Software module to access the 12 LEDs on the Uno32 I/O protection board. These
are arranged in three banks (Bank 1, Bank2, and Bank3) with Bank 1 being red,
Bank 2 yellow, and Bank 3 green.

The individual LEDs can be addressed directly using the appropriate LAT pins
as a bit level manipulation (0 turns LED on, 1 turns LED off), or the provided
functions allow them to be used as banks or a whole 12bit array.

The convention is that LED_BANKx_Y where x is 1, 2, or 3 and Y is 0,1,2, or 3
with the 0 bit being towards the interior of the I/O board and 3 being towards
the edge.

NOTE: In order for the LEDs to light, the LED CONTROL jumper for the corresponding
bank must be in place. SPI_MASTER should be defined if both jumpers on the
SPI CONTROL are towards the top of the board.

LED_TEST (in the .c file) conditionally compiles the test harness for the code.
Make sure it is commented out for module usage.

Created on December 18, 2011, 7:58 PM

LED_Init(unsigned char banks);

@param
banks, use #defined BANKx [1,2, or 3] OR'd together

@return
SUCCESS or ERROR

Function initializes the LED subsystem for which each bank enabled (use
the logical OR of each bank). Make sure that the LED jumpers on the I/O
board is set for each bank in use. Sets pins corresponding to LED banks
to outputs and drives them LOW (turns LED on).

LED_OnBank(unsigned char bank, unsigned char pattern);

@param
bank, use #defined BANKx [1,2, or 3] only one

@return
SUCCESS or ERROR

Sets each LED in banks (BANKx) to ON for every matching 1 on pattern.
0's in the pattern are left unchanged.

LED_OffBank(unsigned char bank, unsigned char pattern);

@param
bank, use #defined BANKx [1,2, or 3] only one

@return
SUCCESS or ERROR

Sets each LED in banks (BANKx) to OFF for every matching 1 on pattern.
0's in the pattern are left unchanged.

LED_InvertBank(unsigned char bank, unsigned char pattern);

@param
bank, use #defined BANKx [1,2, or 3] only one

@return
SUCCESS or ERROR

Inverts each LED in banks (BANKx) for pattern provided.
0's in the pattern are left unchanged.

LED_XLLEH (in the .c file) conditionally compiles the test harness for the code.
@param bank, use #defined BANKx [1,2, or 3] only one
* @return SUCCESS or ERROR
* @remark Toggles each LED in bank (BANKx) for every matching 1 on pattern.
* 0's in the pattern are left unchanged.
* @author Gabriel Hugh Elkaim
* @date 2011.12.25 01:16 */
char LED_InvertBank(unsigned char bank, unsigned char pattern);

/**
* Function: LED_SetBank(unsigned char bank, unsigned char pattern);
* @param bank, use #defined BANKx [1,2, or 3] only one
* @return SUCCESS or ERROR
* @remark Sets each LED in bank (BANKx) to match pattern, 1 turns LED on, 0 turns
* LED off. All pins in the bank are affected.
* @author Gabriel Hugh Elkaim
* @date 2011.12.25 01:16 */
char LED_SetBank(unsigned char bank, unsigned char pattern);

/**
* Function: LED_End(void);
* @param none
* @return SUCCESS or ERROR
* @remark Shuts down the LED subsystem. Returns all pins to inputs.
* @author Gabriel Hugh Elkaim
* @date 2011.12.25 01:16 */
char LED_End(void);
#else