T5L_ASIC Development Guide

Version 2.0

2022/4/13

Contents

1 Summary

The T5L series ASIC is a single-chip and dual-core ASIC designed by DWIN technology co., ltd for AIOT applications with low power consumption, high cost performance, GUI and highly integrated application, including T5L1(low resolution) and T5L2(high resolution). Its main features are as follow:

(1) Mature and stable 8051 core which is the most widely used with the maximum operating frequency of T5L is up to 250MHz, 1T(single instruction cycle)high speed operation.

(2) Separated GUI CPU core running DGUS II System:

➢ High-speed display memory, 2.4GB/S bandwidth, 24bit color display resolution supporting to 800*600(T5L1) or 1366*768(T5L2).

➢ 2D hardware acceleration, the decompression speed of JPEG is up to 200fps@1280*800, the UI with animation and icons as its main feature is extremely cool and smooth.

➢ Images and icons stored in JPEG format. Adopt Low-cost 16Mbytes SPI Flash.

➢ Support CTP or RTP with adjustable sensitivity and maximum 400Hz touch frequency.

➢ High-quality speech compression storage and playback.

➢ 128Kbytes variable storage space for exchanging data with OS CPU core and memory

➢ 1-way 15bit 32Ksps PWM digital power amplifier driver loudspeaker to save power amplifier cost and achieve high signal-to-noise ratio and sound quality restoration.

➢ Support DGUS development and simulation on PC. Support background remote upgrade.

(3) Separated CPU (OS CPU) core runs user 8051 code or DWIN OS system, user CPU omitted in practical application.

- Standard 8051 architecture and instruction set, 64Kbytes code space, 32Kbytes on-chip RAM.
- 64 bit integer mathematical operation unit(MDU), including 64bit MAC and 64 bit divider.
- Built-in software WDT, three 16bit Timers, 12 interrupt signals with the highest four interrupt embedding.
- 28 IOs, 4-channel UARTs, 1-channel CAN, up to 8-channel 12-bit A/Ds and 2-channle 16-bit PWM of adjustable resolution.
- Support IAP online simulation and debugging with unlimited number of breakpoints.
- Upgrade code online through DGUS system.

(4) 1Mbytes on-chip Flash with DWIN patent encryption technology ensure code and data security.

(5) Reduces crystal requirements and PCB design challenges for a variety of inexpensive wide-range tuned impedance crystal oscillators and PLLs.

(6) 3.3V IO voltage, adaptable to 1.8/2.5/3.3 various levels

(7) Support SD interface or UART1 download and configuration. Support SD card file reading and rewriting.

(8) Support DWIN WiFi module to access to DWIN cloud directly, and easily development for various cloud platform applications.

(9) Working temperature ranges from - 40℃ to +85℃ (Customizable IC for -55℃ to 105℃ operating temperature range)

(10) With low power consumption and strong anti-interference ability, it can work steadily on the double-sided PCB design, and easy to pass EMC/EMI test.

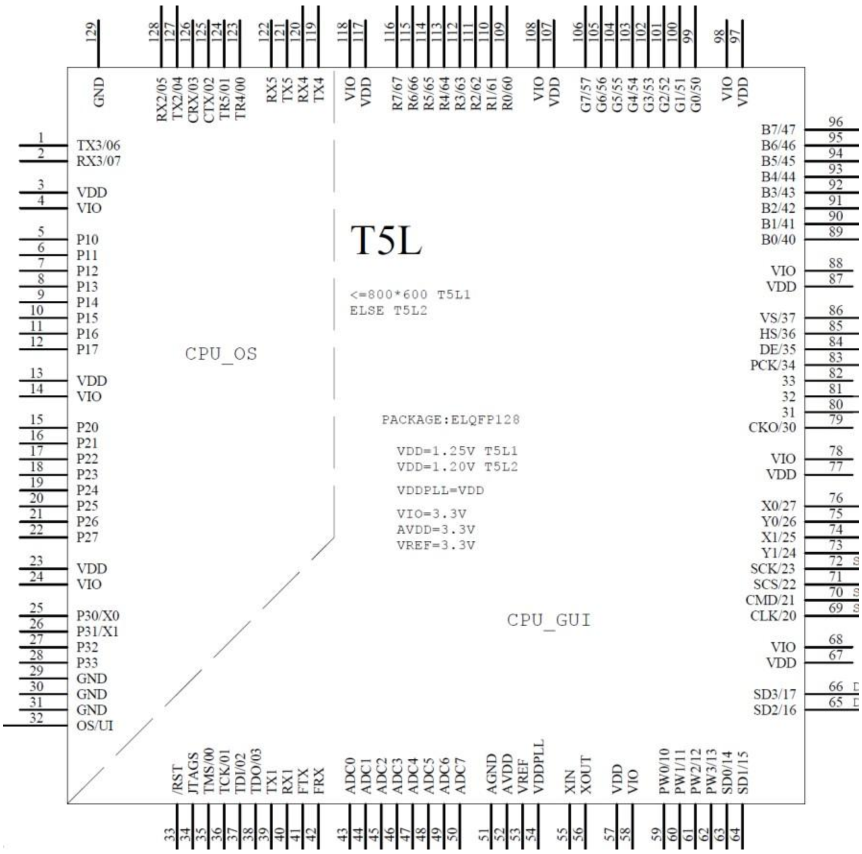
(11) 0.4mm ELQFP128 packaging with low processing difficulty and low cost.

(12) T5L ASIC + LCD + TP + UI design support for industry customers with cost-effective matching program sales and comprehensive technical service support.

2 Hardware Description

2.1 PIN Definition

T5L ASIC is packaged in ELQFP128 (16*16*1.5mm), and pins arrangement are shown below.



T5L pin

CPU	PIN#	Definition					
		Function 1	Instructions	Function 2	Instructions	Function 3	Instructions
OS	119	TX4	UART4 data				
OS	120	RX4	UART4 data				
OS	121	TX5	UART5 data				
OS	122	RX5	UART5 data				
OS	123	P0.0	I/O port				
OS	124	P0.1	I/O port				
OS	125	P0.2	I/O port	CAN_TX	CAN data sending		
OS	126	P0.3	I/O port	CAN_RX	CAN data receiving		
OS	127	P0.4	I/O port	TX2	UART2 data sending		
OS	128	P0.5	I/O port	RX2	UART2 data receiving		
OS	1	P0.6	I/O port	TX3	UART3 data sending		
OS	2	P0.7	I/O port	RX3	UART3 data receiving		
OS	3	VDD	T5L1=1.25V T5L2=1.2V				
OS	4	VIO	3.3V				

OS	5	P1.0	I/O port				
OS	6	P1.1	I/O port				
OS	7	P1.2	I/O port				
OS	8	P1.3	I/O port				
OS	9	P1.4	I/O port				
OS	10	P1.5	I/O port				
OS	11	P1.6	I/O port				
OS	12	P1.7	I/O port				
OS	13	VDD	T5L1=1.25V T5L2=1.2V				
OS	14	VIO	3.3V				
OS	15	P2.0	I/O port				
OS	16	P2.1	I/O port				
OS	17	P2.2	I/O port				
OS	18	P2.3	I/O port				
OS	19	P2.4	I/O port				
OS	20	P2.5	I/O port				
OS	21	P2.6	I/O port				
OS	22	P2.7	I/O port				
OS	23	VDD	T5L1=1.25V T5L2=1.2V				
OS	24	VIO	3.3V				
OS	25	P3.0	I/O port	EX0	External interrupt 0 input		
OS	26	P3.1	I/O port	EX1	External interrupt 1 input		
OS	27	P3.2	I/O port				
OS	28	P3.3	I/O port				
	29	GND					
	30	GND					
	31	GND					
	32	OS/GUI	0:GUI JMARK, 1=OS JMARK				
	33	/RST	System reset				
	34	JMARKS	PIN35#-PIN38# Select: 0=JMARK, 1=I/O port				
GUI	35	P0.0	I/O port	TMS	JMARK interface	TCON_CS	LCD screen TCON interface
GUI	36	P0.1	I/O port	TCK	JMARK interface	TCON_CLK	LCD screen TCON interface
GUI	37	P0.2	I/O port	TDI	JMARK interface	TCON_DATA	LCD screen TCON interface
GUI	38	P0.3	I/O port	TDO	JMARK interface	TCON_RST	LCD screen TCON interface
GUI	39	P0.4	I/O port	TX1	UART1 data sending		
GUI	40	P0.5	I/O port	RX1	UART1 data receiving		
GUI	41	P0.6	I/O port	FSK_TX	FSK transceiver data sending		

