ANSILIC

AS32I401 Datasheet

32bit microcontroller based on RISC-V, using dual-core lock-step architecture, dual backup QSPI-Flash, 192KB SRAM, 20 communication interfaces.

- Core

- 32-bit CPU based on RISC-V

- 32-bit RISC-V RV32IMZicsr instruction set

- Operating frequency: 120MHz

- Single-cycle dual-core lock step architecture design, resistant to common mode faults

- The fail-save design supports single-cycle error rollback

- The internal storage and bus use low-power SECDED encoding for automatic error correction

- The system control module adopts three-mode redundancy

- Storage

- Dual backup QSPI Flash

– 128KB SRAM

- 32KB Cache Instruction memory (ICACHE)

- 32KB Data Tightly Coupled Memory (DTCM)

- Simulation module

- 2 12bit ADC interfaces, 16 multiplexed input channels

- Two 12bit DAC ports

- DMA timer

- 16-channel DMA controller

- 3 16-bit advanced timers, each with up to 6 channels for input capture/output comparison /PWM or single pulse counting

- 1 watchdog timer

- Digital interface

- Up to 4 standard UART interfaces

- Up to 4 SPI interfaces

- A maximum of four IIC ports are provided

- One LIN interface

-Two CAN FD bus interfaces

- Two QEI ports

- One 10/100M Ethernet port

- Two MIL-STD-1553B ports

- Working environment

- -40℃ to +125℃

- Packaging process: LQFP144

- Size: 20*20mm, pitch 0.5mm

Table of Contents

1 Specifications

AS32I401 microcontroller is a high performance and reliable general-purpose microcontroller based on 32-bit RISC-V processor. At its core, it uses a RISC-V processor with dual-core lockstep function and operates at up to 120MHz.

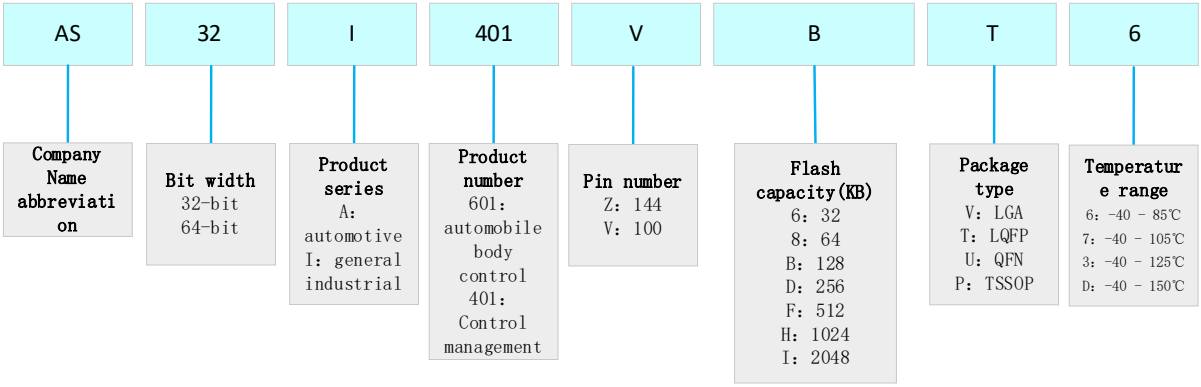
The AS32I401 microcontroller has built-in dual backup QSPI Flash, and the on-chip storage capacity can reach 192KB, including 32KB ICACHE, 32KB DTCM, and 128KB distributed SRAM.

The AS32I401 microcontroller provides a variety of peripheral interfaces that are mounted on the AHB bus. The device supports up to 80 GPIOs, it provides 4 standard UART interfaces, 1 debug UART, 4 SPI interfaces, 4 IIC interfaces, 1 LIN interface, and 2 CAN interfaces FD bus interface, 2 12-bit DAC interface, 2 12-bit ADC interface, 3 16-bit advanced timers, 2 QEI interface, 1 10/100M Ethernet interface, 2 MIL-STD-1553B interfaces.

