

```

/*****
Template header file for Hierarchical State Machines AKA StateCharts
02/08/12 adjustments for use with the Events and Services Framework Gen2
3/17/09 Fixed prototypes to use Event_t
*****/

#ifndef CalibrationServiceSM_H
#define CalibrationServiceSM_H

// typedefs for the states
// State definitions for use with the query function
typedef enum { CalibrationState } TemplateState_t ;

// Public Function Prototypes

ES_Event RunCalibrationServiceSM( ES_Event CurrentEvent );
void StartCalibrationServiceSM ( ES_Event CurrentEvent );
TemplateState_t QueryCalibrationServiceSM ( void );

#include "ES_Types.h"
#include "ES_Configure.h"
#include "ES_Timers.h"
#include "ADS12.h"
#include <hidef.h>
#include <mc9s12e128.h>
#include <Bin_Const.h>
#include <termio.h>
#include <S12eVec.h>
#include <stdio.h>

#include "MasterHSM.h"

#define RED_BIT BIT0HI //AD output port for red status indicator LED
#define GREEN_BIT BIT1HI //AD output port for green status indicator LED
#define BLUE_BIT BIT0HI //AD output port for blue status indicator LED

#define POT_PIN 3 //AD port used for potentiometer setpoint

#define DIPSWITCH_1 BIT4HI //AD input port for dip switch 1
#define DIPSWITCH_2 BIT5HI //AD input port for dip switch 2
#define DIPSWITCH_3 BIT6HI //AD input port for dip switch 3

#define PUSHBUTTON BIT7HI //AD input port for pushbutton

#define CALIBRATION_TIMER 9
#define CALIBRATION_INTERVAL_MS 250

// Public Function Prototypes

bool InitCalibrationService ( uint8_t Priority );
bool PostCalibrationService( ES_Event ThisEvent );

```

```
ES_Event RunCalibrationService( ES_Event ThisEvent );
unsigned char get_cal_pot(void);
unsigned char get_pushbutton(void);
unsigned char get_dipswitch(void);
unsigned char get_dipswitch_1(void);
unsigned char get_dipswitch_2(void);
unsigned char get_dipswitch_3(void);

#define PRINT_CALIBRATION_SM_CALLS //this will spam the terminal window every time the routine runs
//which is frequent if we are polling switches
#define PRINT_CALIBRATION_READS

#endif /*CalibrationServiceSM_H */
```