

```

1  /**-----
2      \file shared.h
3  --
4  --          ECEN 5803 Mastering Embedded System Architecture          --
5  --          Project 1 Module 3                                         --
6  --          Microcontroller Firmware                                   --
7  --          shared.h                                                  --
8  -------
9
10 --
11 --   Designed for:   University of Colorado at Boulder
12 --
13 --
14 --   Designed by:   Tim Scherr
15 --   Revised by:   David James & Ismail Yesildirek
16 --
17 --   Version:   2.0.2
18 --   Date of current revision:   2018-10-08
19 --   Target Microcontroller:   Freescale MKL25ZVMT4
20 --   Tools used:   ARM mbed compiler
21 --                 ARM mbed SDK
22 --                 Freescale FRDM-KL25Z Freedom Board
23 --
24 --
25 --   Functional Description:   Header file for all globals
26 --
27 --           Copyright (c) 2015 Tim Scherr   All rights reserved.
28 --
29 */
30
31   #include "mbed.h"
32
33   /*****
34   * #defines available to all modules included here
35   *****/
36   // #define OFF 0                /* used for readability */
37   // #define ON 1                 /* used for readability */
38   // #define NO 0                /* used for readability */
39   // #define YES 1               /* used for readability */
40   #define TEN 10
41
42   #define TIMER0 TMR0
43   #define SEC 10000              /* 10000 timer0 interrupts per second (100 usec.) */
44
45   #define T100MS 0.1*SEC
46   #define T2S 2*SEC
47
48   #define LED_FLASH_PERIOD .5    /* in seconds */
49
50   #define CLOCK_FREQUENCY MHZ 8
51   #define CODE_VERSION "2.0.2 2018/10/04" /* YYYY/MM/DD */
52   #define COPYRIGHT "Copyright (c) University of Colorado"
53
54   //enum boolean { FALSE, TRUE };      /// \enum boolean
55   enum dmode {QUIET, NORMAL, DEBUG, VERSION};
56
57   typedef unsigned char UCHAR;
58   typedef unsigned char bit;
59   typedef unsigned int uint32_t;
60   typedef unsigned short uint16_t;
61
62   #ifdef __cplusplus
63   extern "C" {
64   #endif
65
66   /*****
67   /*          Global Variable declarations          */
68   *****/
69
70   extern unsigned char Error_status;      // Variable for debugging use
71   extern UCHAR display_timer; // \var 1 second software timer for display

```

```

72     extern UCHAR   display_flag;    // flag between timer interrupt and monitor.c,
73         // like a binary semaphore
74     extern UCHAR   tx_in_progress;
75     extern UCHAR   *rx_in_ptr; /* pointer to the receive in data */
76     extern UCHAR   *rx_out_ptr; /* pointer to the receive out data*/
77     extern UCHAR   *tx_in_ptr; /* pointer to the transmit in data*/
78     extern UCHAR   *tx_out_ptr; /*pointer to the transmit out */
79     #define RX_BUF_SIZE 10           /* size of receive buffer in bytes */
80     #define TX_BUF_SIZE 40           /* size of transmit buffer in bytes */
81
82     /*****
83     * Some variable definitions are done in the module main.c and are externed in
84     * all other modules. The following section is visible to main.c only.
85     *****/
86     #ifdef MAIN
87
88         enum dmode display_mode = QUIET;
89
90         UCHAR serial_flag = 0;
91
92         UCHAR tx_in_progress;
93         UCHAR *rx_in_ptr; /* pointer to the receive in data */
94         UCHAR *rx_out_ptr; /* pointer to the receive out data*/
95         UCHAR *tx_in_ptr; /* pointer to the transmit in data*/
96         UCHAR *tx_out_ptr; /*pointer to the transmit out */
97
98         UCHAR rx_buf[RX_BUF_SIZE]; /* define the storage */
99         UCHAR tx_buf[TX_BUF_SIZE]; /* define the storage */
100
101     #define MSG_BUF_SIZE 10
102     UCHAR msg_buf[MSG_BUF_SIZE]; // define the storage for UART received messages
103     UCHAR msg_buf_idx = 0; // index into the received message buffer
104
105     /*****
106     * Some variable definitions are done in the module main.c and are externed in
107     * all other modules. The following section is visible to all modules EXCEPT
108     * main.c.
109     *****/
110     #else
111
112     /*****
113     /* Declarations */
114     *****/
115
116     extern volatile UCHAR swtimer0;
117     extern volatile UCHAR swtimer1;
118     extern volatile UCHAR swtimer2;
119     extern volatile UCHAR swtimer3;
120     extern volatile UCHAR swtimer4;
121     extern volatile UCHAR swtimer5;
122     extern volatile UCHAR swtimer6;
123     extern volatile UCHAR swtimer7;
124
125     extern UCHAR serial_flag;
126
127     extern enum dmode display_mode;
128
129     extern UCHAR rx_buf[]; /* declare the storage */
130     extern UCHAR tx_buf[]; /* declare the storage */
131
132     #define MSG_BUF_SIZE 10
133     extern UCHAR msg_buf[MSG_BUF_SIZE]; // declare the storage for UART received messages
134     extern UCHAR msg_buf_idx; // index into the received message buffer
135
136     #endif
137
138     /*****
139     * All function prototypes are externed in all the modules.
140     *****/
141     extern void monitor(void); /* located in module monitor.c */
142     extern void timer0(void); /* located in module timer0.c */

```

```
143  extern void serial(void);    /* located in module UART_poll.c */
144
145  extern void UART_put(UCHAR);    /* located in module UART_poll.c */
146  extern UCHAR UART_get(void);    /* located in module UART_poll.c */
147  extern UCHAR UART_input(void);  /* located in module UART_poll.c */
148  extern void UART_direct_msg_put(const char *); /* located in module UART_poll.c */
149  extern void UART_direct_hex_int_put(uint32_t, uint8_t);
150  //extern void UART_msg_put(const char *);    /* located in module UART_poll.c */
151  extern void UART_direct_hex_put(UCHAR);    /* located in module UART_poll.c */
152  extern void UART_direct_put(UCHAR);    /* located in module UART_poll.c */
153  //extern void UART_hex_put(UCHAR);    /* located in module UART_poll.c */
154  extern void UART_low_nibble_direct_put(UCHAR);    /* located in module UART_poll.c */
155  extern void UART_direct_word_hex_put(uint32_t); /* located in module UART_poll.c */
156  extern void chk_UART_msg(void);    /* located in module monitor.c */
157  extern void UART_msg_process(void);    /* located in module monitors.c */
158  extern void status_report(void);    /* located in module monitor.c */
159  extern void set_display_mode(void);    /* located in module monitor.c */
160  extern uint32_t frequency;
161  extern uint32_t temperature;
162  //extern uint32_t velocity;
163  //extern uint32_t St_const;
164  //extern uint32_t Re;
165  //extern uint32_t viscosity;
166  //extern uint32_t rho_density;
167  extern uint32_t Flow;
168
169  #ifdef __cplusplus
170  }
171  #endif
172
173
```