Chip model:

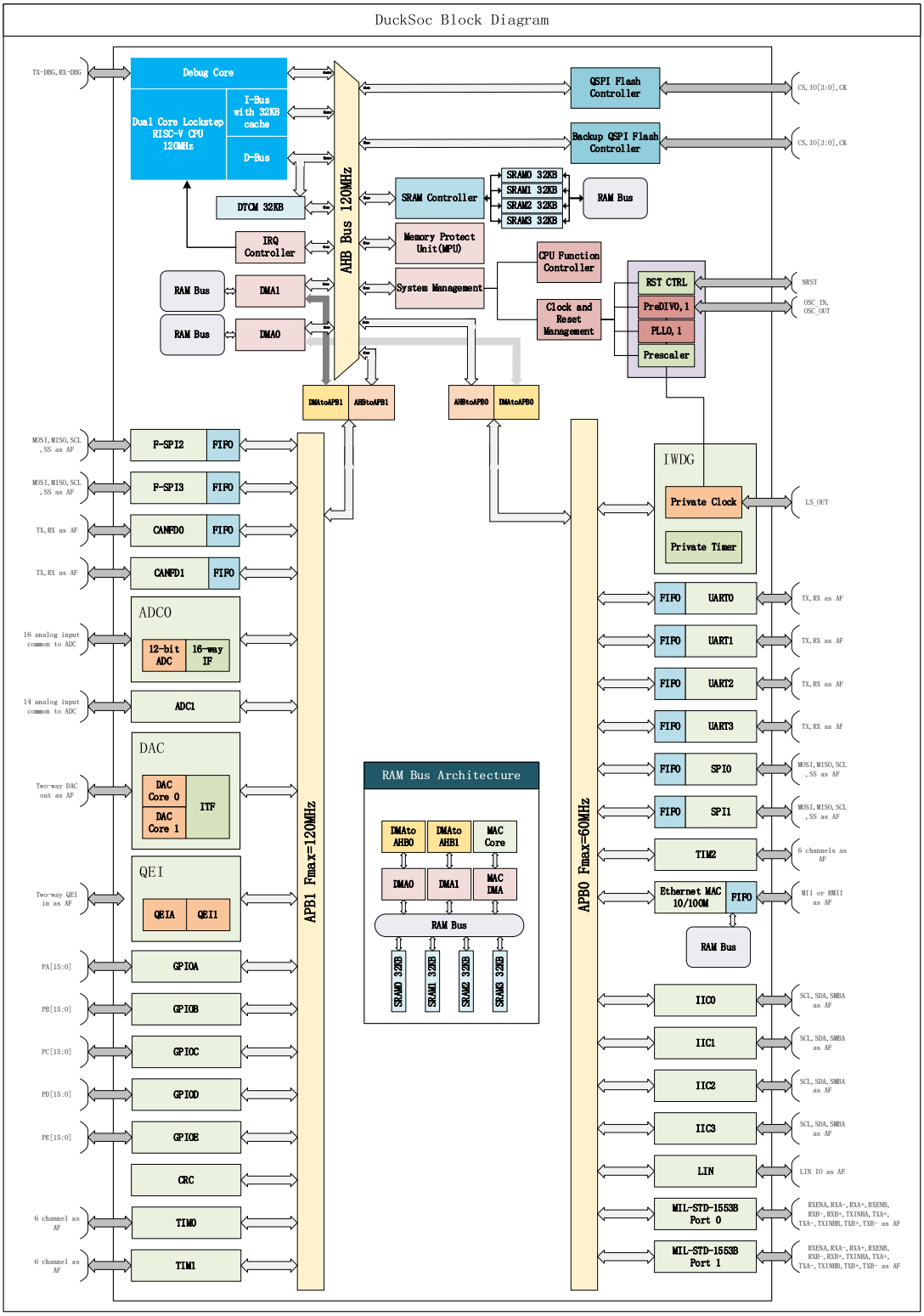
Chip type	Device	Grade	Package
MCU	AS32I401VBT6	Industrial	LQFP100

Chip naming rule



2 Function Overview

Figure 2.1 AS32I401 system architecture diagram



2.1 RISC-V Processor

The AS32I401 microcontroller is a high-performance and highly reliable embedded microcontroller based on 32-bit RISC-V processors. The processor core adopts the RISC-V (Rv32IMZicsr) processor core with dual-core lock step function, which has the characteristics of single-bit code error correction and multi-bit code error detection single-cycle rollback, and has good reliability.

2.2 Memory

- Dual backup QSPI Flash
- 128KB SRAM
- 32KB Cache Instruction memory (ICACHE)
- 32KB Data Tightly Coupled Memory (DTCM)

2.3 Memory Protection Unit (MPU)

The AS32I401 provides users with an MPU to protect memory/peripherals at specific addresses. MPU can be used to prevent programs from destroying mission-critical (such as operating system kernel) data, lock SRAM memory areas (read protection + write protection) to prevent code injection attacks and other operations to protect the entire embedded system, thereby improving system robustness and security.

