

Application Note

AFE7950-SP シングル イベント効果検出



Shaily Jain, Nikhil Jain, Srinivas Murthy, Aleena Paul

概要

AFE7950-SP は、構成レジスタ (ビット) 監視、SNR 監視、RX バイアス監視などの監視手法を使用して、シングル イベント アップセットや機能割り込みなどのシングル イベント効果 (SEE) を検出する内部メカニズムをサポートしています。ファームウェアは、アラームをトリガすることで放射線誘発イベントを検出し、外部ホストはこれらのイベントから回復できます。

構成レジスタ (ビット) 監視をトリガすると、デバイス全体の構成レジスタのセットを定期的に監視し、その値をゴールデンリファレンスと比較できます。ゴールデンリファレンスとは、目的の使用事例に合わせてデバイスを構成した後に計算されたレジスタセットのパリティを指します。ビット反転が検出され、単一事象アップセット (SEU) が発生したことを示す場合、アラームが発生します。

シングル イベント機能割り込み (SEFI) は、SEFI 専用の別の GPIO アラームで通知される SNR および RX バイアス監視により検出できます。SNR モニタリングは、ユーザが指定した帯域ですべてのチャンネルの SNR を提供し、チェックします。監視帯域は、チャンネルごとに異なる場合があります。SNR は、指定された帯域についてデバイス内部の FFT エンジンで算出され、ユーザーが設定可能なスレッショルドと比較されます。スレッショルドを超えた場合、GPIO アラームが発生します。RX バイアス モニタリングは、レシーバおよびフィードバック チャンネルのバイアス電圧を定期的に監視し、仕様範囲外になった場合にアラームを発生させます。

目次

1 はじめに.....	2
2 要件.....	3
3 AFE79xx Latte Python スクリプトの構成.....	4
4 デバイス設定後.....	6
4.1 SNR モニタリング パラメータの設定.....	6
4.2 ゴールデンリファレンスの計算.....	8
4.3 モニタリングの開始.....	8
4.4 アラームの読み取り.....	9
4.5 監視周期の設定.....	10
4.6 監視の停止.....	10
5 動的変更後の構成.....	11
6 結果.....	12
7 復旧アクション.....	13
8 参考資料.....	13

商標

すべての商標は、それぞれの所有者に帰属します。

1 はじめに

AFE7950-SP を目的の用途に合わせて設定した後、ユーザーはファームウェアに指示して、レジスタの現在のパリティ セットを計算させる必要があります。このパリティ セットは、ゴールデン リファレンスとして使用されます。ゴールデン リファレンスの算出後、ユーザーはビット監視機能を有効にする必要があります。

SNR 監視では、対象チャンネルについて、ノイズフロアを監視する周波数帯域とスレッショルドを指定する必要があります。ここでは、各チャンネルについて、監視帯域とスレッショルドを個別に選択できます。周波数帯域を指定する際は、誤警報を防ぐため、監視対象の帯域で信号の受信または送信が行われないことを確認してください。帯域およびしきい値を設定した後、対象チャンネルの監視を開始し、GPIO またはアラーム リードバック CAPI を介してアラームが発生しているかどうかを確認できます。

同様に、RX バイアス監視では、監視対象のチャンネルを有効にして監視を開始し、SEFI イベントが発生していないか確認できます。

SEU または SEFI イベントを検出するため、それぞれを個別に通知する二本の専用 GPIO が用意されています。

SEFI イベントが発生した場合、以下のメカニズムにより、GPIO ピンで確認できるアラームが発生します。以下のいずれかの監視機構が正常に動作しない場合、アラームが発生します。

1. **SnrMonitor:** SNR が監視帯域で設定されたスレッショルドを超えて不良になったことを示します。
2. **RxBiasMonitor:** RX バイアスが仕様範囲外であることを示します。
3. **FwHungStatus:** ウォッチドッグ タイマがタイムアウトしたことを示します。これは、ファームウェアが機能を失っていることを意味します。

SEU イベントが発生した場合、他の GPIO ピンによってエラーを通知できるため、外部ホストは適切なアクションを実行できます。

1. **BitMonitor:** 監視対象レジスタ マップ内のビット反転を示します。

ファームウェアは、SEU に影響を受けるレジスタの性能に致命的な影響があるか、それとも特に問題ないかは判断できません。したがって、TI はホスト システムが適切な是正措置を講じることを推奨しています。

上記の監視機能はすべてバックグラウンドで動作し、データ パスの通常動作、レイテンシ、または JESD に影響を与えません。ただし、TX SNR 監視では、TX 出力信号の一部を分岐してフィードバック チャンネルで監視するため、電力がわずかに低下 (0.4dB 未満) することが想定されます。

