

1. 質問

STM のアクティブな状態を取得するには、どうするのでしょうか？

2. 回答

STM のアクティブな状態を取得するには、状態番号取得機能を使用する方法があります。

**但し、並列ステートに対しては、状態番号取得機能を使用できません。**

状態番号取得機能を使用する手順を、次章より説明します。

### 3. 状態番号取得機能

#### 3.1 プロジェクト作成

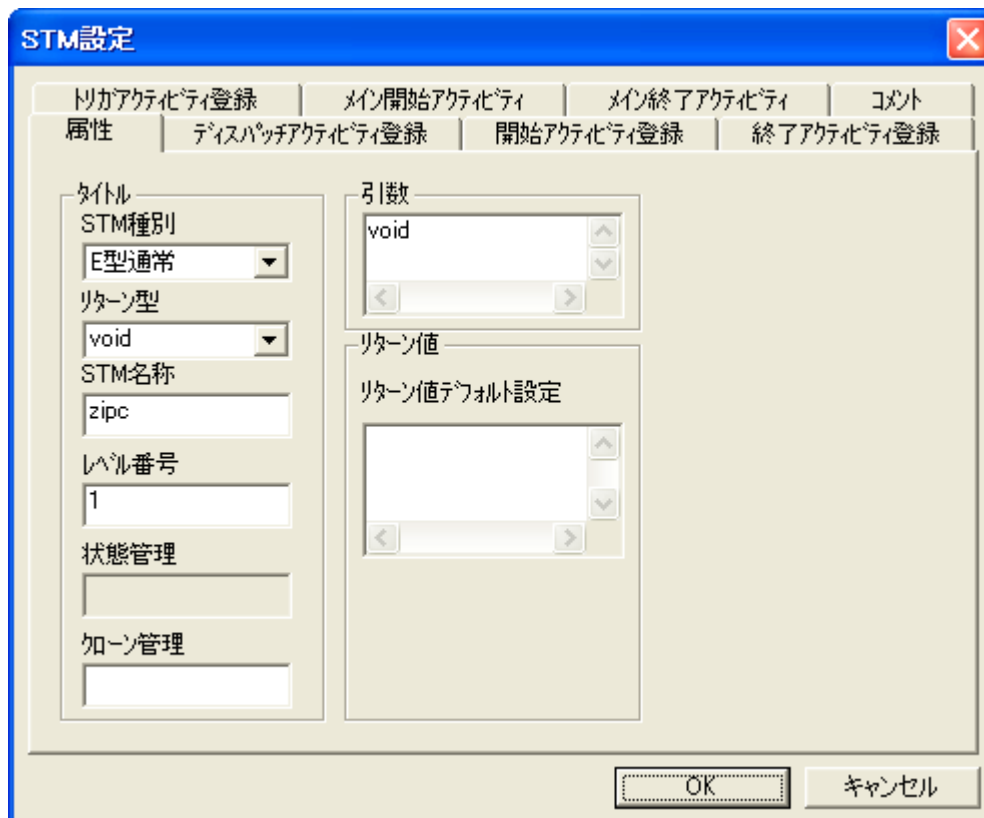
本章では、状態番号取得機能の設定手順を説明します。

##### ■手順

1. ZIPC プロジェクトファイル (状態番号取得.zipf) を新規作成します。
2. "zipc" タスクを登録します。
3. "zipc" タスクに," zipc" STM 設計書(zipc.stm)を登録します。
4. "zipc" STM 設計書の属性を、下記のように設定します。

##### ■"main" STM 設計書属性

No	項目	値
1	STM 種別	E 型通常
2	リターン型	void
3	STM 名称	zipc
4	レベル番号	1
5	引数	void



5. " zipc" STM 設計書を、下図のように編集します。

□1 zipc	S	s0	s1	s2	s3	s4
E		0	1	2	3	4
flg == 1	0	s1 flg = 0;	s2 flg = 0;	s3 flg = 0;	s4 flg = 0;	s0 flg = 0;

