

```
// KOCH Engineering 2003
// Henrik J. Koch

// fault messages to LCD display
#define MSG_NO_SIGNAL      "No Signal! "
#define MSG_UPDATING      "Updating!  "

/* Events sent from Inner State Machine to Outer State Machine */
#define EVENT_ZEROBIT      0x00
#define EVENT_ONEBIT      0x01
#define EVENT_MINUTESYNC  0xFF

/* Inner state machine states */
#define ISM_RESYNCO        0
#define ISM_RESYNCL        1
#define ISM_ACTIVE         2
#define ISM_INACTIVE       3

/* Outer state machine states.  For state 0-58, the state numbers correspond
to the bit number that is expected next. */
#define OSM_START          0 /* Startbit
*/
#define OSM_FUTURE01      1 /* For future use, currently '0'
*/
#define OSM_FUTURE02      2 /* For future use, currently '0'
*/
#define OSM_FUTURE03      3 /* For future use, currently '0'
*/
#define OSM_FUTURE04      4 /* For future use, currently '0'
*/
#define OSM_FUTURE05      5 /* For future use, currently '0'
*/
#define OSM_FUTURE06      6 /* For future use, currently '0'
*/
#define OSM_FUTURE07      7 /* For future use, currently '0'
*/
#define OSM_FUTURE08      8 /* For future use, currently '0'
*/
#define OSM_FUTURE09      9 /* For future use, currently '0'
*/
#define OSM_FUTURE10     10 /* For future use, currently '0'
*/
#define OSM_FUTURE11     11 /* For future use, currently '0'
*/
#define OSM_FUTURE12     12 /* For future use, currently '0'
*/
#define OSM_FUTURE13     13 /* For future use, currently '0'
*/
#define OSM_FUTURE14     14 /* For future use, currently '0'
*/
#define OSM_ANTENNA       15 /* Using backup antenna
*/
#define OSM_BEFORE_DST    16 /* Is set one hour before switching between
normal/DST */
#define OSM_DST           17 /* Daylight saving time active (summer)
*/
#define OSM_NO_DST        18 /* Daylight saving time not active (winter)
*/
#define OSM_LEAP_SECOND   19 /* Announcing leap second
*/
```

```
#define OSM_TIMESTART    20 /* Start bit for time info (always 1)
*/
#define OSM_MIN01       21 /* Minute bit 0
*/
#define OSM_MIN02       22 /* Minute bit 1
*/
#define OSM_MIN04       23 /* Minute bit 2
*/
#define OSM_MIN08       24 /* Minute bit 3
*/
#define OSM_MIN10       25 /* Minute bit 4
*/
#define OSM_MIN20       26 /* Minute bit 5
*/
#define OSM_MIN40       27 /* Minute bit 6
*/
#define OSM_PARITY1     28 /* Parity of bit 21-27
*/
#define OSM_HOUR01      29 /* Hour bit 0
*/
#define OSM_HOUR02      30 /* Hour bit 1
*/
#define OSM_HOUR04      31 /* Hour bit 2
*/
#define OSM_HOUR08      32 /* Hour bit 3
*/
#define OSM_HOUR10      33 /* Hour bit 4
*/
#define OSM_HOUR20      34 /* Hour bit 5
*/
#define OSM_PARITY2     35 /* Parity of bit 29-34
*/
#define OSM_DATE01      36 /* Date bit 0
*/
#define OSM_DATE02      37 /* Date bit 1
*/
#define OSM_DATE04      38 /* Date bit 2
*/
#define OSM_DATE08      39 /* Date bit 3
*/
#define OSM_DATE10      40 /* Date bit 4
*/
#define OSM_DATE20      41 /* Date bit 5
*/
#define OSM_WEEKDAY01   42 /* Weekday bit 0
*/
#define OSM_WEEKDAY02   43 /* Weekday bit 1
*/
#define OSM_WEEKDAY04   44 /* Weekday bit 2
*/
#define OSM_MONTH01     45 /* Month bit 0
*/
#define OSM_MONTH02     46 /* Month bit 1
*/
#define OSM_MONTH04     47 /* Month bit 2
*/
#define OSM_MONTH08     48 /* Month bit 3
*/
#define OSM_MONTH10     49 /* Month bit 4
*/
```

```
#define OSM_YEAR01      50  /* Year bit 0
*/
#define OSM_YEAR02      51  /* Year bit 1
*/
#define OSM_YEAR04      52  /* Year bit 2
*/
#define OSM_YEAR08      53  /* Year bit 3
*/
#define OSM_YEAR10      54  /* Year bit 4
*/
#define OSM_YEAR20      55  /* Year bit 5
*/
#define OSM_YEAR40      56  /* Year bit 6
*/
#define OSM_YEAR80      57  /* Year bit 7
*/
#define OSM_PARITY3     58  /* Parity of bit 36-57
*/
#define OSM_MINUTESYNC  59  /* Awaiting minute sync after correctly received
time */
#define OSM_RESYNC      60  /* Awaiting minute sync after an error
*/

// ***** variables *****
extern bank2 BYTE      ism_state;
extern bank2 BYTE      osm_state;
extern bank2 WORD      ism_ticks;

extern bank2 BYTE      watch_timeset;
extern bank2 BYTE      watch_hour;
extern bank2 BYTE      watch_minute;
extern bank2 BYTE      watch_second;
extern bank2 BYTE      watch_date;
extern bank2 BYTE      watch_month;
extern bank2 BYTE      watch_year;
extern bank2 BYTE      watch_weekday;

extern bit             NewTelegramFlag;
