```
/*******************************************************************************/
/********************************************************************************/
/* Name of the program : PES PROGRAM 2                                          */
/*                                                                              */
/* Authors: Aaksha Jaywant, Sarayu Managoli                                    */
/*                                                                             */
/* Compiler used: MCUXPRESSO IDE                                               */
/*                                                                             */
/*                                                                             */
/* Program statement: You will develop a C program to drive the                */
/*                                multicolor LED through multiple timing cycles */
/*                                on board the Freedom KL25Z. Perform fb_run,   */
/*                                fb_debug,                                     */
/*                                 pc_run, pc_debug                             */
/*                                                                             */
/*                                                                             */
/* Applicable Link: https://www.gnu.org/software/make/     ,                   */
/*                  ECEN5813_001_B_ECCR1B55_9_26_2019 lecture                  */
```

MAKEFILE:

```
#############################
# Command for removing files
RM := rm -rf

#############################
# Compiler, linker and includes

ifeq ($(ACTION), FB_RUN)
CC := arm-none-eabi-gcc
LL := arm-none-eabi-gcc
INCLUDES := \
	-I"CMSIS" \
	-I"sources" \
	-I"board" \
	-I"drivers" \
	-I"utilities"

CC_OPTIONS := \
	-c \
	-std=gnu99 \
	-O0 \
	-fno-common \
	-fmessage-length=0 \
	-g3 \
	-ffunction-sections \
	-fdata-sections \
	-fno-builtin \
	-mcpu=cortex-m0plus \
	-mthumb

OBJS := \
	./debug/main.o \
        ./debug/led.o \
        ./debug/startup_mkl25z4.o \
        ./debug/system_MKL25Z4.o \
        ./debug/board.o \
        ./debug/clock_config.o \
        ./debug/peripherals.o \
        ./debug/pin_mux.o \
        ./debug/fsl_clock.o \
        ./debug/fsl_common.o \
        ./debug/fsl_flash.o \
        ./debug/fsl_gpio.o \
        ./debug/fsl_lpsci.o \
        ./debug/fsl_uart.o \
        ./debug/fsl_smc.o \
        ./debug/fsl_debug_console.o

EXE := \
        ./debug/MKL25Z128xxx4_Project_PES_Project2.axf

LL_OPTIONS := \
        -nostdlib -Xlinker -Map="debug/MKL25Z128xxx4_Project_PES_Project2.map" -
Xlinker --gc-sections -Xlinker -print-memory-usage -mcpu=cortex-m0plus -mthumb -T
linkerfile.ld -o $(EXE)

TEMP_FILES := \
        ./debug/main.d \
        ./debug/led.d \
        ./debug/startup_mkl25z4.d \
        ./debug/system_MKL25Z4.d \
        ./debug/board.d \
        ./debug/clock_config.d \
        ./debug/peripherals.d \
        ./debug/pin_mux.d \
        ./debug/fsl_clock.d \
        ./debug/fsl_common.d \
        ./debug/fsl_flash.d \
        ./debug/fsl_gpio.d \
        ./debug/fsl_lpsci.d \
        ./debug/fsl_uart.d \
        ./debug/fsl_smc.d \
        ./debug/fsl_debug_console.d

BUILD_OPTIONS := \
        -D__USE_CMSIS \
        -Ddebug \
        -DSDK_OS_BAREMETAL \
        -DPRINTF_FLOAT_ENABLE=0 \
        -DSCANF_FLOAT_ENABLE=0 \
        -DPRINTF_ADVANCED_ENABLE=0 \
        -DSCANF_ADVANCED_ENABLE=0 \
        -DFRDM_KL252 \
