Gen 2 Tag Clock Rate—*What You Need to Know*

Executive Summary

The EPCglobal™ Gen 2 specification mandates the functions that a tag must perform. The specific implementation and the tag's corresponding level of performance, though, are left to the tag designer. Not only are critical performance parameters *not* specified, they're not even evaluated as part of the certification process. Certification merely ensures that tags conform to the Gen 2 protocol. As a result, certified tags may still perform poorly if their designers made poor engineering choices.

The selection of a tag's clock frequency is one such engineering choice that is crucial to tag performance. Aim too low and the tag may decode reader commands incorrectly or may backscatter data to the reader at the wrong frequency. Aim too high and the tag will consume excessive power, shortening both read and write ranges.

Fortunately, the minimum clock frequency for a Gen 2 tag can be calculated from purely theoretical considerations. And that minimum value is 1.92 MHz. Unfortunately, some legacy RFID tags use a 1.28 MHz clock frequency. If a Gen 2 tag designer cuts corners and reuses one of their existing 1.28 MHz oscillators, then the tag's performance will be compromised. Worse yet, a tag designer is actually incentivized to use a 1.28 MHz clock, because the resulting longer tag range is easily demonstrated to an end user, whereas degraded command decoding and incorrect backscatter frequency can be conveniently blamed on the reader or on the "noisy environment."

Tag vendors generally don't tell their end customers which clock frequency they use, nor can end customers easily test a tag to uncover performance issues due to a poor clock-frequency choice. The purpose of this tech brief is to prove why 1.92 MHz is the right choice. Armed with this analysis, an end user can simply ask, "What clock frequency does your tag use?"

Background

When a reader communicates with a tag, it first sends the preamble shown in Figure 1. Two key parameters in this preamble are RTcal (reader-to-tag calibration symbol) and TRcal (tag-to-reader calibration symbol), the lengths of which serve to establish the rules for decoding data-1s and data-0s. To this end, you can think of the tag as a stopwatch: The start of RTcal (TRcal) is the start button; the end of RTcal (TRcal) is the stop button. The tag simply counts the number of clock ticks occurring between start and stop. In other words, the respective lengths of RTcal and TRcal are determined by counting the number of internal clock edges during each symbol period.

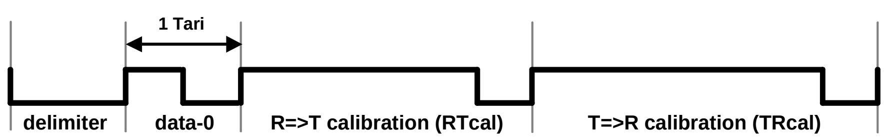

Figure 1 Reader-to-Tag Preamble

RTcal: Readers send commands to tags using binary signaling—a series of data-0 and data-1 symbols. These binary symbols are shown in Figure 2. Notice that a data-0 is shorter than a data-1 (for historical reasons the length of a data-0 is called "Tari," short for Type-A Reference Interval). Before the reader sends such symbols to a tag it first sends RTcal, whose length is equal to the length of a data-0 plus the length of a data-1. The tag measures the length of RTcal, divides by 2, and considers any subsequent symbol shorter than RTcal/2 to be a data-0 and any subsequent symbol longer than RTcal/2 to be a data-1. A simple example can help clarify this. Suppose data-0s are 10µs in length and data-1s are 20µs in length. RTcal is therefore 10µs + 20µs = 30µs in length. The tag computes RTcal/2 = 15µs, and treats symbols shorter than 15µs to be data-0s, and symbols longer than 15µs to be data-1s. Easy enough.

TRcal: Tags can backscatter data to readers at a variety of data rates, depending on the desired mode of operation. The reader specifies the rate by sending TRcal and a parameter called the divide ratio (DR). The tag measures TRcal, divides by DR, and sends data to the reader at a rate given by TRcal/DR. Again, a simple example can help clarify the situation. Suppose TRcal is 100μs in length. Suppose DR=8. In this case, the tag sends data to the reader at a rate given by TRcal/DR = 12.5μs per symbol. Simple.

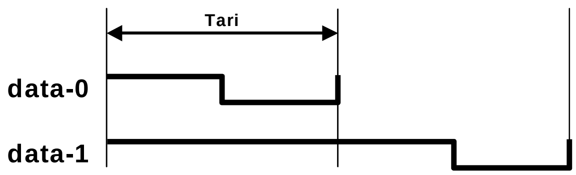

Figure 2 Data Symbols

Choosing Tag Clock Frequency Based on RTcal

If a tag's clock frequency is low, then it won't count many clock ticks during RTcal, and the granularity by which it can decide whether an incoming symbol is a data-0 or a data-1 will be coarse. If the clock frequency is high, it will be better able to resolve a data-0 from a data-1.

