

**S $\mu$ MMIT Anomaly for SKIP Commands Following A 1553 Message Resulting in a No Response Time-Out Message Error****Introduction:**

The UT69151 S $\mu$ MMIT protocol device (See Table 1 for at-risk S $\mu$ MMIT products) operating as a bus controller provides the user with a handy software and bus message scheduling instruction set. The available scheduling instructions are the SKIP, and Load Minor Frame Timer (LMFT) instructions. The SKIP instruction allows the user to halt subsequent command block execution by the S $\mu$ MMIT bus controller for a period of time ranging from 41.67ns to 2.73ms. Because every non-1553 command block requires about 30 $\mu$ s for the S $\mu$ MMIT to execute, loading the SKIP command with a timer value less than the required execution time will have no effect; therefore, the minimum time that the SKIP command can force the S $\mu$ MMIT to wait is 30 $\mu$ s. This command is frequently used to increase the inter-message gap between 1553 bus messages by forcing the S $\mu$ MMIT to wait a specified period of time before initiating another 1553 bus message. The second scheduling instruction provided by the S $\mu$ MMIT is the LMFT instruction. This instruction allows the user to schedule the time that the S $\mu$ MMIT bus controller will execute blocks of instructions. The LMFT instruction can establish the execution of command blocks within a time period ranging from 64 $\mu$ s to 4.194s. The minor frame scheduling is frequently used to control the duty cycle (or frequency) of 1553 activity.

**Table 1: S $\mu$ MMIT Protocol Devices Applicable to the SKIP Anomaly**

<b>Device Name</b>	<b>Generic Part #</b>	<b>SMD#</b>	<b>Protocol Die PIC#</b>	<b>Protocol Chip PIC#</b>
S $\mu$ MMIT E	UT69151 E	5962-92118 02, 04	SJ02, JN01, TJ02, JS01	SJ02, TJ02, JS01
S $\mu$ MMIT DXE	UT69151 DXE	5962-94663 05, 011	TJ02, JS01 JN01	MM010A, MM023A, MM025A, MM027A
S $\mu$ MMIT LXE12	UT69151 LXE12	5962-94663 06	SJ02, JS01, JN01	MM012A, MM024A, MM026A
S $\mu$ MMIT LXE15	UT69151 LXE15	5962-94663 04, 10	TJ02	MM011A
S $\mu$ MMIT XTE	UT69151 XTE	5962-94758 05	SJ02	MM013A
S $\mu$ MMIT XTE12	UT69151 XTE12	5962-94758 06	SJ02	MM014A
S $\mu$ MMIT XTE15	UT69151 XTE15	5962-94758 04	SJ02	MM015A

### **Problem Description:**

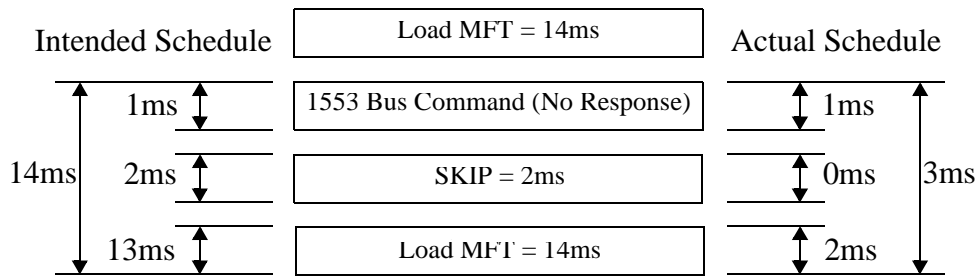
When a SKIP opcode follows a command block requiring MIL-STD-1553 bus communication (see Table 2), and the 1553 communication results in a bus time-out (e.g. no response, short message length, etc.), the specified “SKIP TIME” does not elapse before the next command block is processed. Furthermore, if this situation occurs within a minor frame as determined by a LMFT command, the minor frame will complete in the amount of time specified by the “SKIP TIME” which will effectively lose any excess time established by the LMFT command. This scenario is depicted by the drawing in figure 1.

