STM8TL52x4 STM8TL53x4 Errata sheet

STM8TL52x4 and STM8TL53x4 Rev Z device limitations

Silicon identification

This errata sheet applies to revision “Z” of the STMicroelectronics STM8TL5xx4 devices.

Table 1. Device identification

Sales type	Revision code marked on device(1)
STM8TL52x4, STM8TL53x4	“Z”

1. Refer to STM8TL52x4 and STM8TL53x4 datasheet for the device marking.

Table 2. Device summary

Reference	Part number
STM8TL52x4	STM8TL52F4, STM8TL52G4
STM8TL53x4	STM8TL53C4, STM8TL53F4, STM8TL53G4

Contents

1 Product evolution

Table 3 gives a summary of the fix status.

Legend for *Table 3*:

A = workaround available;
N = no workaround available;
P = partial workaround available;
‘-’ and grayed = fixed.

Table 3. Product evolution summary

Section	Limitation	Rev Z
Section 1.1: Core limitations	*Section 1.1.1: Interrupt service routine (ISR) executed with priority of main process*	N
	Section 1.1.2: Main CPU execution is not resumed after an ISR resets the AL bit	A
	Section 1.1.3: Unexpected DIV/DIVW instruction result in ISR	A
	Section 1.1.4: Incorrect code execution when WFE execution is interrupted by ISR	A
Section 1.2: USART peripheral limitations	*Section 1.2.1: IDLE frame detection not supported in the case of a clock deviation*	N
	Section 1.2.2: PE flag can be cleared in Duplex mode by writing to the data register	A
	Section 1.2.3: PE flag is not set in Mute mode using address mark detection	N
	Section 1.2.4: IDLE flag is not set using address mark detection	N

1.1 Core limitations

1.1.1 Interrupt service routine (ISR) executed with priority of main process

Description

If an interrupt is cleared or masked when the context saving has already started, the corresponding ISR is executed with the priority of the main process.

Workaround

None.

No fix is planned for this limitation.

1.1.2 Main CPU execution is not resumed after an ISR resets the AL bit

Description

If the CPU is in wait for interrupt state and the AL bit is set, the CPU returns to wait for interrupt state after executing an ISR. To continue executing the main program, the AL bit must be reset by the ISR. When AL is reset just before exiting the ISR, the CPU may remain stalled.

Workaround

Reset the AL bit at least two instructions before the IRET instruction.

No fix is planned.

1.1.3 Unexpected DIV/DIVW instruction result in ISR

Description

In very specific conditions, a DIV/DIVW instruction may return a false result when executed inside an interrupt service routine (ISR). This error occurs when the DIV/DIVW instruction is interrupted and a second interrupt is generated during the execution of the IRET instruction of the first ISR. Under these conditions, the DIV/DIVW instruction executed inside the second ISR, including function calls, may return an unexpected result.

The applications that do not use the DIV/DIVW instruction within ISRs are not impacted.

Workaround 1

If an ISR or a function called by this routine contains a division operation, the following assembly code should be added inside the ISR before the DIV/DIVW instruction:

```
push cc
pop a
and a,#$BF
push a
pop cc
```

This sequence should be placed by C compilers at the beginning of the ISR using DIV/DIVW. Refer to your compiler documentation for details on the implementation and control of automatic or manual code insertion.

Workaround 2

To optimize the number of cycles added by workaround 1, you can use this workaround instead. Workaround 2 can be used in applications with fixed interrupt priorities, identified at the program compilation phase:

```
push #value
pop cc
```

where bits 5 and 3 of #value have to be configured according to interrupt priority given by I1 and I0, and bit 6 kept cleared.

In this case, compiler workaround 1 has to be disabled by using compiler directives.

No fix is planned for this limitation.

1.1.4 Incorrect code execution when WFE execution is interrupted by ISR

Description

Two types of failures can occur:

- Case 1: In case WFE instruction is placed in the two MSB of the 32-bit word within the program memory, an event which occurs during the WFE re-execution cycle when returning from ISR handler will cause an incorrect code execution.
- Case 2: An interrupt request, which occurs during the WFE execution cycle will lead to incorrect code execution. This is also valid for the WFE re-execution cycle, while returning from an ISR handler.

The above failures have no impact on the core behavior when the ISR request or events occur in Wait for Event mode itself, out of the critical single cycle of WFE instruction execution.

Workaround

- Case 1: Replace the WFE instruction with:

```
<...>
  WFE
  JRA next
next
<...>
```

- Case 2: It is recommended to avoid any interrupts before WFE mode is entered. This can be done by disabling all interrupts before the device enters Wait for event mode.

```
<...>
  SIM
  WFE
  RIM
<...>
```

  This workaround is also valid for case 1.
- Another solution is to ensure no interrupt request occurs during WFE instruction execution or re-execution cycle by proper application timing.

No fix is planned for this limitation.

1.2 USART peripheral limitations

1.2.1 IDLE frame detection not supported in the case of a clock deviation

Description

An idle frame cannot be detected if the receiver clock is deviated.

If a valid idle frame of a minimum length (depending on the M and Stop bit numbers) is followed without any delay by a start bit, the IDLE flag is not set if the receiver clock is deviated from the RX line (only if the RX line switches before the receiver clock).

Consequently, the IDLE flag is not set even if a valid idle frame occurred.

Workaround

None.

1.2.2 PE flag can be cleared in Duplex mode by writing to the data register

Description

The PE flag can be cleared by a read to the USART_SR register followed by a read or a write to the USART_DR register.

When working in duplex mode, the following event can occur: the PE flag set by the receiver at the end of a reception is cleared by the software transmitter reading the USART_SR (to check TXE or TC flags) and writing a new data into the USART_DR.

The software receiver can also read a PE flag at ‘0’ if a parity error occurred.

Workaround

The PE flag should be checked after the end of reception and before transmission.

1.2.3 PE flag is not set in Mute mode using address mark detection

Description

If, when using address mark detection, the receiver recognizes in Mute mode a valid address frame but the parity check fails, it exits from the Mute mode without setting the PE flag.

Workaround

None.

1.2.4 IDLE flag is not set using address mark detection

Description

The IDLE flag is not set when the address mark detection is enabled, even when the USART is in Run mode (not only in Mute mode).

Workaround

None.

2 Revision history

Table 4. Document revision history

Date	Revision	Changes
13-Oct-2011	1	Initial release.
23-Mar-2012	2	Added STM8TL52G4, STM8TL53F4 and STM8TL52F4 devices.
18-Mar-2015	3	Added a table footnote about the device marking to *Table 1*. Removed the appendix A showing the revision code on device marking.