- Programmable protection widths ranging from 32B to 32KB;
- 16 independent memory protection channels. The protection areas of each channel can be superimposed.
- single period fast exception response;
- locks the data channel upon hit to ensure that no unauthorized access occurs.
- Supports three protection modes: read protection, write protection, or area locked (read protection + write protection).

2.4 Interrupt

The AS32I401 supports 32 interrupt source inputs and supports both software and hardware save and recovery modes. It can be controlled by hardware save and recovery enable bit, and supports a maximum of 8 levels of hardware save and recovery. Interrupts include peripherals, CSR software interrupts, and CSR timer interrupts. The interrupt entry is in vector mode, and each interrupt request has its own user-configured entry function address. PLIC is the interrupt controller, which supports the enable and priority setting of interrupts.

- 32 interrupt sources are supported. Each interrupt source is assigned an ID. ID 0 indicates no interrupt.

- Priority Supports software Settings, ranging from 0 to 31 levels, and can be set to the same priority. Level 0 is the highest. If the priority is the same, the priority with the smaller ID is executed first.
- Output is also a single-period pulse signal, the same time only one highest priority interrupt output, execute interrupt function does not respond to the same interrupt;
- All interrupt sources are triggered by rising edge;
- The interrupt is in vector mode (hardware interrupt mode);
- Supports interrupt nesting (not supported when saving or recovering a scene);
- Each interrupt source has its own enable bit. This is different from the interface private disable bit. Both types of interrupt bits need to be enabled.
- Supports interrupt priority threshold setting.

2.5 Clock and Reset Management

The AS32I401 is clocked by a 20MHz passive crystal oscillator. The clock source is used as a pair of mutual backup PLLS, which can automatically switch to the other PLL when one of them is damaged, and its maximum frequency can be doubled to 240MHz.

The pre-divider allows the CPU and AHB clock domains to use up to 120MHz clock frequency, the high-speed APB domain (APB1) to use up to 120MHz clock frequency, and the low-speed APB domain (APB0) to use up to 60MHz clock frequency.

There are four control reset modes for the AS32I401: external reset, watchdog reset, debug unit reset, and SMU control reset. The external reset is reset by an external control signal to reset the entire system, the watchdog reset acts on all parts of the system except the SMU, while the debug unit reset only acts on the CPU, the SMU control reset programmable control reset all parts except itself.

2.6 Startup Mode

The AS32I401 provides three BOOT sources, which can be selected through boot Settings:

- Start from the primary QSPI Flash.
- Start from the backup QSPI Flash.
- Boot from the internal SRAM.

2.7 Direct Memory Access Controller (DMA)

DMA controllers provide a hardware way to transfer data between peripherals and memory, or between memory and memory, without the involvement of the CPU, so that the CPU can focus on processing other system functions. The DMA controller has 16 channels (DMA0 and DMA1 each have 8 channels). Each channel is dedicated to processing memory access requests from one or more peripherals. An arbitrator is implemented inside the DMA controller to arbitrate multiple DMA requests.

DMA controllers use separate peripheral buses. In most cases DMA can access the same address space at the same time as the CPU. Only when DMA and CPU access the same peripheral, DMA access may block CPU access to the system bus for a few bus cycles. A circular arbitration algorithm is implemented in the peripheral bus matrix to allocate DMA and CPU access rights, which can ensure that the CPU gets at least half of the peripheral bandwidth.

Peripherals that DMA can use: UART, IIC, SPI, ADC, DAC.

2.8 General Purpose Input/Output Interface (GPIO)

The AS32I401 supports up to 80 general purpose I/O pins (GPIOs), PA0 to PA15, PB0 to PB15, PC0 to PC15, PD0 to PD15, and PE0 to PE15, which are used by each on-chip device to implement logical input/output functions. Each GPIO port has associated control and configuration registers to meet the needs of the specific application. Peripheral GPIO pins have their own independent interrupt control.

GPIO ports share pins with other spare functions (Afs) for maximum flexibility in specific packages. The GPIO pin can be used as an alternate function pin by configuring the relevant register, either input or output. Each GPIO pin can be configured by software for output (push-pull or leak-open), input, peripheral backup function, or analog mode. Each GPIO pin can be configured to pull up, pull down, or pull down/pull down.

