

W7500x Errata Sheet

Document History

Ver 1.0.0 (July.11, 2016)	First release (erratum 1) - W7500x I2C
Ver 1.1.0 (Jun.18, 2018)	erratum 2 - W7500P Transmission Delay Case
Ver 1.2.0 (May.12. 2019)	Erratum 3 - IAP Function Call Failure Erratum 4 - Two Image Banks Failure Erratum 5 - Cold Booting Failure

© 2016 WIZnet Co., Inc. All Rights Reserved.

For more information, visit our website at <http://www.wiznet.co.kr>

Erratum 1	
W7500x I2C	
Phenomenon	I2C에 반복적인 DATA가 들어올 경우 이후에는 멈춘다.
Condition	<p>W7500x가 I2C통신중에 같은 데이터를 연속적으로 수신하게 되면, 처음 데이터만 수신하고 이후의 같은 데이터들은 인식하지 못하고 데이터를 버리는 문제. 그리고 이로 인한 데이터 손실이 발생한다.</p>
Solution & Recommendation	<p>이 Erratum을 피하기 위해서는 W7500x의 I2C를 GPIO로 변경하여 사용해야 한다. GPIO로 사용할 경우, SCL은 최대 100KHz를 초과할 수 없다.</p> <p>Example pseudo code:</p> <pre> Function Initialize_I2C () { ... scl_port_num = I2C_PORT(conf->scl); scl_pin_index = I2C_PIN_INDEX(conf->scl); sda_port_num = I2C_PORT(conf->sda); sda_pin_index = I2C_PIN_INDEX(conf->sda); //SCL setting GPIO_InitDef.GPIO_Pin = scl_pin_index; GPIO_InitDef.GPIO_Mode = GPIO_Mode_OUT; if(scl_port_num == 0) { GPIO_Init(GPIOA, &GPIO_InitDef); GPIO_SetBits(GPIOA, scl_pin_index); } ... //SDA setting </pre>

```

GPIO_InitDef.GPIO_Pin = sda_pin_index;
GPIO_InitDef.GPIO_Mode = GPIO_Mode_IN;
if(sda_port_num == 0)
{
    GPIO_Init(GPIOA, &GPIO_InitDef);
    GPIO_ResetBits(GPIOA, sda_pin_index);
}
...
}

/* SCL function */
Function I2C_SCL()
{
    ...
    if(scl_port_num == 0)
    {
        if(data == 1)
            GPIO_SetBits(GPIOA, scl_pin_index);
        else
            GPIO_ResetBits(GPIOA, scl_pin_index);
    }
    ...
}

/* SDA function */
Function I2C_SDA()
{
    ...
    if(sda_port_num == 0)
    {
        if(data == 1)
            GPIOA->OUTENCLR = sda_pin_index;
        else
            GPIOA->OUTENSET = sda_pin_index;
    }
    ...
}

```

```
}  
/* START function */  
Function I2C_START()  
void I2C_Start(I2C_ConfigStruct* conf)  
{  
    I2C_WriteBitSCL(conf, 1);  
    I2C_WriteBitSDA(conf, 1);  
  
    I2C_WriteBitSDA(conf, 0);  
    I2C_WriteBitSCL(conf, 0);  
}  
  
/* STOP function */  
Function I2C_STOP()  
void I2C_Stop(I2C_ConfigStruct* conf)  
{  
    I2C_WriteBitSCL(conf, 0);  
    I2C_WriteBitSDA(conf, 0);  
  
    I2C_WriteBitSCL(conf, 1);  
    I2C_WriteBitSDA(conf, 1);  
}  
.....
```

Erratum 2

W7500P Transmission Delay Case

W7500P가 일부 스위치 또는 라우터와의 연동 시 Half Duplex로 인식되면서 TX가 늦게 나가는 현상이 발견됨. (아래는 “TP_LINK AC750” 라우터로 테스트한 결과이다.)