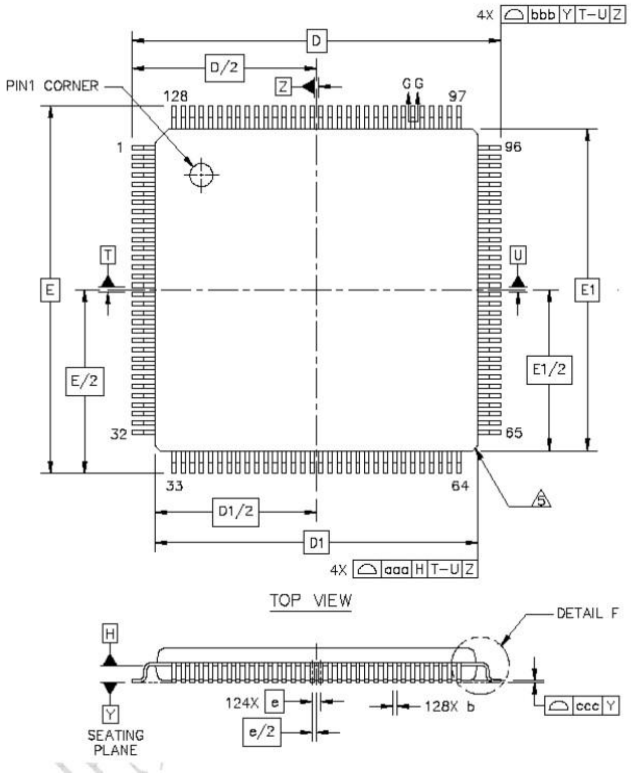
GUI	42	P0.7	I/O port	FSK_RX	FSK transceiver data receiving		
GUI	43	ADC0	AD input	RTP_X0	4-wire resistance touch screen interface		
GUI	44	ADC1	AD input	RTP_Y0	4-wire resistance touch screen interface		
GUI	45	ADC2	AD input	RTP_X1	4-wire resistance touch screen interface		
GUI	46	ADC3	AD input	RTP_Y1	4-wire resistance touch screen interface		
GUI	47	ADC4	AD input	IF_0.4	DC/DC 0.4V voltage feedback		
GUI	48	ADC5	AD input	VF_1.25	DC/DC 1.25V voltage feedback		
GUI	49	ADC6	AD input	IF_0.4			
GUI	50	ADC7	AD input	VF_1.25			
GUI	51	AGND	AD GND				
GUI	52	AVDD	AD power supply, 3.3V, close to 470pJ (C0G material) in parallel with 105 filter capacitor				
GUI	53	VREF	AD reference power supply, up to AVDD+0.3V, close to 470pJ (C0G material) in parallel with 105 filter capacitor				
GUI	54	VDDPLL	T5L1=1.25V T5L2=1.2V, close to 470pJ (C0G material) in parallel with 105 filter capacitor				
GUI	55	XIN	Crystal,10MHz-12MHZ	CLK_IN	3.3V clock input		
GUI	56	XOUT	Crystal				
GUI	57	VDD	1.1V				
GUI	58	VIO	3.3V				
GUI	59	P1.0	I/O port	PWM0	16bit PWM output		
GUI	60	P1.1	I/O port	PWM1	16bit PWM output	PWM_V	LCD screen AVDD DC/DC
GUI	61	P1.2	I/O port	PWM2	16bit PWM output	PWM_I	LCD backlight DC/DC
GUI	62	P1.3	I/O port	PWM3	16bit PWM output	BUZZ&SPK	Buzzer/speaker drive
GUI	63	P1.4	I/O port	SDD0	SD card interface: data	SPI_D0	SPI Flash interface: data
GUI	64	P1.5	I/O port	SDD1	SD card interface: data	SPI_D1	SPI Flash interface: data
GUI	65	P1.6	I/O port	SDD2	SD card interface: data	SPI_D2	SPI Flash interface: data
GUI	66	P1.7	I/O port	SDD3	SD card interface: data	SPI_D3	SPI Flash interface: data
GUI	67	VDD	T5L1=1.25V T5L2=1.2V				
GUI	68	VIO	3.3V				
GUI	69	P2.0	I/O port	SDCK	SD card interface: clock	PA_EN	Amplifier power switch for speech playback
GUI	70	P2.1	I/O port	SDCK	SD card interface: instructions		

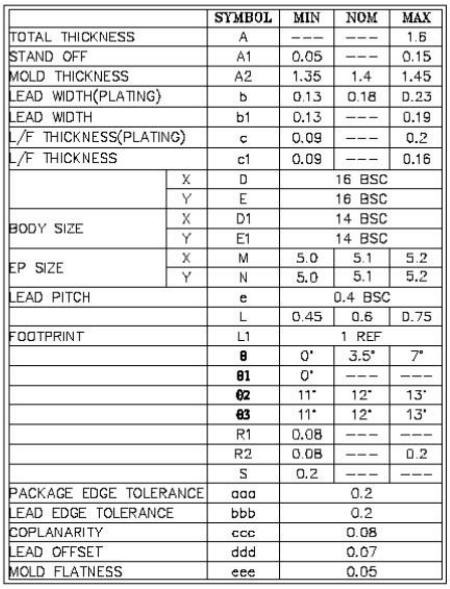
GUI	71	P2.2	I/O port	SPI_CS	SPI Flash interface: chip selection		
GUI	72	P2.3	I/O port	SPI_CLK	SPI Flash interface: clock		
GUI	73	P2.4	I/O port	RTP_Y1	4-wire resistance touch screen interface	CTP_SDA	Capacitance touch screen interface
GUI	74	P2.5	I/O port	RTP_X1	4-wire resistance touch screen interface	CTP_INT	Capacitance touch screen interface
GUI	75	P2.6	I/O port	RTP_Y0	4-wire resistance touch screen interface	CTP_SCL	Capacitance touch screen interface
GUI	76	P2.7	I/O port	RTP_X0	4-wire resistance touch screen interface	CTP_RST	Capacitance touch screen interface
GUI	77	VDD	T5L1=1.25V T5L2=1.2V				
GUI	78	VIO	3.3V				
GUI	79	P3.0	I/O port	CLK_OUT	System clock frequency division output		
GUI	80	P3.1	I/O port	FSK_TR	T/R switching signal for half-duplex use of SFK transceiver		
GUI	81	P3.2	I/O port				
GUI	82	P3.3	I/O port				
GUI	83	P3.4	I/O port	LCD_PCLK	LCD interface		
GUI	84	P3.5	I/O port	LCD_DE	LCD interface		
GUI	85	P3.6	I/O port	LCD_HS	LCD interface		
GUI	86	P3.7	I/O port	LCD_VS	LCD interface		
GUI	87	VDD	T5L1=1.25V T5L2=1.2V				
GUI	88	VIO	3.3V				
GUI	89	P4.0	I/O port	LCD_B0	LCD interface		
GUI	90	P4.1	I/O port	LCD_B1	LCD interface		
GUI	91	P4.2	I/O port	LCD_B2	LCD interface		
GUI	92	P4.3	I/O port	LCD_B3	LCD interface		
GUI	93	P4.4	I/O port	LCD_B4	LCD interface		
GUI	94	P4.5	I/O port	LCD_B5	LCD interface		
GUI	95	P4.6	I/O port	LCD_B6	LCD interface		
GUI	96	P4.7	I/O port	LCD_B7	LCD interface		
GUI	97	VDD	T5L1=1.25V T5L2=1.2V				
GUI	98	VIO	3.3V				
GUI	99	P5.0	I/O port	LCD_G0	LCD interface		
GUI	100	P5.1	I/O port	LCD_G1	LCD interface		
GUI	101	P5.2	I/O port	LCD_G2	LCD interface		

GUI	102	P5.3	I/O port	LCD_G3	LCD interface		
GUI	103	P5.4	I/O port	LCD_G4	LCD interface		
GUI	104	P5.5	I/O port	LCD_G5	LCD interface		
GUI	105	P5.6	I/O port	LCD_G6	LCD interface		
GUI	106	P5.7	I/O port	LCD_G7	LCD interface		
GUI	107	VDD	T5L1=1.25V T5L2=1.2V				
GUI	108	VIO	3.3V				
GUI	109	P6.0	I/O port	LCD_R0	LCD interface		
GUI	110	P6.1	I/O port	LCD_R1	LCD interface		
GUI	111	P6.2	I/O port	LCD_R2	LCD interface		
GUI	112	P6.3	I/O port	LCD_R3	LCD interface		
GUI	113	P6.4	I/O port	LCD_R4	LCD interface		
GUI	114	P6.5	I/O port	LCD_R5	LCD interface		
GUI	115	P6.6	I/O port	LCD_R6	LCD interface		
GUI	116	P6.7	I/O port	LCD_R7	LCD interface		
GUI	117	VDD	T5L1=1.25V T5L2=1.2V				
GUI	118	VIO	3.3V				

Note that the pads on the bottom of the IC must be reliably grounded, otherwise the performance of the IC will be affected.

