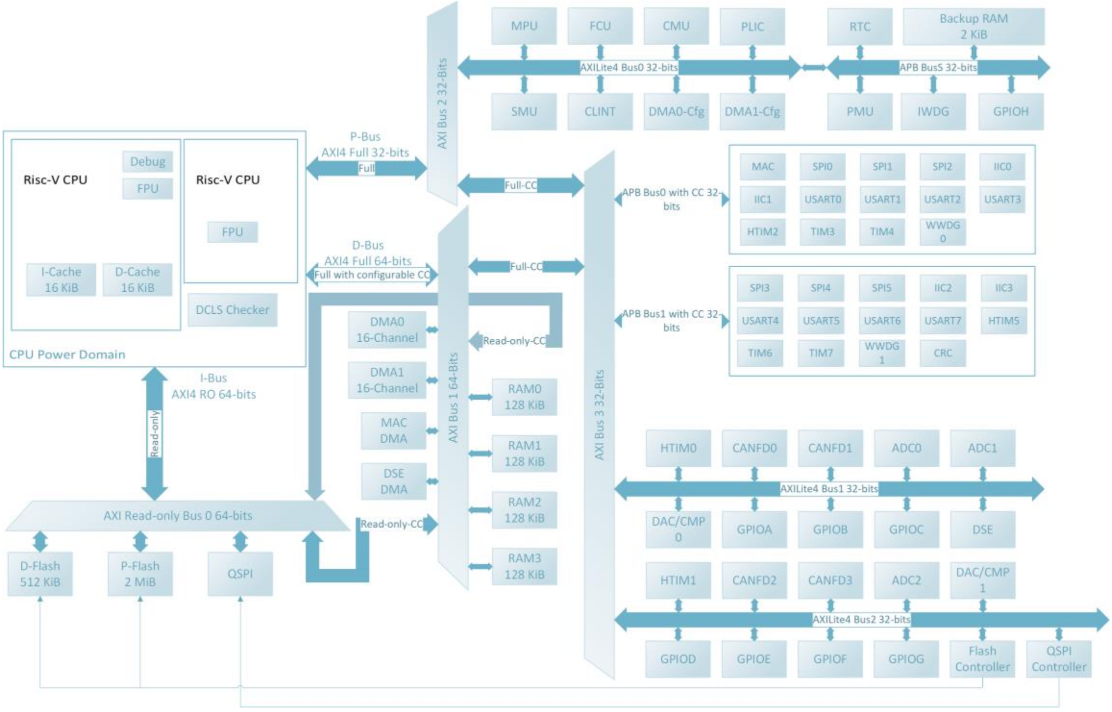
- 8-stage dual-emission pipeline
- Dynamic branch prediction
- The cache of the Harvard architecture (16 KB I-cache and 16 KB D-cache)
- 64-bit AXI4 bus interface

The processor supports the following memory interfaces :

- Instruction AXI interface (AXII)
- Data AXI interface (AXID)
- Low latency peripheral AXI interface (AXIP)

Its built-in dual floating-point FPU (floating-point unit) can accelerate the operation of floating-point related software.

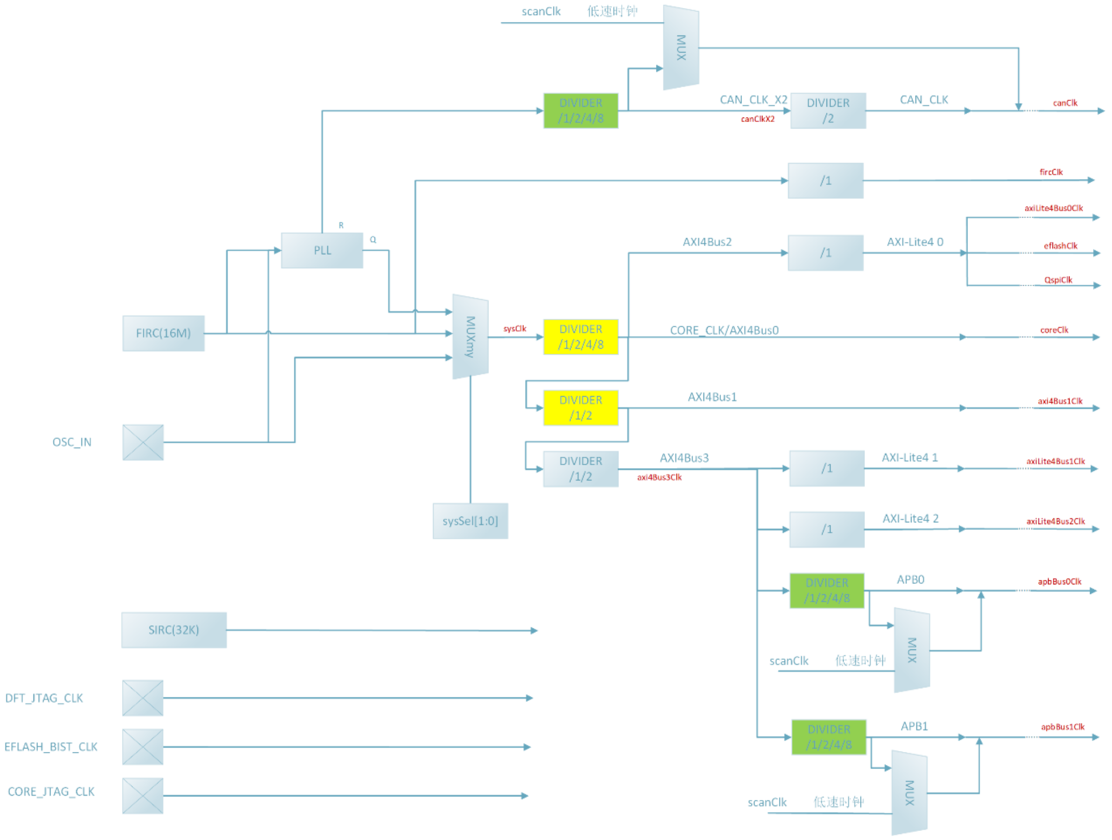
2.2 Bus Architecture



The AXI Crossbar of the MCU bus is a bus matrix used to interconnect the CPU core with the system memory and peripheral module access (read and write).