Phenomenon

```

Command Prompt - ping 192.168.11.2 -t
Reply from 192.168.11.2: bytes=32 time=936ms TTL=128
Reply from 192.168.11.2: bytes=32 time=157ms TTL=128
Reply from 192.168.11.2: bytes=32 time=552ms TTL=128
Reply from 192.168.11.2: bytes=32 time=215ms TTL=128
Reply from 192.168.11.2: bytes=32 time=189ms TTL=128
Reply from 192.168.11.2: bytes=32 time=2177ms TTL=128
Reply from 192.168.11.2: bytes=32 time=493ms TTL=128
Reply from 192.168.11.2: bytes=32 time=594ms TTL=128
Reply from 192.168.11.2: bytes=32 time=85ms TTL=128
Reply from 192.168.11.2: bytes=32 time=453ms TTL=128
Reply from 192.168.11.2: bytes=32 time=586ms TTL=128
Reply from 192.168.11.2: bytes=32 time=586ms TTL=128
Reply from 192.168.11.2: bytes=32 time=466ms TTL=128
Reply from 192.168.11.2: bytes=32 time=578ms TTL=128
Reply from 192.168.11.2: bytes=32 time=574ms TTL=128
Reply from 192.168.11.2: bytes=32 time=476ms TTL=128
Reply from 192.168.11.2: bytes=32 time=34ms TTL=128
Reply from 192.168.11.2: bytes=32 time=563ms TTL=128
Reply from 192.168.11.2: bytes=32 time=492ms TTL=128
Reply from 192.168.11.2: bytes=32 time=555ms TTL=128
Reply from 192.168.11.2: bytes=32 time=1319ms TTL=128
Reply from 192.168.11.2: bytes=32 time=7ms TTL=128
Reply from 192.168.11.2: bytes=32 time=721ms TTL=128
Reply from 192.168.11.2: bytes=32 time=261ms TTL=128
Reply from 192.168.11.2: bytes=32 time=208ms TTL=128
    
```

위의 결과와 같이 핑 응답이 3 초 이상 지연되는 현상이 불규칙적으로 발생하는 것을 확인할 수 있다.

Condition

이와 같은 현상은 NC(Not Connected) 패드 및 칩 내부의 PHY MII 신호와 관련된 연결 문제로 인해 발생된다. (W7500P는 silicon-in-package 제품이며 내부에 W7500 및 Ethernet PHY를 포함한다.); duplex mode의 잘못된 검출로 인한 Collision 처리에 의해 전송 패킷이 지연된다.

Solution & Recommendation

이 현상을 해결하기 위해서, 사용자는 **반드시** 아래의 PHY 초기화 코드를 삽입해야 한다.

```

void PHY_Init(void)
{
#ifdef __W7500P__ // W7500P only
    // PB_12
    *(volatile uint32_t *) (0x41003070) = 0x61; // RXDV: set pull down
    // PB_05
    
```

```

*(volatile uint32_t *) (0x41002054) = 0x01;
*(volatile uint32_t *) (0x41003054) = 0x61;
// PB_06
*(volatile uint32_t *) (0x41002058) = 0x01;
*(volatile uint32_t *) (0x41003058) = 0x61;
// PHY reset pin pull-up (PD_06)
*(volatile uint32_t *) (0x410020D8) = 0x01;
*(volatile uint32_t *) (0x410030D8) = 0x02;
*(volatile uint32_t *) (0x45000004) = 0x40;
*(volatile uint32_t *) (0x45000010) = 0x40;
mdio_init(GPIOB, W7500x_MDC, W7500x_MDIO); // MDIO Init
mdio_write(GPIOB, PHYREG_CONTROL, CNTL_RESET); // PHY Reset
#endif
}
    
```

W7500P의 DUP pin(pin 15)는 스위치 또는 라우터와 연동된 duplex mode를 나타내며 value는 다음과 같다.

- DUP pin = '1' (HIGH) : Full duplex mode
- DUP pin = '0' (LOW) : Half duplex mode

Erratum 3	
IAP Function Call Issue	
Phenomenon	내부 Flash 메모리에 Write 하기위해서 IAP function을 호출할 때, Processor가 멈추는 현상
Condition	MCU 오류는 외부 Clock source 를 사용할 때 발생한다.
Solution & Recommendation	<p>해결책은 내부 8MHz Clock을 사용하는 것이다.</p> <ol style="list-style-type: none"> 1. Clock 소스를 내부 8MHz를 변경한다. 2. 내부 Flash write operation을 위한 IAP functions을 호출한다. 3. IAP 호출이 끝나면 원래 외부 Clock 소스로 복구한다. <p>다음은 IAP functions을 호출하기 전과 후의 예제이다.</p> <pre> static void flash_update_start(void) { /* System Core Clock Update */ SystemCoreClockUpdate_User(CLOCK_SOURCE_INTERNAL, PLL_SOURCE_8MHz, SYSTEM_CLOCK_8MHz); /* SysTick_Config */ SysTick_Config((GetSystemClock()/1000)); /* Backup Interrupt Set Pending Register */ temp_interrupt = (NVIC->ISPR[0]); (NVIC->ISPR[0]) = (uint32_t)0xFFFFFFFF; } /* System Core Clock Update - Restore */ static void flash_update_end(void) { /* System Core Clock Update */ SystemCoreClockUpdate_User(DEVICE_CLOCK_SELECT, DEVICE_PLL_SOURCE_CLOCK, DEVICE_TARGET_SYSTEM_CLOCK); /* SysTick_Config */ </pre>

```
SysTick_Config((GetSystemClock()/1000));  
  
/* Restore Interrupt Set Pending Register */  
(NVIC->ISPR[0]) = temp_interrupt;  
}
```