2.9 Universal Asynchronous Transceiver (UART)

The AS32I401 provides four universal asynchronous transceivers (UART) for flexible and convenient serial data exchange, and data frames can be transmitted through full-duplex asynchronous mode.

The UART provides a programmable baud rate generator that splits the system clock to produce the specific frequencies required by the UART transmitter and receiver.

UART supports the DMA function, which can realize high-speed data communication and reduce the burden on the CPU.

- NRZ standard format
- full-duplex asynchronous communication
- half-duplex single-wire communication
- Programmable baud rate generator
- The clock is generated by frequency division of the APB0 clock
- Adjustable oversampling rate
- Fully programmable serial port features
- Generation/detection of even, odd and no check bits
- Data bits (7 or 8 bits)
- 256-bit read/write FIFO
- DMA accesses the data buffer

- Transmission detection flags: Receive buffer not empty (RXNE), send buffer empty (TXE), send completed (TC), receive completed (IDLE)
- Error detection flags: Overload error (ORES), Frame Format Error (FERS), Parity error (PERS)

2.10 Internal Integrated Circuit Bus Interface (IIC)

The AS32I401 provides four IIC bus interfaces with an industry-standard two-wire serial interface for communication between MCU and external IIC devices. The IIC bus uses two serial lines: the serial data line SDA and the serial clock line SCL.

The IIC bus interface supports the master-slave mode, and two-way data transmission can be carried out between the master and slave. The IIC bus interface provides a transfer rate of up to 100KHz in standard mode and up to 400KHz in fast mode, supporting PMBus (Power Management Bus).

IIC interface module supports DMA mode, which can effectively reduce the burden of CPU.

- The same interface can realize both host and slave functions
- Two-way data transmission between master and slave machines
- Supports 7-bit address addressing mode
- Supports IIC multi-host mode
- Configurable SCL active pull down in slave mode
- Support DMA mode
- There are three types of interrupts: send interrupts, receive interrupts, and transfer step interrupts

2.11 Serial Peripheral Interface (SPI)

The AS32I401 provides four SPI interfaces and supports half duplex/full duplex synchronous serial communication. The SPI interface supports master/slave mode, which can be configured as host or slave mode. When configured in host mode, it can provide communication clock (SCK) for external slave devices, SPI2 and SPI3 interfaces support 4 slave machines, with 4 slave chip options, SPI0 and SPI1 support 1 slave machine.

The SPI interface supports DMA.

- Support for Motorola Mode Serial Peripheral Interface (SPI)
- 4-wire full-duplex synchronous transmission (3-wire half-duplex)
- Select from 4 to 32 bits transmission frame format
- The SPI clock rate is configurable
- Master mode or slave mode operation
- Maximum frequency PCLK/2 in master mode

- Maximum frequency PCLK/8 when in slave mode
- Data sending sequence MSB is first
- Programmable clock polarity and phase
- Dedicated send and receive flags that trigger interrupts

2.12 Local Area Internet Network (LIN)

The AS32I401 supports one LIN interface.

LIN is a low-cost serial communication protocol based on UART/SCI, which can be used in many fields such as automobiles, home appliances, and office equipment. The network consists of a host Node and multiple Slave nodes, the Master Node contains the Master Task and the Slave Task, and the Slave Node contains only the slave task. The LIN protocol specifies a maximum bit rate of 20kbps. Due to physical layer restrictions, a LIN network can connect up to 16 nodes, typical applications are generally less than 12 nodes, host nodes have and only one, slave node has 1 to 15.

- Host write (send header and reply frame)
- Host read (frame header only)
- Slave write
- The slave task sends or receives the data terminal and checksum
- Synchronization interval end and synchronization segment detection, frame header detection
- Programmable baud rate and automatic baud rate detection
- The number of bytes in the data field ranges from 1 to 8 bytes
- Standard check and enhanced check Settings
- Sync interval segment width, interval delimiter width, reply interval, byte spacing, frame interval width Settings
- Bus error detection
- P1 and P0 check errors
- Checksum error
- Response error detection (frame response interrupted by new frame)
- Interrupt output
- The bus receives digital filtering
- P1 and P0 check bits are automatically calculated for sending and receiving
- Send and receive checksums are calculated automatically

2.13 TIMER

The AS32I401 provides three sets of advanced timers. The advanced timer includes a 16-bit automatic overload counter that is driven by a programmable pre-divider and supports multiple count modes. The timer is a 6-channel timer, which supports input capture, output comparison and single pulse mode, and can measure pulse width of input signal and generate PWM waveform. The main features of timers are as follows:

- Total number of channels: 6;
- Automatic overload counter with 16-bit increasing count, decreasing count and center count;
- 16-bit programmable pre-divider, used to divide the frequency of the counter clock, can be modified in the run, the frequency division coefficient range is 1~65536;
- Automatic reloading function;
- Support single pulse mode;
- Each channel can be configured: input capture mode, output comparison mode, programmable PWM mode.