By analogy, imagine that you have to time a 50-yard dash, where the racers run one at a time. If you have a digital stopwatch that only reads in 10-second increments, every runner will be timed at 10 seconds. As a result, the runners' performances will be indistinguishable from one another. On the other hand, if you have a digital stopwatch that reads in tenths of seconds, you will be able to determine which runner was the fastest. This same notion of sampling resolution applies to tags that must discern with adequate margin the difference between a data-0 and a data-1.

Figure 3 shows, for a tag using a 1.28 MHz clock, the resulting decoding margin versus Tari for both data-0 and data-1 symbols. In this figure, decoding margin is a reference to the number of clock ticks that separate RTcal/2 from the measured symbol. Notice that, for small Tari values, there is zero decoding margin, meaning that the length of the measured symbol is equal to RTcal/2 (the decision threshold). The tag cannot tell whether the symbol is a data-0 or a data-1, so the command will fail. Figure 4 shows the same situation for a tag with a 1.92 MHz clock. Notice that the decoding margin is always greater than zero, so the incoming symbol is never indeterminate.

Choosing Tag Clock Frequency Based on TRcal

If a tag's clock frequency is low, then it won't count many clock ticks during TRcal, and the granularity by which it can decide its backscatter data rate will be coarse. If the clock frequency is high, then it will be better able to choose the right backscatter frequency. Using the right backscatter frequency is important, because Gen 2 specifies the maximum tag backscatter rate error, and if a tag responds with a frequency outside the maximum error then a reader may (in fact, *should*) ignore it.

Figure 5 shows the backscatter frequency error for a tag with a 1.28 MHz clock. The brown boundary lines are per the Gen 2 spec requirements. Figure 5 shows that if a tag uses a 1.28 MHz clock, it cannot meet the backscatter frequency requirements. Figure 6, on the other hand, shows the backscatter frequency error for a tag with a 1.92 MHz clock. Figure 6 shows that if a tag uses a 1.92 MHz clock, it can meet the Gen 2 requirements with margin.