2.2 Packaging Dimension





		SYMBOL	MIN	NOM	MAX
TOTAL THICKNESS		A	---	---	1.6
STAND OFF		A1	0.05	---	0.15
MOLD THICKNESS		A2	1.35	1.4	1.45
LEAD WIDTH(PLATING)		b	0.13	0.18	D.23
LEAD WIDTH		b1	0.13	---	0.19
L/F THICKNESS(PLATING)		c	0.09	---	0.2
L/F THICKNESS		c1	0.09	---	0.16
BODY SIZE	X	D	16 BSC		
	Y	E	16 BSC		
	X	D1	14 BSC		
	Y	E1	14 BSC		
EP SIZE	X	M	5.0	5.1	5.2
	Y	N	5.0	5.1	5.2
LEAD PITCH		e	0.4 BSC		
FOOTPRINT		L	0.45	0.6	D.75
		L1	1 REF		
		θ	0°	3.5°	7°
		θ1	0°	---	---
		θ2	11°	12°	13°
		θ3	11°	12°	13°
		R1	0.08	---	---
		R2	0.08	---	0.2
		S	0.2	---	---
PACKAGE EDGE TOLERANCE		aaa	0.2		
LEAD EDGE TOLERANCE		bbb	0.2		
COPLANARITY		ccc	0.08		
LEAD OFFSET		ddd	0.07		
MOLD FLATNESS		eee	0.05		

For PCB design, please refer to the DWIN official device packaging and reference design.

2.3 Basic Performance Parameters

Parameter	Unit	Minimum	Typical value	Maximum	Instructions
CPU core voltage	V	1.10	1.20	1.30	T5L2
	V	1.20	1.25	1.35	T5L1
CPU core current	mA		100		Dual-core 200MHz full speed operation
IO voltage (VIO)	V	1.8	3.3	3.6	5V TTL/CMOS level input requires voltage division or clamp protection
AD operating voltage(VADD)	V	1.8	3.3	3.6	
AD reference voltage (Vref)	V			VADD+0.3	
AD input voltage	V			Vref+0.3	
IO high level output amplitude(VOH)	V	3.0			VIO=3.3V,IO load current 8mA
IO low level 1utput amplitude(Vol)	V			0.3	VIO=3.3V,IO load current 8mA
IO high level output current	mA	-10			VIO=3.3V, VOH=3V
IO low level output current	mA	10			VIO=3.3V, VOL=0.3V
IO port turnover speed	MHz		100		
IO high level recognition voltage(VIH)	V	1.6			
IO low level recognition voltage(VIL)	V			0.6	
External crystal frequency	MHz	10.0	11.0592	12.0	
CPU main frequency (CPU_CLK)	MHz		206.4384		Corresponding to 11.0592MHz crystal, CPU main frequency = crystal frequency* 56/3. 350MHz version code can be customized.
Working temperature	℃	-40		+85	
Storage temperature	℃	-55		+105	
ESD protection capability	KV		2		HBM

2.4 Notices for Hardware Design

(1) The core power voltage must be stable, otherwise it will lead to abnormal CPU operation.

(2) Reset is recommended to be handled by a low-level reset IC like SGM809S, instead of a simple RC reset circuit. The T5L has a built-in watchdog (WDT) for each CPU core, so there is no need for an external WDT IC.

(3) When designing for dual panel applications, connect a 470pF in parallel with 104 (or 105) filter capacitors as close as possible to the IC supply pins to reduce noise emission.

(4) When IO input signal is over 0.3V of VIO voltage, IO must be protected by voltage divider or clamp, otherwise it may cause abnormal signal or damage IC.

(5) All IO ports are floating input when they are configured as input mode, without internal pull-up or pull-down.

All IO ports are in the input mode during the reset. If they are output, they can be pulled down or pulled up externally to ensure that the reset has a definite level.

(6) The 4-bit bus speed of T5L and external SPI Flash is 100MHz, and thus The wiring should be as close as possible and 470pF must be arranged on the power pins of the SPI Flash in parallel with 105 filter capacitor.

3 OS CPU

The T5L OS CPU adopt the 8051 kernel, which is the most widely used in industry and has longest history in mass production.

On the basis of the good real-time performance, fast IO rate and stable reliability of 8051, DWIN has significantly improved the 8051 memory by optimizing the code processing, expanding the SFR bus and enhancing the hardware math processor.

For applications mainly for computing and data processing, users can also run the OS CPU and conduct secondary application development on the DWIN OS platform.

Refer to "DWIN OS Development Guide" for specific development methods.

3.1 Initial Configuration

When the 8051 kernel is powered on, the special function register (SFR) in the following table must be initialized correctly.

SFR name	Address	Initial values	Instructions
CKON	0X8E	0X00	CPU runs in 1T mode
T2CON	0XC8	0X70	Configure extended interrupt system; configure timer T2 to run in autoload mode
DPC	0X93	0X00 or 0x01	The change mode of DPTR after MOVX instruction operation must be configured to 0x00 if developed with C51. 0x00: No change. 0x01: DPTR=DPTR+1. 0x03: DPTR=DPTR-1.
PAGESEL	0X94	0X01	64KB code space
D_PAGESEL	0X95	0X02	32KB RAM space accessed by MOVX: 0x8000-0xFFFF
MUX_SEL	0XC9	0x60 orconfiguration according to application needs	Peripheral multiplexing selection: .7 1 = CAN interface leads to P0.2, P0.3, 0 = CAN interface does not lead out, and it works as an IO interface; .6 1 = UART2 interface leads to P0.4, P0.5, 0 = UART2 interface does not lead out, and it works as an IO interface; .5 1 = UART3 interface leads to P 0.6, P 0.7, 0 = UART3 interface does not lead out, and it works as an IO interface; .4-.2 Reserved; .1 WDT control 1=open 0=close; .0 WDT feed dog, 1=feed the dog one time(The WDT count becomes zero, and the watchdog's overflowing time is one second.);
PORTDRV	0XF9	0x01	Driver capability configuration of IO port output mode. 0x00=4mA 0X01=8mA 0X02=16mA 0X03=32mA
RAMMODE	0XF8	0X00	DGUS variable memory access interface control

3.2 Memory

The 8051 kernel of OS can access seven different kinds of memory, which are shown as below.

Memory type	Size	Address space	Access mode
Code memory	64KBytes	0x0000-0xFFFF	It can only be read by MOVC instruction, same as standard 8051.
Data register	256Bytes	0x00-0xFF	The same as standard 8051
SFR register	128Bytes	0X80-0XFF	The same as standard 8051. DWIN can provide user SFR definition files(.INC or .H header file).
Extended SFR register	64Bytes	0x00-0x3F	Accessible using the SFR register interface defined by EXADR, EXDATA.
Data memory	32KBytes	0x8000-0xFFFF	Accessible using MOVX instruction. When DPC is configured as 0x00, same as standard 8051.
DGUS variable memory	256KBytes	0x00:0000-0x00:FFFF	Accessible using DGUS variable memory interface.
CAN Communication interface	48Bytes	0xFF:0060-0xFF:006B	Accessible using DGUS variable memory interface.

3.2.1 Code Memory(64KBytes)

Functional partitioning and definition of the code memory space are shown in the following table.

Address	Definition	Instructions
0x0000	Reset_PC	After reset, the program starts running address .
0x0003	EX0_ISR_PC	External interrupt 0 program interface
0x000B	T0_ISR_PC	Timer0 interrupt program interface
0x0013	EX1_ISR_PC	External interrupt 1 program interface
0x001B	T1_ISR_PC	Timer1 interrupt program interface
0x0023	UART2_ISR_PC	UART2 TX/RX interrupt program interface
0x002B	T2_ISR_PC	Timer2 interrupt program interface
0x004B	CAN_ISR_PC	CAN interface interrupt program interface
0x0053	UART4_TX_ISR_PC	UART4 TX interrupt program interface
0x005B	UART4_RX_ISR_PC	UART4 RX interrupt program interface
0x0063	UART5_TX_ISR_PC	UART5 TX interrupt program interface
0x006B	UART5_RX_ISR_PC	UART5 RX interrupt program interface
0x0083	UART3_ISR_PC	UART3 TX/RX interrupt program interface
0x00F8	JMARK interface enabled	0xFFFF will allow connection to JMARK interface for simulation debugging, and other values will be prohibited.
0x00FA	"DWINT5"	Code identification, illegal values will cause OS 8051 to stop running.
0x0100	Application code start	Maximum 63.75KB

The OS 8051 code is stored in the 0x01:0000-0x01:FFFF position of the 1Mbytes on chip Flash. After power-on reset, the system loads and runs in RAM.