        -DFREEDOM \
        -DFSL_RTOS_BM \
        -DCR_INTERGER_PRINTF \
        -DCPU_MKL25Z128VLK4 \
        -DCPU_MKL25Z128VLK4_cm0plus \
        -DSDK_debugCONSOLE_ITM \
        -D__MCUXPRESSO \
        -D__REDLIB__ \
        -DSDK_debugCONSOLE=1 \
        -specs=redlib.specs

 else ifeq ($(ACTION), FB_DEBUG)
 CC := arm-none-eabi-gcc
 LL := arm-none-eabi-gcc
 INCLUDES := \
        -I"CMSIS" \
        -I"sources" \
        -I"board" \
        -I"drivers" \
        -I"utilities"

 CC_OPTIONS := \
        -c \
        -std=gnu99 \
        -O0 \
        -fno-common \
        -fmessage-length=0 \
        -g3 \
        -ffunction-sections \
        -fdata-sections \
        -fno-builtin \
        -mcpu=cortex-m0plus \
        -mthumb

 OBJS := \
        ./debug/main.o \
        ./debug/led.o \
        ./debug/startup_mkl25z4.o \
        ./debug/system_MKL25Z4.o \
        ./debug/board.o \
        ./debug/clock_config.o \
        ./debug/peripherals.o \
        ./debug/pin_mux.o \
        ./debug/fsl_clock.o \
        ./debug/fsl_common.o \
        ./debug/fsl_flash.o \
        ./debug/fsl_gpio.o \
        ./debug/fsl_lpsci.o \
        ./debug/fsl_uart.o \
        ./debug/fsl_smc.o \
        ./debug/fsl_debug_console.o

 EXE := \
        ./debug/MKL25Z128xxx4_Project_PES_Project2.axf

 LL_OPTIONS := \
        -nostdlib -Xlinker -Map="debug/MKL25Z128xxx4_Project_PES_Project2.map" -
 Xlinker --gc-sections -Xlinker -print-memory-usage -mcpu=cortex-m0plus -mthumb -T
 linkerfile.ld -o $(EXE)

 TEMP_FILES := \
        ./debug/main.d \
        ./debug/led.d \
        ./debug/startup_mkl25z4.d \
        ./debug/system_MKL25Z4.d \
        ./debug/board.d \
        ./debug/clock_config.d \
        ./debug/peripherals.d \
        ./debug/pin_mux.d \
        ./debug/fsl_clock.d \
        ./debug/fsl_common.d \
        ./debug/fsl_flash.d \
        ./debug/fsl_gpio.d \
        ./debug/fsl_lpsci.d \
        ./debug/fsl_uart.d \
        ./debug/fsl_smc.d \
        ./debug/fsl_debug_console.d

 BUILD_OPTIONS := \
        -D__USE_CMSIS \
        -Ddebug \
        -DSDK_OS_BAREMETAL \
        -DPRINTF_FLOAT_ENABLE=0 \
        -DSCANF_FLOAT_ENABLE=0 \
        -DPRINTF_ADVANCED_ENABLE=0 \
        -DSCANF_ADVANCED_ENABLE=0 \
        -DFRDM_KL252 \
        -DFREEDOM \
        -DFSL_RTOS_BM \
        -DCR_INTERGER_PRINTF \
        -DCPU_MKL25Z128VLK4 \
        -DCPU_MKL25Z128VLK4_cm0plus \
        -DSDK_debugCONSOLE_ITM \
        -D__MCUXPRESSO \
        -D__REDLIB__ \
        -DSDK_debugCONSOLE=1 \
        -specs=redlib.specs

 else ifeq ($(ACTION), PC_RUN)
 CC := gcc
 LL := gcc
 INCLUDES := \
        -I"C:\MinGW\include"

 CC_OPTIONS := \
        -c \
        -std=gnu99

 OBJS := \
        ./debug/main.o

 EXE := \
        ./debug/MKL25Z128xxx4_Project_PES_Project2.exe

 LL_OPTIONS := \
         -g -o $(EXE)

 TEMP_FILES := \
        ./debug/main.d

 else ifeq ($(ACTION), PC_DEBUG)
 CC := gcc
 LL := gcc
 INCLUDES := \
        -I"C:\MinGW\include"
```

```
CC_OPTIONS := \
       -c \
       -std=gnu99

OBJS := \
       ./debug/main.o

EXE := \
       ./debug/MKL25Z128xxx4_Project_PES_Project2.exe

LL_OPTIONS := \
        -g -o $(EXE)

TEMP_FILES := \
       ./debug/main.d

endif


############################
# Include generated dependcy files (only if not clean target)
ifneq ($(MAKECMDGOALS),clean)
ifneq ($(strip $(TEMP_FILES)),)
-include $(TEMP_FILES)
endif
endif


all: $(EXE)
       @echo "*** finished building ***"


clean:
       -$(RM) $(EXECUTABLES) $(OBJS) $(TEMP_FILES) $(EXE)