さまざまなチャンネルの SNR 監視メカニズムを図 1-1 と図 1-2 に示します。

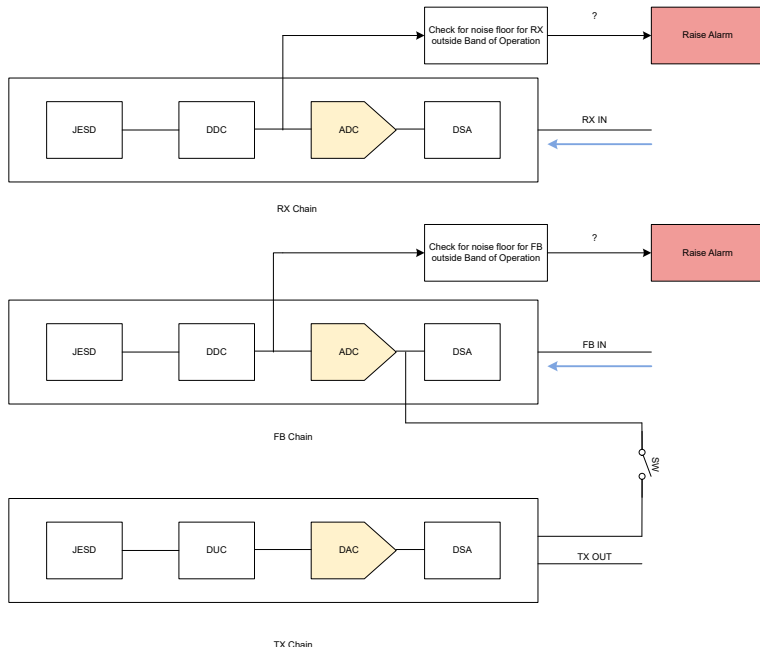


図 1-1. RX および FB での SNR のチェック

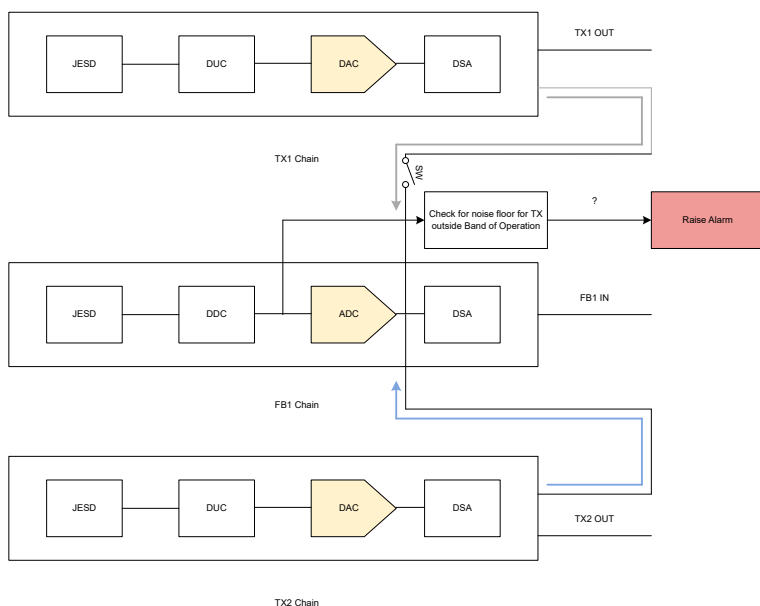


図 1-2. FB 上のループバックによる TX での SNR の確認

2 要件

監視機能を使用する際に、以下の条件が満たされていることを確認する必要があります。

1. フィードバック チャンネルは **TX SNR** 監視に使用されます。したがって、データ受信には使用しないでください。
2. 両方のフィードバック チャンネルが外部の **100Ω** 差動終端を備えていることを確認する必要があります。
3. **RX** チャンネルには、このような要件はありません。
4. **SNR** 監視帯域には、信号が存在せず、外部ブロックを含まない帯域を選択する必要があります。帯域の選択が不適切であったり、監視帯域に信号が漏洩したりすると、誤警報が発生する可能性があります。

3 AFE79xx Latte Python スクリプトの構成

SEFI 検出ファームウェアは、AFE79xx Latte V3.0 から統合されています。Latte をインストールした後、デバイスを目的のモードに構成する際に、以下のパラメータを設定します。

1. `sysParams.isAFE7950SP = 1`
2. `sysParams.intPinsParams`: 複数のブロック (例: JESD、PLL、ビット監視、SNR 監視、RX バイアス監視、FW ウォッチドッグ タイムアウト) からのアラームを有効化し、監視できます。AFE7950-SP には、二本の割り込みピンが用意されています: (ALARM1、ALARM2) で構成されています。SEFI および SEU イベントは、それぞれ ALARM1 および ALARM2 でトリガできます。

フォーマット: `sysParams.intPinsParams[alarmNo][<parameter>] = True/False`

例: `sysParams.intPinsParams[0]['SNRMonitor'] = True`

True = ピンでアラームがイネーブルになっています

False = ピンでアラームは有効ではありません

表 3-1. 各種アラームの説明

パラメータ	デフォルト値とフォーマット	説明
BitMonitor	1/True	構成レジスタセット内の SEU イベント。
FwHungStatus	1/True	ウォッチドッグ タイマのタイムアウト。
JESD	1/True	JESD エラー。
PLL	1/True	PLL ロック喪失。
SNRMonitor	1/True	SNR が特定のスレッショルドを下回ると不良になります。
RxBiasMonitor	1/True	RX バイアスは仕様範囲外になります。