6. " zipc" タスクに、" zipc" RAM 設計書(zipc.ram)を登録します。

7. " zipc" RAM 設計書に、下記コードを追記します。

■追記するコード

```
unsigned char flg;
```

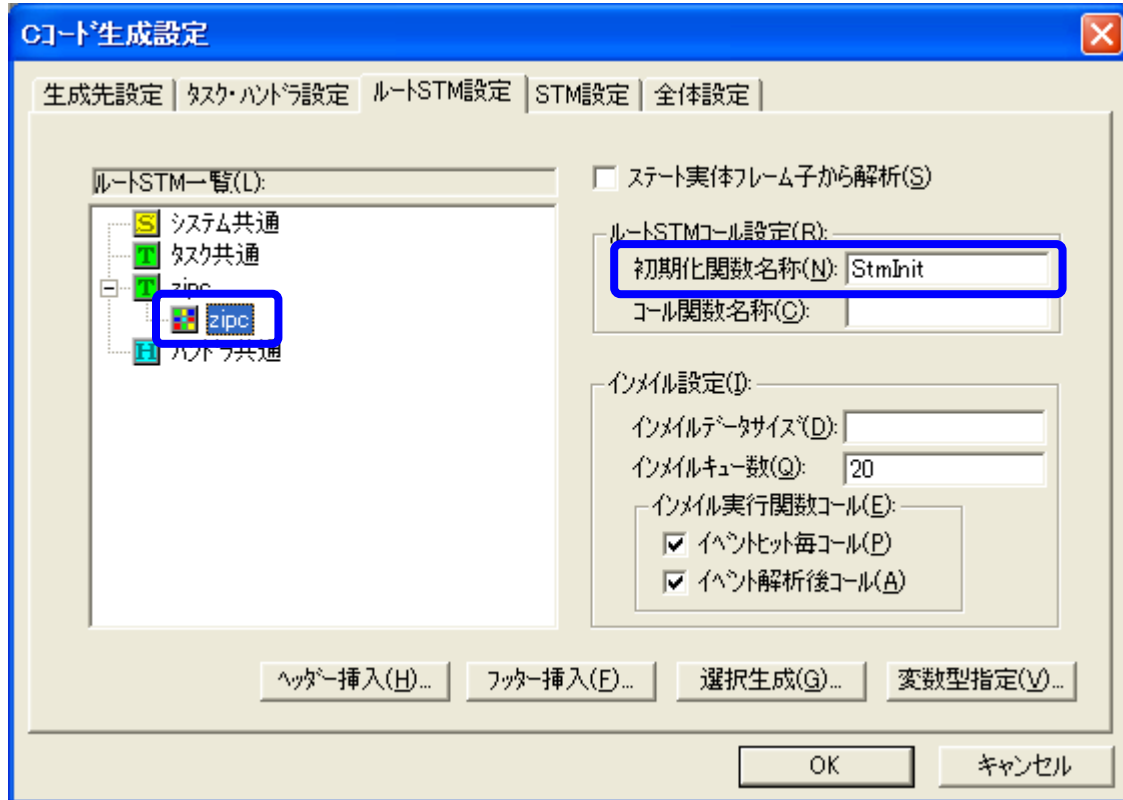
8. " zipc" タスクに," zipc" FNC 設計書(zipc. fnc)を登録します。
9. " zipc" FNC 設計書に、下記コードを追記します。