For reference,

https://github.com/Wiznet/WIZ750SR/blob/master/Projects/S2E_App/src/Configuration/segcp.c, line 1438

Erratum 4	
Two Image Banks Issue	
Phenomenon	<p>펌웨어 업그레이드와 같은 기능을 위해 내부 Flash 메모리가 두개의 image bank로 분리되어 있을 때, Bank 2의 코드를 수행하는 도중에 MCU가 멈추는 현상.</p>
Condition	<p>이 문제는 W7500x가 'interrupt vector table remap'을 지원하지 않기 때문에 발생하는 것이다.</p> <p>내부 Flash 메모리가 두개의 image bank로 나뉘어져 있을 때, Bank 2에 있는 코드가 수행될 때 인터럽트가 발생하면 interrupt vector table내의 reference가 유효하지 않아서 문제가 발생한다.</p>
Solution & Recommendation	<p>Bank 2에 있는 interrupt vector table을 Address 0에 복사함으로써 이 문제를 회피할 수 있다.</p> <p>이를 위해서는 다음 조건을 만족해야 한다.</p> <ol style="list-style-type: none"> 1. 대부분의 동작은 Bank 2 코드에서 수행되고 Bank 1 코드는 내부 Flash 를 업데이트 할 때만 사용되는 경우. 2. Bank 1 code 는 인터럽트를 사용하지 않는다. 3. Interrupt vector table 을 복사하는 동안에 Erratum 3 에 적용된 Solution 을 사용한다. <p>다음은 예제 코드이다.</p> <pre style="background-color: #1a3d54; color: #e0e0e0; padding: 10px;">void Copy_Interrupt_VectorTable(uint32_t start_addr) { uint32_t i; uint8_t flash_vector_area[SECT_SIZE]; for (i = 0x00; i < 0x08; i++) { flash_vector_area[i] = *(volatile uint8_t *) (0x00000000+i); } for (i = 0x08; i < 0xA8; i++) {</pre>

```

    /* Actual address range; Interrupt vector table is located here */
    flash_vector_area[i] = *(volatile uint8_t*)(start_addr+i);
  }

  for (i = 0xA8; i < SECT_SIZE; i++) {
    flash_vector_area[i] = *(volatile uint8_t*)(0x00000000+i);
  }

  /* Global interrupt disabled */
  __disable_irq();

  /* Erase the interrupt vector table area : Sector 0 */
  DO_IAP(IAP_ERAS_SECT, 0x00000000, 0, 0);

  /* Write the applicaion vector table to 0x00000000 */
  DO_IAP(IAP_PROG, 0x00000000, flash_vector_area , SECT_SIZE);

  /* Global interrupt enabled */
  __enable_irq();
}

```

For reference,

https://github.com/Wiznet/WIZ750SR/blob/master/Projects/S2E_Boot/src/main.c, line 532

Erratum 5	
Cold Booting Failure	
Phenomenon	Cold booting failure
Condition	입력전원이 Operating supply voltage(2.7V) 까지 상승하는 데까지 걸리는 시간이 20ms 보다 길거나, 전원 인가전에 일반 Peripheral pins에 1V이상의 전압이 걸려있을 때 MCU의 내부 reset logic이 정상동작 하지 않아서 발생하는 현상.
Solution & Recommendation	<p>POR(Power On Reset)칩을 추가하는 것이 가장 좋은 솔루션.</p> <div style="border: 2px solid red; padding: 10px; margin: 10px 0;"> <p style="text-align: center; color: blue; font-weight: bold;">W7500P requires an external Power On Reset (POR) IC.</p> <p style="text-align: center; font-weight: bold; color: blue;">RESET</p> </div> <p>POR 을 사용하지 않기 위해서는 입력전원 상승 시간이 20ms 이하, 일반 Peripheral pins 과 연결된 회로가 동시에 전압이 공급되는 조건이 되어야 함.</p> <p>For reference, https://github.com/Wiznet/Hardware-Files-of-WIZnet/tree/master/01_iMCU/W7500P/Reference%20Schematic</p>