以下の例を参照して、ALARM1 で「SNRMonitor」、「RxBiasMonitor」、「FwHungStatus」、ALARM2 で「BitMonitor」を使用します。

```
sysParams.intPinsParams[0]['BitMonitor']=0
sysParams.intPinsParams[0]['FwHungStatus']=1
sysParams.intPinsParams[0]['SNRMonitor']=1
sysParams.intPinsParams[0]['RxBiasMonitor']=1
sysParams.intPinsParams[1]['SNRMonitor']=0
sysParams.intPinsParams[1]['RxBiasMonitor']=0
sysParams.intPinsParams[1]['BitMonitor']=1
sysParams.intPinsParams[1]['FwHungStatus']=0
```

3. `sysParams.gpioMapping = {'M15': 'ALARM1', 'L14': 'ALARM2'}`

割り込みを構成した後、GPIO M15 と L14 に割り当てます。GPIO M15 および L14 にそれぞれ ALARM1 および ALARM2 割り込みが出力される以下の例を参照してください。

4. `FbEnable = [True, True]`

TX を監視するには、FB チャンネルを使用するため、起動構成でイネーブルにする必要があります。TX1 と TX2 は FB1 を使用して監視し、TX3 と TX4 は FB2 を使用して監視します。したがって、必要に応じて FbEnable を設定します。フィードバック チャンネルに関連するその他のパラメータは、RX と同様に設定できます。

