DUE-CORE
DUE-CORE Datasheet
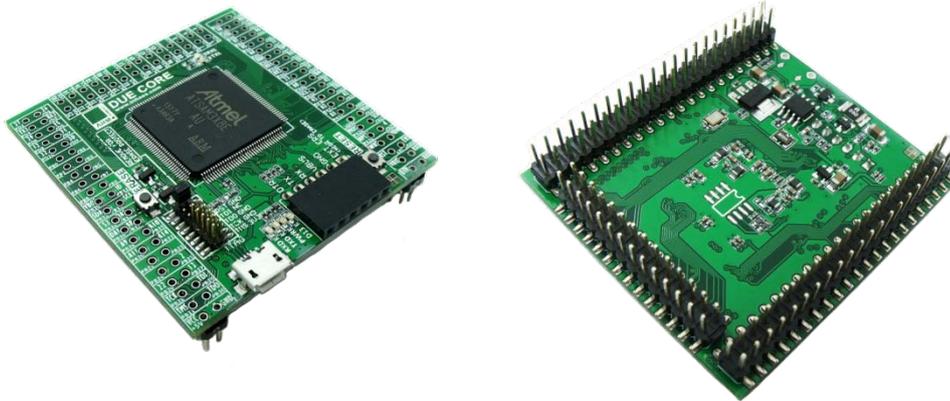
INHAOS™
Everything done inhouse

1   Summary

The Arduino Due is a microcontroller board based on the Atmel SAM3X8E ARM Cortex-M3 CPU. It is the first Arduino board based on a 32-bit ARM Cortex-M3 core microcontroller.

DUE-CORE is a compact version of DUE , It integrates all peripherals required for MCU , and all GPIO are connected to the 2.54mm connectors for users. As a standard MCU core board, apart from main features with DUE also has the following features:

- **Compact size:** All components are put on a 54 x 58mm 4-Layers PCB. All IO are connected to 116pin 2.54 standard connector. User can apply the PCBA to any application even the size is very compact.
- **Easy to use:** All IO are connected to 116pin 2.54 standard connector. Only a 5V power to the board to make it work.
- **Stable design:** adopt high-cost 4-layer PCB layout , two 5V to 3.3V LDO onboard , one is for digital part and one is for analog part. Separate AVCC and AGND, to ensure the best analog performance.
- **Easy to set up the development environment:** Fully development on Arduino IDE , uploading sketch through standard Arduino 6pin UART interface, standard Micro usb connector , full use of existing resources .
- **User-friendly design:** Rich LED status indication, two onboard buttons, one is for MCU reset , and other one is for Flase Erase. **Unique jumper erase protection against the flash erased by mistake operation.**
- **Rich resource :** All IO are connected out for the user, even if not defined Arduino. Onboard I2C EEPROM designed to compensate for the shortcomings of SAM3X8E no built-in EEPROM.

2   Parameters

Microcontroller:	AT91SAM3X8E LQFP-144
Operation Voltage:	3.3V
Input voltage:	5V
Total connector pins:	44 + 44 + 28 = 116Pins
Flash:	2 x 256 Kbytes
SRAM:	64 + 32 Kbytes
Clock Speed:	84 MHz
Number of PIOs:	103
NAND Flash Controller (NFC):	Yes
SHDN Pin:	Yes
EMAC:	MII/RMII
External Bus Interface:	16-bit data / 8 chip selects / 23-bit address
Central DMA:	6
12-bit ADC:	16 ch
12-bit DAC:	2 ch

|	
32-bit Timer:	9 ch
PDC Channels:	17
USART/UART:	3 / 2 (USART3 in UART mode (RXD3 and TXD3 available))
SPI:	1 SPI controller,4 chip selects +3 USART with SPI mode
HSMCI:	1 slot, 8 bits
Clock Speed:	84 MHz
Board dimension:	54 x 58mm
Arduino resource support:	https://www.arduino.cc/en/Main/ArduinoBoardDue