       -$(RM) ./debug/*.map
       -@echo ' '


$(EXE): $(OBJS) linkerfile.ld
       @echo 'Building target: $@'
       @echo 'Invoking: Linker'
       $(LL) $(LL_OPTIONS) $(OBJS) $(LIBS)
       @echo 'Finished building target: $@'
       @echo ' '

ifeq ($(ACTION), FB_RUN)
./debug/%.o: ./source/%.c
       $(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) -DFB_RUN $(INCLUDES) -MMD -MP -
MF"$(@:%.o=%.d)" -MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./board/%.c
       $(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./CMSIS/%.c
       $(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./startup/%.c
       $(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./drivers/%.c
```

```
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./utilities/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
endif

ifeq ($(ACTION), FB_DEBUG)
./debug/%.o: ./source/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) -DFB_DEBUG $(INCLUDES) -MMD -MP -
MF"$(@:%.o=%.d)" -MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./board/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./CMSIS/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./startup/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./drivers/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
./debug/%.o: ./utilities/%.c
	$(CC) $(CC_OPTIONS) $(BUILD_OPTIONS) $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)"
-MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
endif

ifeq ($(ACTION), PC_RUN)
./debug/%.o: ./source/%.c
	$(CC) $(CC_OPTIONS) -DPC_RUN $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)" -
MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
endif

ifeq ($(ACTION), PC_DEBUG)
./debug/%.o: ./source/%.c
	$(CC) $(CC_OPTIONS) -DPC_DEBUG $(INCLUDES) -MMD -MP -MF"$(@:%.o=%.d)" -
MT"$(@:%.o=%.o)" -MT"$(@:%.o=%.d)" -o "$@" "$<"
	Endif
```

CODE:

MAIN.C

```
/*******************************************************************************/
/********************************************************************************/
/* Name of the program : PES PROGRAM 2                                          */
/*                                                                              */
/* Authors: Aaksha Jaywant, Sarayu Managoli                                    */
/*                                                                             */
/* Compiler used: MCUXPRESSO IDE                                               */
/*                                                                             */
/*                                                                             */
/* Program statement: You will develop a C program to drive the                */
/*                               multicolor LED through multiple timing cycles */
/*                                on board the Freedom KL25Z. Perform fb_run,    */
                                  fb_debug,                                      */
/*                                 pc_run, pc_debug                              */
/*                                                                               */
/*                                                                               */
/* Applicable Link: https://www.gnu.org/software/make/     ,                     */
/*                   ECEN5813_001_B_ECCR1B55_9_26_2019 lecture                   */



/**
 * @file    MKL25Z128xxx4_Project.c
 * @brief   Application entry point.