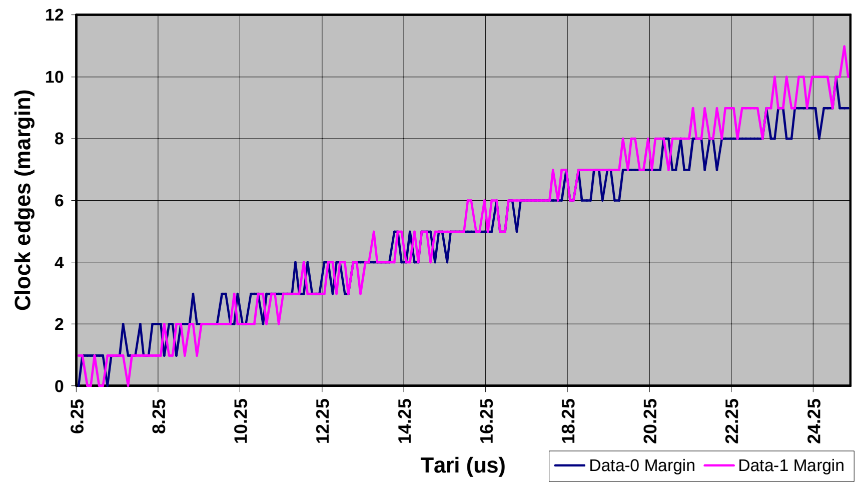

Figure 3 Data Decoding at 1.28 MHz Clock Rate

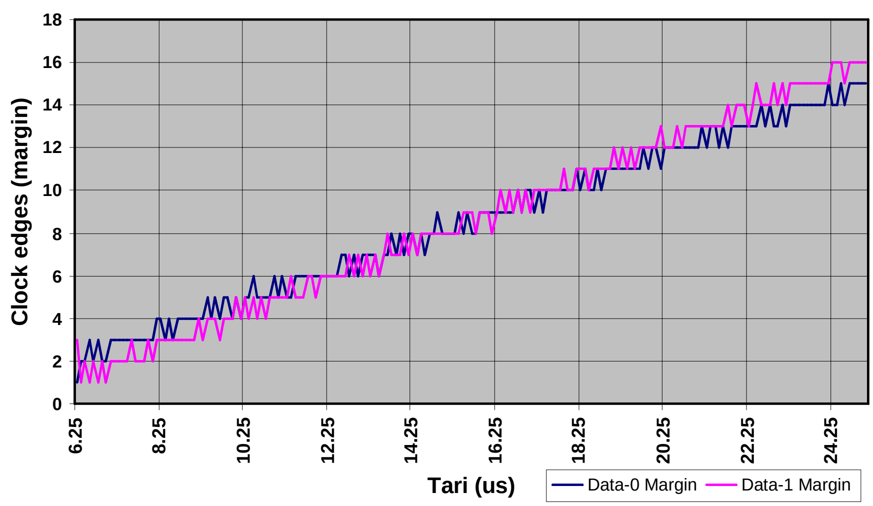

Figure 4 Data Decoding at 1.92 MHz Clock Rate

The curves in Figures 5 and 6 are theoretical—that is, they include oscillator uncertainty but not myriad other tag error sources. Consequently, fielded tags that use a 1.28 MHz clock are guaranteed to perform *worse* than predicted by Figure 5. Unfortunately, EPCglobal's certification process does not test whether a vendor's tag meets the Gen 2 frequency requirements, but instead allows a vendor to self-certify ("guarantee by design") that their tag meets the requirements. If a tag vendor performs an overly simplified analysis and blindly picks a low clock frequency, the RFID system performance will surely suffer.

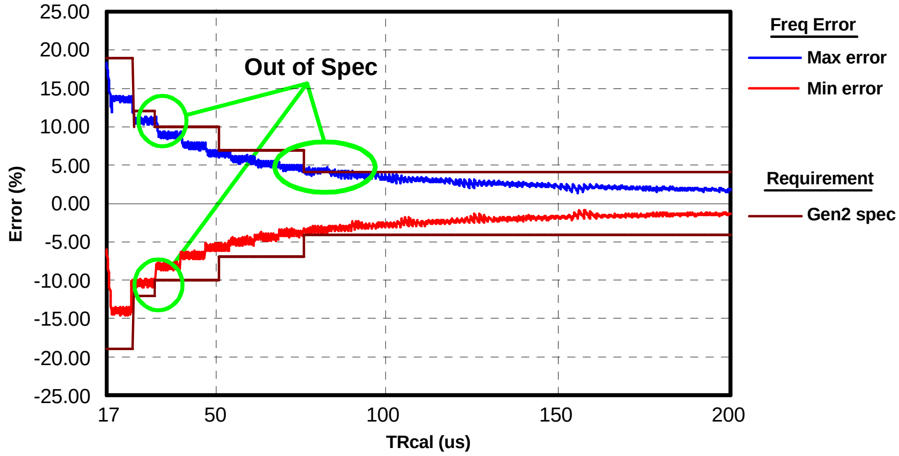

Figure 5 Tag Backscatter Frequency Error at 1.28 MHz Clock Rate

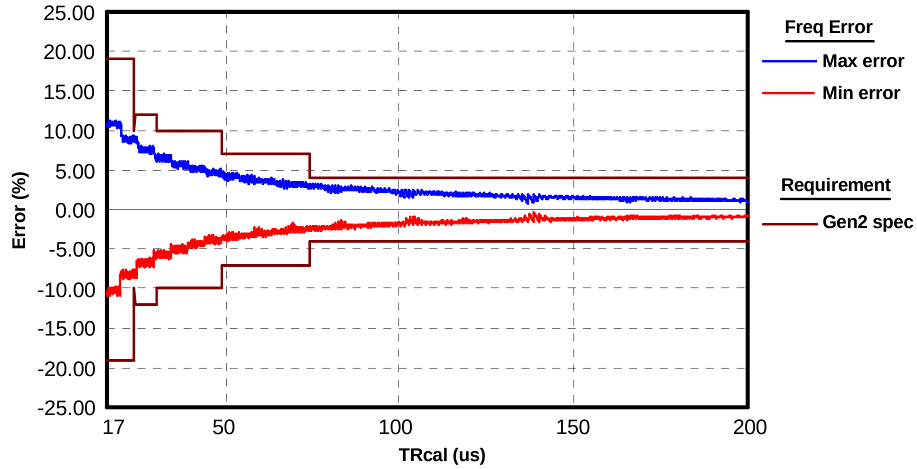

Figure 6 Tag Backscatter Frequency Error at 1.92 MHz Clock Rate

Conclusion

Why do the calculations for RTcal and for TRcal both conclude that a tag should use a 1.92 MHz clock? The answer is simple: The team that wrote the Gen 2 specification performed this exacting analysis before finalizing the standard. The more pertinent question, though, is: Does the choice of clock frequency really matter? The answer is a resounding “YES.” Tags with poor symbol decoding margin or that backscatter at the wrong frequency simply will not perform as well in the real world as tags that get it right. More insidiously, their performance will “appear” to be better in laboratory testing, because their long range, achieved by choosing too low a clock frequency, will make them *seem* superior. And in the field, it is easy to blame the reader rather than the tag for readability problems.

So what should an end user do? Well, for the same reasons that no one would ever buy a computer without knowing its clock frequency, users shouldn’t procure tags without appreciating the vital importance of this same fundamental attribute. Simply ask your tag vendor: What clock frequency does your tag use?

For *RFID that just works*, contact Impinj, or visit www.impinj.com