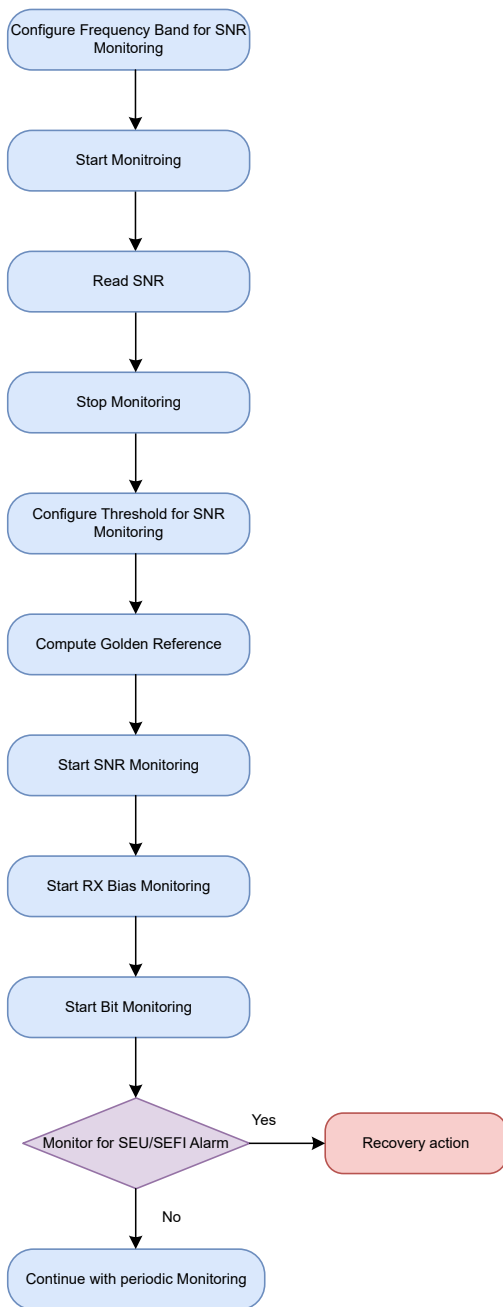


図 3-1. 最初の起動後のシーケンス

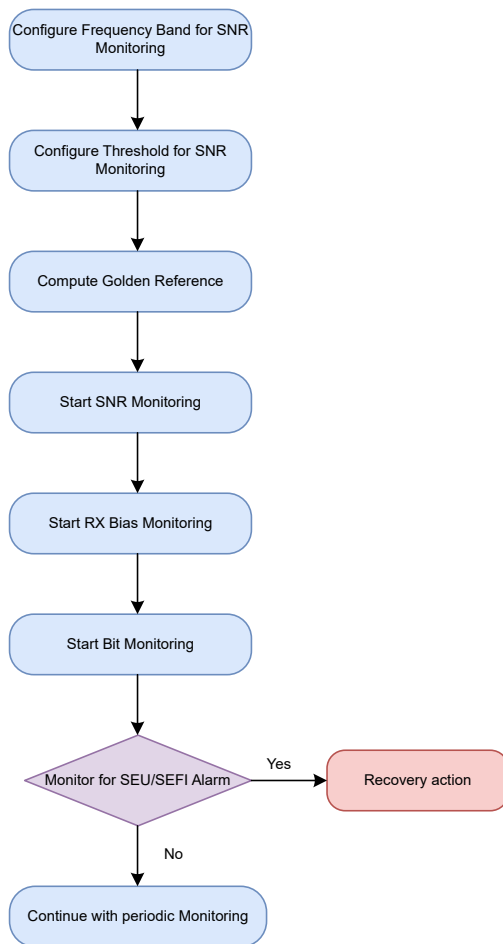


図 3-2. その後の起動後のシーケンス

4 デバイス設定後

前のセクションの手順に従ってデバイスを設定し、立ち上げが完了した後、以下の手順を実行してください。

1. CAPI `configSnrBand` を使用して、すべてのチャンネルの SNR モニタリング バンドを一度に指定するか、RX、FB、および TX のチャンネル選択を使用して個別に設定します。
2. CAPI `startSnrMonitoring` を使用して、選択したチャンネルのモニタリングを開始します。
3. RX、FB、TX の SNR を読み取るには、それぞれ CAPI `readRxSnr`、`readFbSnr`、`readTxSnr` を使用します。この CAPI を呼び出す前に、必ず `startSnrMonitoring` CAPI が呼び出されていることを確認します。レポートされた値を使用して、スレッシュホールドを設定します。
4. 値の読み取り後に CAPI `stopSnrMonitoring` を実行して監視を停止します。
5. チャンネル選択オプションを使用して、選択したチャンネルのスレッシュホールドを設定するには、CAPI `configSnrThreshold` を使用します。RX および FB チャンネルの場合、スレッシュホールドは報告された値より少なくとも 10dB 高い値に設定します。一方、TX では、報告された値よりも少なくとも 5dB 高いスレッシュホールドが設定されます。
6. 必要に応じて、SNR の `setSnrMonitorPeriod`、RX Bias の `setRxBiasMonitorPeriod`、および構成監視の `setBitMonitorPeriod` をそれぞれ使用してモニタ期間を変更します。それぞれのデフォルトの監視周期は 1 秒に設定されています。
7. 目的の帯域、スレッシュホールド、および監視パラメータが設定されたら、CAPI `startSnrMonitoring` を使用して、選択したチャンネルの定期的なモニタリングを開始します (デフォルトの周期は約 1 秒)。
8. CAPI `startRxBiasMonitoring` を使用して、選択したチャンネルの定期的なモニタリング (デフォルトの周期は約 1 秒) を開始します。
9. CAPI `computeGolden` を使用して、構成されたレジスタ セットのゴールデン リファレンス パリティを計算します。
10. CAPI `startMonitoring` を使用して、定期的なモニタリング (周期は約 1 秒) を開始します。
11. アラームの SEFI 検出専用の GPIO ピン M15 を監視します。さらに、CAPI `getSnrAlarm` を使用して、SNR が低下した場合や、CAPI `getRxBiasAlarm` を使用して各チャンネルのアラーム ステータスを読み取ることができます。バイアスが仕様外の場合には、RX および FB チャンネルのアラーム ステータスを取得できます。同様に、構成レジスタ セットのいずれかのビットが反転した場合に、CAPI `getBitAlarm` を使用してアラーム ステータスを取得します。
12. CAPI `stopSnrMonitoring` および `stopRxBiasMonitoring` を使用して、SEFI および `stopMonitoring` のモニタリングを停止し、SEU のモニタリングを停止します。

上記の手順に従う必要があるのは、特定のモードで初めてデバイスを起動する場合のみです。以降の起動では、ステップ 2、3、4 をスキップし、モニタリング帯域が同一であれば、最初の実行から取得したスレッシュホールドを設定できます。[図 3-1](#) と [図 3-2](#) を参照してください。

CAPI シーケンスを呼び出すには、次のコマンドを使用します。

4.1 SNR モニタリング パラメータの設定

1. フォーマット: `CAFE.configSnrBand (afInst, rxSelect, fbSelect, txSelect, startFreq, stopFreq)`

a. `Example 1.1: CAFE.configSnrBand(0, 15, 3, 0, 9300000, 9400000)`

b. `Example 1.2: CAFE.configSnrBand(0, 0, 0, 15, 9450000, 9550000)`

2. フォーマット: `CAFE.startSnrMonitoring (afInst, rxEnable, fbEnable, TxEnable)`

a. `Example 2: CAFE.startSnrMonitoring(0,15,3,15)`

3. フォーマット: `CAFE.readRxSnr (afInst, chNo)`

フォーマット: `CAFE.readFbSnr (afInst, chNo)`

フォーマット: `CAFE.readTxSnr (afInst, chNo)`

a. `Example 3.1: CAFE.readRxSnr(0,0)`

結果:-75

他の RX チャンネルについても同様です。

b. `Example 3.2 CAFE.readFbSnr(0,0)`

結果:-80

別の FB チャンネルでも同様です。

c. `Example 3.3: CAFE.readTxSnr(0,0)`

結果:-55

他の TX チャンネルでも同様です。

4. フォーマット: *CAFE.stopSnrMonitoring (afelInst)*

a. `Example 4: CAFE.stopSnrMonitoring(0)`

5. フォーマット: *CAFE.configSnrThreshold (afelInst, rxSelect, fbSelect, txSelect, Threshold)*