- The host of Crossbar can actively initiate data access requests, while the slave can only passively accept access
- There are ECC codec modules between each host / slave and bus of Crossbar
- Each port has a separate control bus, address bus and data bus
- Different host can access different slaves at the same time, so as to ensure the working bandwidth of MCU system
- The protection of the host to any slave is protected by MPU

2.3 Clock Management



The clock management module provides clock selection for MCU. It includes fast internal clock (FIRC), internal slow clock (SIRC), external crystal oscillator (OSC) and PLL as the clock source. The reference clock of PLL can be derived from FIRC or OSC.

The default clock of CPU working after reset is from FIRC. FIRC is a 16MHz internal RC oscillator with an error accuracy of about 1 %. Then the application can choose FIRC or external 8-40MHz crystal oscillator as the clock source. If the clock monitoring module detects that the OSC clock is unstable, it automatically switches to FIRC and notifies the application through an interrupt. The frequency doubling clock of PLL can be as high as 500MHz. When the clock monitoring module detects an error in the PLL output clock, it will also inform the application through an interrupt.

2.4 Security Overview

The system is designed for the related systems that need high security integrity to ensure the functional safety of the MCU. For different modules, the protection measures are also different :

- For the device of the kernel class as the host to operate the system, the delayed locking method is used to ensure security
- The security of memory and memory and peripheral data paths is protected by end-to-end ECC. For peripherals, end-to-end protection ends at the APB transfer bridge
- The clock is monitored by multiple discrete CMUs
- The power supply is monitored by PMU and ADC
- The security of peripherals is ensured by application-level measures (multiple peripherals access the same unit at the same time). The hardware supports this application-level redundancy by providing IO modules connected to different peripheral bridges to maximize the independence between monitored and monitored resources
- MBIST and LBIST are used to avoid potential fault accumulation in functional logic and security mechanisms. The dedicated mechanism is used to check the availability of the security mechanism and the function of each error response path
- The fault collection unit is responsible for collecting faults and responding to them
- For error events (including ECC correction and detection, software configuration verification errors) FDU (Fault Detect Unit) is responsible for collecting and reporting system error events to the FCU (Fault Control Unit)
- Common Cause Failure (CCF) is handled through a set of measures to control and avoid CCF across system-level methods (temperature and non-functional signal monitoring) and back-end technologies (physical isolation of silicon regions, routing restrictions)
- Run interference protection is ensured by a layered processor protection method to allow software running in parallel with different ASIL standards

The security design of MCU integration can detect single-point faults and potential faults in a high diagnostic range. But not all common mode faults can be detected. In order to achieve security requirements, it is necessary to make certain requirements on the hardware environment and software environment at the system level. Please refer to the security manual for this requirement.

2.5 Storage System

2.5.1 Built-in SRAM

- Up to 512KB internal SRAM
- Provides 16KB ICache
- Provides 16KB DCache
- Error correction (ECC) is supported

2.5.2 On-chip Flash Memory

- The Flash interface can manage CPU AXI access to the EFlash. This interface can perform erase and programming operations on the EFlash, and implement read/write protection mechanisms. The EFlash structure is as follows:
- Four 512 KB P-Flash blocks
- One 512 KB D-Flash block

Flash controller features :

- Support Flash programming / erase operation
- Support Flash read and programming / erase protection mechanism
- Support error correction (ECC) : 64-bit Flash word 8 ECC bits
- Multiple storage areas support simultaneous operation : read and program / erase operations can be performed in parallel in two storage areas
- Built-in prefetching buffer (Cache) to improve the reading efficiency, the fastest can achieve a single cycle to read data

2.6 DMA

The direct memory access (DMA) controller is used for high-speed data transmission between peripherals and memory, or between memory and memory. Data can move quickly through DMA without CPU intervention.

Two sets of general DMA are integrated on the MCU, and each set of DMA presets 16 channels.

DMA assigns request signals through a unified RMUX unit, which allows each DMA to access any peripheral through software configuration.

DMA supports the following peripherals :

- SPI
- USART
- TIMER
- DAC
- ADC
- CMP
- QSPI(Supports only from QSPI to peripherals)
- E-Flash (Supports only from E-Flash to peripherals)

In addition, the Ethernet MAC and DSE use their own independent DMA.

2.7 MPU

As one of the security mechanisms of MCU, the storage and peripheral protection unit (MPU) provides regional protection functions. The MPU can limit the bus host 's access to memory / peripherals in the selected address area. Different access rights of the protected memory area can be configured independently for the hosts on each bus.