2.14 Ethernet (ETH: Media Access Control MAC through a DMA controller)

The AS32I401 provides one 10M/100M Ethernet interface. The MAC interface implements the Media Access Control (MAC) over Ethernet connection collision detection (CSMA/CD) algorithm defined by IEEE 802.3. Communication with the external host is achieved through a set of control and status registers and a direct Memory access (DMA) controller for the external shared RAM. The MAC interface implements data transmission with the host through DMA. It automatically gets the send buffer data and stores the receive buffer data to the external RAM. It implements a variety of memory allocation schemes by managing the list of receive and send descriptors. The MAC interface internal RAM 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;
- Communicate with an external fast Ethernet PHY via an IEEE 802.3-compliant MII interface;
- Support full duplex and half duplex operation;
- Header and frame start data (SFD) are inserted in the send path and deleted in the receive path;
- CRC and PAD generation can be controlled frame-by-frame;
- PAD/CRC is automatically removed when receiving frames;
- Supports a variety of flexible address filtering modes;
- Two sets of cache FIFOs (receiving FIFO and sending FIFO);
- DMA moves data between memory and the MAC's cache.

2.15 Controller Area Network (CAN)

The AS32I401 provides two CAN ports. The CAN interface communicates according to ISO11898-1:2015 and Bosch CAN FD specifications. Connecting to the physical layer requires additional transceiver hardware.

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

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

- The design conforms to ISO11898-1/2015 specification;
- Supports CAN and CAN FD frames;
- Supports the CAN FD frame format specified in ISO11899:2015 specification;
- Support 64 bytes CAN FD frame;
- Supports variable data rates up to 8Mb/s;
- Support normal data rate of 1Mb/s;
- Supports transmitter delay compensation of up to three data bits;
- Supports configurable send and receive mailbox buffers;
- Supports two 64-frame message buffers and 32 ID filtering masks;
- Messages with low ids are preferentially sent.
- Support for outgoing message elimination;
- Individual error logging for fast data rates.

2.16 Cyclic Redundancy Check Computing Unit (CRC)

A CRC computing unit is integrated into the AS32I401. Cyclic Redundancy Check (CRC) is a channel coding technique that generates short, fixed-bit check codes based on data such as network packets or computer files to detect or verify possible errors after data transmission or storage. It uses the principle of division and remainder for error detection.

CRC (Cyclic redundancy Check) cells use a polynomial generator to generate CRC codes from 8-bit / 16-bit / 32-bit data words. The main features of CRC computing unit are as follows:

- Use fully programmable polynomials with pre-set bits (7, 8, 16 and 32 bits);
- Can handle 8-bit, 16-bit, 32-bit data size;
- Programmable CRC initial value;
- Single input/output 32-bit data register;

- Input buffer prevents bus blocking during computation;
- Reversible options for I/O data;
- XOR computation is supported.

2.17 Orthogonal Encoder Interface (QEI)

The AS32I401 provides two QEI interfaces. The Orthogonal Encoder Interface (QEI) module provides an interface for incremental encoders to obtain mechanical position data. Orthogonal encoders (also known as incremental encoders or photoelectric encoders) are used to detect the position and speed of a rotating motion system. Orthogonal encoders can be used to achieve closed-loop control for a variety of Motor control applications, such as Switched Reluctance (SR) motors and AC Induction Motor (ACIM).

- Two phase signals (QEA and QEB) and one index signal (INDX)
- Motion direction detection with directional change interruption
- Programmable QEA, QEB and INDX input noise filters
- 32-bit speed and position register
- Two location update modes (x2 and x4)
- Speed measurement and high-speed measurement after programmable pulse frequency division
- Position counter interrupt
- Speed update interrupt

2.18 Analog-to-Digital Converter (ADC)

The AS32I401 offers two 12-bit analog/digital converters (ADCs) with 16 multiplexed channels that can convert analog signals from multiple external channels. The analog watchdog allows the application to detect if the input voltage exceeds a user-set high-low threshold. A/D conversion of various channels can be configured in single, continuous, scan or intermittent conversion mode. The result of the ADC conversion can be stored in a 16-bit data register either left-justified or right-justified. Adcs support the use of DMA operations.