■追記するコード

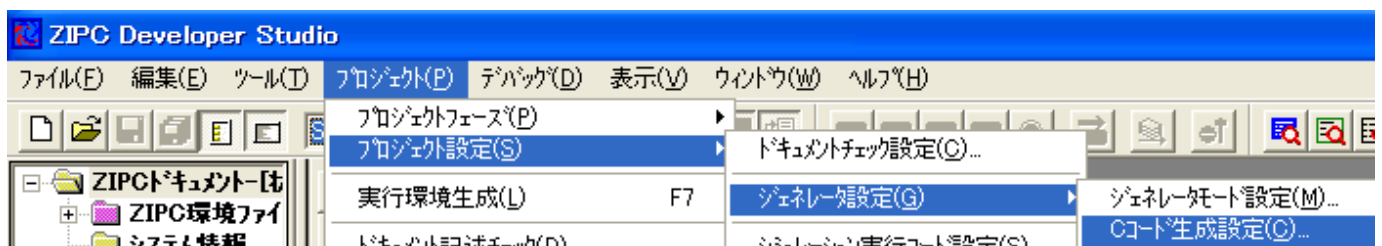
```
/*ZIPC FUNC HEADER*/  
void main( void )  
{  
    char state_now = -1;  
    char state_before = -1;  
  
    /* STM を初期化する */  
    StmInit();  
  
    while(1){  
        /* STM の状態番号を出力する */  
        state_now = GetStateNum();  
        if( state_now != state_before ){  
            printf( "STM 状態番号 : %d\n", state_now );  
            state_before = state_now;  
        }  
        else{  
            /* none */  
        }  
  
        □zipc;  
    }  
}
```

備考 : GetStateNum 関数が," zipc" STM 設計書の状態取得関数になります。

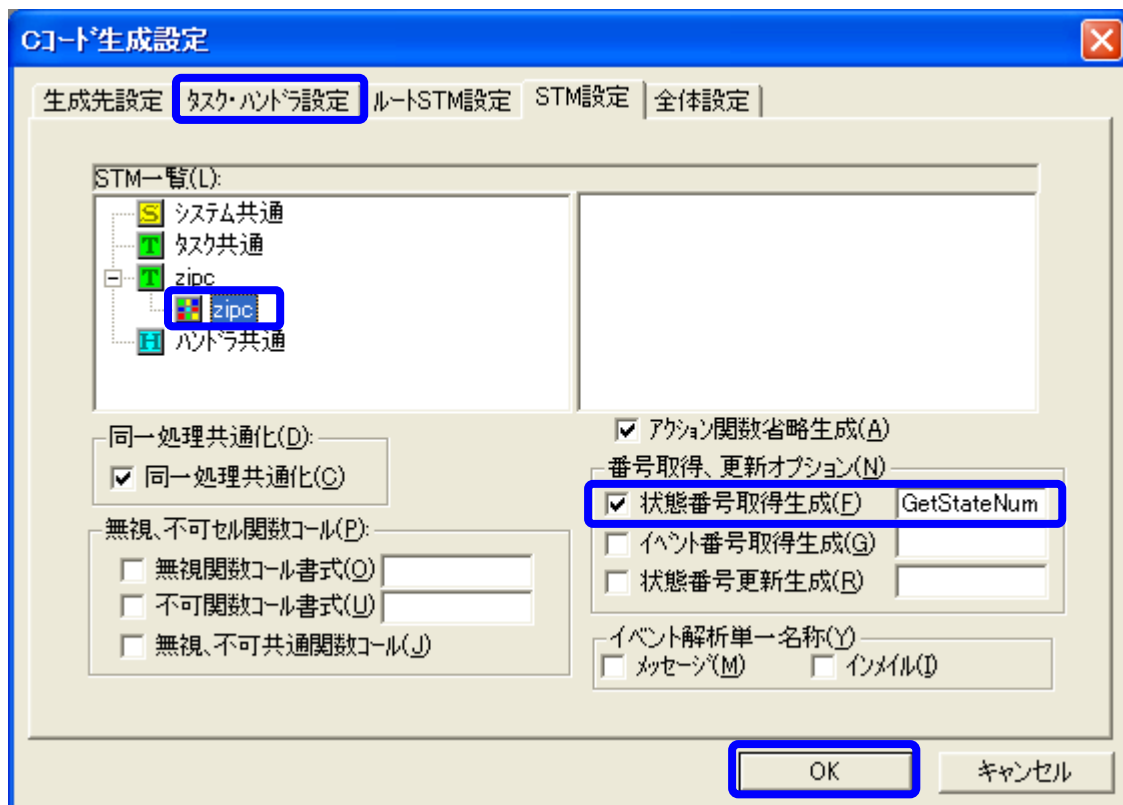
10. シミュレーションフェーズに移行します。
11. "zipc" STM 設計書の初期化関数を、"StmInit" に設定します。