extern bit             DCF_bit_received;
extern BYTE            LastReceived;
extern bit             LedTimerOn;
extern bank1 int       led_on_cnt;
extern bank2 BYTE      no_of_ones;
extern bank1 BYTE      event_copy;

// ***** functions *****

// dcf77 set time
extern void dcf77_settime (BYTE hour,      /* 0-23 */
                          BYTE minute,    /* 0-59 */
                          BYTE date,      /* 1-31 */
                          BYTE month,     /* 1-12 */
                          BYTE year,      /* 0-99 */
                          BYTE weekday);  /* 1-7, 1=Monday */

// outer state machine
extern void dcf77_osm (BYTE event);

// inner state machine
extern void dcf77_ism (void);
```

```
// initialiation
extern void dcf77_init (void);

struct dcf77_byte
{
    BYTE byte1;    // LSB
    BYTE byte2;    //
    BYTE byte3;    //
    BYTE byte4;    //
    BYTE byte5;    //
    BYTE byte6;    //
    BYTE byte7;    //
    BYTE byte8;    // MSB
};

struct dcf77_bit
{
//  BYTE bit1  : 1;    // LSB   reserved
//  BYTE bit2  : 1;    //      -
//  BYTE bit3  : 1;    //      -
//  BYTE bit4  : 1;    //      -
//  BYTE bit5  : 1;    //      -
//  BYTE bit6  : 1;    //      -
//  BYTE bit7  : 1;    //      -
//  BYTE bit8  : 1;    //      -
//  BYTE bit9  : 1;    //      -
//  BYTE bit10 : 1;    //      -
//  BYTE bit11 : 1;    //      -
//  BYTE bit12 : 1;    //      -
//  BYTE bit13 : 1;    //      -
//  BYTE bit14 : 1;    //      reserved
    BYTE bit15 : 1;    //      R   Antenna
    BYTE bit16 : 1;    //      Z1  Time Zone Change Annoncement
    BYTE bit17 : 1;    //      Z1  Time Zone
    BYTE bit18 : 1;    //      Z2  Time Zone
    BYTE bit19 : 1;    //      A2  Leap Second Annoncement
    BYTE bit20 : 1;    //
    BYTE bit21 : 1;    //
    BYTE bit22 : 1;    //
    BYTE bit23 : 1;    //
    BYTE bit24 : 1;    //
    BYTE bit25 : 1;    //
    BYTE bit26 : 1;    //
    BYTE bit27 : 1;    //
    BYTE bit28 : 1;    //
    BYTE bit29 : 1;    //
    BYTE bit30 : 1;    //
    BYTE bit31 : 1;    //
    BYTE bit32 : 1;    //
    BYTE bit33 : 1;    //
    BYTE bit34 : 1;    //
    BYTE bit35 : 1;    //
    BYTE bit36 : 1;    //
    BYTE bit37 : 1;    //
    BYTE bit38 : 1;    //
    BYTE bit39 : 1;    //
    BYTE bit40 : 1;    //
    BYTE bit41 : 1;    //
};
```

```
    BYTE bit42 : 1;    //
    BYTE bit43 : 1;    //
    BYTE bit44 : 1;    //
    BYTE bit45 : 1;    //
    BYTE bit46 : 1;    //
    BYTE bit47 : 1;    //
    BYTE bit48 : 1;    //
    BYTE bit49 : 1;    //
    BYTE bit50 : 1;    //
    BYTE bit51 : 1;    //
    BYTE bit52 : 1;    //
    BYTE bit53 : 1;    //
    BYTE bit54 : 1;    //
    BYTE bit55 : 1;    //
    BYTE bit56 : 1;    //
    BYTE bit57 : 1;    //
    BYTE bit58 : 1;    //
};

struct dcf77_struct
{
    // ialt 44 bits
    BYTE reserved_bits1      : 7;    // 1-14    reserved
    BYTE reserved_bits2      : 7;
    BYTE antenna             : 1;    // 15      ("R" 0=normal,
1=backup)
    BYTE time_zone_change_announce : 1;    // 16      ("A1" 0=nothing,
1=change)
    BYTE time_zone           : 2;    // 17-18   ("Z1","Z2"
00b=+0h, 01b=+1h, 10b=+2h, 11b=+3h)
    BYTE leap_second_announce : 1;    // 19      ("A2" 0=nothing,
1=change)
    BYTE start               : 1;    // 20      ("S" always 1)
    BYTE minute              : 7;    // 21-27   (minutes BCD
coded)
    BYTE parity1             : 1;    // 28      ("P1" even
parity for bits 21-27)
    BYTE hour                : 6;    // 29-34   (hours BCD
coded)
    BYTE parity2             : 1;    // 35      ("P2" even
parity for bits 29-34)
    BYTE day                 : 6;    // 36-41   (day of month
BCD coded)
    BYTE day_of_week         : 3;    // 42-44   (day of week BCD
coded)
    BYTE month               : 5;    // 45-49   (month BCD
coded)
    BYTE year                : 8;    // 50-57   (year. Ones and
tens only !)
    BYTE parity3             : 1;    // 58      ("P3" even
parity for bits 36-57)
};

union dcf77_union
{
    struct dcf77_byte        bytes;    // 8 bytes
    struct dcf77_bit         bits;    // 58 bits
    struct dcf77_struct      records; // blanding af records
};

extern bank3 struct dcf77_struct    dcf77_telegram;
```

File: C:\Documents and Settings\Henrik\Skrivebord\BlueBird_ny\til PDF\dcf77.h
29-01-2003, 23:56:11