3   Pin Description

< TOP VIEW >

DUE-CORE
DUE-CORE Datasheet

INHAOS™
Everything done inhouse
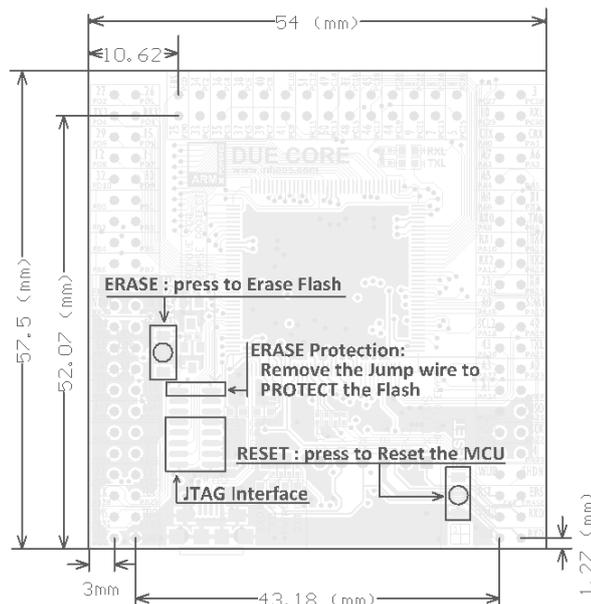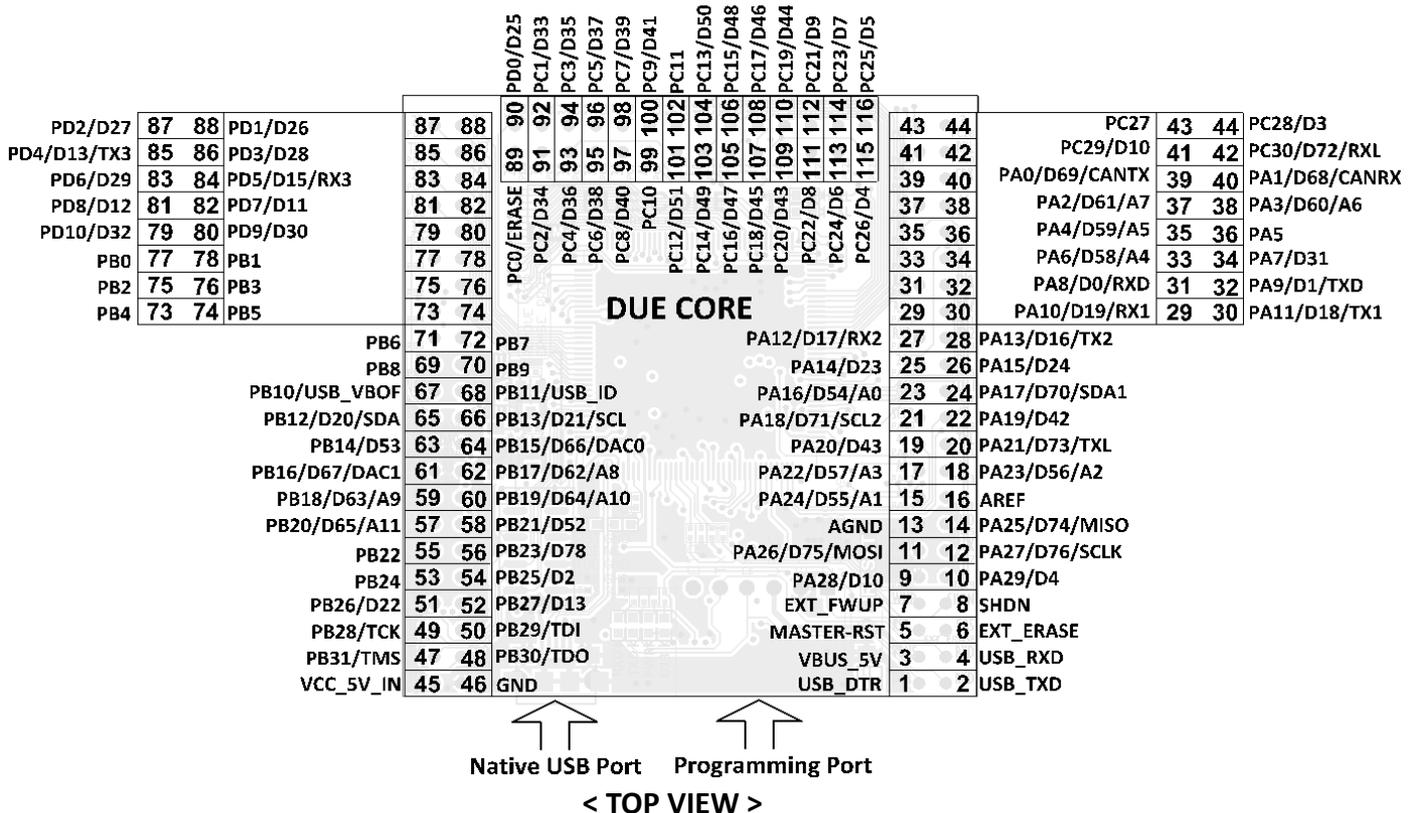
| Pin No.	MCU Pin Function	Arduino Pin Fuction	Pin No.	MCU Pin Function	Arduino Pin Fuction
1	USB DTR (Pull low this pin will reset the MCU)		59	PB18/RD/PWML2/AD11	D63/A9
2	USB TXD (USB UART bridge's RXD <-> MCU PA8/RXD0)		60	PB19/RK/PWML3/AD12	D64/A10
3	VBUS_5V (USB 5V)		61	PB16/TCLK5/PWML0/DAC1	D67/DAC1
4	USB RXD (USB UART bridge's RXD <-> MCU PA9/TXD0)		62	PB17/RF/PWML1/AD10	D62/A8
5	Master-Reset (MCU Reset output signal for periphery device)		63	PB14/CANTX1/PWMH2	D53
6	EXT_Erase (Pull High this pin for Erase the Flash of MCU)		64	PB15/CANRX1/PWMH3/DAC0/WKUP12	D66/DAC0
7	EXT_FWUP	---	65	PB12/TWD1/PWHH0/AD8	D20/SDA
8	SHDN	---	66	PB13/TWCK1/PWMH1/AD9	D21/SCL
9	PA28/SPI0_NPCS0/PCK2/WKUP11	D10	67	PB10/UOTGVBOF/A18	USB_VBOF
10	PA29/SPI0_NPCS1/NRD	D4	68	PB11/UOTGID/A19	USB_ID
11	PA26/SPI0_MOSI/A19	D75/MOSI	69	PB8/EMDC/PWML6	---
12	PA27/SPI0_SCPK_A20_WKUP10	D76/SCLK	70	PB9/DMDIO/PWML7	---
13	AGND		71	PB6/ERX1/PWML4	---