**Table 2: Opcodes Requiring 1553 Communication and are at Risk of Bus Time-Out**

<b>Opcode (Binary)</b>	<b>1553 Communication Types<sup>1</sup></b>
0100	BC to RT, RT to BC, RT to RT, Broadcast
0101	BC to RT, RT to BC, RT to RT, Broadcast
0110	BC to RT, RT to BC, RT to RT, Broadcast
0111	BC to RT, RT to BC, RT to RT, Broadcast
1000	BC to RT, RT to BC, RT to RT, Broadcast
1001	BC to RT, RT to BC, RT to RT, Broadcast

Note 1: All MIL-STD-1553 broadcast commands have a built-in time-out mechanism, and will therefore cause the SKIP anomaly to occur.

If the 1553 bus command in figure 1 completes without a bus time-out, the minor frame time will execute with the time scheduling shown under the column labeled “Intended Schedule”. If the 1553 message exits with a bus time-out, then the minor frame command schedule will execute according to the timing under the “Actual Schedule” label. This problem is solved by establishing a new MFT schedule that avoids the use of the SKIP command, or by implementing a set of command blocks that takes advantage of the MFT and SKIP relationship and therefore allows the program to wait the specified “SKIP TIME” and then restores the remaining minor frame time.



**Figure 1.** 1553 Scheduling with the MFT and SKIP Commands when a 1553 No Response Occurs

## **Solution:**

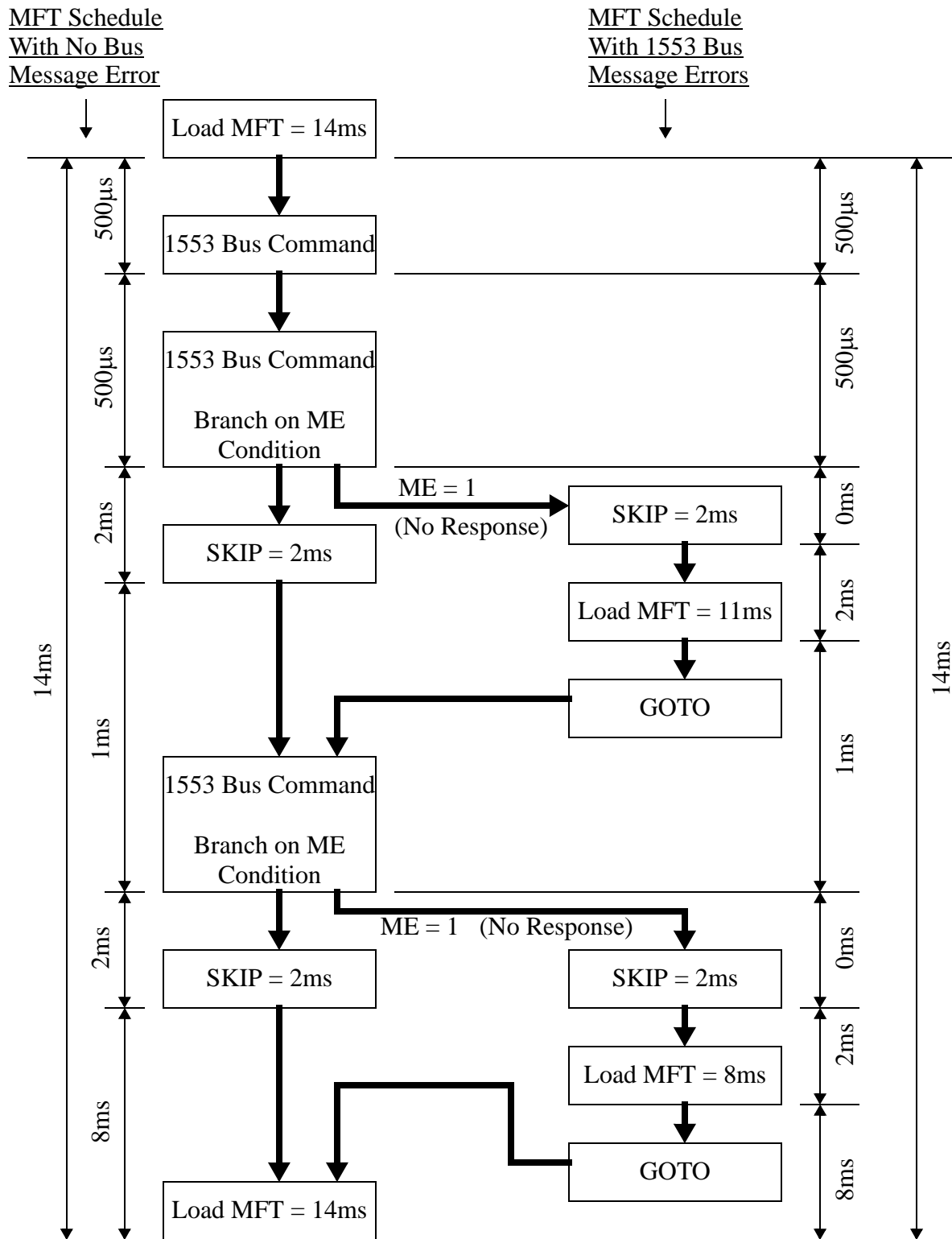
UTMC has identified 3 solutions that allow the user to maintain the desired minor frame time sequencing and achieve the desired “SKIP TIME” independent of the exit status of a MIL-STD-1553 message. The user should understand that each of these solutions has a different set of benefits and must be evaluated for their applicability to the current application. The last solution provides a method to guarantee the desired “WAIT TIME” for application code executing outside of a minor frame time block.