Code can only be written to on-chip Flash through SD interface or UART1 interface (or WIFI network interface, etc).

3.2.2 Variable Memory(256KBytes)

(1) 0x00:0000-0x00:7FFF addresses correspond to the 128Kbytes variable memory space currently used by the DGUS II system.

For example, the two DGUS II variable memories, 0x1000 and 0x1001, correspond to the 0x0800 address of the OS 8051 DGUS variable memory, where D3 corresponds to the 0x1000 variable high byte and D2 corresponds to the 0x1000 variable low byte; D1 corresponds to 0x1001 variable high byte and D0 corresponds to 0x1001 variable low byte.

(2) 0x00:8000-0x00:FFFF addresses are not used by DGUS II system at present, and can be defined by users as variable memory.

(3) 0xFF:0060-0xFF:006B address, the configuration and transceiver buffer of CAN interface.

Access to the DGUS variable memory uses the SFR register interface in the following table.

SFR name	Address	Instruction
RAMMODE	0xF8	DGUS variable memory access interface control, which can be bit addressed. .7 Write 1=request occupation of DGUS variable memory to read/write. Must be cleared when not occupied. .6 APP_EN Write 1=initiate read/write once. It is cleared after hardware execution. .5 APP_RW 1=Read variable memory 0=Write variable memory .4 APP_ACK Hardware answer to 8051 request to occupy variable memory, 1=OK, 0=BUSY. .3-.0 Corresponding to DATA3:DATA0 Write enable 1=corresponding bytes written, 0=corresponding byte not written.
ADR_H	0XF1	DGUS variable memory high 8-bit address,A23:A16.
ADR_M	0XF2	DGUS variable memory middle 8-bit address,A15:A8.
ADR_L	0XF3	DGUS variable memory low 8-bit address,A7:A0.
ADR_INC	0xF4	The automatic increment of address after reading and writing of DGUS variable memory. That is, ADR_H:M:L after reading and writing = ADR_H:M:L+ADR_INC before.
DATA3	0xFA	DGUS variable data interface, write selection corresponds to RAMMODE.3.
DATA2	0XFB	DGUS variable data interface, write selection corresponds to RAMMODE.2.
DATA1	0XFC	DGUS variable data interface, write selection corresponds to RAMMODE.1.
DATA0	0XFD	DGUS variable data interface, write selection corresponds to RAMMODE.0.

DGUS variable memory must be read and written according to the steps below.(If you want to use interrupt in the application, the interrupt must be closed when the main application reads and writes DGUS variable data, not embedded.)

(1) Configured address and address increment;

(2) Set RAMMODE as 0x8F (write) or 0xAF (read), check if RAMMODE. 4=1 to confirm access to read and write control.

(3) Read and write data, set RAMMODE as 0X00 after reading and writing.

Example

Read and write two double words in 0x0800 address (corresponding to DGUSII variable memory address 0x1000).

```
        MOV     ADR_H,#00H              ;Configure DGUS variable memory address
        MOV     ADR_M,#08H
        MOV     ADR_L,#00H
        MOV     ADR_INC,#01H            ;Configuration address increment
        MOV     RAMMODE,#0AFH           ;Start read mode
        JNB     APP_ACK,$               ;Waiting for confirmation
        MOV     R0,#TEST_BUF            ;Reading demo
        MOV     R1,#2
 RDVP:  SETB    APP_EN                  ;Start reading data once
        JBAPP_EN,$
        MOV     @R0,DATA3
        INC     R0
        MOV     @R0,DATA2
        INC     R0
        MOV     @R0,DATA1
        INC     R0
        MOV     @R0,DATA0
        INC     R0
        DJNZ    R1,RDVP
        CLR     APP_RW                  ;Writing mode; writing demo
        MOV     ADR_L,#00H              ;Adjust the address to 0x08:0000
        MOV     R0,#TEST_BUF
        MOV     R1,#2
WRVP:   MOV     DATA3,@R0
        INC     R0
        MOV     DATA2,@R0
        INC     R0
        MOV     DATA1,@R0
        INC     R0
        MOV     DATA0,@R0
        SETB    APP_EN                  ;Start writing data once
        JNB     APP_EN,$
        INC     R0
        DJNZ    R1,WRVP
        MOV     RAMMODE, #00H           ;Variable memory read and write ends
```

3.2.3 Data Memory(32KBytes)

The OS 8051 CPU of T5L has 32KBytes RAM as data memory, corresponding to addresses 0x8000-0xFFFF, and the related interface SFRs are listed below.

SFR name	Address	Instructions

DPC	0x93	DPTR change mode after MOVX instruction operation.
		DPC=0X00:DPTR remains unchanged after MOVX instruction operation. If developed using C51, DPTR must be configured as 0x00
		DPC=0X01 After MOVX instruction operation, DPTR=DPTR+1
		DPC=0X03 After MOVX instruction operation, DPTR=DPTR-1

DPH	0X83	DPTR data pointer
DPL	0X82	

The address space from 0x0000 to 0x7FFF prohibit using MOVX instructions to read and write, which may cause code to run abnormally.

The T5L's MOVX instruction is 3 instruction cycles (3T, 14.5nS at 11.0592MHz crystal), and the DPC can be configured with DPTR auto increment (or decrement) mode, making the T5L much faster than the standard 8051 for reading and writing data memory, especially for inverse order memory read and write applications.

Example

```
MOV         DPC,#01H          ;DPTR++
MOV         DPTR,#8000H
MOVX         A,@DPTR           ;A=@8000
MOVX         A,@DPTR           ;A=@8001, DPTR=8002 after reading
```

3.2.4 Extended SFR Register

Extended SFR registers use EXADR, EXDDATA register to read and write.

SFR name	Address	Instruction
EXADR	0xFE	Extend the SFR address and automatically add 1 to the next SFR after each reading and writing.
EXDATA	0xFF	Expanding SFR data interface

The extended SFR register defines the register interface of the Mathematical Operating Unit (MDU) and provides 48 additional data storage for users. The following table is defined:

EXADR	Definition	Instruction
0x00	MDU_A7	Maximum bit of MDU A register(64bit)
......		
0x07	MDU_A0	Minimum bit of MDU A register(64bit)
0x08	MDU_B7	Maximum bit of MDU B register(64bit)
......		
0x0F	MDU_B0	Minimum bit of MDU B register(64bit)
0x10	MDU_C7	Maximum bit of MDU C register(64bit)
......		
0x17	MDU_C0	Minimum bit of MDU C register(64bit)
0x18	EXR0	First extended data register.
0x19	EXR1	Second extended data register
0x3F	EXR39	Fortieth extended data register

If you need to read or write the extended SFR in the interrupt application, the main program must turn off the interrupt when reading or writing the extended SFR and cannot be embedded.

3.3 Mathematical Operating Unit(MDU)

Due to the limited computing power of 8051, the hardware mathematic unit (MDU) is extended in T5L to improve computing power. OS 8051 applications can access hardware MAC and hardware divider. The related SFR definitions are as follows:

SFR name	Address	Instruction

MAC_CN	0xE5	The MAC hardware multiplier-adder control register is defined as follows.
		.7 MAC enable. Write 1 to perform a calculation, and clear after hardware execution;
		.6 MAC mode. 1 for multiplier-adder mode: C=A*B+C; 0 for multiplier mode: C=A*B;
		.5 Write 0;
		.4 1=64bit MAC 0=32bit MAC(A3:0/B3:0/C7:0;note that C is still 64bit);
		.3 1 = Signed number; 0 = Unsigned number;
		.2-.0 Write 0;
		The A, B, C register group is the MDU_A, MDU_B, MDU_C register group of extended SFR registers.

DIV_CN	0XE6	DIV hardware divider control register (division C/A, quotient A, remainder B) is defined as follows:
		.7 DIV enable. Write 1 to perform a calculation, and clear after hardware execution;
		.6 DIV mode 1: Rounded 0: Not rounded;
		.5-.4 Undefined, write 0;
		.3 1 = Signed number 0 = unsigned number;
		.2-.0 Write 0;
		The A, B, C register group is the MDU_A, MDU_B, MDU_C register group of extended SFR registers.

