Thermal Qualification Specification

FN-STD-TQR Rev. 2 2014-10

1. Purpose

The FNmega128 platform is shared across multiple thermal product families. To validate temperature-controlled equipment, the manufacturer runs thermal qualification cycles. On successful completion of a qualification cycle, the controller generates a Thermal Qualification Record (TQR) in volatile memory.

2. Qualification Cycle

A qualification cycle requires the chamber to reach the qualification temperature of +125 °C while the product core reaches the core criterion of +72 °C. The cycle must be completed on the actual simulated equipment; the record is generated by the control core, not by setting a status flag.

NOTE: The qualification temperature in FN-TEMP8 is 0x7D (+125 °C). This constant is used by the diagnostic signature encoding.

3. Record Layout

The TQR is stored in the diagnostic buffer region (external SRAM, 0x1000–0x3FFF). Field offsets are relative to the record base.

Offset	Field	Size	Description
+0	Identifier	3	ASCII 'TQR'
+3	Version	1	Record version
+4	Seed	1	Diagnostic signature seed
+5	Cycle	1	Qualification cycle counter
+6	Status	1	'P' pending / 'Q' qualified
+7	Peak chamber	1	FN-TEMP8 peak chamber temperature
+8	Peak core	1	FN-TEMP8 peak core temperature
+9	Sig length	1	Signature length N
+10	Signature	N	Diagnostic signature (rolling-XOR encoded)
+10+N	Checksum	2	CRC16 of the signature (little-endian)

4. Qualification States

- Pending ('P') — no qualification cycle has completed; the signature field holds synthetic diagnostic data.
- Qualified ('Q') — a qualification cycle has completed; the signature field holds the generated diagnostic signature.

5. Diagnostic Signature Encoding (Rolling XOR)

The diagnostic signature is encoded with a temperature-dependent rolling XOR. This is a legacy diagnostic format and is not a secure cipher. The key byte rolls forward after each byte, mixing in the qualification temperature (FN-TEMP8) and the plaintext byte.

```
k = seed
for i, p in enumerate(plaintext):
    cipher[i] = p XOR k
```

© 1987–2015 Frost-Nord Industrial Systems. All rights reserved.Thermal Qualification Specification

```
        k = (k + QUAL_TEMP + p) &amp;amp; 0xFF

    where QUAL_TEMP = 0x7D (FN-TEMP8 of +125 °C)
    and 'seed' is the record Seed field (offset +4).
```

Decoding recovers each plaintext byte before rolling the key:

```
    k = seed
    for i, c in enumerate(cipher):
        plain[i] = c XOR k
        k = (k + QUAL_TEMP + plain[i]) &amp;amp; 0xFF
```

6. Worked Example (generic)

Encoding the ASCII string 'DIAG-OK' with seed 0x3C and QUAL_TEMP 0x7D:

i	plain	key	cipher	next key
0	44 ('D')	3C	78	(3C+7D+44) & FF = FD
1	49 ('I')	FD	B4	(FD+7D+49) & FF = C3
2	41 ('A')	C3	82	(C3+7D+41) & FF = 81
3	47 ('G')	81	C6	(81+7D+47) & FF = 45
4	2D ('-')	45	68	(45+7D+2D) & FF = EF
5	4F ('O')	EF	A0	(EF+7D+4F) & FF = BB
6	4B ('K')	BB	F0	(BB+7D+4B) & FF = 83

Resulting signature bytes:

```
    78 B4 82 C6 68 A0 F0
    CRC16 checksum: 29CA  (stored little-endian: CA 29)
```

NOTE: This example is unrelated to any specific unit's record and is provided only to illustrate the format.

7. Checksum

The checksum is CRC16/CCITT-FALSE (polynomial 0x1021, initial value 0xFFFF) computed over the encoded signature bytes, stored little-endian.

© 1987–2015 Frost-Nord Industrial Systems. All rights reserved.Thermal Qualification Specification