


```

typedef struct
{
    uint8 red[8];    // 8 bytes (64 bits) for 64 red leds
    uint8 green[8]  // 8 bytes (64 bits) for 64 green leds
} PATTERN;

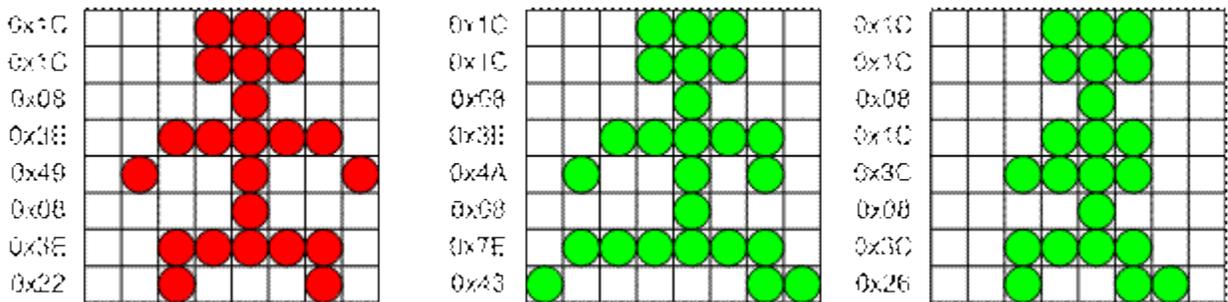
int No_Seq = 3;
int seq_length[] = {1, 2, 2}    // length of the sequences
int duration [] = {20, 10, 10, 5, 5};    // define the display duration for each pattern
PATTERN seq [] = {.....}; // define 3 sequences and each may have 2 patterns

```

The example shows there are 3 sequences of patterns. It then declares the length of the sequence and the display duration for each pattern (assuming the unit is 0.1s). Note that there are a total 5 patterns in the 3 sequences and the display durations in duration[] are listed sequentially according to the orders of sequences and patterns. Finally, it declares the patterns for the sequences.

When your program starts to run, the LED matrix should show the 1st sequence. Upon a SW1 interrupt, the display switches to the next sequence and wraps around when finishing all sequences. To display a sequence, the list of patterns are shown repeatedly and each pattern is displayed for the specific duration.

An example of the animation you can display is the crosswalk sign in the following diagram. The little red guy standing still signals no crosswalk. The two green dot matrices can be displayed alternately to show the motion of walking. Also, you can turn on both LEDs in the dot matrices and display in a fast pace to signal that pedestrians must walk fast. Thus, there are 3 sequences of length 1, 2 and 2, respectively, to illustrate the crosswalk signs of red, green, and yellow lights.



Your project, named "led_display", should be built for "console debug" target and the system clock should be initialized to 80MHz. The project should consist of a main.c program and additional .h and .c files. You can use several existing routines from 5211DEMO stationary files, including clock_pll (to initialize clock), mcf5xxx_irq_enable (to enable interrupts), and mcf5xxx_set_handler (to set an interrupt handler), etc.