Example: 32bit MAC calculate 0x1234*0x5678-0x2000

```
MOV     EXADR,#04H       ;write A3:A0=0x 00 00 12 34
MOV     EXDATA,#00H
MOV     EXDATA,#00H
MOV     EXDATA,#12H
MOV     EXADR,#34H       ;write B3:B0=0x 00 00 56 78
MOV     EXDATA,#00H
MOV     EXDATA,#00H
MOV     EXDATA,#56H
MOV     EXDATA,#78H
MOV     EXADR,#10H       ;write C7:C0=0xFF FF FF FF FF FF
                         E0 00 (-0x2000)
MOV     EXDATA,#0FFH
MOV     EXDATA,#0FFH
MOV     EXDATA,#0FFH
MOV     EXDATA,#0FFH
MOV     EXDATA,#0FFH
MOV     EXDATA,#0FFH
MOV     EXDATA,#0E0H
MOV     EXDATA,#00H
MOV     MAC_CN,#0C8H     ;32bit integer MAC
MOV     A,MAC_CN
MOV     ACC,7,WTMAC
MOV     EXADR,#10H       ;read results00 00 00 00 06 25 E0 60
MOV     R7,EXDATA
MOV         R6,EXDATA
MOV         R5,EXDATA
MOV         R4,EXDATA
MOV         R3,EXDATA
MOV         R2,EXDATA
MOV         R1,EXDATA
MOV         R0,EXDATA
```

3.4 Timer

T5L OS 8051 has three timers: T0/T1/T2, of which T0/T1 are consistent with standard 8051, and T2 can only work in 16 bit autoload mode.

The clocks of T0 and T1 are 1/12 of the CPU main frequency. The clocks of T2 can be configured as 1/12 or 1/24 of the CPU main frequency.

The relevant SFRs are shown in the table below

SFR name	Address	Instructions
TCON	0x88	T0, T1 control registers, same as standard 8051, can be addressable by bit .7=TF1 .6=TR1 .5=TF0 .4=TR0 .3=IE1 .1=IEO .0=ITO IT1 and IT0 are external interrupt trigger mode selection: 0 = low level trigger 1 = down-jump edge trigger.
TMOD	0x89	T0, T1 mode selection, same as standard 8051.
TH0	0x8C	
TL0	0x8A	
TH1	0x8D	
TL1	0x8B	

T2CON	0xC8	The T2 control register can be addressable by bit.
		.7:clock frequency division selection. 0=CPU main frequency /12; 1=CPU main frequency/24;
		.6-.4:must write 1;
		.3-.1:must write 0;
		.0:TR2. 1=T2 run; 0=T2 close;

TH2	0xCD	T2 running value, automatically loaded every time counting overflow TH2=CRCH TL2=CRCL.
TL2	0xCC	
TRL2H	0xCB	The reload value of T2 = 65536-T2 timer interval (uS)*T2 clock frequency (MHz). CPU main frequency = crystal frequency * 56/3, corresponding CPU main frequency/12 = crystal frequency * 14/9, CPU main frequency/24 = crystal frequency * 7/9. For example, the CPU main frequency is 206.438 MHz, T2 chooses 1/12 frequency division, and the setting value of 1 mS timer interval is 48333 (0xBCCD).
TRL2L	0xCA	

The relevant settings of timer interrupt are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
T0 interruption	0x000B	TF0(TCON.5)	IEN0.1	Automatic clear TF0 in interrupt response
T1 interruption	0x001B	TF1(TCON.7)	IEN0.3	Automatic clear TF0 in interrupt response
T2 interruption	0x002B	TF2(TCON.6)	IEN0.5	After interrupt response, TF2 needs to be cleared by software, otherwise interrupt will be triggered again.

Example

In 11.0592 MHz crystal (corresponding to the main frequency of CPU 206.4384 MHz), T2 1mS interruption is set to output 500 MHz square wave at P 1.0.

```
            ORG       002BH               ;T2 interrupt program  entry
            LJMP      T21NT

T21NT:      CLR       TF2                 ;T2 interrupt program
            CPL       P1.0
            RETI

    ;Initialization of T2-related SFR
    MOV   CRCH,#HIGH(48333)               ;1mS timer

    MOV     CRCL,#LOW(48333)
    MOV     T2CON,#71H                    ;TR2=1
    ORL      IEN0,#20H                    ;ET2=1   turn on T2 interrupt
```

3.5 Watchdog Timer(WDT)

In order to monitor the operation of the software and ensure that that a system reset is automatically generated to restore normalcy in the event of an abnormality, T5L OS 8051 is equipped with a software watchdog (WDT) timer whose counting reset time is set to 1 second (corresponding to 11.0592 MHz crystal).

Once the WDT is turned on, the software needs to feed the dog in the counting reset time, otherwise a system reset will occur.

Reset does not influence the contents of 32KB data memory and 0x008000-0x00FFFF space 128KB DGUS variable memory.

The relevant reference codes for WDT operations are as follows.

```
ORL       MUX_SEL,#02H              ;open WDT
ANL       MUX_SEL,#0FDH             ;close WDT
ORL       MUX_SEL,#01H              ;WDT reset (feeding dog)
```

3.6 IO

T5L OS 8051 has three 8 bit parallel ports(P0-P2) and a 4 bit parallel port(P3.0-P3.3), a total of 28 IO ports. P0-P0.7 of P0 port is multiplexed with UART and CAN interface, and multiplexing function or IO function can be selected through MUX_SEL register.

The input of all IO ports is always open, the input is floating, and there is no internal pull-up or pull-down. When IO port is used as output, it is necessary to open the output control, and the output driving ability can also be configured.

Because the IO interface of power-on reset process is input mode, when used as output, it must be pulled up or pulled down externally to ensure that the reliable output of power-on process is ensured, and will not malfunction due to IO floating.

P3.0 is external interrupt 0, and P3.1 is external interrupt 1 input. It can be configured as low level trigger (0) or jump down edge trigger (1) through IT0 and IT1.

In addition to the need to control the output switch, output strength and peripheral multiplexing power-on initialization configuration, the subsequent use of IO is consistent with the standard 8051.

IO-related SFR are shown in the following table:

SFR name	Address	Instructions
P0	0x80	Addressable by bit, same as standard 8051
P1	0x90	Addressable by bit, same as standard 8051
P2	0xA0	Addressable by bit, same as standard 8051
P3	0xB0	It can be addressable by bit, the same as standard 8051, only P3.0-P3.3 is valid.

POMDOUT	0xB7	P0 Interface Output Configuration:
		.7 0=P0.7Output shutdown 1=P0.7Output Open (push-pull output);
		.6 0=P0.6Output shutdown 1=P0.6Output Open (push-pull output);
		.5 0=P0.5Output shutdown 1=P0.5Output Open (push-pull output);
		.4 0=P0.4Output shutdown 1=P0.4Output Open (push-pull output);
		.3 0=P0.3Output shutdown 1=P0.3Output Open (push-pull output);
		.2 0=P0.2Output shutdown 1=P0.2Output Open (push-pull output);
		.1 0=P0.1Output shutdown 1=P0.1Output Open (push-pull output);
		.0 0=P0.0Output shutdown 1=P0.0Output Open (push-pull output);

P1MDOUT	0xBC	P1 interface output configuration
P2MDOUT	0xBD	P2 interface output configuration
P3MDOUT	0xBE	P3 interface output configuration

MUX_SEL	0xC9	Selection of peripheral reuse. .7 1 = CAN interface leads to P 0.2 and P 0.3; 0 = CAN interface is not lead out, and it works as an IO port. .6 1 = UART2 interface leads to P 0.4 and P 0.5; 0 = UART2 interface is not lead out, and it works as an IO port. .5 1 = UART3 interface leads to P 0.6 and P 0.7; 0 = UART3 interface is not lead out, and it works as an IO port. .4-.2 Reserved; .1 WDT control. 1=open; 0=close; .0 WDT feed dog. 1=feed the dog one time(The WDT count becomes zero, and the watchdog's overflowing time is 1);
PORTDRV	0xF9	Driver capability configuration of IO port output mode, 0x00 = 4mA 0x01=8mA(recommended values) 0x02=16mA 0x03=32mA

The relevant settings of IO port external interrupt are as follows.

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
EX0 interrupt	0x0003	IE0(TCON.1)	IEN0.0	IE0 is cleared automatically when interrupt response, corresponding to P3.0. IT0 (TCON.0) = 0 low level trigger interrupt, IT0 = 1 jump down edge trigger interrupt.
EX1 interrupt	0x0013	IE1(TCON.3)	IEN0.2	IE1 is cleared automatically when interrupt response, corresponding to P3.1. IT1 (TCON.2) = low level trigger interrupt, IT1 = 1 jump down edge trigger interrupt.

3.7 UART

3.7.1 UART2

UART2 related SFR are shown in the following table.