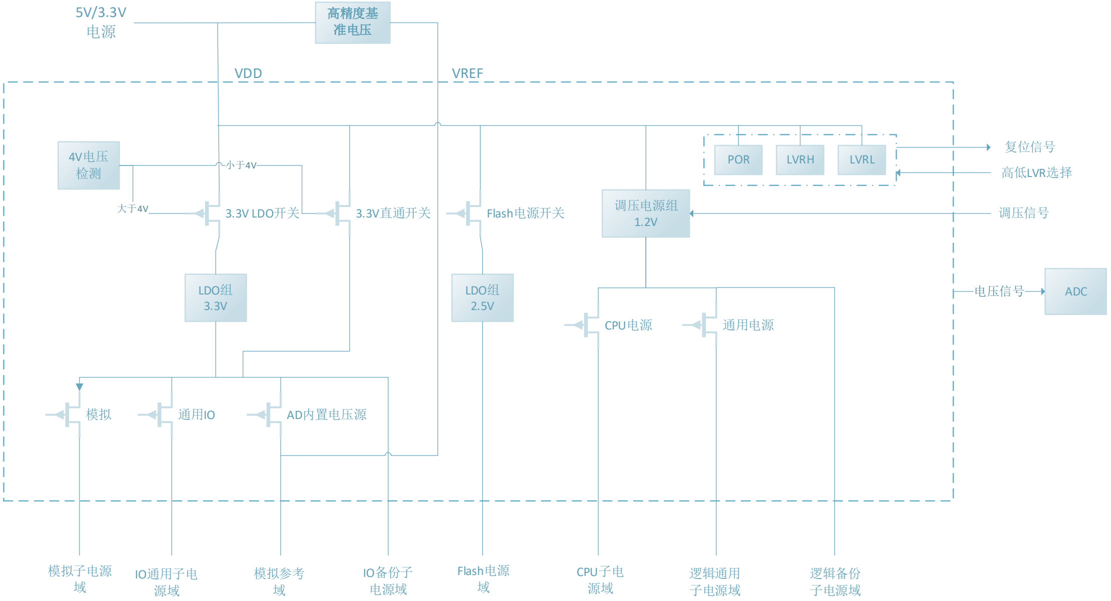
MPU is especially helpful for applications that must protect certain critical or authentication codes from other task misconducts. It is usually managed by a real-time operating system (RTOS). If the program accesses the storage location prohibited by MPU, RTOS can detect it and take action. In the RTOS environment, the kernel can dynamically update the MPU region setting according to the process to be executed.

The MPU has the following characteristics：

- Arbitrary protection width, allowing for any protection range within specific thresholds
- 16 independent memory protection channels (CPU I/D/P MPU) / 8 independent channels (DMA MPU), with overlapping protection regions between channels
- Single-cycle fast exception handling
- Address offset upon a hit to prevent illegal access
- Supports three protection modes: read protection, write protection, or locked region (read protection + write protection)

2.8 PMB

PMU is a whole power management module, which can control the power switch and adjust the voltage of 1.2V power domain, and has the function of power on reset (POR) and low voltage reset (LVR). The overall block diagram is as follows :



2.9 PMU

The Power Management Unit (PMU) is responsible for switching between various MCU power modes and managing functionality within each power mode：

- Run Mode (RUN) - The CPU runs at full speed, which is the default mode of the system.
- Slow Run Mode (SRUN) - The CPU runs at a lower speed, with the PLL and

external crystal oscillator turned off, and only the internal high-speed oscillator is used.

- Sleep Mode (SLEEP) - The CPU and system clocks are turned off, but backup domain devices remain powered on.
- Deep Sleep Mode (DEEPSLEEP) - Most power and clocks, except for the backup domain, are turned off, and the backup domain is driven by an internal low-speed clock.

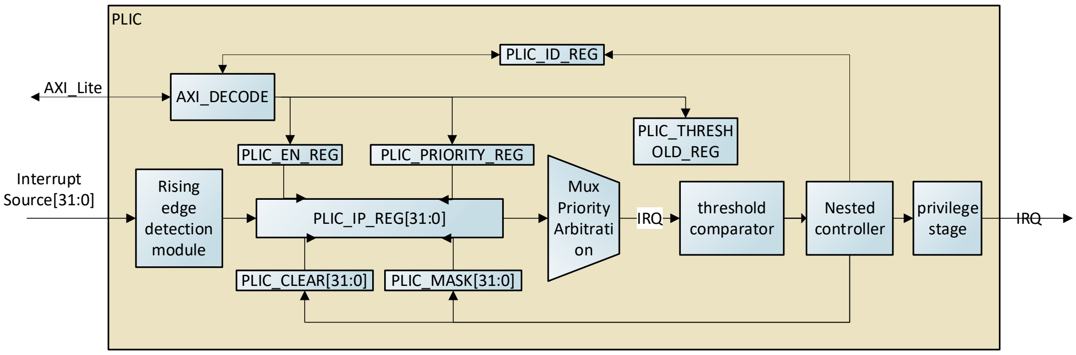
2.10 DSE

DSE (Device Security Engine) is a hardware encryption and decryption module, which conforms to the HIS-SHE security specification standard.

- The key length of AES supports 128bit, 192bit and 256bit
- AES encryption supports ECB, CBC, and OFB
- AES CMAC generation and verification are supported
- Supports SM2, SM3 and SM4
- The generation of true random number (TRNG) is supported

2.11 PLIC

The Platform Level Interrupt Controller (PLIC) is the interrupt controller of the CPU, which mainly samples the interrupt source, prioritizes arbitration and distribution. Peripherals, CSR software interrupts and CSR timer interrupts are uniformly connected to PLIC, and PLIC manages and outputs interrupt request IRQ _ REQ to the kernel. PLIC dispatches one interrupt request with the highest priority each time and supports interrupt nesting.



