WM_W600_Arduino EV Board_UserManual V1.0

Beijing Winner Microelectronics Co., Ltd.

Addr: 18th Floor, Yindu Building, No.67 Fucheng Road, Haidian District, Beijing, China

Tel: +86-10-62161900

Website: www.winnermicro.com

Document History

Version	Complete Date	Revision Record	Author	Auditor
V1.0	2018-10-17	Initial Release		

Content

1 Introduction

This document describes W600 Arduino EV board's interface definition, function description and interface reuse which can be used as master and slave equipment. At the end of this document, there is the schematic diagram of this EV board.

2 Interface Description

Following figure is the layout sketch map of this EV board:

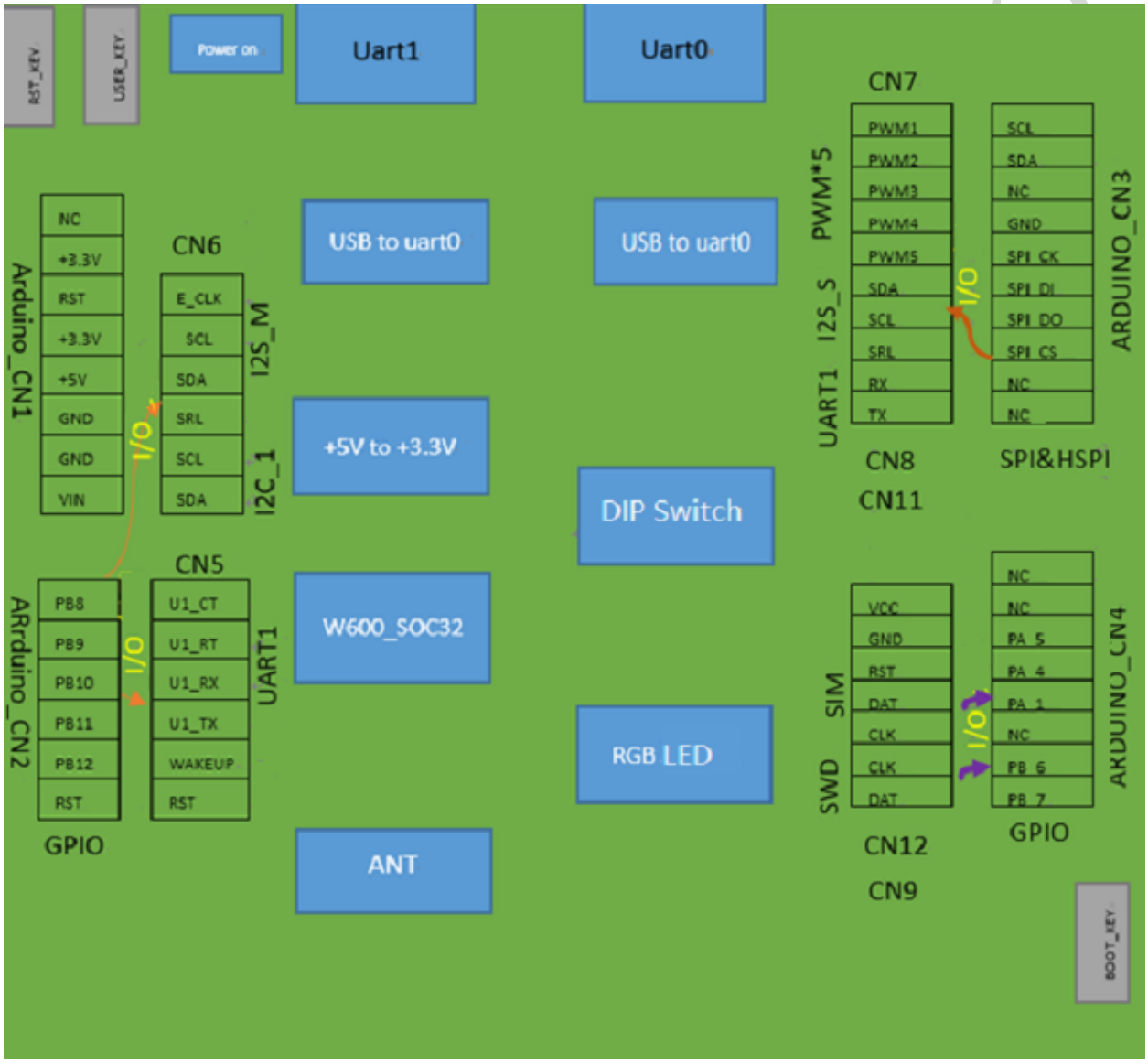

Figure1 Layout sketch map

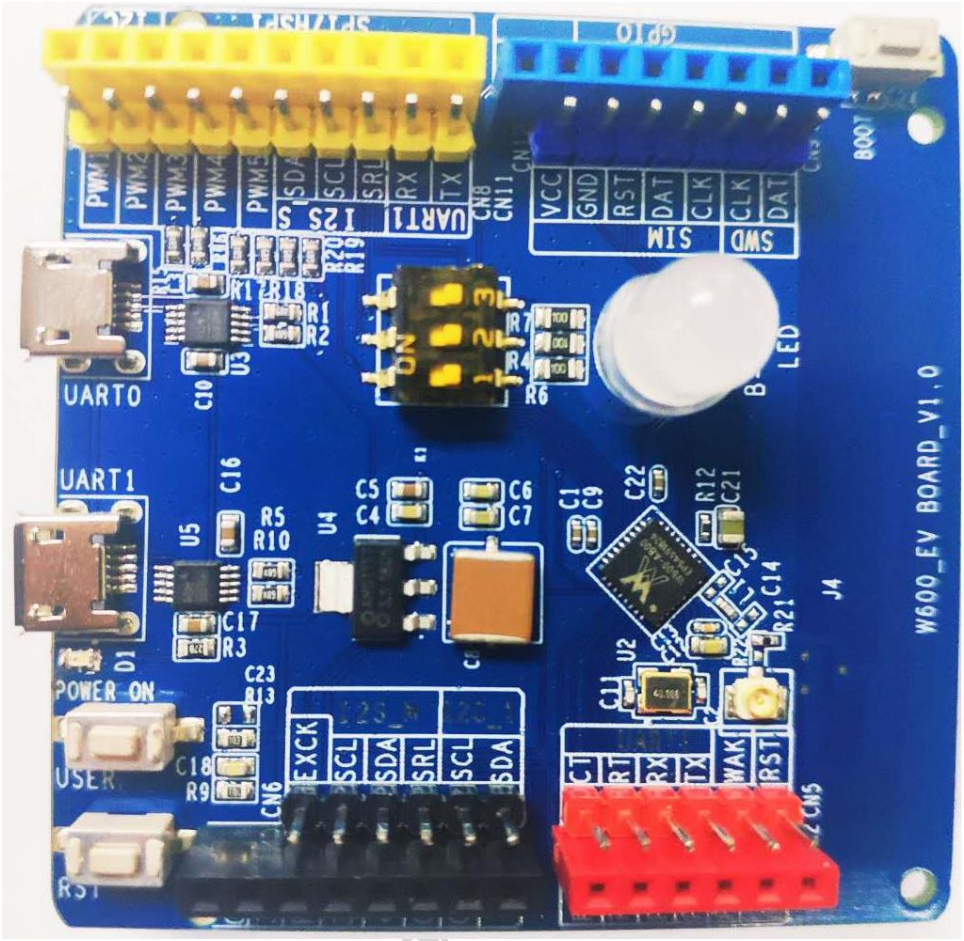

Figure2 Real board

Interfaces' function description (The detailed description for the order of interfaces in Section 4.3)

- UART0: used for power supply and interact debugging information
- UART1: used for power supply and interact controlling information
- CN1~CN4: Arduino standard interfaces
- CN5: UART1, wakeup, reset
- CN6: I²S master interface, I²C interface
- CN7: PWM interface
- CN8: I²S slave interface
- CN9: SW debugging download interface
- CN11: UART1 receive and transmit interface
- CN12: 3.3V, GND and SIM interface

- Dial switch: such switch should be turned to ON when using PWM function to control tricolor LED
- Keys' function:
  - RST: Reset W600 chipset
  - USER: control high or low level of PB_7
  - BOOT: control PA_0 to high or low level. When this pin is low level during starting up, the chipset will jump into ROM and run with the firmware updating mode.

3 Application Scenario

3.1 Used for master equipment

Arduino EV board supports following interfaces:

- I²C & I²S interface
- Uart0 & SWD debugging interface
- SPI & Uart1 & Uart2 interface
- PWM
- SIM
- GPIO
- Micro USB

When this EV board is used for master equipment, Micro USB interface can be used for debugging and communication. Uses can develop with the interfaces on EV board. This EV board is compatible with standard Arduino interfaces and uses can connect with other Arduino equipment directly.