SFR name	Address	Instructions
MUX_SEL	0xC9	.6 1 = UART2 interface leads to P 0.4 and P 0.5; 0 = UART02 interface does not lead out, it works as an IO port .
SCONO	0x98	UART2 control interface, the same as standard 8051, can be addressable by bit. .7=SM0 .6=SM1 .5=SM2(multiprocessor communication bit) .4=REN0 .3=TB80 .2=RB80 .1=TI0 .0=RI0.
SBUF0	0x99	UART2 transceiver data interface
ADCON	0xD8	Baud rate generator selection, 0x00 = T1 timer (standard 8051), 0x80 = SRELOH: L.
PCON	0x87	.7 SMOD baud rate frequency doubling selection. 0 = no frequency doubling 1 = frequency doubling.

SRELOH	0xBA	When ADCON = 0x80, SRELOH:L is used to set the baud rate without occupying T1.
		SMOD=0 SREOH:L=1024-CPU main frequency/(64*baud rate)
		SMOD=1 SREOH:L=1024-CPU main frequency/(32*baud rate)
		CPU main frequency = crystal frequency * 56/3, 11.0592 MHz crystal corresponds to 206.4384 MHz main frequency.

SRELOL	0xAA	

The relevant settings for UART2 interruption are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks

UART2 interrupt	0x0023	RIO(SCON0.0)
		TIO(SCON0.1)
			IEN0.4	After interruption, software needs to clear the interruption trigger mark.

3.7.2 UART3

UART3 related SFRs are shown in the following table.

SFR name	Address	Instructions
MUX_SEL	0xC9	.5 1 = UART3 interface leads to P 0.6, P 0.7, 0 = UART3 interface does not lead out, it is IO port.
SCON1	0x9B	UART3 control interface, it is not addressable by bit. .7 0=9bit UART; 1=8bit UART; .6 Undefined; .5=SM2(multiprocessor communication bit) .4=REN .3=TB8 .2=RB8 .1=TI .0=RI. Clearing the SCON1 bit mark requires two consecutive writings, such as ANL SCON1,#0FEH ANL SCON1,#0FEH
SBUF1	0x9C	UART3 transceiver data interface
SREL1H	0xBB	UART3 baud rate setting (CPU main frequency = crystal frequency * 56/3, 11.0592 crystal corresponding to 206.4384 MHz main frequency): SRE1H:L=1024-CPU main frequency/(32*baud rate)
SREL1L	0x9D	

The relevant settings for UART3 interruption are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
UART3 interrupt	0x0083	SCON1.0,SCON1.1	IEN2.0	After interruption, software needs to clear the interruption trigger mark.

3.7.3 UART4

UART4 related SFR are shown in the following table.

SFR name	Address	Instructions

SCON2T	0x96	UART4 sending control:
		.7 UART4 sending enable. 0=close;1=open;
		.6 0=8bit mode,1=9bit mode;
		.5 TB8, 9th bit sent in 9bit mode;
		.4-.1 Write0;
		.0 TI, send mark. The position at which the stop bit is sent.

SCON2R	0x97	UART4 receive control:
		.7 UART4 sending enable. 0=close;1=open;
		.6 Write0;
		.5 RB8, 9th bit received in 9bit mode;
		.4-.1 Write 0;
		.0 R RI, receive mark. Set when the stop bit is received when a valid stop bit is received.

SBUF2_TX	0x9E	UART4 sending data interface
SBUF3_RX	0x9F	UART4 receiving data interface
BODE2_DIV_H	0XD9	UART4 baud rate setting BODE2_DIV_H:L=CPU main frequency/(8*baud rate)
BODE2_DIV_L	0XD7	

The relevant settings of UART4 interruption are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
UART4 send interrupt	0x0053	SCON2T.0	IEN1.2	After interruption, clear the interruption trigger mark by software.
UART4 receive Interrupt	0x005B	SCON2R.0	IEN1.3	After interruption, clear the interruption trigger mark by software.

3.7.4 UART5

The relevant settings of UART5 are as follows:

SFR name	Address	Instructions

SCON3T	0xA7	UART5 sending control:
		.7 UART5 sending enables. 0=close;1=open;
		.6 0=8bit mode,1=9bit mode;
		.5 TB8, 9th bit sent in 9bit mode;
		.4-.1 Write 0;
		.0 TI, send flag. The position at which the stop bit is sent.

SCON3R	0xAB	UART5 receive control:
		.7 UART5 sending enables. 0=close;1=open;
		.6 Write 0;
		.5 TB8, 9th bit received in 9bit mode;
		.4-.1 Write 0;
		.0 R RI, receive mark. Set when the stop bit is received when a valid stop bit is received.

SBUF3_TX	0xAC	UART5 sending data interface
SBUF4_RX	0xAD	UART5 receiving data interface
BODE3_DIV_H	0xAE	UART5 baud rate setting BODE3_DIV_H:L=CPU main frequency/(8*baud rate)
BODE3_DIV_L	0xAF	

The relevant settings of UART5 interruption are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
UART5 send interrupt	0x0063	SCON3T.0	IEN1.4	After interruption, clear the interruption trigger mark by software.
UART5 receive Interrupt	0x006B	SCON3R.0	IEN1.5	After interruption, clear the interruption trigger mark by software.

3.8 CAN

The SFRs associated with the CAN interface are shown in the following table :

SFR name	Address	Instructions
MUX_SEL	0xC9	.7 1 = CAN interface leads to P 0.4 and P 0.5,0 = CAN interface does not lead out, it is IO port .

CAN_CR	0x8F	CAN interface control register
		.7 CAN interface enable.1=open; 0=close;
		.6 CAN interface mode. 1=software reset; 0=working normally;
		.5 Write 1 to configure data of CAN interface once (0xFF:0060-0xFF:0062), and clear after hardware processing;
		.4 Set speed mode. 1 = 1 sampling; 0 = 3 sampling;
		.3 Setting filter mode.1=double; 0=single;
		.2 Write 1 to send once, and clear after hardware processing (sending success, arbitration failure, EI (CAN_IR.3) occurred, software reset);
		.1-0 Undefined, write 0.

CAN_IR	0x91	CAN interface interrupt status register .7 RF_IF. Remote frame receiving interrupt mark, and clear after hardware placement; .6 CAN_RX_IF. CAN receiving interrupt mark, and clear after hardware placement. During the placement, the hardware can no longer update the data; .5 CAN_RX_IF, CAN sending interrupt mark, and clear after hardware placement; .4 OI, receiving overflow markers, hardware blanking, software clearance is required; .3 EI, error mark. CAN_ET[4:0] has an error generated when this bit is pulled high and needs to be cleared by software .2 JI, send arbitration mark. 1 = send arbitration failure; 0 = send arbitration success; .1-.0 Undefined.

CAN_ET	0xE8	CAN interface error type register and clear after hardware placement. Addressable by bit.
		.7 Node suspension identification;
		.6 Active error identification;
		.5 Passive error identification;
		.4 CRC check error identification;
		.3 Response error identification;
		.2 Format error identification;
		.1 Bit filling error identification;
		.0 Bit error identification.

The CAN communication interface is defined in terms of DGUS variable space. The related variables are as below.

Address	Position	Length	Definition	Instructions
0xFF:0060	D3	1	BRP	BRP: Baud rate frequency divider register BTR0: [7:5] is the synchronous jump width sjw, [3:0] prop propagation time T0. BTR1: [7:4] is phase buffer segment 1, T1, [3:0] is phase buffer segment 2, T2. T0+T1+T2=CPU main frequency/(baud rate*(BRP*2+1))-4
	D2	1	BTR0	
	D1	1	BTR1	
	D0	1	Undefined, write 0	
0xFF:0061	D3:D0	4	ACR3:0	Acceptance code register

0xFF:0062	D3:D0	4	AMR3:0	Acceptance code register
0xFF:0063	D3	1	RXERR	Receiving error count register
	D2	1	TXERR	Sending error count register
	D1:D0	2	Undefined	Write 0
0xFF:0064	D3	1	CAN_TX_BUFFER	[7]IDE,[6]RTR,[3:0]—DLC, frame data length
	D2:D0	3	Undefined	
0xFF:0065			ID	ID, 29bit is valid for extended frame and 11bit is valid for standard frame.
	D3	1		ID, first byte, standard frame and extended frame.
	D2	1		ID, second byte. The [7:5] is the high 3 bits of the standard frame and the 2nd byte of the extended frame.
	D1	1		ID, third byte, invalid standard frame, third byte of extended frame.
	D0	1		ID, fourth byte, invalid standard frame, [7:3] extended frame high 5 bit.
0xFF:0066	D3:D0	4	Data	Send data,DATA1-DATA4
0xFF:0067	D3:D0	4	Data	Send data,DATA5-DATA8
0xFF:0068	D3	1	CAN_RX_BUFFER	[7]IDE,[6]RTR,[3:0]—DLC, frame data length
	D2:D0	3	Undefined	
0xFF:0069			ID	ID, 29bit is valid when expanding frames, 11bit is valid for standard frames.
	D3	1		ID, first byte, standard frame and extended frame.
	D2	1		ID, the second byte, [7:5] standard frame height 3 bit, and the second byte of the extended frame.
	D1	1		ID, third byte, invalid standard frame, third byte of extended frame.
	D0	1		ID, fourth byte, invalid standard frame, [7:3] extended frame height 5 bit.
0xFF:006A	D3:D0	4	Data	Receive data,DATA1-DATA4
0xFF:006B	D3:D0	4	Data	Receive data,DATA5-DATA8

The relevant settings of CAN interface interruption are as follows:

Interrupt type	Program entry address	Trigger marker	Interrupt enabling control	Remarks
CAN interface interruption	0x004B	CAN_IR	IEN1.1	After interruption, software needs to clear the interruption trigger mark.