- Supports 32 interrupt sources
- Each interrupt source is assigned an ID, starting from 1, 0 means no interrupt
- supports setting the priority of each interrupt source, a total of 32 levels, which can be set to the same priority, with the highest level of 31 and the lowest level of 0. When the priority is the same, the ID number is small
- Support to set the interrupt enable bit, pay attention to the difference between the private interface interrupt enable, these two need to be turned on
- Support for setting interrupt priority threshold

- Output single cycle interrupt request to the kernel, only one highest priority interrupt output at the same time, the execution of the interrupt function does not respond to the same interrupt
- The rising edge of the interrupt source is triggered
- The interrupt entry is vector mode
- Interrupt nesting is supported (interrupt nesting is not supported when saving the scene and restoring the scene)
- Save the recovery site fixed as a software way
- Supports RISC-V standard CSR timers and CSR software interrupts
- Support interrupt delegation mechanism
- Support machine mode and management mode

2.12 CRC

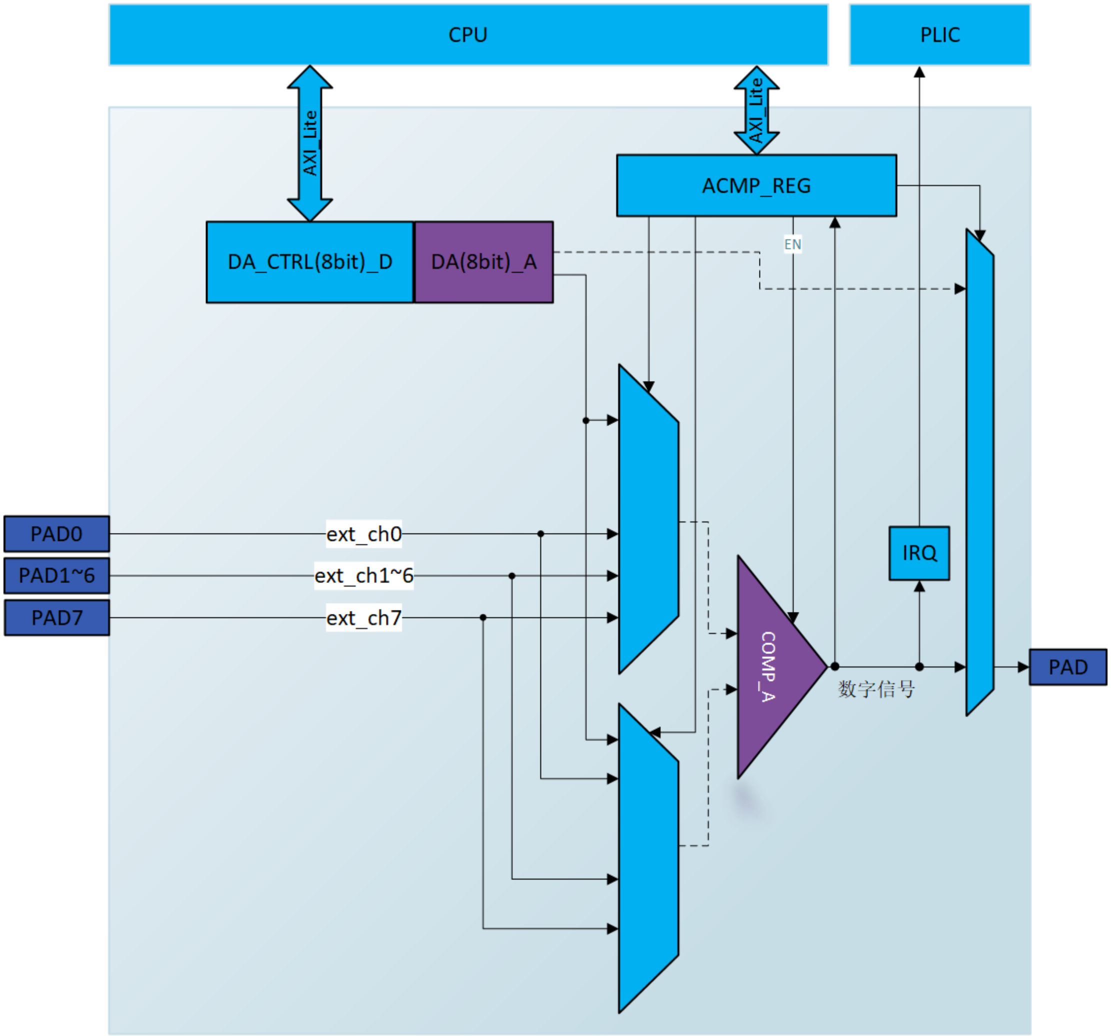
Cyclic Redundancy Check (CRC) is a channel coding technique that generates a short fixed-digit check code based on data such as network packets or computer files. It is mainly used to detect or verify errors that may occur after data transmission or storage. It uses the principle of division and remainder for error detection.

The CRC (Cyclic Redundancy Check) computing unit uses a polynomial generator to generate CRC codes from 8-bit / 16-bit / 32-bit data words.

In many applications, CRC-based technology is also commonly used to verify the integrity of data transmission or storage. According to the functional safety standards, these technologies provide a way to verify the integrity of Flash. The CRC computing unit helps to calculate the signature of the software during operation, and compares the signature with the reference signature generated and stored in the specified storage unit at the time of the link.

- The fully programmable polynomial (7-bit, 8-bit, 16-bit and 32-bit) can be preset by using the number of bits
- Processing 8-bit, 16-bit, 32-bit data size
- Programmable CRC initial value
- Single input / output 32-bit data register
- Input buffer can avoid bus blocking during calculation
- For 32-bit data size, CRC calculation is completed within 4 APB clock cycles (PCLK)
- 8-bit universal register (can be used for temporary storage)
- I / O data reversibility options
- Supports XOR calculation

2.13 DAC



The blue part is a digital circuit, and the purple part is an analog circuit.