14	PA25/SPI0_MISO/A18	D74/MISO	72	PB7/ERXER/PWML5	---
15	PA24/MCDA3/PCK1/AD6	D55/A1	73	PB4/ECRSDV/ERXDV/TIOA5	---
16	AREF		74	PB5/ERX0/TIOB5	---
17	PA22/MCDA1/TCLK3/AD5	D57/A3	75	PB2/EXT0/TIOA4	---
18	PA23/MCDA2/TCLK4/AD5	D56/A2	76	PB3/EXT1/TIOB4	---
19	PA20/MCCDA/PWML2	D43	77	PB0/ETXCK/EREFCK/TIOA3	---
20	PA21/MCDA0/PWML0	D73/TXL	78	PB1/EXTEN/TIOB3	---
21	PA18/TWCK0/A20/WKUP9	D71/SCL2	79	PD10/NWR1/NBS1	D32
22	PA19/MCCK/PWMH1	D42	80	PD9/A22/NANDCLE/TCLK8	D30
23	PA16/SPCK1/TD/AD7	D54/A0	81	PD8/A21/NANDALE/TIOB8	D12
24	PA17/TWD0/SPCK0	D70/SDA1	82	PD7/A17/BA1/TIOA8	D11
25	PA14/RTS1/TK	D23	83	PD6/A16/BA0/PWMF12	D29
26	PA15/CTS1/TF/WKUP8	D24	84	PD5/A15/RXD3	D13/RX3
27	PA12/RXD1/PWML1/WKUP7	D17/RX2	85	PD4/A14/TXD3	D13/TX3
28	PA13/TXD1/PWMH2	D16/TX2	86	PD3/A13/MCDA7	D28
29	PA10/RXD0/DATRG/WKUP5	D19/RX1	87	PD2/A12/MCDA6	D27
30	PA11/TXD0/ADTRG/WKUP6	D18/TX1	88	PD1/A11/MCDA5	D26
31	PA8/URXD/PWMH0/WKUP4	D0/RXD	89	PC0/ERASE	ERASE
32	PA9/UTXD/PWMH3	D1/TXD	90	PD0/A10/MCDA4	D25
33	PA6/TIOB2/NCS0/AD3	D58/A4	91	PC2/D0/PWML0	D34
34	PA7/TCLK2/NCS1/WKUP3	D31	92	PC1	D33
35	PA4/TCLK1/NWAIT/AD2	D59/A5	93	PC4/D2/PWML1	D36