3.9 Interrupt System

3.9.1Interrupt Control SFR

T5L OS CPU has 12 interrupts. The related control SFRs list is as follows:

SFR name	Address	Instructions
IEN0	0xA8	The interrupt enable controller 0 can be addressed by bit. .7 Interrupt master control bit. 0=all interrupts closed; 1=whether an interrupt is opened is controlled by the control bit of each interrupt; .6 Must write 0; .5 ET2 T2 timer interrupt enable control bit; .4 ES0 UART2 interrupt enable control bit; .3 ET1 T1 timer interrupt enable control bit; .2 EX1 external interrupt 1 (P3.1 pin) interrupt enabling control bit; .1 ET0 T0 timer interrupt enable control bit; .0 EX0 external interrupt 1 (P3.0 pin) interrupt enabling control bit.
IEN1	0xB8	The interrupt enable controller 1 can be addressed by bit. .7-.6 Write 0; .5 ES3R UART5 receiving interrupt enabled control bit; .4 ES3T UART5 receiving interrupt enabled control bit; .3 ES2R UART4 receiving interrupt enabled control bit; .2 ES2R UART4 receiving interrupt enabled control bit; .1 ECAN CAN communication interrupt enabling control bit; .0 Write 0.
IEN2	0x9A	Interrupt enabling controller 2 .7-.1 Must write 0 .0 ESI USRT3 interrupt enabling control bit
IEN3	0xD1	Interrupt enabling controller 3, must write 0x00
IP0	0xA9	Interrupt priority controller 0
IP1	0xB9	Interrupt priority controller 1

3.9.2 Interrupt Priority

The interrupt priority of T5L OS CPU is handled according to the following rules.

(1) Twelve interrupts are divided into six groups with two interrupts in each group. The priority in the same group is fixed. For example, the priority of interrupt 0 is higher than that of UART3, as shown in the table below.

Grouping	IP0 correspondence	IP1 correspondence	Interrupt correspondence	
			High priority	Low priority
G0	.0	.0	External interrupt 0	UART3 interrupt
G1	.1	.1	T0 timer interrupt	CAN communication interrupt
G2	.2	.2	External interrupt 1	UART4 send interrupt
G3	.3	.3	T1 timer interrupt	UART4 receive interrupt
G4	.4	.4	UART2 interrupt	UART5 send interrupt
G5	.5	.5	T2 timer interrupt	UART5 receive interrupt

(2) There are four levels of priority among the six groups, which can be configured by the corresponding bits of IP0 and IP1 according to the table below.

Inter group priority	IP1 counterpart	IP0 counterpart
3(highest)	1	1
2	1	0
1	0	1
0(lowest)	0	0

For example, if you want to set the G2 group's T2 timer interrupt and UART5 receive interrupt priority to the highest, you can set 1P1=0x20, 1P0=0x20.

(3) If the configurations have the same priority (IP1 = 0x00 IP0 = 0x00), the G0 group has the highest priority and the G5 group has the lowest priority. The interrupt priority weights with the same configuration are as follows:

Weight	11	10	9	8	7	6	5	4	3	2	1	0
Priority	Maximum											Minimum
Interrupt	EX0	UART3	ET0	CAN	EX1	UART4-TX	ET1	UART4-RX	UART2	UART5-TX	ET2	UART5-RX

(4) High priority interrupts can be embedded in low priority interrupts (i.e. interrupts with small weights can be interrupted by interrupts with large weights), and at most four levels can be embedded.

Note

The T5L OS CPU is fast (1uS can execute 130-150 instructions on average) and the interrupt execution time is short, so the real-time performance is already very high.

It is not recommended to use interrupt embedding that makes program architecture more complex.

Users can directly turn off interrupts (EA=0) when each interrupt service program is executed, and turn on interrupts (EA=1) when exiting.

3.10 T5L ASIC 8051 Instruction Set

Instruction format	Instruction length	Instruction cycle	Instruction format	Instruction length	Instruction cycle
ADD/ADDDC A,Rn	1	1	MOV @Ri,direct	2	2
ADD/ADDDC A,direct	2	2	MOV @Ri,#data	2	2
ADD/ADDDC A,@Ri	1	2	MOV DPTR,#data16	3	3
ADD/ADDDC A,@data	2	2	MOVC A,@A+DPTR	1	3
SUBB A,Rn	1	1	MOVC A,@A+PC	1	3
SUBB A,direct	2	2	MOVX A,@Ri	1	3
SUBB A,@Ri	1	2	MOVX A,@DPTR	1	3
SUBB A,#data	2	2	MOVX A,@Ri,A	1	3
INC/DEC A	1	1	MOVX @DPTR,A	1	3
INC/DEC Rn	1	1	PUSH/POP	1	2
INC/DEC direct	2	2	XCH A,Rn	1	1
INC/DEC @Ri	1	2	XCH A,direct	2	2
INC DPTP	1	1	XCH A,@Ri	1	2
MUL AB	1	4	XCHD A,@Ri	1	2
DIV AB	1	4	ACALL addr11	2	2
DA A	1	1	LCALL addr16	3	3
ANL/ORL/XRL A,Rn	1	1	RET/RETI	1	4
ANL/ORL/XRL A,direct	2	2	AJMP addr11	2	2
ANL/ORL/XRL A,@Ri	1	2	LJMP addr16	3	3
ANL/ORL/XRL A,#data	2	2	SJMP rel	2	2
ANL/ORL/XRL direct,A	2	2	JMP @A+DPTR	1	3
ANL/0RL/XRL direct,#data	3	3	JZ/JNZ/JC/JNC rel	2	3
CLR A	1	1	JB/JNB/JBC	3	4
CPL A	1	1	CJNE A,direct,rel	3	4
RL/RR A	1	1	CJNE A,#data,rel	3	4
RLC/RRC A	1	1	CJNE Rn,#data,rel	3	4
SWAP A	1	1	CJNE @Ri,#data,rel	3	5
MOV A,Rn	1	1	DJNZ Rn,rel	2	3
MOV A,direct	2	2	DJNZ direct,rel	3	4
MOV A,@Ri	1	2	NOP	1	1
MOV A,#data	2	2	CLR/SETB C	1	1
MOV Rn,A	1	1	CLR/SETB bit	2	2
MOV Rn,direct	2	2	CPL C	1	2
MOV Rn,#data	2	2	CPL bit	2	2
MOV direct,A	2	2	ANL C,bit	2	2
MOV direct,Rn	2	2	ANL C,/bit	2	2
MOV direct1,direct2	3	3	ORL C,bit	2	2
MOV direct,@Ri	2	2	ORL C,/bit	2	2
MOV direct,#data	3	3	MOV C,bit	2	2

MOV @Ri,A	1	1	MOV bit,C	2	2

CPU main frequency = crystal frequency * 56/3.

11.0592MHz crystal corresponds to 206.438MHz main frequency, corresponding to an instruction cycle (1T) of 4.844nS.