The first solution to circumventing the SKIP anomaly is to implement the 1553 command string such that all SKIP commands are protected by a subroutine that restores the minor frame timer in the event of a 1553 message error preceding the SKIP command (see figure 2). This will allow the SuMMIT to wait the amount of time specified in the SKIP command, and maintain the original minor frame timing schedule. In order to implement this solution, the software designer determines the amount of time that elapses from the start of the minor frame time block to the point that the minor frame time value is restored. Specifically, the minor frame time restoration value is calculated as follows:

$$\text{New MFT} = \text{Old MFT} - (\text{SuMMIT Execution Time to SKIP Command} + \text{SKIP TIME})$$

The “SuMMIT Execution Time to SKIP Command” is the amount of time needed for the SuMMIT BC to execute all instructions after the Load MFT instruction and before the SKIP instruction which is at risk of the anomaly. This will vary depending on the number of 1553 commands, 1553 message lengths, and the number of retries attempted. You can assume 30µs of execution time for each non-1553 instruction. Figure 2 below depicts the new program flow that will solve the SKIP/MFT anomaly.

The benefit of this solution is that it provides a method to establish the desired “SKIP TIME” while maintaining the overall time schedule of minor frame blocks. However, this solution is only applicable for designs that include a SµMMIT protocol device listed in table 1 above. Using this code for SµMMIT devices not listed in the above table will adversely affect the user’s desired scheduling between minor frames.