36	PA5/TIOA2/PWMFI0/WKUP2	---	94	PC3/D1/PWMH0	D35
37	PA2/TIOA1/NANDRDY/AD0	D61/A7	95	PC6/D4/PWML2	D38
38	PA3/TIOB1/PWMFI1/AD1/WKUP1	D60/A6	96	PC5/D3/PWMH1	D37
39	PA0/CANTX0/PWML3	D69/CANTX	97	PC8/D6/PWML3	D40
40	PA1/CANRX0/PCK0/WKUP0	D68/CANRX	98	PC7/D5/PWMH2	D39
41	PC29/A8/TIOB7	D10	99	PC10/D8/ECRS	---
42	PC30/A9/TCLK7	D72/RXL	100	PC9/D7/PWMH3	D41
43	PC27/A6/TCLK6	---	101	PC12/D10/ERX3	D51
44	PC28/A7/TIOA7	D3	102	PC11/D9/ERX2	---
45	VCC_5V_IN (Module power supply) 4.75~5.25V		103	PC14/D12/ERXCK	D49
46	GND		104	PC13/D11/ECOL	D50
47	PB31/TMS/SWDIO	TMS	105	PC16/D14/ETX3	D47
48	PB30/TDO/TRACESWO	TDO	106	PC15/D13/ETX2	D48
49	PB28/TCK/SWCLK	TCK	107	PC18/NWR0/NWE/PWMH6	D45
50	PB29/TDI	TDI	108	PC17/D15/ETXER	D46
51	PB26/CTS0/TCLK0/WKUP15	D22	109	PC20/NANDWE/PWMH4	D43
52	PB27/NCS3/TIOB0	D13	110	PC19/NANDOE/PWMH5	D44
53	PB24/SCK2/NCS2	---	111	PC22/A1/PWML5	D8
54	PB25/RTS0/TIOA0	D2	112	PC21/A0/NBS0/PWML4	D9
55	PB22/RTS2/PCK0	---	113	PC24/A3/PWML7	D6
56	PB23/CTS2/SPI0_NPCS3/WKUP14	D78	114	PC23/A2/PWML6	D7
57	PB20/TXD2/SPI0_NPCS1/AD13	D65/A11	115	PC26/A5/TIOB6	D4
58	PB21/RXD2/SPI0_NPCS2/AD14/WKUP13	D52	116	PC25/A4/TIOA6	D5