3.2 Used for Wi-Fi module to connect with other Arduino master board

Arduino EV board supports following interfaces as a slave equipment:

- Arduino interfaces
- Uart0 & SWD debugging interfaces
- HSPI & Uart1 communication interfaces
- Micro USB

When this EV board is usd for slave equipment, it can be connected with other Arduino master board such as STM32 Nucleo. And this W600 EV board can support Wi-Fi communication

function and so on.

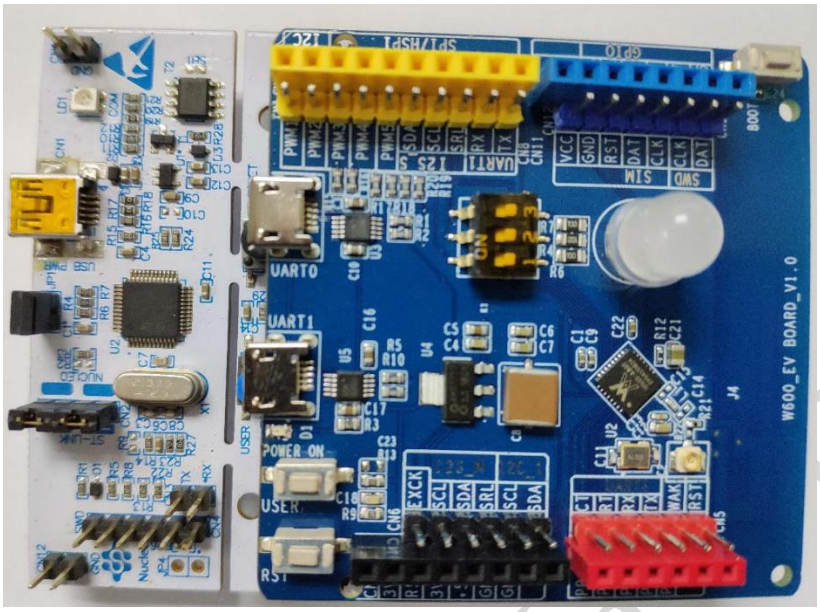

Figure3 Connected with STM32 Nucleo

4 Functions and using description

4.1 W600 EV board's functions and usage introduction

- This EV board supports following interfaces, and some interfaces have reuse functions, detailed information is in Section4.2.
  - I²C & I²S
  - Uart0 & SWD
  - SPI & Uart1 & Uart2
  - PWM
  - SIM
  - GPIO
  - Micro USB
- Power supply of this EV board
  - +5V can be inputed through the "+5V" pin of Uart0 or Uart1 or CN1 interface on this Arduino board. Then the LDO will convert +5V to +3.3V for the circuits on the board. All the GPIO's high level power are +3.3V.
- Indicator light on the board
  - Power on: the POWER ON LED will be lighted up when +3.3V has been inputed.
  - RGB LED lighting: there is a RGB tricolor LED light on the board. The corresponding GPIOs' duty cycle can be adjusted to change LED color. When LED function is working, the dial switch should be turned to ON.

- Wi-Fi communication function
  - Support GB15629.11-2006、IEEE802.11 b/g/e/i/d/k/r/s/w/n standard
  - Support frequency range: 2.4~2.4835 GHz
  - Support Wi-Fi WMM/WMM-PS/WPA/WPA2/WPS
  - Support Wi-Fi Direct
  - Support EDCA channel access
  - Support 20/40M bandwidth
  - Support STBC, GreenField, Short-GI and reverse transmission
  - Support RIFS interframe spaces
  - Support AMPDU, AMSDU
  - Support IEEE802.11n MCS 0~7, MCS32, transmission rate is up to 150Mbps
  - Support Short Preamble in 2/5.5/11 Mbps
  - Support HT-immediate Compressed Block Ack, Normal Ack, No Ack
  - Support CTS to self
  - Support STA/AP/AP+STA functions
  - As AP in BSS, the sum of sites and groups is up to 32 and in IBss is up to 16
  - Support up to 32 multicast networks with different encryption methods in BSS