a. `Example 5.1: CAFE.configSnrThreshold(0, 15, 0, 0, -65)`

b. `Example 5.2: CAFE.configSnrThreshold(0, 0, 3, 0, -70)`

c. `Example 5.3: CAFE.configSnrThreshold(0, 0, 0, 15, -50)`

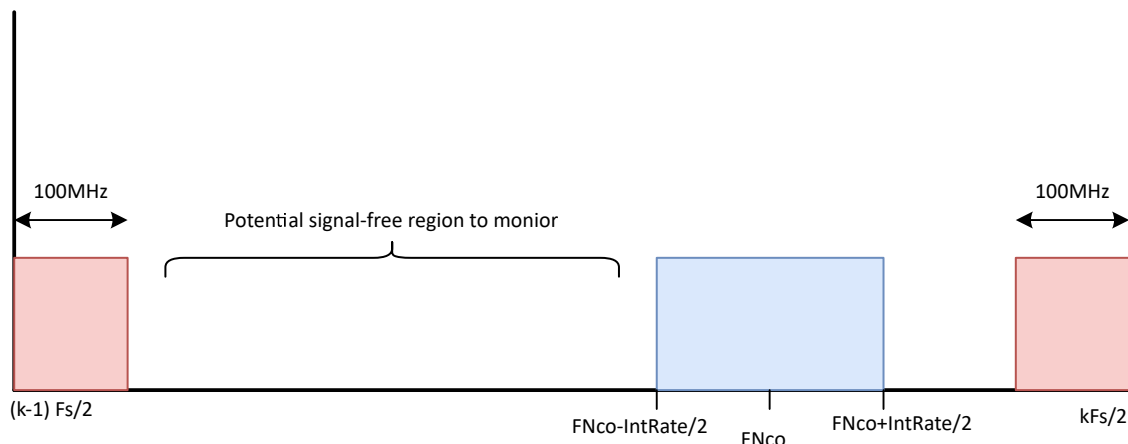


図 4-1. 100MHz の監視帯域は、青色および赤色の領域外で、かつ信号を受信しているナイキスト帯域と同一の帯域内に設定する必要があります

4.1.1 RX のモニタリング パラメータの設定

開始周波数と停止周波数は、次のように選択する必要があります。

1. 監視帯域は、ナイキスト周波数から少なくとも 100MHz 離れている必要があります。
2. 監視対象の帯域は、信号を受信しない帯域であり、かつ DDC 帯域の外側に設定する必要があります。外部ブロッカまたはジャマーの影響は排除する必要があります。
3. 監視帯域は、NCO 周波数と同じナイキストで選択する必要があります。

例えば、ADC のサンプリング周波数 (F_s) が 3000MHz、NCO 周波数が 9500MHz、インターフェイス レートが 200MHz の場合、前述の条件を満たす方に応じて、開始周波数として 9300MHz または 9700MHz のいずれかを選択します。停止周波数は開始周波数から 100MHz 離す必要があります、監視帯域幅が 100MHz であることを確認します。スレッショルドを設定するには、CAPI readRxSnr を使用します。スレッショルドは、報告された SNR よりも少なくとも 10dB 高く設定す

ることを推奨します。これにより、10dB 以上の SNR の劣化を検出できます。上記の例 3.1 を参照してください。すべての RX チャンネルに対してシーケンスを実行します。

4.1.2 FB のモニタリング パラメータの構成

両方の FB チャンネルについて、開始周波数 (kHz) および停止周波数 (kHz) を RX チャンネルと同じ値に設定します。監視帯域幅は 100MHz である必要があります。スレッシュホールドを設定するには、CAPI readFbSnr を使用します。TI は、スレッシュホールドを報告された SNR よりも 10dB 以上高く設定する必要があることを推奨しています。これにより、10dB 以上の SNR の劣化を検出できます。上記の例 3.2 を参照してください。すべての FB チャンネルに対してシーケンスを実行します。

4.1.3 TX のモニタリング パラメータの設定

TX チャンネルの監視は、内部ループバック機構を使用して送信信号を FB チャンネルにループバックすることで行われます。開始周波数と停止周波数は、次のように選択する必要があります。

1. 監視帯域は、FB チャンネルのナイキスト帯域エッジから少なくとも 100MHz 離れている必要があります。
2. 監視対象の帯域は、TX 信号帯域の外側に設定し、信号が送信されておらず、ほかのスプリアス成分も含まれていない帯域としてください。また、監視する TX 帯域を選択する際は、FB ADC を介して TX 信号を観測したときに生じる TX DAC イメージのエリヤス成分が、その帯域に含まれていないことを確認します。
3. 監視帯域は、送信するものと同じナイキスト領域内にある必要があります。

