```
/****************************************************************************
 LOCMaster header file for Hierarchical Sate Machines
 ****************************************************************************/

#ifndef LOCMaster_H
#define LOCMaster_H

// typedefs for the states
// State definitions for use with the query function
typedef enum { SAMPLE_STATE, SAMPLE_STATE_TWO, WAITING, GAME_STATUS_SENDING_TO_LOC,
GAME_STATUS_RECEIVING_FROM_LOC, SENDING_TO_LOC_AT_STAGING, RECEIVING_FROM_LOC_AT_STAGING,
WAITING_FOR_200MS_TIMEOUT } LOCMasterState_t ;

// Public Function Prototypes
ES_Event RunLOCMasterSM( ES_Event CurrentEvent );
void StartLOCMasterSM ( ES_Event CurrentEvent );
bool PostLOCMasterSM( ES_Event ThisEvent );
bool InitLOCMasterSM ( uint8_t Priority );
uint32_t QueryGameStatus(void);

// SPI function
void SPI_Init(void);
uint8_t SPI_Read(void);
void SPI_Write(uint8_t data);
void SPI_Interupt_Response(void);
void InitInputCapture_Hall( void );
void InputCaptureResponse_Hall( void );
void InitOneShotInt_Hall( void );
static void StartOneShot_Hall( void );
void OneShotIntResponse_Hall( void );
uint8_t frequency_map(uint32_t period);

// Functions used to interact with other modules
void check_active_event();
uint32_t queryStatusBytes(void);
uint8_t queryActiveStagingGreen(void);
uint8_t queryActiveStagingRed(void);
uint8_t queryActiveShootingGreen(void);
uint8_t queryActiveShootingRed(void);
uint8_t quickQueryActiveLocation(void);
uint32_t queryStagingCurrentXDestination(void);
uint32_t queryStagingCurrentYDestination(void);
uint8_t queryGoalGreen(void);
uint8_t queryGoalRed(void);
#endif
```