12. [プロジェクト] - [プロジェクト設定] - [ジェネレータ設定] - [Cコード生成設定] メニューを選択します。



13. [Cコード生成設定] ダイアログにて、[STM 設定] タブを選択します。
14. [Cコード生成設定] ダイアログにて [STM 設定] タブにて、[STM 一覧] 項目から” zipc” STM 設計書を選択します。
15. [Cコード生成設定] ダイアログにて [STM 設定] タブにて、[番号取得、更新オプション] - [状態番号取得生成] 項目をチェックします。
16. [Cコード生成設定] ダイアログにて [STM 設定] タブにて、[番号取得、更新オプション] - [状態番号取得生成] 項目に” GetStateNum” と設定します。
17. [Cコード生成設定] ダイアログにて、[OK] ボタンを選択します。



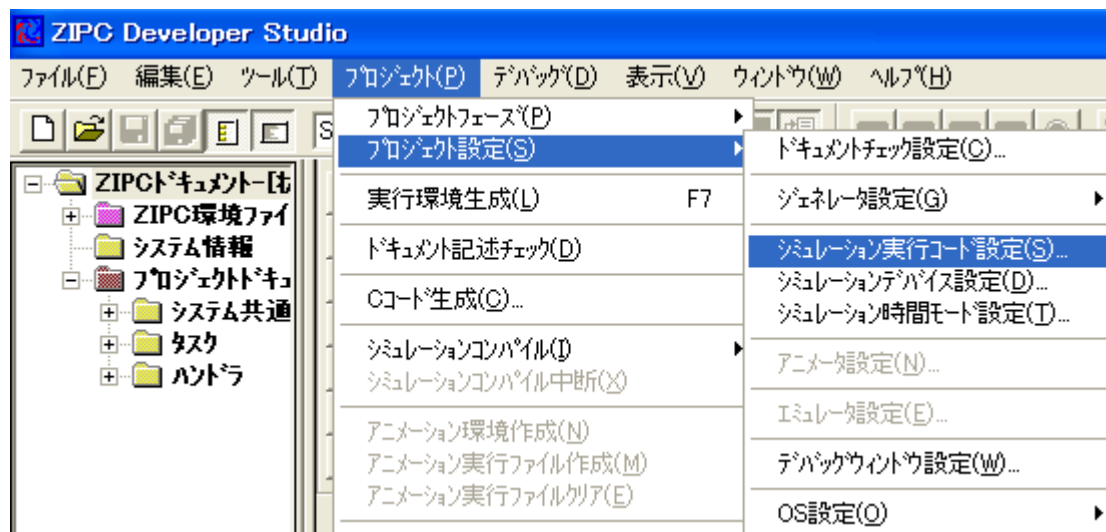
これで、状態番号取得機能の設定は完了です。

### 3.2 シミュレーション

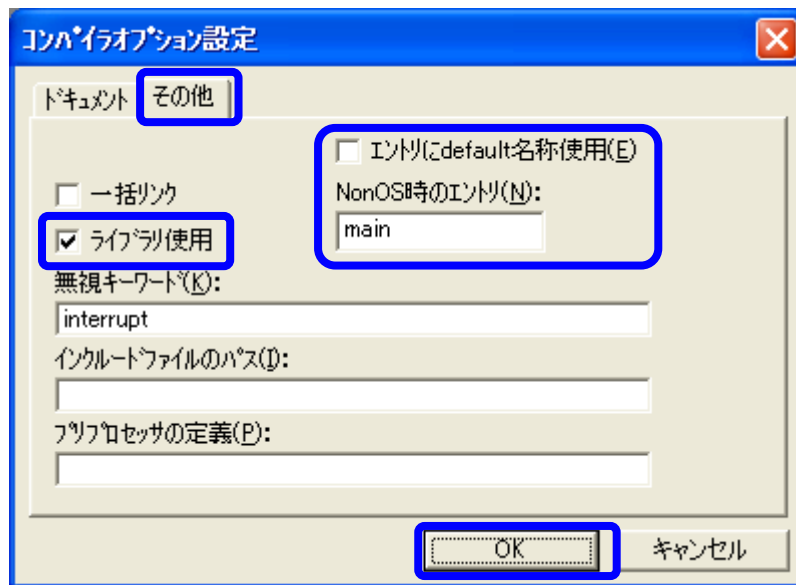
前章までで作成したプロジェクトでは、シミュレーションデバッグすることができません。  
本章では、本プロジェクトにてシミュレーションデバッグを行う為に必要な手順を説明します。

#### ■手順

1. [プロジェクト] - [プロジェクト設定] - [シミュレーション実行コード設定] メニューを選択します。



2. [コンパイラオプション設定] ダイアログにて、[その他] タブを選択します。
3. [コンパイラオプション設定] ダイアログにて、[その他] タブにて、[ライブラリ使用] 項目をチェックします。
4. [コンパイラオプション設定] ダイアログにて、[その他] タブにて、[エントリに default 名称使用] 項目のチェックをはずします。
5. [コンパイラオプション設定] ダイアログにて、[その他] タブにて、[NonOS 時のエントリ] 項目を” main” に設定します。
6. [コンパイラオプション設定] ダイアログにて、[OK] ボタンを選択します。