**Figure 2.** 1553 No Response Scheduling Solution Using a Subroutine to Implement the SKIP and Restore the MFT

## Aeroflex UTMC Errata

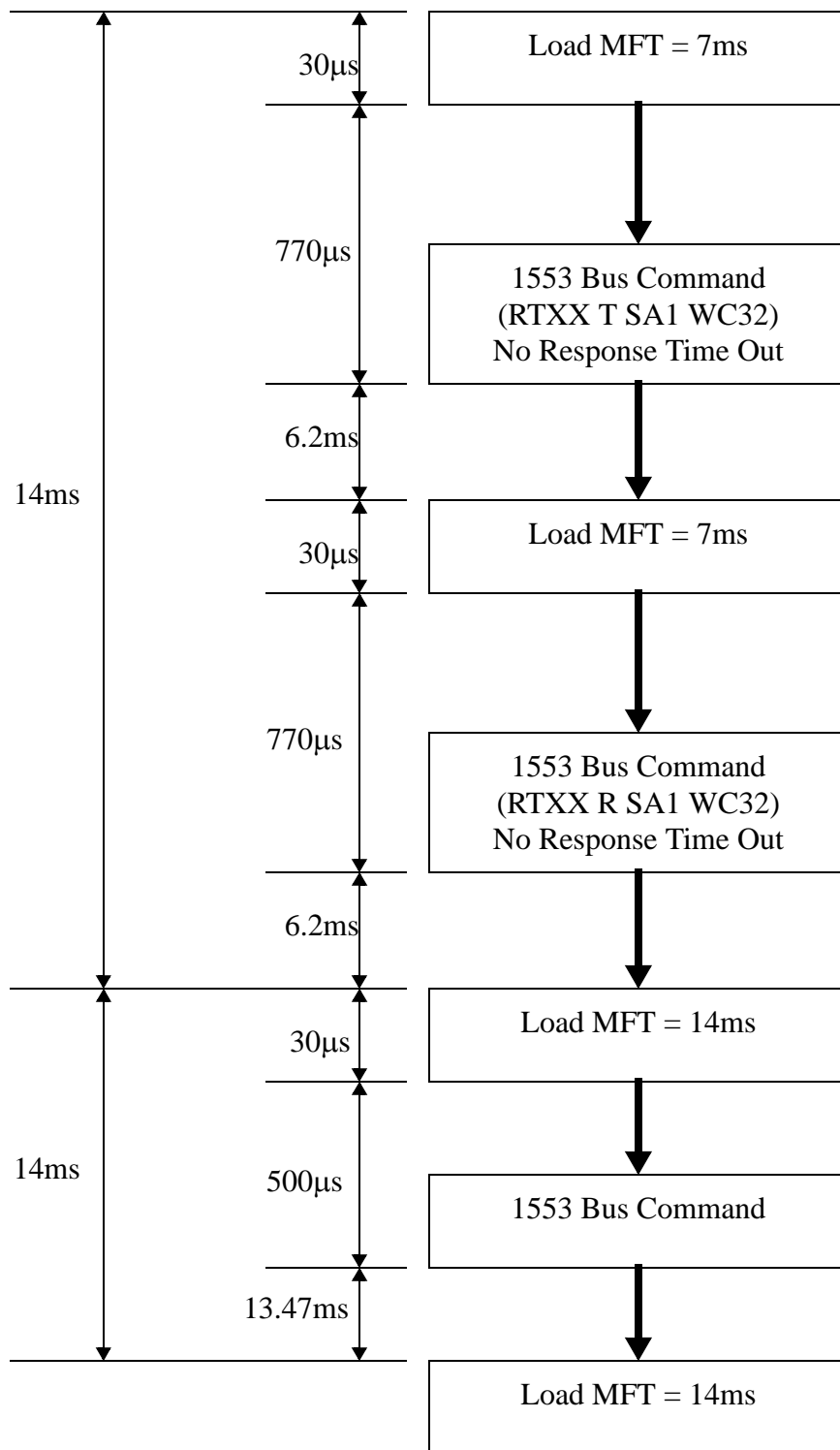
---

The second solution to avoid the SKIP anomaly within a minor frame block is to establish a new minor frame schedule that establishes the desired message duty cycle while providing a sufficiently long inter-message gap without using the SKIP command. Figure 3 shows an example of the code flow that implements a modified minor frame scheduling program supporting the second solution to the SKIP anomaly.

For this solution, the user would replace all SKIP command blocks separating 1553 command blocks within a minor frame block with LMFT block. Next, the user will re-size the time schedule of the LMFT command blocks to allow the S $\mu$ MMIT to execute all required 1553 messages at the desired frequency, and provide a minimum inter-message gap time.

Using the example depicted in figure 3, let us assume that the S $\mu$ MMIT is to initiate two MIL-STD-1553 commands within 14ms, and the inter-message gap time can not be less than 2ms. With this in mind, the first thing to do is place a LMFT command block in between the 1st and 2nd 1553 bus command block (instead of a SKIP command). Next, the LMFT command blocks should be resized to ensure that both MIL-STD-1553 message occur within 14ms and each message should have a minimum inter-message gap of at least 2ms. As shown in figure 3, this is accomplished by sizing the 1st LMFT to time out in 7ms, then size the LMFT command between the two 1553 command blocks to time out in 7ms also. As seen by the timing measurements shown in figure 3, the 1st MIL-STD-1553 message will retire 770 $\mu$ s after the minor frame is established, and the next MIL-STD-1553 message will begin about 6.2ms later. Looking at the associated timing, both messages will occur during the desired 14ms minor frame, and each message will be separated by more than 2ms of inter-message gap time.

The benefit of this solution is that it allows the user to maintain control of the MIL-STD-1553 message duty cycle, and establish an extended inter-message gap. Furthermore, this solution requires little calculations in order to identify the new sizes of the LMFT command blocks. Lastly, by using this solution, the software will work for all S $\mu$ MMIT devices, which includes those not listed in table 1. The only disadvantage to this solution is that it is difficult to establish a constant inter-message gap with high resolution. This occurs because the intervening SKIP command is not used to place a definite inter-message gap between the two MIL-STD-1553 commands. Under this solution, the inter-message gap is variable based on the length of the 1st MIL-STD-1553 command, and the number of retries that are attempted.