2.14 TIMER

The chip contains a total of 8 timers, including 4 advanced timers and 4 general timers. Advanced Timer Features :

- 32-bit increasing, decreasing, increasing / decreasing automatic reloading counter
- 32-bit programmable prescaler
- Repetitive counter
- 6 independent channels, which can be used for input capture, output comparison, PWM generation, single pulse mode output
- Generate interrupts and DMA requests
- Support the complementary output of programmable dead zone
- Support orthogonal encoder and Hall sensor circuit
- It supports external signal control timer, which can realize the synchronization circuit of multiple timer interconnection
- Supports two open-circuit inputs to ensure that the timer output signal is in a safe state

General timer characteristics :

- 16-bit increasing, decreasing, increasing / decreasing automatic reloading counter
- 16-bit programmable prescaler
- 4 independent channels, can be used for input capture, output comparison, PWM generation, single pulse mode output
- Generate interrupts and DMA requests
- Support orthogonal encoder and Hall sensor circuit
- Support external signal control timer, can realize the synchronization circuit of multiple timer interconnection

2.15 FlexCAN

The controller area network (CAN) communicates according to ISO11898-1 : 2015 and Bosch CAN FD specifications. Connecting to the physical layer requires additional transceiver hardware.

All the functions related to processing messages are implemented by the receiving handler and the sending handler. The receiving handler manages messages to be filtered, sends the received messages from the CAN core to the message RAM, and provides the status information of the received messages. The sending handler is responsible for sending messages from the message RAM to the CAN core and providing the sending status information.

Acceptance filtering is achieved by a combination of up to 32 filter elements, each of which can be configured as a range, bit mask, or dedicated ID filter.

- The design conforms to the ISO11898-1 / 2015 specification
- Support CAN and CAN FD frame
- Support the CAN FD frame format specified in the ISO11899 : 2015 specification
- Supports 64-byte CAN FD frames
- Supports variable data rates up to 8Mb / s
- Support normal data rate 1Mb / s
- Support for transmitter delay compensation up to three data bits
- Support configurable sending and receiving mailbox buffers
- It supports two buffers with a depth of 64 frames and 32 ID filtering masks
- Support low ID message priority sending
- Supports message elimination to be sent
- Support for individual error records at fast data rates

2.16 USART

USART can flexibly exchange full-duplex data with external devices to meet the requirements of external devices for the industrial standard NRZ asynchronous serial data format. USART can achieve multiple baud rates.

USART not only supports synchronous one-way communication and half-duplex single-line communication, but also supports LIN (Local Area Network) and CTS / RTS (Modem Operation). High-speed data communication can be achieved by configuring multiple buffers using DMA (direct memory access).

- Support full-duplex / half-duplex asynchronous communication
- Support NRZ standard format
- 16 times oversampling and 8 times oversampling can be configured
- Automatic baud rate detection
- Two internal FIFOs for sending and receiving data, which can be enabled / disabled by software
- The length of the data word is programmable, which can realize 6-bit, 7-bit, 8-bit and 9-bit data words
- Configurable stop bits for 1, 1.5, or 2 stop bits
- Clock input / output for synchronous communication
- Support SPI mode
- Support DMA continuous data transmission
- The transmitter and receiver are supported to enable separately
- Support for sending and receiving a separate signal polarity control
- TX / RX pin switching support
- Support the modulation and demodulation of RS-485 transmitter hardware flow control
- Support parity check control
- Support LIN master-slave mode, configurable interrupt character

2.17 FlexSPI

SPI full name is serial peripheral interface, serial peripheral interface. AS32A601 provides four SPI interfaces to support half-duplex / full-duplex synchronous serial communication. The interface can be configured as host or slave mode. When configured as host mode, it can provide communication clock (SCK) for external slave devices. SPI2 and SPI3 interfaces support 4 slaves, with 4 slave chip options, SPI0 and SPI1 support 1 slave.

- Supports Motorola Mode Serial Peripheral Interface (SPI)
- 4-line full-duplex synchronous transmission (3-line half-duplex)
- 4 ~ 32 bit transmission frame format selection
- SPI clock rate can be configured
- Main mode or slave mode operation
- The highest frequency is PCLK / 2 in the main mode
- The highest frequency of PCLK / 8 is from the mode
- The data transmission order MSB is in the front
- Programmable clock polarity and phase
- Dedicated send and receive flags that trigger interrupts
- Support DMA function

2.18 FlexI2C

I2C (Inter-Integrated Circuit) bus is a simple, two-way two-wire synchronous serial bus, through the clock line (SCL) and data line (SDA) for data transmission. SCL is a clock signal generated by the host driver, and SDA is a two-way data signal, which can be generated by both the main device (host) and the slave device (slave).

- Support master-slave integration mode
- Support 7bit range address and 10bit range address
- Support slave monitoring function
- Support multi-host arbitration
- Support programmable burr filtering
- Support SDA / SCL low-level timeout interrupt

2.19 MAC

It implements the Media Access Control (MAC) defined by IEEE 802.3 Collision Detection over Ethernet Connections (CSMA / CD) algorithm. Communication with external hosts is achieved through a set of control and state registers and direct memory access (DMA) controllers for external shared RAM. The MAC interface sends data to the host through DMA. It automatically acquires sending buffer data and stores receiving buffer data to external RAM. It implements a variety of memory allocation schemes by managing the receiving and sending descriptor list. The RAM inside the MAC interface is used as a configurable FIFO memory block, and there are separate memory blocks for sending and receiving processes.