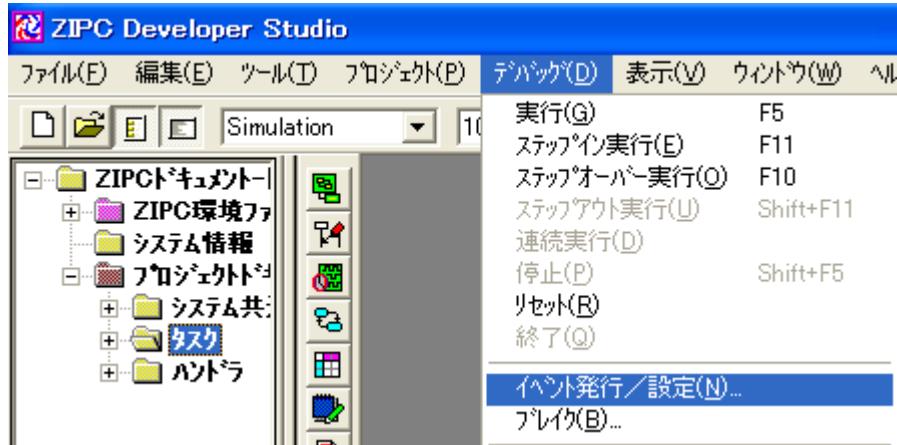
7. シミュレーション環境を構築します。

以上の手順だけでも、ZIPC シミュレーション環境にて、変数ウインドウで変数の値を設定することにより、デバッグを行うことができます。

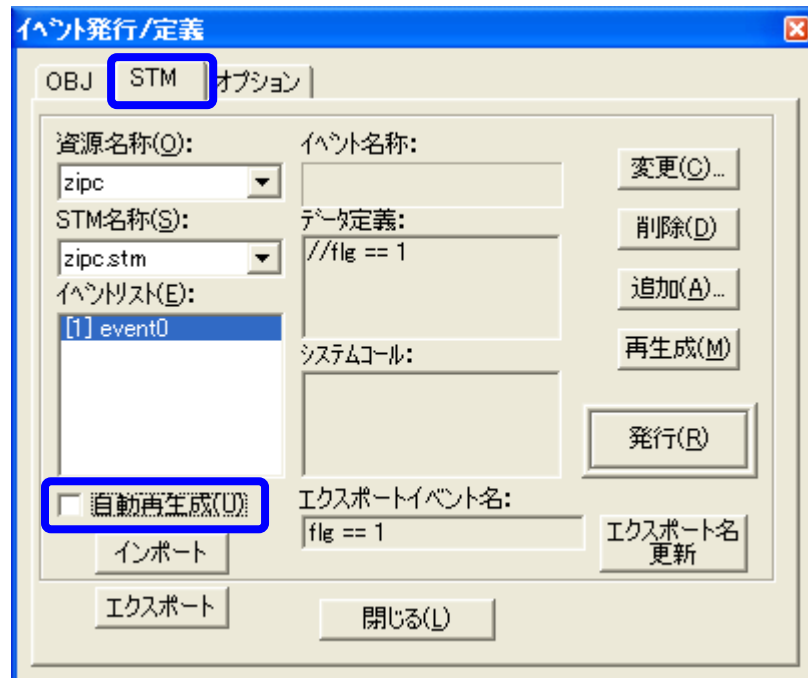
しかし、メッセージイベントと同様にSTM 上からイベントを発行することはできません。STM 上からイベントを発行させるには、下記の手順にてイベント発行定義を行います。



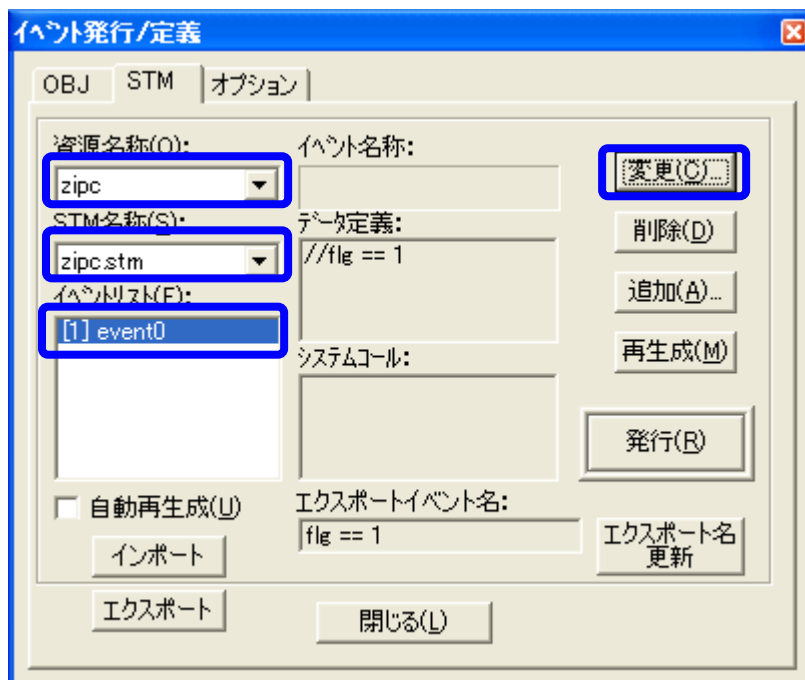
8. [デバッグ] - [イベント発行/設定] メニューを選択して、[イベント発行/定義] ダイアログを起動します。



9. [イベント発行/定義] ダイアログにて、[STM] タブを選択します。
10. [イベント発行/定義] ダイアログ [STM] タブにて、[自動再生成] 項目のチェックを外します。



11. [イベント発行/定義] ダイアログ [STM] タブにて、[資源名称] 項目を” zipc” を選択します。
12. [イベント発行/定義] ダイアログ [STM] タブにて、[STM 名称] 項目を” ポーリング機構.stm” に選択します。
13. [イベント発行/定義] ダイアログ [STM] タブにて、[イベントリスト] 項目を” [1]event0” に選択します。
14. [イベント発行/定義] ダイアログ [STM] タブにて、[変更] ボタンを選択して、[イベント定義] ダイアログを起動します。