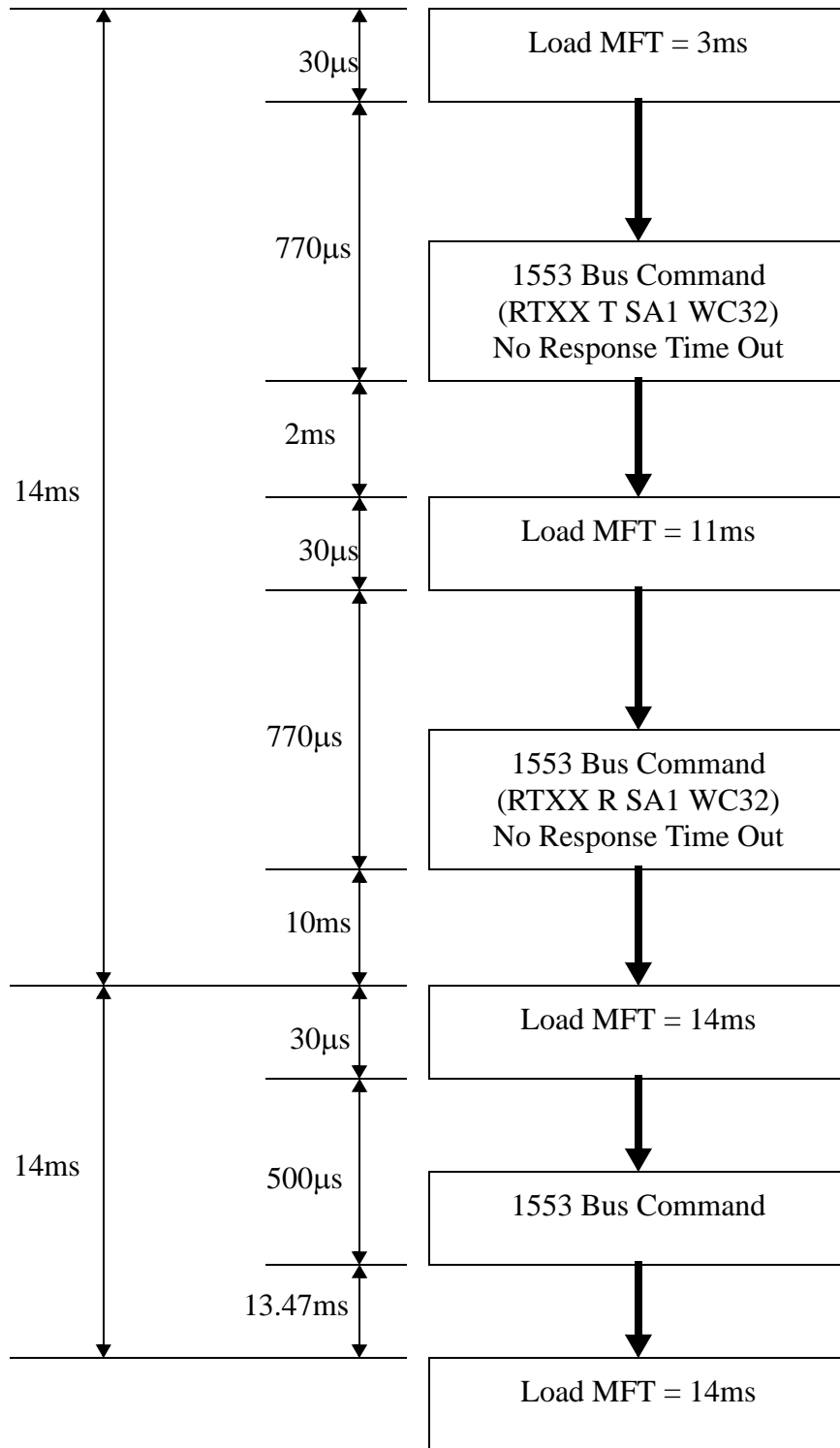
**Figure 3.** 1553 No Response Scheduling Solution Using a Modified MFT Scheduling (No Minor Frame Scheduling with the SKIP Command)