4.2 Reuse Functions of Interfaces

Table1 Reuse Functions of W600 on the EV board

W600	I²C	I²S_M	I2S_	SPI	HSPI	PWM	UART	UART	SWD	SIM	GPIO	Others
PB_6									DAT	CLK	GPIO	
PB_7									CK		GPIO	User key
PB_8		SLC									GPIO	
PB_9		SDA						CTS			GPIO	
PB_10		SRL						RTS			GPIO	
PB_11	SCL							RX			GPIO	
PB_12	SDA							TX			GPIO	
PB_13	SCL											
PB_14	SDA		SDA		INT	PWM5						
PB_15			SCL	CS	CS	PWM4						
PB_16			SRL	CK	CK	PWM3						LED RED
PB_17				DI	DI	PWM2		RX				LED GREEN

PB_18				DO	DO	PWM1		TX				LED BLUE
PA_0												BOOT KEY
PA_1										DAT	GPIO	
PA_4							TX			RST	GPIO	
PA_5		EXTCL					RX				GPIO	

4.3 Arduino interfaces definitions

Table2 Arduino Interfaces on Arduino EV board

Name	No.	Definition	Name	No.	Definition
CN1	1	NC	CN3	1	I²C_SCL
	2	+3.3V		2	I²C_SDA
	3	RESET		3	NC
	4	+3.3V		4	GND
	5	+5V		5	SPI_CK
	6	GND		6	SPI_DI
	7	GND		7	SPI_DO
	8	VIN		8	SPI_CS
CN2	1	PB_8		9	NC
	2	PB_9		10	NC
	3	PB_10	CN4	1	NC
	4	PB_11		2	NC
	5	PB_12		3	PA_5
	6	RESET		4	PA_4
				5	PA_1
				6	NC
				7	PB_7
				8	PB_6

Table3 Other Interfaces

Name	No.	Definition	Name	No.	Definition
CN5	1	UART1_CTS	CN8	1	I²S_S_SDA
	2	UART1_RTS		2	I²S_S_SCL
	3	UART1_RX		3	I²S_S_SRL
	4	UART1_TX	CN9	1	SWDAT
	5	WAKEUP		2	SWCK
	6	RESET	CN11	1	UART1_RX
CN6	1	I²S_M_EXTCLK		2	UART1+TX
	2	I²S_M_SCL	CN12	1	+3.3V
	3	I²S_M_SDA		2	GND
	4	I²S_M_SRL		3	SIM_RST
	5	I²C_SCL_1		4	SIM_DATA
	6	I²C_SDA_1		5	SIM_CLK
CN7	1	PWM_1			
	2	PWM_2			
	3	PWM_3			
	4	PWM_4			
	5	PWM_5			

5 Application example for controlling RGB tricolor LED by PWM

Application Steps:

1， Open the project file in the director of Tools in WM_SDK_RGB_Control_Sample.

2， Compiling the project and generate the target file. Such target file is in the Bin director.

3， Update the generated target file to W600 Arduino EV board through serial port tools or programming tools.

4， Start up the new firmware and the LED light will change color every 0.5 second.

5， UserMain() is the program's entry function. A task named pwm_task has been created in UserMain. And this task is used for lighting LED (the example code is in APP/main.c).

```
void UserMain(void)
{
    printf("\n user task\n");

    tls_os_task_create(NULL, NULL,
                       pwm_task,
                       NULL,
                       (void *)TaskStk,
                       TASK_SIZE * sizeof(u32),
                       31,
                       0);
#if DEMO_CONSOLE
    CreateDemoTask();
#endif
//用户自己的task
}
```

6，Following code let the LED change color every 0.5 second.

```
static void pwm_task(void *sdata)
{
    unsigned char duty[3] = {0};

    while (1)
    {
        // Get 3 random data for duty[3] array for 3 PWM channels
        random_get_bytes(duty, 3);

        // PWM wave from channel 0 for output only mode, @freq=100Hz,duty=duty[0], it will reload automatically.
        pwm_work(0, 100, duty[0], 4, 0); //blue led

        // PWM wave from channel 1 for output only mode, @freq=100Hz,duty=duty[1], it will reload automatically.
        pwm_work(1, 100, duty[1], 4, 0); //green led

        // PWM wave from channel 2 for output only mode, @freq=100Hz,duty=duty[2], it will reload automatically.
        pwm_work(2, 100, duty[2], 4, 0); //red led

        // delay 0.5second
        tls_os_time_delay(HZ/2);
    }
}
```

6 Appendix: Schematic Diagram