2.19 Digital-to-Analog Converter (DAC)

The AS32I401 offers two 12-bit analog/digital converters (Dacs). The DAC can convert 12-bit digital data into a voltage output on an external pin. When externally triggered is enabled, DMA can be used to update input digital data. At output voltage, the DAC output buffer can be utilized to obtain higher drive capability. The two Dacs can work independently or concurrently. DAC supports the following functions:

- 12-bit resolution, data aligned left or right;
- Synchronous update conversion;
- External events trigger transitions;

- Input reference voltage, VREF+;
- Noise wave generation (LSFR noise mode and triangle noise mode);
- Support DMA operation;
- Dual DAC concurrent mode.

2.20 Watchdog (WDT)

A watchdog timer (WDT) is used to prevent system locking when software gets stuck in a deadlock. It features a 32-bit downward counter that allows external independent clock input, and a 12-bit pre-divider. It can only generate a general reset or processor reset. In addition, it can be stopped when the processor is in debug mode or sleep mode (idle mode).

- Support 32-bit watchdog timer;
- Preset divider, divider value up to 1024;
- Support independent external clock input 20MHz;
- Support external input reset (feed the dog).

2.21 1553B Controller (MIL-STD-1553B)

The AS32I401 supports two 1553B bus controllers, 1553B0 and 1553B1.

The 1553B bus adopts asynchronous data transmission mode, the code rate is 1Mbps, the data encoding adopts Manchester II code, differential transmission, and generally adopts shielded twisted pair wire as the transmission medium.

- Supports three modes: BC, RT, and BM
- The input clock is 20M, and the internal generated send clock is 1M
- Register configuration mode
- Interrupt logging function
- Programmable message delay
- Cyclic cache mode
- The dedicated memory is 16k bytes in size and is used to store message lists, data, and interrupt log lists

3 Memory address mapping

Table 3.1 AS32I401 memory address mapping table

Content	Address	Size
FLASH	0x0000_0000~0x0FFF_FFFF	256MB
BK_FLASH	0x5000_0000~0x5FFF_FFFF	256MB
RAM	0x1000_0000~0x1FFF_FFFF	256MB
SRAM0	0x1000_0000~0x1000_7FFF	32KB
SRAM1	0x1000_8000~0x1000_FFFF	32KB
SRAM2	0x1001_0000~0x1001_7FFF	32KB
SRAM3	0x1001_8000~0x1001_FFFF	32KB
SMU	0x2000_0000~0x21FF_FFFF	32MB
DMA0	0x2200_0000~0x22FF_FFFF	16MB
DMA1	0x2300_0000~0x23FF_FFFF	16MB
MPU	0x2400_0000~0x24FF_FFFF	16MB
ICACHE		32KB
PLIC	0x2500_0000~0x25FF_FFFF	16MB
DTCM	0x2600_0000~0x2600_7FFF	32KB
UART0	0x3000_0000~0x30FF_FFFF	16MB
UART1	0x3100_0000~0x31FF_FFFF	16MB
UART2	0x3200_0000~0x32FF_FFFF	16MB
UART3	0x3300_0000~0x33FF_FFFF	16MB
SPI0	0x3400_0000~0x34FF_FFFF	16MB
SPI1	0x3500_0000~0x35FF_FFFF	16MB
TIM2	0x3600_0000~0x36FF_FFFF	16MB
MAC	0x3700_0000~0x37FF_FFFF	16MB
IIC0	0x3800_0000~0x38FF_FFFF	16MB
IIC1	0x3900_0000~0x39FF_FFFF	16MB
IIC2	0x3A00_0000~0x3AFF_FFFF	16MB
IIC3	0x3B00_0000~0x3BFF_FFFF	16MB
LIN	0x3C00_0000~0x3CFF_FFFF	16MB
MIL-STD-1553B BUS 0	0x3D00_0000~0x3DFF_FFFF	16MB
MIL-STD-1553B BUS 1	0x3E00_0000~0x3EFF_FFFF	16M
WATCHDOG	0x3F00_0000~0x3FFF_FFFF	16M

