Application Note

RF measurements with the DTA-2137(C)

1. Introduction

The DTA-2137(C) is a DVB-S(2) receiver card for PCI Express that can receive Transport Streams BBFRAMEs from an L-Band modulated DVB-S(2) signal. While receiving these signals, the DTA-21 is capable of measuring several statistics, such as symbol rate, Bit Error Rate (BER), Reed-Solomo errors counters and Modulation Error Rate (MER). This application note contains the availa measurements for the DTA-2137(C) and provides some background information about thes measurements.

2. Measurements supported by the DTA-2137(C)

Measurement	Supported for		Comment
	DVB-S	DVB-S2	
Receiver lock	yes	yes	Overall lock status
MER / SNR / MER	yes	yes	Please refer to §3 - MER estimation in the DTA-2137(C) for more information
Eb/N0	yes	CCM only	Energy per bit to noise power spectral density ratio
Es/N0	yes	yes	Energy per symbol to noise power spectral density
Link margin	yes	CCM only	Difference between SNR of the received signal and the SNR at which the receiver cannot demodulate the signal any more
RF level	yes	yes	
Reed-Solomon error counter	yes	n.a.	The number of packets declared erroneous by the Reed Solomon FEC. This will be a continuous counter and this counter will not be reset upon a read
BER pre-Viterbi	yes	n.a.	Pre-Viterbi bit error rate
BER post-Viterbi	yes	n.a.	Post-Viterbi bit error rate
BER pre-Reed-Solomon	yes	n.a.	Pre-Reed-Solomon bit error rate
BER pre-LDPC	n.a.	CCM only	Pre-LDPC bit error rate. Please refer to §5 for more information
BER post-LDPC	n.a.	yes	Post-LDPC bit error rate
BER pre-BCH	n.a.	yes	Pre-BCH bit error rate
BER post-BCH	n.a.	yes	Post-BCH bit error rate
Spectrum inversion	yes	yes	
Occupied bandwidth	yes	yes	
Roll-off factor	yes	yes	

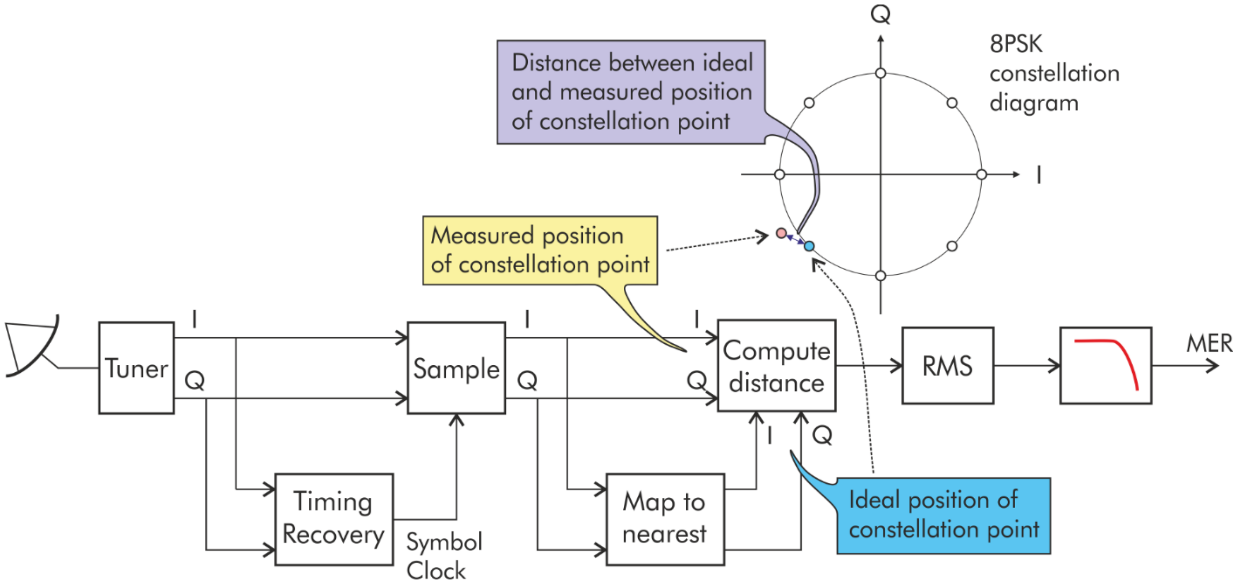
n.a. = not applicable. E.g BER pre-BCH is not applicable to DVB-S, as DVB-S does not use BCH coding.