## **Aeroflex UTMC Errata**

---

The last solution (shown in figure 4) is a modified version of solution 2 above where the user tries to generate a specific inter-message gap by sizing the LMFT command blocks based on the execution time of the 1st MIL-STD-1553 message. Instead of sizing both LMFT commands equally, the first LMFT command creates a minor frame block that includes the execution time of the MIL-STD-1553 message and an additional 2ms delay time. The after the MIL-STD-1553 message completes, the subsequent LMFT will have to wait 2ms for the first LMFT to time out, thus inserting a 2ms inter-message gap between the 1st and 2nd MIL-STD-1553 message. Additionally, the second LMFT command must be sized to establish the remainder of the desired message duty cycle. For this example, the second minor frame time is sized to time out in after 11ms. As a result, both MIL-STD-1553 messages will occur within the 14ms time frame, and both messages will be separated by about 2ms.

The benefit of this solution is that it allows the user to maintain control of the MIL-STD-1553 message duty cycle, and establish a specific inter-message gap, although not always as accurately as can be afforded by the SKIP command. Furthermore, by using this solution, the software will work for all S $\mu$ MMIT devices, which includes those not listed in table 1. One disadvantage to this solution is that it is difficult to establish a constant inter-message gap with high resolution because the intervening SKIP command is not used to place a definite inter-message gap between the two MIL-STD-1553 commands. Under this solution, the inter-message gap is variable based on the length of the 1st MIL-STD-1553 command, and the number of retries that are attempted. In order to meet a specified inter-message gap time, the user will have to determine the time require to complete the 1st MIL-STD-1553 message, and size the first LMFT command to expire at the desired inter-message gap time after the bus message completes.



**Figure 4.** 1553 No Response Scheduling Solution Using a Modified MFT Scheduling While Establishing a 2ms Minor Frame Schedule

