

Tag Encoding for the Ultra and Joey Timing Systems

February 2018 RFID Race Timing Systems

A UHF Gen 2 tag has several sections of memory where data can be saved. This data is of course binary data of various lengths. The data section that the Ultra reads is called the EPC (Electronic Product Code). The EPC is typically 96 bits long but can vary depending on the tags IC (ie. Monza r6). To ensure complete compatibility between the tag and the Ultra/Joey there are certain rules that should be followed when encoding the EPC.

To ensure that the EPC is correctly filled the data should be 'padded' to the full 96bit (12 byte) length. An example is shown below;

Decimal 1001 (as printed on the tag) will be Hexadecimal 03E9 which when padded will be 000000000000000000000003E9

Note that the EPC has been padded to 24 characters in length. Because the code shown is in hexadecimal there will be 24 characters as there are 2 characters per byte and a byte is 8 bits. The padding can be done in Excel with a formula like;

=REPT("0",24-LEN(C1))&C1 ;where C1 is the cell with the hexadecimal number we want encoded

The ThingMagic USB Encoding software supplied with the reader as a package ensures that the correct padding is done.

Decimal versus Hexadecimal

There is still confusion as to what way to encode the tags for correct reading on the Ultra and Joey systems. When tags are encoded in decimal sequence then they will be the following;

Dec	Hex
1001	03E9
1002	03EA
1003	03EB

Note that if we printed the Hex code onto the UHF tag then we would be getting the characters A...F as well as 0...9. Humans have trouble reading that format unlike computers. Scoring programs will also have no issue telling the difference between hexadecimal numbers as they are often treated as a string. Obviously, the code needs to be unique to each race entrant.

Often when people ask to encode tags in hexadecimal format they are asking that the human readable number excludes any of the A...F characters. So the sequence could look like this;

Dec	Hex
518	206
519	207
520	208
521	209
528	210
529	211

In this case the human readable number (or race number) is the hex number but the decimal number jumps when we get to the hex numbers that have characters in them rather than numbers. In reality the number encoded into the EPC is of binary format (1's and 0's) and you can represent the same number in binary, decimal or hexadecimal. Computers and integrated circuits only deal in binary data.

We recommend that the easiest way is to encode tags is in decimal sequence and set the Ultra or Joey to Decimal output. There really is no advantage in outputting hexadecimal but if you create the correct bib chip file (hex code versus race number) then this will not be a problem.

Prefix's

Often it is a good idea to add a prefix to the sequential tag encoding. The reason is to make the tag code more unique and avoid the chance of a foreign UHF tag in the area being read with the same tag code. This would then be a potential disaster for scoring. The timer can ask for any prefix but make sure the total number (in decimal) is no greater than 32 bits long. That is FFFFFFFF in hexadecimal and 4294967295 in decimal. The reason is that the Ultra and Joey only use the lower 32 bits of the EPC tag code. An example of a tag sequence with a prefix would be;

202180001
202180002
202180003

Where the first 5 digits (20218) is the prefix and the last 4 digits (0001) is the race number. Note that this number range is less than the maximum number that fits into 32 bits (less than 4294967295).

Bib Chip Files

All scoring software packages require a bib to chip file. Bib is synonymous with race number (the race bib number) and is always a unique number in a race. The chip code we have just discussed. So the bibchip file is often a key between bib number and chip code. It can be as simple as an Excel document with bib number in one column and the corresponding chip code in the other. With UHF tags it is easy to sequentially encode the tags so that there is a sequence between bib number and chip code. Using the chip codes from above the bib chip file would look like this;

Chip Code	Race Number (or bib)
202180001	1
202180002	2
202180003	3

The last 4 digits of the chip code does not need to be the same as the race number. In fact the chip code can be anything (but needs to be unique) and you can have random chip codes associated with a sequence of race numbers. But because UHF tags are easy to encode in sequence it is best practice to make it easier for the human to understand what chip code is associated with a race number without even knowing the bib chip file.

Some other timing systems use different parts of the EPC and/or other pages of memory on the tag to protect it from being read with their system unless an unlock code checks out (like a CRC checksum). The systems from RFID Race Timing Systems do not use any locking on the tags so this is nothing to worry about. Tag locking in Gen2 UHF terminology is a different mechanism by which tags can be encoded then locked so they cannot be written to again without the 4 digit password. It is

possible to ask your supplier to lock the tags on encoding but remember you will not be able to re-encode them without the password. The necessity for tags to be locked in this way is debatable as writing to a tag needs the tag to be in close proximity to the reader and away from other tags – an unlikely scenario in a race but possible no less.