- Support external PHY interface to achieve 10 / 100Mbit / s data transmission rate
- It communicates with external Fast Ethernet PHY through IEEE 802.3 compliant MII interface
- Support full-duplex and half-duplex operation
- The header and frame start data (SFD) are inserted in the sending path and deleted in the receiving path
- The automatic generation of CRC and PAD can be controlled frame by frame
- PAD / CRC can be automatically removed when receiving frames
- Support a variety of flexible address filtering mode
- Two sets of cache FIFO (receiving FIFO and sending FIFO)
- DMA performs data migration in the memory and MAC caches

2.20 GPIO

Each GPIO pin can be configured as output mode (push-pull output, open-drain output), input mode (floating input, pull-up input, pull-down input) or peripheral multiplexing mode by software. Most GPIO multiplexing has the function of multiple digital modules / analog modules, and the maximum multiplexing function can reach 6 groups. All GPIO have a high current output (20mA) capability. Each IO has an independent interrupt control enablement, and can be configured as rising edge, falling edge or double edge trigger. Each GPIO is equipped with an optional digital filter, and the clock cycle of the filter is configurable.

2.21 DEBUG

The debugger is based on Debug Spec 0.3.2 of RISC-V Foundation. The external interface can access the internal registers and memories, and control the operation / stop / reset of the program.

- JTAG debugging interface
- 16 hardware breakpoints

3 Appendix

3.1 Address Mapping

The address mapping table is as follows：

Space	Start Address	End Address	Space Size	Description
AXI Bus 0	0x0000_0000	0x1FFF_FFFF	512 MiB	Instruction Bus
	0x0000_0000	0x0000_0FFF	4 KiB	BOOT-ROM
	0x0100_0000	0x0107_FFFF	512 KiB	D-FLASH
	0x0200_0000	0x023F_FFFF	2 MiB	P-FLASH
	0x1000_0000	0x1FFF_FFFF	256 MiB	QSPI-FLASH
AXI Bus 1	0x2000_0000	0x2FFF_2FFF	256 MiB	Data Bus
	0x2000_0000	0x2001_FFFF	128KiB	SRAM0
	0x2002_0000	0x2003_FFFF	128KiB	SRAM1
	0x2004_0000	0x2005_FFFF	128KiB	SRAM2
	0x2006_0000	0x2007_FFFF	128KiB	SRAM3
AXi Bus 2	0x3000_0000	0x7FFF_FFFF	1.25 GiB	Configuration /Peripheral Bus
AxiLite4 Bus0	0x3000_0000	0x3FFF_FFFF	256MiB	Setting bus
	0x3000_0000	0x3000_03FF	1 KiB	MPU0-IBUS
	0x3000_0400	0x3000_07FF	1 KiB	MPU1-DBUS
	0x3000_0800	0x3000_0BFF	1 KiB	MPU2-PBUS
	0x3000_0C00	0x3000_0FFF	1 KiB	MPU3-DMA0
	0x3000_1000	0x3000_13FF	1 KiB	MPU4-DMA1
	0x3000_2000	0x3000_23FF	1 KiB	FCU
	0x3000_3000	0x3000_33FF	1 KiB	CMU0
	0x3000_3400	0x3000_37FF	1 KiB	CMU1
	0x3000_3800	0x3000_3BFF	1 KiB	CMU2
	0x3000_3C00	0x3000_3FFF	1 KiB	CMU3