ここでは、FB チャンネルを使用して TX 出力が監視されているため、FB ADC のサンプリング レートに従ってナイキスト周波数を決定する必要があります。バンドを選択する際には、**エイリアシング**の効果も考慮する必要があります。

例えば、TX NCO 周波数が 9250MHz、インターフェース レートが 200MHz の場合、前述の条件を満たす方に応じて、開始周波数として 9050MHz または 9450MHz のいずれかを選択できます。監視の帯域幅が確実に 100MHz であるように、停止周波数を開始周波数から 100MHz 離す必要があります。スレッシュホールドを設定するには、CAPI readTxSnr を使用します。TI はスレッシュホールドが報告された SNR より 5dB 高く設定することを推奨しています。上記の例 3.3 を参照します。すべての TX チャンネルに対してシーケンスを実行します。

4.2 ゴールデン リファレンスの計算

フォーマット: *CAFE.computeGolden (afelInst)*

Example 6: *CAFE.computeGolden(0)*

ゴールデン リファレンスとは、目的の使用事例に合わせてこのデバイスが構成された後に計算されたレジスタ セットのパリティを指します。破損に対するレジスタ セットの比較のためのリファレンスとして機能するゴールデン リファレンスを計算するため、上記のコマンドを呼び出します。

4.3 モニタリングの開始

フォーマット: *CAFE.startSnrMonitoring (afelInst, rxEnable, fbEnable txEnable)*

Example 7.1: *CAFE.startSnrMonitoring(0,15,3,15)*

rxEnable、txEnable、および fbEnable の各パラメータを使用して、それぞれ SNR 監視の対象となる RX、TX、FB チャンネルを選択します。各パラメータはチャンネル インデックスをビット単位で表したもので、最下位ビットが第一チャンネルに対応し、以降の各ビットも順番に各チャンネルへ対応します。

フォーマット: *CAFE.startRxBiasMonitoring (afelInst, rxEnable, fbEnable)*

Example 7.2: *CAFE.startRxBiasMonitoring(0,15,3)*

rxEnable パラメータと fbEnable パラメータを使用して、監視用の RX チャンネルと FB チャンネルをそれぞれ選択します。各パラメータは、チャンネル インデックスのビット単位表現であり、最下位ビットは最初のチャンネルに対応しています。

フォーマット: *CAFE.startMonitoring (afelInst)*

Example 7.3: *CAFE.startMonitoring(0)*

上記のコマンドを呼び出して、ビット モニタリングを開始します。

4.4 アラームの読み取り

上記の手順を実行して監視を有効にすると、割り当てられた GPIO でアラームの発生を確認できます。さらに、次の方法でアラーム読み戻し機能呼び出すことができます。

フォーマット: *CAFE.getSnrAlarm (afelInst)*

Example 8.1: *CAFE.getSnrAlarm(0)*

フォーマット: *CAFE.getRxBiasAlarm (afelInst)*

Example 8.2: *CAFE.getRxBiasAlarm(0)*

フォーマット: *CAFE.getBitAlarm (afelInst)*

Example 8.3: *CAFE.getBitAlarm(0)*

CAPI *getSnrAlarm* は三つの値を返します。一番目の値は RX アラーム、二番目の値は FB アラーム、三番目の値は TX アラームに対応します。RX アラームはビット単位で解釈できます。ビット 0 は RX1 のアラームに対応し、ビット 1 は RX2 のアラームに対応し、以下同様です。同様に、FB アラームはビット単位で解釈できます。ここで、ビット 0 は FB1 のアラームに対応し、ビット 1 は FB2 のアラームに対応します。TX アラームは、RX アラームと同様に解釈できます。

以下の例を参照してください。

Output: [11,1,2]
Binary representation: [1011,01,0010]

これは、SNR がチャネルの設定スレッシュホールドを超えたことを意味します:RX1、RX2、RX4、FB1 および TX2。

CAPI *getRxBiasAlarm* は、最初の値が RX Alarms に対応し、二番目の値が FB アラームに対応する二つの値を報告します。RX アラームはビット単位で解釈できます。ビット 0 は RX1 のアラームに対応し、ビット 1 は RX2 のアラームに対応し、以下同様です。同様に、FB アラームはビット単位で解釈できます。ここで、ビット 0 は FB1 のアラームに対応し、ビット 1 は FB2 のアラームに対応します。

以下の例を参照してください。

Output: [3,1]
Binary Representation: [0011,01]

これは、バイアスが以下の点について仕様外になったことを意味します。RX1、RX2、FB1。

CAPI *getBitAlarm* は、放射線イベントによって、構成レジスタ セットが破損していたかどうかを報告します。出力が 1 の場合、構成レジスタ セットの一回のビット反転または複数回の反転を示し、0 はレジスタ セットに影響を与えないことを示します。