Notes:

- D4 connected to both PA29 andPC26
- D10 connected to both PA28 andPC29
- The board can be powered by Pin45/46 VCC_5V_IN and GND , also can be powered by USB or Programming port.
- If VCC_5V_IN and USB or Programming port are connected at the same time , the board will VCC_5V_IN is priority.
- The VCC_5V_IN must limit tolerance within +/-5% , which means the voltage range is 4.75-5.25V
- The GND(Pin-46) and AGND(Pin-13) are connected by onboard 0R resistor.

4   Typical application

Power Connection 1: powered by Programming port, in this case , the module can be powered by VBUS_5V form the USB to UART cable (Recommend UC-2102). The connection as below :
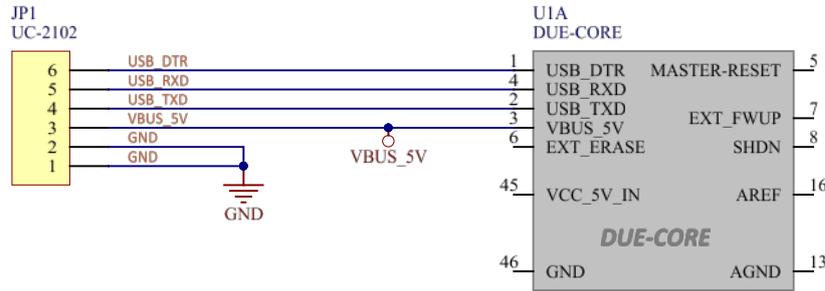
Power Connection 2: powered by VCC_IN_5V, in this case , The VCC_5V_IN must limit tolerance within +/-5% , which means the voltage range is 4.75-5.25V. The connection as below :
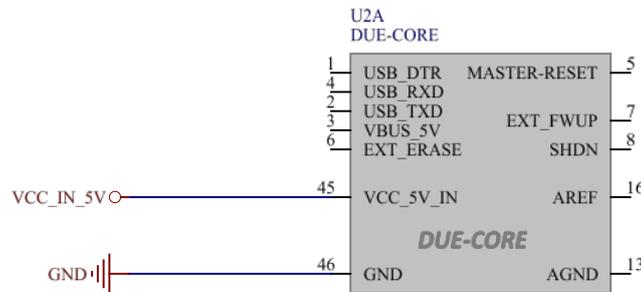
Power Connection 3: Both Programming port and VCC_5V_IN are connected , In this case , the VCC_IN_5V is priority, The connection as below :
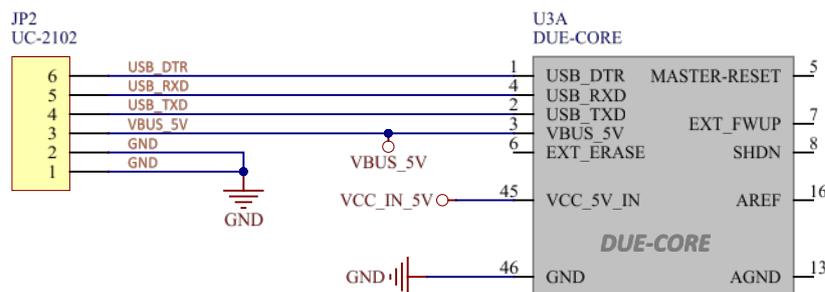
NOTE: THE POWER SUPPLY TO THE MODULE IS 5V , BUT THE MCU IS WOKING UNDER 3.3V , THE ONBOARD LDO WILL REGULATE THE 5V POWER TO 3.3V.

5 Uploading Sketch

The AT91SAM3X8E is already build in bootlaoder , user can be uploading sketch by two ways with Arduino IDE:

Native port: To use this port, select "Arduino Due (Native USB Port)" as your board in the Arduino IDE. The Native USB port is connected directly to the SAM3X. Connect the Due's Native USB port to your computer. Opening and closing the Native port at 1200bps triggers a 'soft erase' procedure: the flash memory is erased and the board is restarted with the bootloader. If the MCU crashed for some reason it is likely that the soft erase procedure won't work as this procedure happens entirely in software on the SAM3X. Opening and closing the native port at a different baudrate will not reset the SAM3X.

Programming port: To use this port, select "Arduino Due (Programming Port)" as your board in the Arduino IDE. Connect the Due's programming port to your computer through a standard USB to Serial convertor (UC-2102). The UC-2102 use CP-2102 as a USB-to-serial chip connected to the first UART of the SAM3X (RX0 and TX0). The UC-2102 DTR pin is contorl the Reset pins of the SAM3X. Press the "ERASE button" on the module will triggers a "hard erase" procedure of the SAM3X chip, activating the Erase pins on the SAM3X before communicating with the UART. This is the recommended port for programming the Due. It is more reliable than the "soft erase" that occurs on the Native port, and it should work even if the main MCU has crashed.

Erase Protection: In order to prevent misoperation to erase Flash data, DUE CORE designed a jumper to connection ERASE buttons and MCU Erase Pin, remove the jumper ERASE buttons and MCU pins can be disconnected, even if this time the ERASE button is pressed, it will not erase Flash data. When your program development is completed, officially delivered to the user in use, remove the jumper, it can be prevent Flash data loss caused by user errors operation.

6 For more about DUE , please refer link: https://www.arduino.cc/en/Main/ArduinoBoardDue

INHAOS Headquarter：：

    1111 Oakmont Drive #C, San Jose, CA   95117
    E-mail ：support@inhaos.com

INHAOS China office：：
No.6 Building,Songke Estate,Songshan Lake National Hi-tech Industrial Development Zone,Dongguan,Guangdong Province，523808,China
E-mail：Support@inhaos.com

Doc ID: DOC-DUE-CORE-V01-20160611   **www.inhaos.com**   Page: 5 of 5