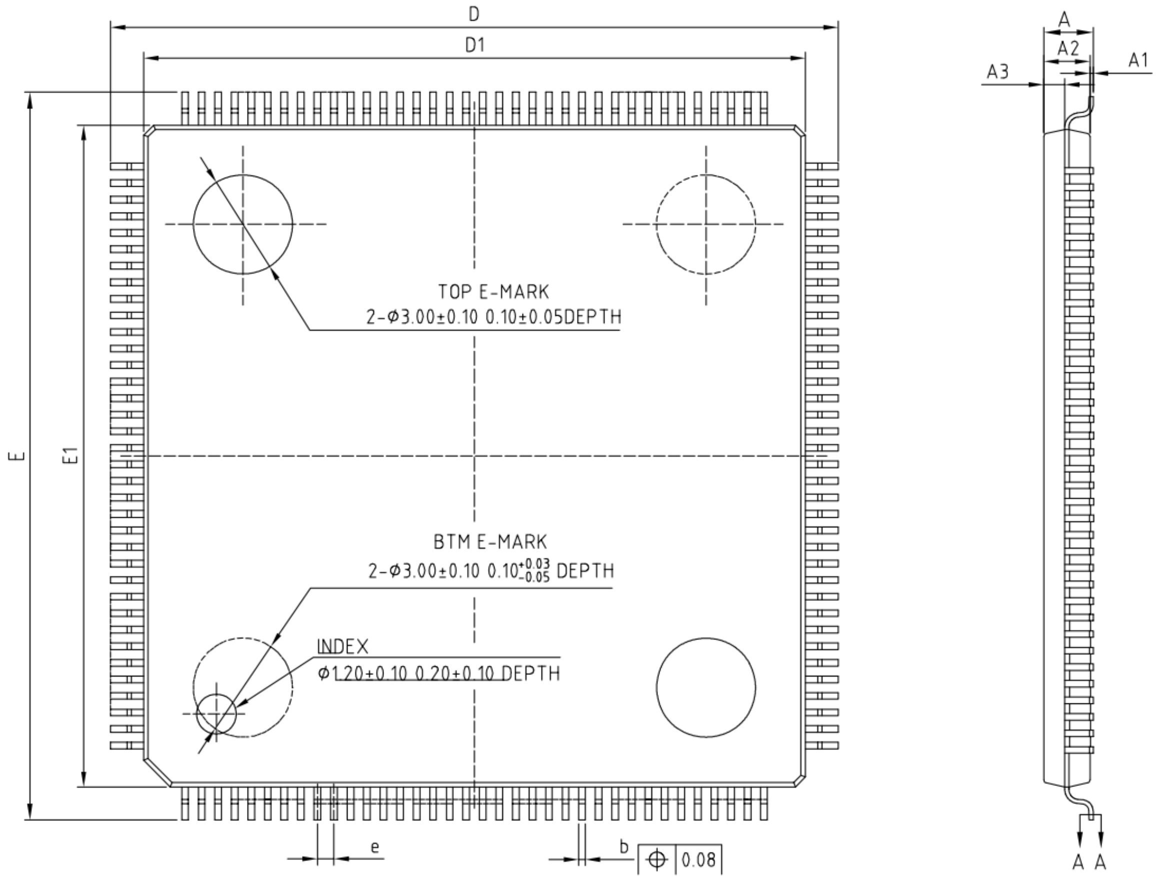
	0x3000_4000	0x3000_43FF	1 KiB	DMA0-CFG
	0x3000_4400	0x3000_47FF	1 KiB	DMA1-CFG
	0x3000_5000	0x3000_53FF	1 KiB	IWDG
	0x3000_6000	0x3000_63FF	1 KiB	SMU
	0x3001_0000	0x3001_FFFF	64 KiB	CLINT
	0x3002_0000	0x3002_FFFF	64 KiB	PLIC
APB Bus S	0x3010_0000	0x301F_FFFF	1 MiB	Backup Domain Bus
	0x3010_0000	0x3010_07FF	2 KiB	BACKUP-RAM
	0x3010_1000	0x3010_13FF	1 KiB	RTC
	0x3010_2000	0x3010_23FF	1 KiB	GPIOH
	0x3010_3000	0x3010_33FF	1 KiB	PMU
AXI Bus 3	0x4000_0000	0x7FFF_FFFF	1 GiB	Peripheral bus
	0x4000_0000	0x4000_03FF	1 KiB	DMAMUX0
AxiLite4 Bus1	0x4100_0000	0x41FF_FFFF	16 MiB	Peripheral Lite bus 1
	0x4100_0000	0x4100_03FF	1 KiB	TIM0
	0x4101_0000	0x4101_FFFF	16 KiB	CANFD0
	0x4102_0000	0x4102_FFFF	16 KiB	CANFD1
	0x4100_1000	0x4100_13FF	1 KiB	ADC0
	0x4100_2000	0x4100_23FF	1 KiB	ADC1
	0x4100_3000	0x4100_33FF	1 KiB	DAC/CMP0
	0x4100_4000	0x4100_43FF	1 KiB	GPIOA
	0x4100_5000	0x4100_53FF	1 KiB	GPIOB
	0x4100_6000	0x4100_63FF	1 KiB	GPIOC
	0x4110_0000	0x414F_FFFF	1 MiB	DSE
AxiLite4 Bus2	0x4200_0000	0x42FF_FFFF	16 MiB	Peripheral Lite bus 2
	0x4200_0000	0x4200_03FF	1 KiB	TIM1
	0x4201_0000	0x4201_FFFF	16 KiB	CANFD2
	0x4202_0000	0x4202_FFFF	16 KiB	CANFD3

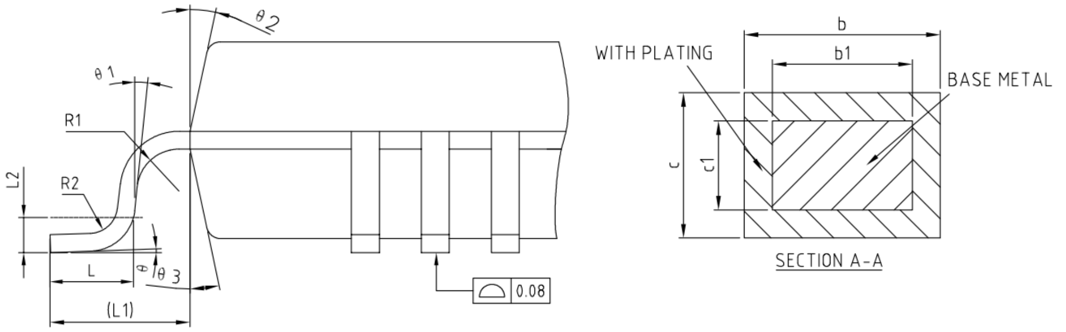
	0x4200_1000	0x4200_13FF	1 KiB	ADC2
	0x4200_2000	0x4200_23FF	1 KiB	DAC/CMP1
	0x4200_3000	0x4200_33FF	1 KiB	GPIOD
	0x4200_4000	0x4200_43FF	1 KiB	GPIOE
	0x4200_5000	0x4200_53FF	1 KiB	GPIOF
	0x4200_6000	0x4200_63FF	1 KiB	GPIOG
	0x4210_0000	0x4210_03FF	1 KiB	D-FLASH CTRL
	0x4210_0400	0x4210_07FF	1 KiB	P-FLASH CTRL
	0x4210_0800	0x4210_0BFF	1 KiB	QSPI CTRL
APBBus0	0x5000_0000	0x50FF_FFFF	16 MiB	Peripheral APB bus 0
	0x5000_0000	0x5000_03FF	1 KiB	MAC
	0x5000_1000	0x5000_13FF	1 KiB	SPI0
	0x5000_2000	0x5000_23FF	1 KiB	SPI1
	0x5000_3000	0x5000_33FF	1 KiB	SPI2
	0x5000_4000	0x5000_43FF	1 KiB	IIC/IIS0
	0x5000_5000	0x5000_53FF	1 KiB	IIC/IIS1
	0x5000_6000	0x5000_63FF	1 KiB	USART0
	0x5000_7000	0x5000_73FF	1 KiB	USART1
	0x5000_8000	0x5000_83FF	1 KiB	USART2
	0x5000_9000	0x5000_93FF	1 KiB	USART3
	0x5000_A000	0x5000_A3FF	1 KiB	TIM2
	0x5000_B000	0x5000_B3FF	1 KiB	TIM3
	0x5000_C000	0x5000_C3FF	1 KiB	TIM4
	0x5000_D000	0x5000_D3FF	1 KiB	WWDG0
	0x5000_E000	0x5000_E3FF	1 KiB	DMAMUX1
APBBus1	0x5100_0000	0x51FF_FFFF	16 MiB	Peripheral APB bus 1
	0x5100_0000	0x5100_03FF	1 KiB	SPI3
	0x5100_1000	0x5100_13FF	1 KiB	SPI4