Content	Address	Size
SPI2	0x4000_0000~0x40FF_FFFF	16M
SPI3	0x4100_0000~0x41FF_FFFF	16M
CAN FD0	0x4200_0000~0x42FF_FFFF	16M
CAN FD1	0x4300_0000~0x43FF_FFFF	16M
ADC0	0x4400_0000~0x44FF_FFFF	16M
ADC1	0x4500_0000~0x45FF_FFFF	16M
GPIOA	0x4600_0000~0x46FF_FFFF	16M
GPIOB	0x4700_0000~0x47FF_FFFF	16M
GPIOC	0x4800_0000~0x48FF_FFFF	16M
GPIOD	0x4900_0000~0x49FF_FFFF	16M
GPIOE	0x4A00_0000~0x4AFF_FFFF	16M
CRC	0x4B00_0000~0x4BFF_FFFF	16M
TIM0	0x4C00_0000~0x4CFF_FFFF	16M
TIM1	0x4D00_0000~0x4DFF_FFFF	16M
DAC	0x4E00_0000~0x4EFF_FFFF	16M
QEI	0x4F00_0000~0x4FFF_FFFF	16M

4 Pin description

Table 4.1 Pin definition of AS32I401

Pin	Pin name	Pin type	IO	Alternate functions 0	Alternate functions 1	Additional functions
1	PE2	I/O	FT	IIC0_SCL	SPI1_CS	—
2	PE3	I/O	FT	IIC0_SDA	SPI1_SCK	—
3	PE4	I/O	FT	TIM2_CH5	SPI1_MISO	—
4	PE5	I/O	FT	TIM2_CH6	SPI1_MOSI	—
5	PE6	I/O	FT	TIM2_CH1	—	—
6	Vbat	S	—	—	—	—
7	PC13	I/O	FT	TIM2_CH2	—	—
8	PC14	I/O	FT	TIM2_CH3	—	—
9	PC15	I/O	FT	TIM2_CH4	—	—
10	Vss	S	—	—	—	—
11	Vdd	S	—	—	—	—
12	OSC_IN	I	—	—	—	—
13	OSC_OUT	I	—	—	—	—
14	NRST	I	RST	—	—	—
15	PC0	I/O	TTa	—	—	ADC12_IN10
16	PC1	I/O	TTa	—	—	ADC12_IN11
17	PC2	I/O	TTa	ADC1_RTRG	ADC2_RTRG	ADC12_IN12
18	PC3	I/O	TTa	ADC1_LTRG	ADC2_LTRG	ADC12_IN13
19	Vdd	S	—	—	—	—
20	Vssa	S	—	—	—	—
21	Vref+	S	—	—	—	—
22	Vdda	S	—	—	—	—
23	PA0	I/O	TTa	DAC_TRG1	—	ADC12_IN0/WKUP
24	PA1	I/O	TTa	DAC_TRG2	—	ADC12_IN1
25	PA2	I/O	TTa	ETH_TXEN	—	ADC12_IN2
26	PA3	I/O	TTa	ETH_TXER	—	ADC12_IN3
27	Vss	S	—	—	—	—
28	Vdd	S	—	—	—	—
29	PA4	I/O	TTa	—	—	ADC1_IN14/DAC_OUT 1
30	PA5	I/O	TTa	—	—	ADC1_IN15/DAC_OUT 2
31	PA6	I/O	TTa	ETH_RXDV	—	ADC12_IN4
32	PA7	I/O	TTa	ETH_TXCLK	—	ADC12_IN5
33	PC4	I/O	TTa	ETH_RXCRS	—	ADC12_IN6
34	PC5	I/O	TTa	ETH_RXCLK	—	ADC12_IN7