## **Aeroflex UTMC Errata**

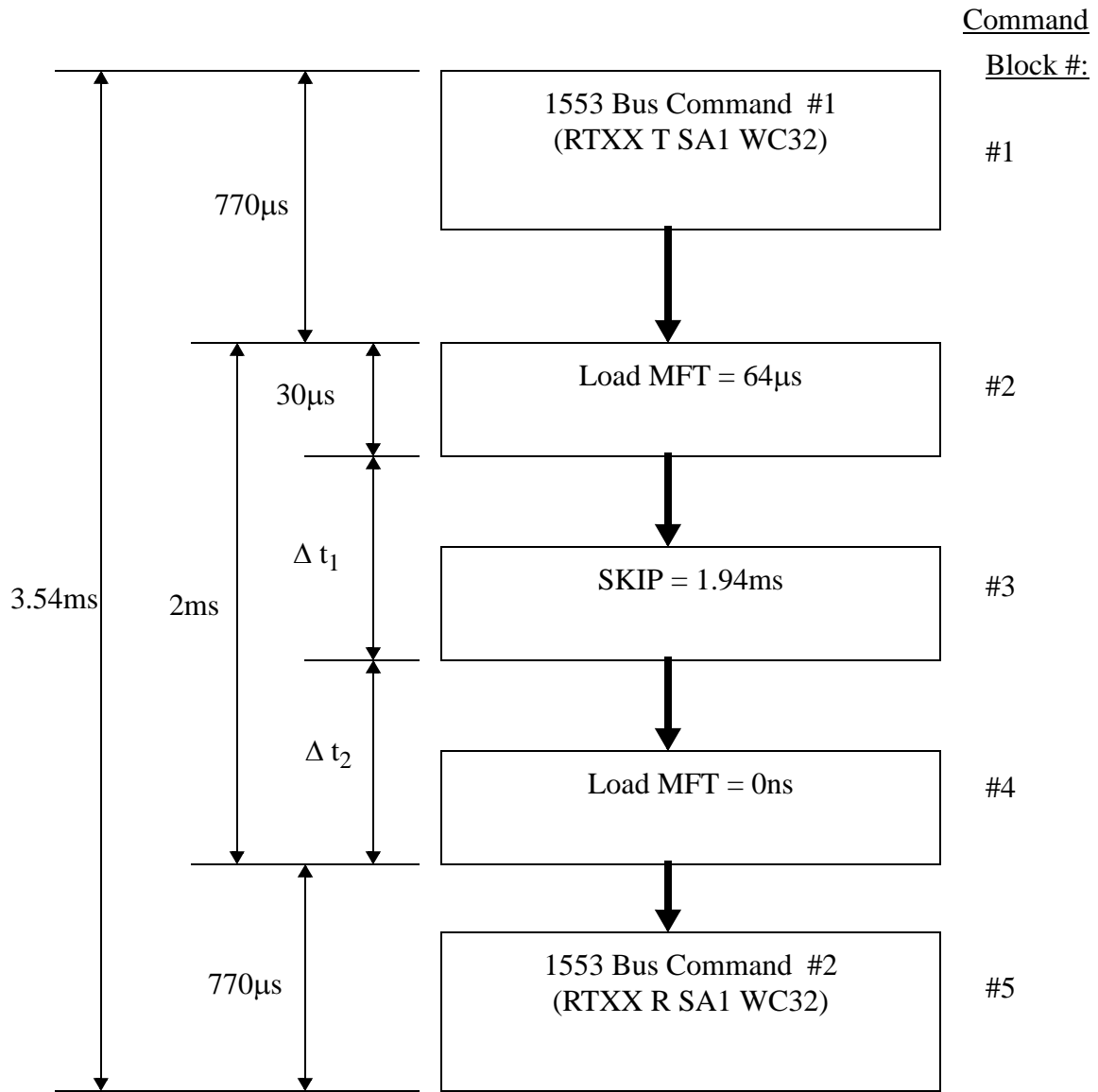
---

The final work-around described in this errata is for protecting the SKIP command. This solution only works outside of a minor frame block. Inside a minor frame block the user should use one of the three solutions discussed previously. When the user requires a SKIP delay, outside of a minor frame block, the SKIP command block can be protected by surrounding it with LMFT command blocks. This methodology will work for all S $\mu$ MMIT devices, including those not listed in table 1. The only caveat is that the minimum inter-message gap that can be established is 90 $\mu$ s. For this work-around the user takes advantage of the fact that upon a bus message error the LMFT command shown in command block #2 is loaded with a time out value specified in the SKIP command block, and the second LMFT command shown in command block #4 will delay subsequent command execution until the original LMFT times out. Conversely, when the MIL-STD-1553 bus command executes without a message error, then the LMFT command in command block #2 is loaded with a value smaller than the SKIP delay and is therefore immaterial. The second LMFT command will execute immediately following the SKIP delay time and will cause about 30ms of error from the SKIP delay time. However, this error can be avoided if the second LMFT execution time is considered in the SKIP delay.

Looking at figure 5, for the case that no message error occurs, the first LMFT will be loaded with a 64 $\mu$ s time out value. Immediately following this LMFT command, the SKIP command will be loaded with a 1.94ms delay time. Since no message error occurred, the SKIP command will create a 1.94ms delay between the execution of the two LMFT commands, and in the process, the first LMFT command will expire having no effect on the execution of the second LMFT command. Furthermore, the SKIP command is sized to incorporate the 30 $\mu$ s execution time of each of the LMFT commands. The final result is a 2ms delay between the end of the 1st MIL-STD-1553 message and the start of the second.

Now, assume that the 1st MIL-STD-1553 command exits with a message time out status. For this scenario, the first LMFT command will execute in about 30 $\mu$ s, and will load the minor frame timer with a time out value of 64 $\mu$ s. Since the SKIP command will be executed before the minor frame time out occurs, and the message error occurred, the minor frame timer will be updated with a time out value of 1.94ms. As a result, the next LMFT command will wait the 1.94ms before executing. Taking into consideration the 30 $\mu$ s execution times of the two LMFT commands and the 1.94ms delay, the MIL-STD-1553 bus will see a 2ms inter-message gap.

This work around does require a little extra coding, but will guarantee protection for the SKIP command, and works for all S $\mu$ MMIT devices. The user should note that this work-around is only applicable when the S $\mu$ MMIT is not executing code from within a minor frame. Using this solution within a minor frame will affect the overall minor frame block scheduling.



**Figure 5.** Establishing the SKIP Delay Outside of a Minor Frame, Independent of the Exit Status of a Preceding 1553 Bus Command