Example

Under 11.0592 MHz crystal, the following code will output 206.4384/((2+2)*2)=25.8048 MHz square wave on the P3.3 IO interface.

```
OUTTEST:CPL     P3.3            ;2T
        AJMP    OUTTEST         ;2T
```

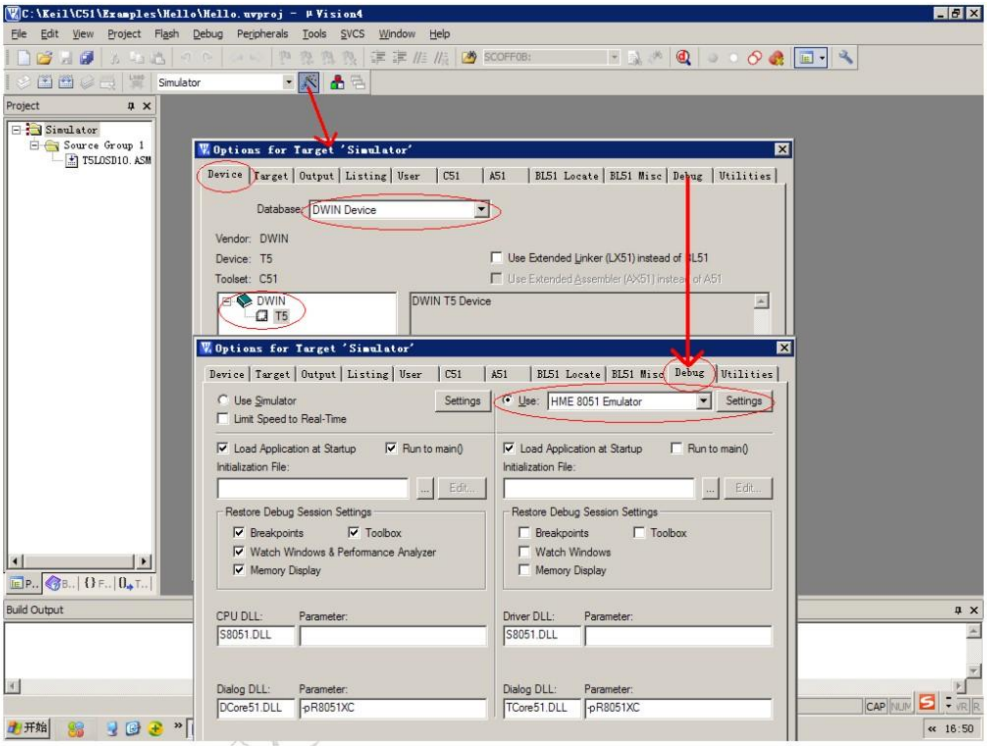
4 Simulation Debug

With the help of HME05 simulator (need to install the corresponding USB driver), users can connect T5L JMARK interface and debug code IAP and run simulation under Keil development environment.

Note

(1) JMARK interface must be selected to OS CPU, and must be JMARK mode, that is, OS/GUI (PIN#32) = 0 JMARKS (PIN#34) = 1 .And when the 4.3 inch evaluation board is used for simulation, the jumper pad on the JMARK interface side is disconnected.

(2) The AGDI driver is installed to enable Keil to support T5 and HME05 simulators. After installation, select and configure according to the following figure. After installation, copy the header file of T5L OS CPU (*. INC or *. h) to KEIL/C51/INC/DWIN directory.


C:\Keil\C51\Examples\Hello\Hello.uvproj - µVision4
File Edit View Project Flash Debug Peripherals Tools SVCS Window Help
SCOFF0B:
Simulator
Project
Simulator
Source Group 1
T5LOSD10.ASM
Build Output
Options for Target 'Simulator'
Device | Target | Output | Listing | User | C51 | A51 | BL51 Locate | BL51 Misc | Debug | Utilities
CAP NUM
16:50

Options for Target 'Simulator' – Device

Database: DWIN Device

Vendor: DWIN

Device: T5

Toolset: C51

□ Use Extended Linker (LX51) instead of BL51

□ Use Extended Assembler (AX51) instead of A51

DWIN / T5 — DWIN T5 Device

Options for Target 'Simulator' – Debug

○ Use Simulator — Settings

□ Limit Speed to Real-Time

☑ Load Application at Startup ☑ Run to main()

Initialization File:

Restore Debug Session Settings: ☑ Breakpoints ☑ Toolbox ☑ Watch Windows & Performance Analyzer ☑ Memory Display

CPU DLL: S8051.DLL Parameter:

Dialog DLL: DCore51.DLL Parameter: -pR8051XC

● Use: HME 8051 Emulator — Settings

☑ Load Application at Startup □ Run to main()

Initialization File:

Restore Debug Session Settings: □ Breakpoints □ Toolbox □ Watch Windows □ Memory Display

Driver DLL: S8051.DLL Parameter:

Dialog DLL: TCore51.DLL Parameter: -pR8051XC

(3) Before setting the breakpoint to read the content of data storage (XDATA), DPC = 0x00 must be ensured, otherwise the data will be misaligned.

(4) Before simulation, it is necessary to ensure that the OS CPU code address 0x00F8 of T5L Flash starts with 0xFF FF 44 57 49 4E 54 35, otherwise the JMARK interface of OS CPU is prohibited and HME05 emulator cannot be connected.

(5) When porting code from standard C or other 8051 platforms, users should be aware that the SFR header file should be loaded with the T5L compliant .INC or .H file during compilation. If the SFR definition in the customer code is different from the T5L definition, you can modify the code or the T5L SFR header file to keep the same.

(6)HME05 realizes hardware simulation by downloading code to Code RAM of T5L OS CPU. The code is not burned to Flash in the chip. To burn code into the chip, it is necessary to use SD card interface or UART1 debugging interface. During SD card download, T5L underlying software will automatically change the location of OS code 0x00F8 to 0x0000 (JMARK interface is forbidden) 44 57 49 4E 54 35.

T5L OS CPU adopts the standard 8051 architecture. Except for the slight difference between SFR and extended peripheral access, the instruction set is identical.

When porting the user's original 8051 code, it can be done quickly by taking care of the following aspects.

(1) According to the hardware design, after reset, the startup.A51 (C51 startup code) or initcpu () program provided by DWIN is used to simply modify and configure T5L-specific SFR and parameter settings.

Typical differences are in IO, main frequency of timer, baud rate, interrupt, WDT, etc.

(2) T5L IO output mode is controllable. When switching between input and output modes, PxMDOUT registers should be configured accordingly, otherwise errors will occur.

(3) Turn off interrupt embedding. EA = 0 for each interrupt service program and EA=1when exiting.

(4) When using off-chip RAM (XRAM) as data storage in the code, note that the starting address of T5L 32KB data RAM should start at 0x8000.

(5) Code 0x00F8 position plus 0xFFFF (or 0x0000 forbid JMARK interface) 44 57 49 4E 54 35. (6) Optimize the algorithm of the original code with MDU hardware computing acceleration; move the UI and Internet (e.g., access to DWIN cloud through DWIN WiFi module) functions to DGUS II platform, user code can be processed by simply reading and writing DGUS variable space, which can significantly improve product performance and enhance R&D efficiency.

5 EK043 Evaluation Board

EK043 evaluation board uses T5L1 to drive 4.3 inch 480*272 TFT screen, and matches capacitor touch screen. Its main features include:

(1) 11.0592 MHz crystal, 206.4384 MHz main frequency operation, support HME05 simulator.

(2) External 16MB SPI Flash, run standard DGUS II UI;

(3) Power supply and debug by USB interface;

(4) The OS CPU 8051 program can be downloaded through the USB interface and automatically reset by hardware;

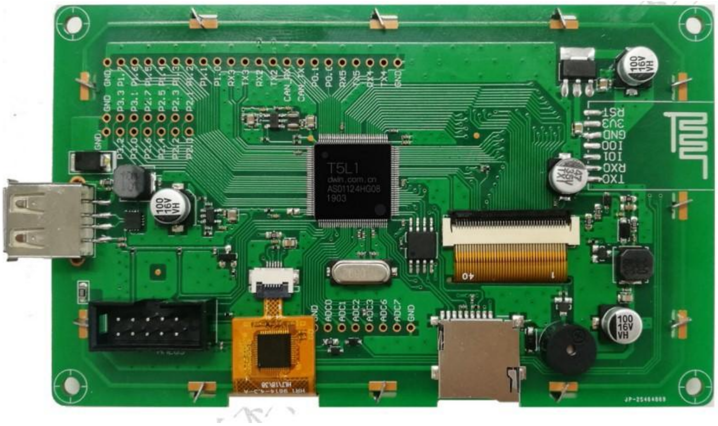
(5) Read and write DGUS variable memory and download pictures and word libraries through USB interface to facilitate DGUS debugging;

(6) The assembly location of DWIN WiFi module is reserved for easy access to DWIN cloud (both WiFi module and USB occupy the UART with a baud rate of 921600 bps);

(7) All interfaces of T5L OS CPU are lead out with 2.54mm spacing through-hole bonding pad and marked.

(8) 6 AD, ADC0-ADC3, ADC6-ADC7 which can be used by users are lead out and marked with 2.54mm spacing through-hole bonding pad.

(9)



If you have any questions during the use of this document or DWIN products, or want to know more about the latest information of DWIN products, please contact us.

Hotline: 400 018 9008

Whatsapp: + 8619192113166

DWIN website: https://www.dwin-global.com/

Thank you for your continuous support to DWIN, your support is the driving force for our progress!