3. MER estimation in the DTA-2137(C)

The modulation error ratio or MER is a measure used to quantify the performance of a digital rece MER is closely related to the signal-to-noise ratio. If we assume that the transmitter does not intr systematic errors such as I/Q unbalance (a valid assumption in practice), MER is equivalent to the signal-to-noise ratio of the signal.

> **Note**: SNR, CNR, Es/N0 and MER are equivalent if the signal contains no systematic errors

The functional block diagram below shows how the demodulator on the DTA-2137(C) estimates M The tuner receives the DVB-S2 signal and demodulates it into I and Q. A Timing Recovery block extracts the symbol clock, which is used to sample I and Q. The result is the "measured" position the constellation points, which are also mapped to the nearest "ideal" points in the conste diagram. The difference between the two is the "error vector". The Root Mean Square of the erro vector is taken, followed by a low-pass filter. The final result is a MER estimate.



Note: If the signal power is low or there is a lot of noise (low value of SNR/MER), the n constellation point may not always be the correct one. The demodulator will correct the error, bu will not reconstruct the originally intended symbol. What this means is that at very low SNR/MER, MER estimate of the demodulator will be a bit too high. The actual MER is lower (=worse) than th MER estimate.

4. Accuracy of MER measurements in the DTA-2137(C)

In order for the DTA-2137(C) to perform an accurate MER measurement, the power level o applied RF signal needs to be in the range of -30dBm to -50dBm. Please find below the accuracy MER measurements in the DTA-2137(C), for RF signals with the indicated power level.

The RF power level is between -30dBm and -40dBm:

Measured MER	Accuracy	Description
0dB to 22dB	±2dB	• The measured MER is accurate in this range
22dB and above	-	• The quality of the signal is good • The measured MER is not accurate in this range

The RF power level is between -40dBm and -50dBm:

Measured MER	Accuracy	Description
0dB to 17dB	±2dB	• The measured MER is accurate in this range
17dB and above	-	• The quality of the signal is good • The measured MER is not accurate in this range

The RF power level is -50dBm or lower:

Measured MER	Accuracy	Description
-	-	• The measured MER is not accurate in this range

5. Relation between MER and pre-LDPC BER

In DVB-S2, the pre-LDPC BER is the bit error rate before any error correction has been applied. It be computed from the MER for QPSK and 8PSK using the formulas below. The BER is dependent o the modulation type used, so the formulas for QPSK and 8PSK are different.

$$\mathrm{BER}(\mathrm{SNR},\mathrm{QPSK}) = \frac{1}{2} * \mathrm{erfc}\left(\sqrt{\frac{1}{10^{\left(\frac{-\mathrm{SNR}}{10}\right)}}} * \frac{1}{\sqrt{2}}\right)$$

$$\mathrm{BER}(\mathrm{SNR},\mathrm{8PSK}) = \frac{1}{3} * \mathrm{erfc}\left(\sqrt{\frac{1}{10^{\left(\frac{-\mathrm{SNR}}{10}\right)}}} * \sin\left(\frac{\pi}{8}\right)\right)$$

{

Notes

- The function erfc(x) is the complementary error function, see e.g. Wikipedia, “Error function”;
- Instead of SNR, MER can also be used as input parameter in these formulas, as under circumstances SNR and MER are identical;
- These formulas assume that the noise distribution is Gaussian (AWGN channel);
- The BER for APSK schemes is more complex and is also dependent on the code rate used.

These formulas have been validated using DekTec’s advanced demodulator simulation software[1], see the table below. The correspondence between theoretical and measured values is good.