 */
#include <stdio.h>
#include <stdint.h>
#include "time.h"
#ifdef PC_RUN
#include <windows.h>
#endif

#ifdef FB_RUN
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"

#include "led.h"
#endif

#ifdef FB_DEBUG
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#include "led.h"
#endif


int main(void)
{

uint8_t i=0,j=0;
time_t rawtime;
struct tm *info;


#ifdef FB_RUN
       BOARD_InitBootPins();
       BOARD_InitBootClocks();
       BOARD_InitBootPeripherals();
       /* Init FSL debug console. */
       BOARD_InitDebugConsole();
    LED_RED_INIT(1);
    LED_BLUE_INIT(1);
    LED_GREEN_INIT(1);
    while(1)
    {
      LED();
    }
#endif


#ifdef FB_DEBUG
    BOARD_InitBootPins();
    BOARD_InitBootClocks();
    BOARD_InitBootPeripherals();
    /* Init FSL debug console. */
    BOARD_InitDebugConsole();
    LED_RED_INIT(1);
    LED_BLUE_INIT(1);
    LED_GREEN_INIT(1);
    while(1)
    {
      LED();
    }
#endif

#ifdef PC_RUN
    while(1)
    {
      for(j=0;j<3;j++)
      {
             printf("\nLED RED ON");
             Sleep(table[i]);
             i++;
             printf("\nLED RED OFF");
             Sleep(table[i]);
             i++;
      }
      for(j=0;j<3;j++)
      {
             printf("\nLED GREEN ON");
             Sleep(table[i]);
             i++;
             printf("\nLED GREEN OFF");
             Sleep(table[i]);
             i++;
      }
      for(j=0;j<3;j++)
      {
             printf("\nLED BLUE ON");
             Sleep(table[i]);
             i++;
             printf("\nLED BLUE OFF");
             Sleep(table[i]);
             i++;
      }
             if(i==200) break;
    }
#endif

#ifdef PC_DEBUG
```

```
    while(1)
    {
      for(j=0;j<3;j++)
      {
             time( &rawtime );
                    info = localtime( &rawtime );
                    printf("\nLED GREEN ON\t%s\t%lu\n",asctime(info),table[i]);
                    Sleep(table[i]);
                    i++;
                    time( &rawtime );
                    info = localtime( &rawtime );
                    printf("\nLED GREEN OFF\t%s\t%lu\n",asctime(info),table[i]);
                    i++;
      }
      for(j=0;j<3;j++)
      {
             time( &rawtime );
             info = localtime( &rawtime );
             printf("\nLED GREEN ON\t%s\t%lu\n",asctime(info),table[i]);
             Sleep(table[i]);
             i++;
             time( &rawtime );
             info = localtime( &rawtime );
             printf("\nLED GREEN OFF\t%s\t%lu\n",asctime(info),table[i]);
             i++;
      }
      for(j=0;j<3;j++)
      {
             time( &rawtime );
             info = localtime( &rawtime );
             printf("\nLED GREEN ON\t%s\t%lu\n",asctime(info),table[i]);
             Sleep(table[i]);
             i++;
             time( &rawtime );
             info = localtime( &rawtime );
             printf("\nLED GREEN OFF\t%s\t%lu\n",asctime(info),table[i]);
             i++;
      }
      if(i==200) break;
    }
#endif

    return 0;
}
```

MAIN.H

```
/*******************************************************************************/
/********************************************************************************/
/* Name of the program : PES PROGRAM 2                                          */
/*                                                                              */
/* Authors: Aaksha Jaywant, Sarayu Managoli                                    */
/*                                                                             */
/* Compiler used: MCUXPRESSO IDE                                               */
/*                                                                             */
/*                                                                             */
/* Program statement: You will develop a C program to drive the                */
/*                               multicolor LED through multiple timing cycles */
/*                              on board the Freedom KL25Z. Perform fb_run,   */
                                fb_debug,                                     */
/*                               pc_run, pc_debug                             */
/*                                                                            */
/*                                                                            */
/* Applicable Link: https://www.gnu.org/software/make/     ,                  */
/*                 ECEN5813_001_B_ECCR1B55_9_26_2019 lecture                  */

#ifndef SOURCE_MAIN_H_
#define SOURCE_MAIN_H_



#endif /* SOURCE_MAIN_H_ */
```

LED.C

```
#include "led.h"
#include <stdio.h>
#include <stdint.h>

#ifdef FB_RUN
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#endif

#ifdef FB_DEBUG
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#endif

void LED(void);

uint8_t i=0;
uint16_t table[]=
{3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200,1000,400,20
00,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200,1000,
400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200
,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,10
00,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,
100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,10
0,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,
500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,50
0,100,500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,