Pin	Pin name	Pin type	IO	Alternate functions 0	Alternate functions 1	Additional functions
35	PB0	I/O	TTa	ETH_RXCOL	—	ADC12_IN8
36	PB1	I/O	TTa	ETH_RXER	—	ADC12_IN9
37	PB2	I/O	FT	—	—	BOOT1
38	PE7	I/O	FT	ETH_RXD0	CAN1_RX	—
39	PE8	I/O	FT	ETH_RXD1/CAN1_TX	—	—
40	PE9	I/O	FT	ETH_RXD2	UART0_RX	—
41	PE10	I/O	FT	ETH_RXD3/UART0_TX	—	—
42	PE11	I/O	FT	ETH_MDC/UART1_RX	—	—
43	PE12	I/O	FT	ETH_MDIO	UART1_TX	—
44	PE13	I/O	FT	ETH_TXD3/QEI0_A	—	—
45	PE14	I/O	FT	ETH_TXD2/QEI0_B	—	—
46	PE15	I/O	FT	ETH_TXD1/QEI0_ID	—	—
47	PB10	I/O	FT	ETH_TXD0	—	—
48	PB11	I/O	FT	LIN	—	—
49	Vssc	S	-	—	—	—
50	Vdd	S	-	—	—	—
51	PB12	I/O	FT	ETH_SCLK	UART2_TX	—
52	PB13	I/O	FT	ETH_SCS/UART2_RX	—	—
53	PB14	I/O	FT	ETH_SDI	—	—
54	PB15	I/O	FT	ETH_SDO	—	—
55	PD8	I/O	FT	SPI3_CS	1553B1_A_IN_EN	—
56	PD9	I/O	FT	SPI3_SCK	1553B1_A_IN_P	—
57	PD10	I/O	FT	SPI3_MOSI	1553B1_A_IN_N	—
58	PD11	I/O	FT	SPI3_MISO	1553B1_B_IN_EN	—
59	PD12	I/O	FT	SPI3_MCS1/1553B1_B_IN_P	—	—
60	PD13	I/O	FT	SPI3_MCS2/1553B1_B_IN_N	—	—
61	PD14	I/O	FT	SPI3_MCS3	1553B1_A_OUT_P	—
62	PD15	I/O	FT	QEI1_A/1553B1_A_OUT_N	—	—
63	PC6	I/O	FT	QEI1_B/1553B1_B_OUT_P	—	—
64	PC7	I/O	FT	QEI1_ID/1553B1_B_OUT_N	—	—
65	PC8	I/O	FT	IIC3_SCL	1553B1_A_OUTIN	—
66	PC9	I/O	FT	IIC3_SDA	1553B1_B_OUTIN	—
67	PA8	I/O	FT	TIM1_CH1	—	—
68	PA9	I/O	FT	TIM1_CH2	—	—
69	PA10	I/O	FT	TIM1_CH3	—	—
70	PA11	I/O	FT	TIM1_CH4	—	—
71	PA12	I/O	FT	TIM1_CH5	—	—

Pin	Pin name	Pin type	IO	Alternate functions 0	Alternate functions 1	Additional functions
72	PA13	I/O	FT	TIM1_CH6	—	—
73	Vddc	S	—	—	—	—
74	Vss	S	—	—	—	—
75	Vdd	S	—	—	—	—
76	PA14	I/O	FT	DBG_RX	—	—
77	PA15	I/O	FT	DBG_TX	—	—
78	PC10	I/O	FT	SPI0_CS	1553B0_A_IN_EN	—
79	PC11	I/O	FT	SPI0_SCK	1553B0_A_IN_P	—
80	PC12	I/O	FT	SPI0_MISO	1553B0_A_IN_N	—
81	PD0	I/O	FT	SPI0_MOSI	1553B0_B_IN_EN	—
82	PD1	I/O	FT	UART3_TX/1553B0_B_IN_P	-	—
83	PD2	I/O	FT	UART3_RX	1553B0_B_IN_N	—
84	PD3	I/O	FT	IIC1_SCL	1553B0_A_OUT_P	—
85	PD4	I/O	FT	IIC1_SDA	1553B0_A_OUT_N	—
86	PD5	I/O	FT	SPI2_CS	1553B0_B_OUT_P	—
87	PD6	I/O	FT	SPI2_SCK	1553B0_B_OUT_N	—
88	PD7	I/O	FT	SPI2_MISO	1553B0_A_OUTIN	—
89	PB3	I/O	FT	SPI2_MOSI	1553B0_B_OUTIN	—
90	PB4	I/O	FT	TIM0_CH1	SPI2_MCS1	—
91	PB5	I/O	FT	TIM0_CH2	SPI2_MCS2	—
92	PB6	I/O	FT	TIM0_CH3	SPI2_MCS3	—
93	PB7	I/O	FT	TIM0_CH4	—	—
94	BOOT0	I	B	—	—	—
95	PB8	I/O	FT	TIM0_CH5	—	—
96	PB9	I/O	FT	TIM0_CH6	—	—
97	PE0	I/O	FT	IIC2_SCL	CAN2_RX	—
98	PE1	I/O	FT	IIC2_SDA	CAN2_TX	—
99	Vss	S	—	—	—	—
100	Vdd	S	—	—	—	—

5 Historical document versions

Date	Version	Modification
2023.3.23	1.00	Original version