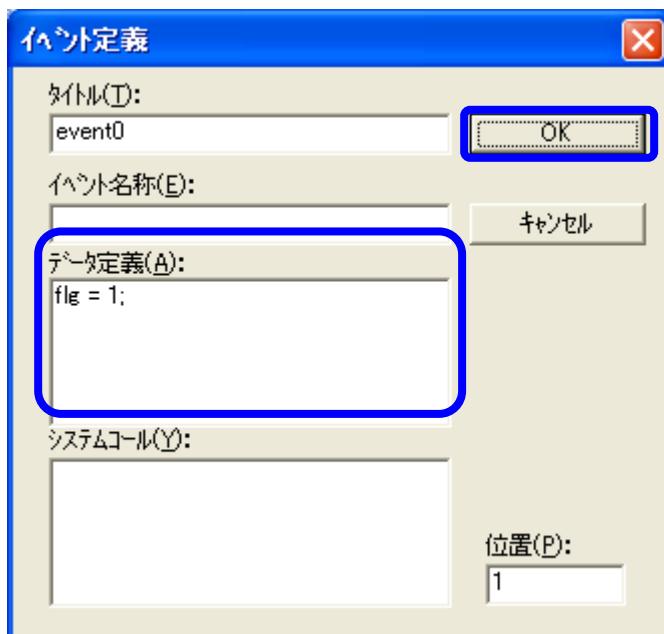


15. [イベント定義] ダイアログにて、[データ定義] 項目を下記コードに設定します。

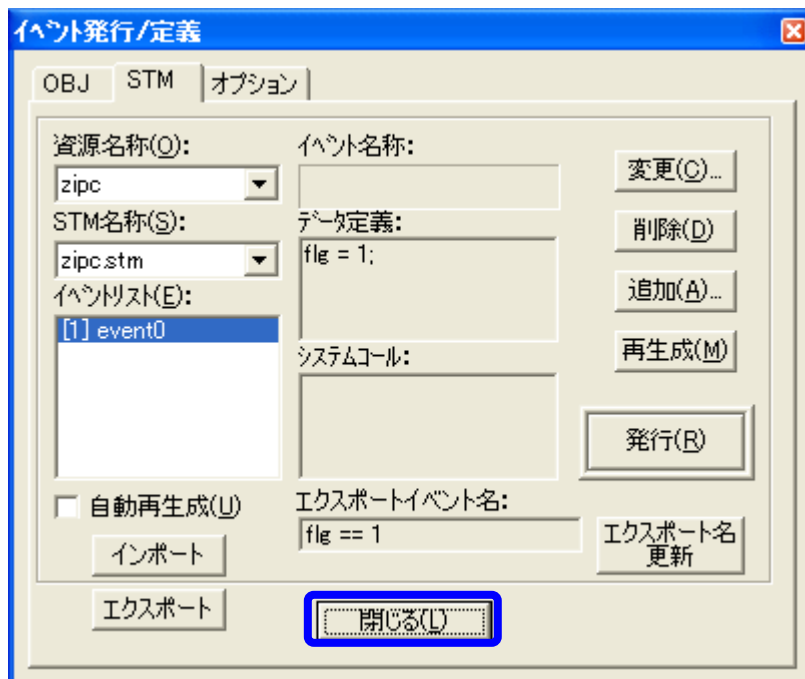
■データ定義設定コード

```
flg = 1;
```

16. [イベント定義] ダイアログにて、[OK] ボタンを選択します。



17. [イベント発行/定義] ダイアログにて、[閉じる] ボタンを選択します。



18. シミュレーション環境を構築します。

これで、メッセージイベントと同様に STM 上からイベントを発行することはできます。