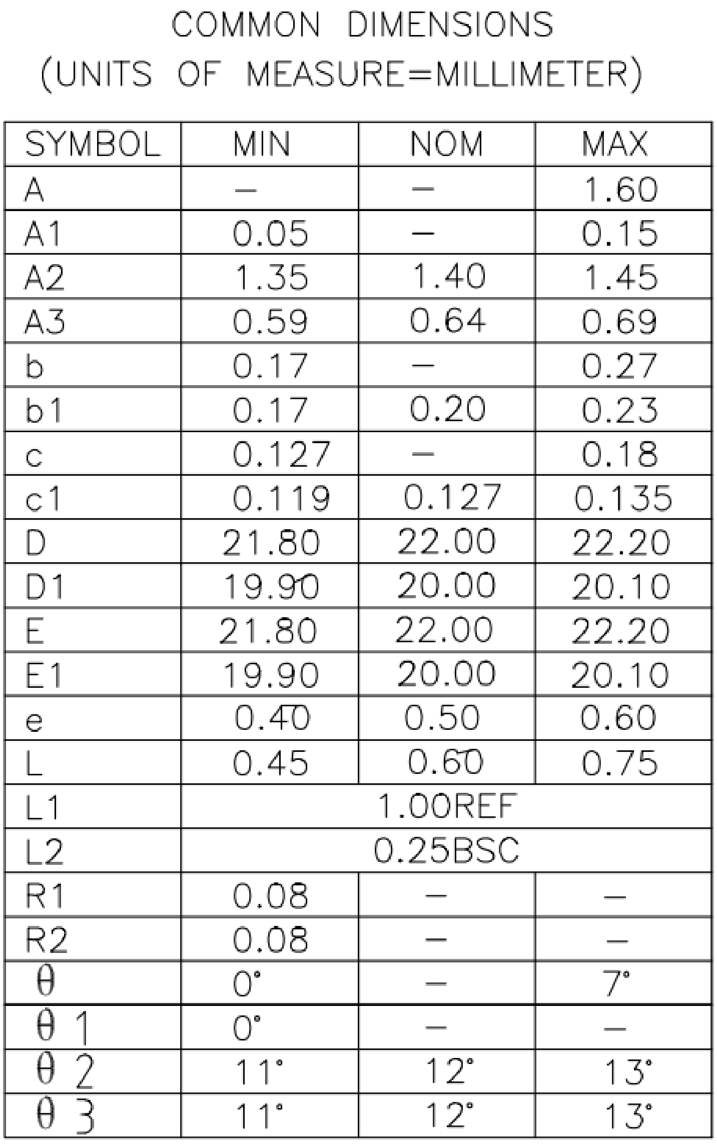
	0x5100_2000	0x5100_23FF	1 KiB	SPI5
	0x5100_3000	0x5100_33FF	1 KiB	IIC/IIS2
	0x5100_4000	0x5100_43FF	1 KiB	IIC/IIS3
	0x5100_5000	0x5100_53FF	1 KiB	USART4
	0x5100_6000	0x5100_63FF	1 KiB	USART5
	0x5100_7000	0x5100_73FF	1 KiB	USART6
	0x5100_8000	0x5100_83FF	1 KiB	USART7
	0x5100_9000	0x5100_93FF	1 KiB	TIM5
	0x5100_A000	0x5100_A3FF	1 KiB	TIM6
	0x5100_B000	0x5100_B3FF	1 KiB	TIM7
	0x5100_C000	0x5100_C3FF	1 KiB	WWDG1
	0x5100_D000	0x5100_D3FF	1 KiB	DMAMUX2

4 Packaging

The package type is LQFP144, and the package size is as follows :


D
D1
E
E1
TOP E-MARK
2-Φ3.00±0.10 0.10±0.05DEPTH
BTM E-MARK
2-Φ3.00±0.10 0.10+0.03/-0.05 DEPTH
INDEX
Φ1.20±0.10 0.20±0.10 DEPTH
e
b
0.08
A
A2
A3
A1
A A





COMMON DIMENSIONS
(UNITS OF MEASURE=MILLIMETER)

SYMBOL	MIN	NOM	MAX
A	–	–	1.60
A1	0.05	–	0.15
A2	1.35	1.40	1.45
A3	0.59	0.64	0.69
b	0.17	–	0.27
b1	0.17	0.20	0.23
c	0.127	–	0.18
c1	0.119	0.127	0.135
D	21.80	22.00	22.20
D1	19.90	20.00	20.10
E	21.80	22.00	22.20
E1	19.90	20.00	20.10
e	0.40	0.50	0.60
L	0.45	0.60	0.75
L1	1.00REF		
L2	0.25BSC		
R1	0.08	–	–
R2	0.08	–	–
θ	0°	–	7°
θ 1	0°	–	–
θ 2	11°	12°	13°
θ 3	11°	12°	13°