これらのアラームは本質的に粘着性があります。これは、アラームが高くなった場合に、その状態を保持することを意味します。クリア アラーム機能を使用して、これらのアラームをクリアし、アラーム読み取り機能を再度呼び出して、現在の状態を取得できます。アラームをクリアするには、次の例を参照してください。

フォーマット: *CAFE.clearSnrAlarm (afelInst)*

Example 10.1: *CAFE.clearSnrAlarm(0)*

フォーマット: **CAFE.clearRxBiasAlarm (afelInst)**

```
Example 10.2: CAFE.clearRxBiasAlarm(0)
```

フォーマット: **CAFE.clearBitAlarm (afelInst)**

```
Example 10.3: CAFE.clearBitAlarm(0)
```

4.5 監視周期の設定

このメカニズムは、監視周期を選択する柔軟性も提供します。周期はミリ秒単位で設定し、必ず 50 ミリ秒を超える値にします。バックグラウンドで監視を実行している場合、監視の周期は変更しないでください。したがって、この CAPI を呼び出すときは、モニタリングが停止状態であることを確認する必要があります。したがって、周期を変更する前に、**stop monitoring CAPI** を呼び出します。モニタリング周期を 3 秒に設定するには、以下の例を参照します。

フォーマット: **CAFE.setBitMonitorPeriod (afelInst, period)**

```
Example 11.1: CAFE.setBitMonitorPeriod (0,3000)
```

フォーマット: **CAFE.setSnrMonitorPeriod (afelInst, period)**

```
Example 11.2: CAFE.setSnrMonitorPeriod(0,3000)
```

フォーマット: **CAFE.setRxBiasMonitorPeriod (afelInst, period)**

```
Example 11.3: CAFE.setRxBiasMonitorPeriod (0,3000)
```

4.6 監視の停止

監視を停止するには、以下の CAPI を使用します。

フォーマット: **CAFE.stopMonitoring (afelInst)**

```
Example 12.1: CAFE.stopMonitoring(0)
```

フォーマット: **CAFE.stopSnrMonitoring (afelInst)**

```
Example 12.2: CAFE.stopSnrMonitoring(0)
```

フォーマット: **CAFE.stopRxBiasMonitoring (afelInst)**

```
Example 12.3: CAFE.stopRxBiasMonitoring(0)
```

5 動的変更後の構成

デバイスの立ち上げ後に、NCO や DSA の更新などの動的な変更が必要な場合は、CAPI を使用して監視対象のレジスタセットを変更できます。その結果、ファームウェアがこの変更を SEU イベントとして解釈し、誤警報が発生します。これらの誤警報を防止するため、次のシーケンスを推奨します。

1. CAPI stopMonitoring を使用して監視を停止します。
2. 目的の動的更新 (NCO Change など) を実行します。
3. CAPI computeGolden を使用して再計算するゴールデン パリティ。
4. CAPI startMonitoring を使用してモニタリングを再開します。

バックグラウンド監視中に、監視帯域、スレッシュホールド、周期、その他の監視関連パラメータを変更する場合、対応する CAPI を直接呼び出すとバックグラウンド監視に干渉し、ファームウェアが停止する可能性があります。

このような状況を回避するため、以下の手順に従うことを推奨します。

1. それぞれの CAPI を使用してモニタリングを停止します (例 12 を参照)。
2. 対応する CAPI を使用して目的の更新を実行します。
3. Start Monitoring CAPI を使用して監視を再開します (例 7 を参照)。