SNR (dB)	BER (QPSK)		BER (8PSK)	
	measured	formula	measured	formula
-3.0	2.3e-01	2.4e-01		
-2.0	2.1e-01	2.1e-01		
-1.0	1.8e-01	1.9e-01		
0.0	1.6e-01	1.6e-01		
1.0	1.3e-01	1.3e-01		
2.0	1.1e-01	1.0e-01		
3.0	8.2e-02	7.9e-02		
4.0	5.8e-02	5.6e-02	1.4e-01	1.3e-01
5.0	3.8e-02	3.8e-02	1.1e-01	1.1e-01
6.0	2.3e-02	2.3e-02	9.3e-02	9.3e-02
7.0	1.3e-02	1.3e-02	7.4e-02	7.5e-02
8.0	5.8e-03	6.0e-03	5.7e-02	5.8e-02
9.0	2.3e-03	2.4e-03	4.2e-02	4.2e-02
10.0	7.7e-04	7.8e-04	2.9e-02	2.9e-02
11.0	1.9e-04	1.9e-04	1.8e-02	1.8e-02
12.0	3.4e-05	3.4e-05	1.0e-02	1.0e-02
13.0	3.9e-06	4.0e-06	5.0e-03	5.2e-03
14.0	1.9e-07	2.7e-07	2.1e-03	2.2e-03
15.0	0.0e+00	9.4e-09	7.4e-04	7.8e-04
16.0	0.0e+00	1.4e-10	2.1e-04	2.1e-04
17.0	0.0e+00	7.2e-13	4.1e-05	4.2e-05
18.0	0.0e+00	9.8e-16	5.9e-06	5.7e-06

[1] This software has been used amongst others in the DVB working groups for the definition of DVB-T2 and DVB-C2

The text box below shows a piece of C/C++ code to compute BER from SNR for QPSK and 8PSK.

```
static double Q(double x)
{
    return 0.5 * erfc(x / sqrt(2.0));
}

double get_ber_qpsk(double snr)
{
    double N0;
    N0 = pow(10.0, -snr / 10.0);
    return Q(sqrt(1.0 / N0));
}

double get_ber_8psk(double snr)
{
    double N0, Ps;
    N0 = pow(10.0, -snr / 10.0);
    Ps = 2 * Q(sqrt(2.0/N0)*sin(M_PI / 8));
    return Ps / 3.0;
}
```

The text box below shows a piece of C/C++ code to compute BER from SNR for QPSK, 8PSK, 16AP
and 32APSK, which uses a table with pre-computed values for the minimum distance between the
constellation points.

```
static double Q(double x)
{
    return 0.5 * erfc(x / sqrt(2.0));
}

double get_ber_psk(double snr, int constellation_scheme, int coderate)
{
    int modulation_index;
    int bps;
    switch (constellation_scheme)
    {
    case QPSK:      modulation_index = 0; bps = 2; break;
    case 8PSK:      modulation_index = 1; bps = 3; break;
    case 16APSK:
        bps = 4;
        switch (coderate)
        {
        case 2_3:   modulation_index = 2; break;
        case 3_4:   modulation_index = 3; break;
        case 4_5:   modulation_index = 4; break;
        case 5_6:   modulation_index = 5; break;
        case 8_9:   modulation_index = 6; break;
        case 9_10:  modulation_index = 7; break;
        }
        break;
```

```
    case 32APSK:
        bps = 5;
        switch (coderate)
        {
        case 3_4:   modulation_index = 8; break;
        case 4_5:   modulation_index = 9; break;
        case 5_6:   modulation_index = 10; break;
        case 8_9:   modulation_index = 11; break;
        case 9_10:  modulation_index = 12; break;
        }
        break;
    }

    double tab_d[13] =
    {
        2.0,        // QPSK
        0.585786,   // 8PSK
        0.297691,   // 16APSK 2/3
        0.334570,   // 16APSK 3/4
        0.337731,   // 16APSK 4/5
        0.341644,   // 16APSK 5/6
        0.347320,   // 16APSK 8/9
        0.346937,   // 16APSK 9/10
        0.134409,   // 32APSK 3/4
        0.145289,   // 32APSK 4/5
        0.149584,   // 32APSK 5/6
        0.159906,   // 32APSK 8/9
        0.160651,   // 32APSK 9/10
    };

    double N0, Ps;
    N0 = pow(10.0, -snr / 10.0);
    Ps = 2 * Q(sqrt(tab_d[modulation_index] / (2.0 * N0)));
    return Ps / bps;
}
```