6 結果

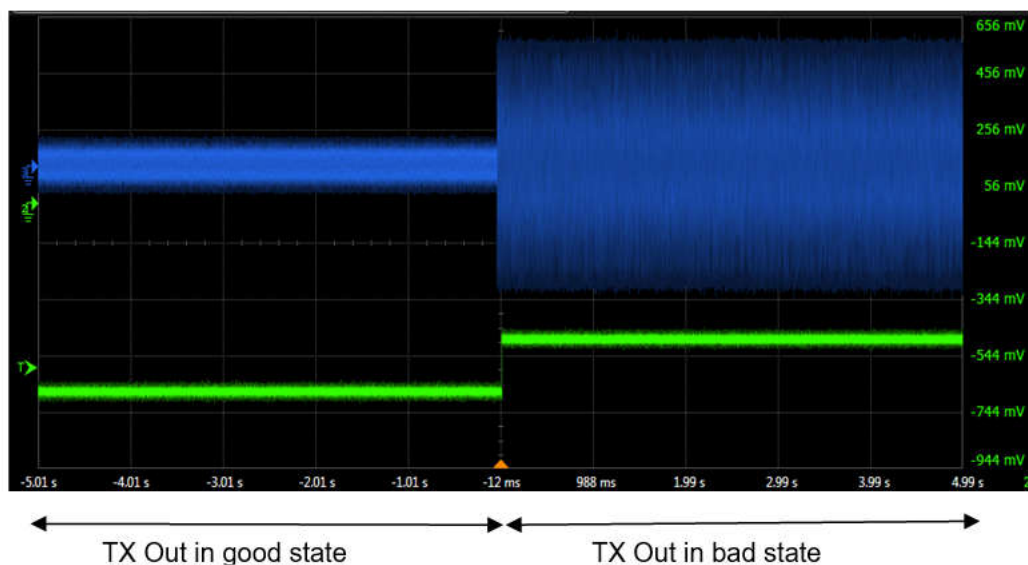


図 6-1. TX 出力は青色で、GPIO アラームは緑色で示されています

監視を有効にすると、SNR が設定したスレッシュホールドを超えた場合にアラームが発生し、このアラームは GPIO M15 で確認できます。(図 6-1 を参照)。

7 復旧アクション

アラームがトリガーされた後、ユーザーはアラームのタイプに応じて適切なアクションを実行する必要があります。アラームが SEU アラームの場合、AFE ハードウェアリセットでアップセット イベントを軽減して、進行中の動作を続行できるようにする必要があります。リセット完了後、設定を再ロードしてデバイスを再度立ち上げ、その後、診断用ファームウェアを起動するための手順を実行する必要があります。

SEFI アラームの場合、SEU の回復手順に加えて、[図 7-1](#) に示すように、コンポーネントを回復するには、電源を入れ直す必要があります。

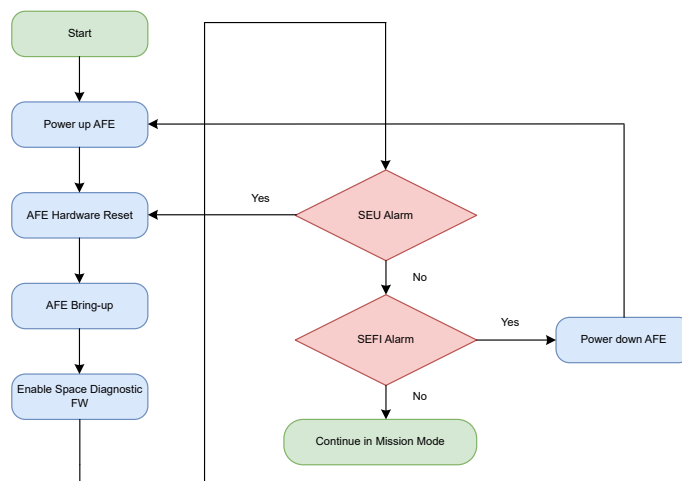


図 7-1. SEU および SEFI アラームの復旧アクション

8 参考資料

- テキサスインスツルメンツ、『[電子機器向け放射線ハンドブック](#)』、e-book。
- テキサス インスツルメンツ、『[AFE7950-SP シングル イベント効果](#)』、放射線レポート。

重要なお知らせと免責事項

TI は、技術データと信頼性データ (データシートを含みます)、設計リソース (リファレンス デザインを含みます)、アプリケーションや設計に関する各種アドバイス、Web ツール、安全性情報、その他のリソースを、欠陥が存在する可能性のある「現状のまま」提供しており、商品性および特定目的に対する適合性の黙示保証、第三者の知的財産権の非侵害保証を含みいかなる保証も、明示的または黙示的にかかわらず拒否します。

これらのリソースは、TI 製品を使用する設計の経験を積んだ開発者への提供を意図したものです。(1) お客様のアプリケーションに適した TI 製品の選定、(2) お客様のアプリケーションの設計、検証、試験、(3) お客様のアプリケーションに該当する各種規格や、その他のあらゆる安全性、セキュリティ、規制、または他の要件への確実な適合に関する責任を、お客様のみが単独で負うものとし、TI は一切の責任を拒否します。

上記の各種リソースは、予告なく変更される可能性があります。これらのリソースは、リソースで説明されている TI 製品を使用するアプリケーションの開発の目的でのみ、TI はその使用をお客様に許諾します。これらのリソースに関して、他の目的で複製することや掲載することは禁止されています。TI や第三者の知的財産権のライセンスが付与されている訳ではありません。お客様は、これらのリソースを自身で使用した結果発生するあらゆる申し立て、損害、費用、損失、責任について、TI およびその代理人を完全に補償するものとし、TI は一切の責任を拒否します。

TI の製品は、[TI の販売条件](#)、[TI の総合的な品質ガイドライン](#)、[ti.com](https://www.ti.com) または TI 製品などに関連して提供される他の適用条件に従い提供されます。TI がこれらのリソースを提供することは、適用される TI の保証または他の保証の放棄の拡大や変更を意味するものではありません。TI がカスタム、またはカスタマー仕様として明示的に指定していない限り、TI の製品は標準的なカタログに掲載される汎用機器です。

お客様がいかなる追加条項または代替条項を提案する場合も、TI はそれらに異議を唱え、拒否します。

Copyright © 2026, Texas Instruments Incorporated

最終更新日：2025 年 10 月