200,500,100,500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400
,1000,200,500,100,500,100,500,100,1000,200,1000,400,2000,600};

int delay(uint64_t k)
{
	k=k*1000*2.4;
```

```
        while(k!=0)
        {
                k--;
        }
        return 0;
}

#ifdef FB_RUN
void LED(void)
{
        uint8_t j=0;
        for(j=0;j<3;j++)
        {
                LED_RED_ON();
                delay(table[i]);
                i++;
                LED_RED_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
                for(j=0;j<3;j++)
        {
                LED_GREEN_ON();
                delay(table[i]);
                i++;
                LED_GREEN_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
        for(j=0;j<3;j++)
        {
                LED_BLUE_ON();
                delay(table[i]);
                i++;
                LED_BLUE_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }

}
#endif
```

```
#ifdef FB_DEBUG
void LED(void)
{
	uint8_t j=0;
	uint64_t waittime=0;
	for(j=0;j<3;j++)
	{
		LED_RED_ON();
		PRINTF("\nLED RED ON");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
		LED_RED_OFF();
		PRINTF("\nLED RED OFF");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
	}
	if(i==200)
	{
		LED_RED_OFF();
		LED_GREEN_OFF();
		LED_BLUE_OFF();
	}
	for(j=0;j<3;j++)
	{
		LED_GREEN_ON();
		PRINTF("\nLED GREEN ON");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
		LED_GREEN_OFF();
		PRINTF("\nLED GREEN OFF");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
	}
	if(i==200)
	{
		LED_RED_OFF();
		LED_GREEN_OFF();
		LED_BLUE_OFF();
	}
	for(j=0;j<3;j++)
	{
		LED_BLUE_ON();
		PRINTF("\nLED BLUE ON");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
		LED_BLUE_OFF();
		PRINTF("\nLED BLUE OFF");
		delay(table[i]);
		PRINTF("\t%d\n",table[i]);
		i++;
	}
	if(i==200)
	{
```

```
             LED_RED_OFF();
             LED_GREEN_OFF();
             LED_BLUE_OFF();
       }

}
#endif
```

LED.H

```
/*******************************************************************************/
/********************************************************************************/
/* Name of the program : PES PROGRAM 2                                          */
/*                                                                              */
/* Authors: Aaksha Jaywant, Sarayu Managoli                                    */
/*                                                                             */
/* Compiler used: MCUXPRESSO IDE                                               */
/*                                                                             */
/*                                                                             */
/* Program statement: You will develop a C program to drive the                */
/*                              multicolor LED through multiple timing cycles */
/*                              on board the Freedom KL25Z. Perform fb_run,   */
                                fb_debug,                                      */
/*                               pc_run, pc_debug                              */
/*                                                                             */
/*                                                                             */
/* Applicable Link: https://www.gnu.org/software/make/     ,                   */
/*                  ECEN5813_001_B_ECCR1B55_9_26_2019 lecture                  */



#ifndef SOURCE_LED_H_
#define SOURCE_LED_H_



#endif /* SOURCE_MAIN_H_ */
```

LED.C

```
/*******************************************************************************/
/********************************************************************************/
/* Name of the program : PES PROGRAM 2                                          */
/*                                                                              */
/* Authors: Aaksha Jaywant, Sarayu Managoli                                    */
/*                                                                             */
/* Compiler used: MCUXPRESSO IDE                                               */
/*                                                                             */
/*                                                                             */
/* Program statement: You will develop a C program to drive the                */
/*                              multicolor LED through multiple timing cycles */
/*                              on board the Freedom KL25Z. Perform fb_run,   */
                                fb_debug,                                      */
/*                               pc_run, pc_debug                              */
/*                                                                             */
/*                                                                             */
/* Applicable Link: https://www.gnu.org/software/make/     ,                   */
/*                 ECEN5813_001_B_ECCR1B55_9_26_2019 lecture                 */


#include "led.h"
#include <stdio.h>
#include <stdint.h>

#ifdef FB_RUN
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#endif

#ifdef FB_DEBUG
#include "board.h"
#include "peripherals.h"
#include "pin_mux.h"
#include "clock_config.h"
#include "MKL25Z4.h"
#include "fsl_debug_console.h"
#endif

void LED(void);

uint8_t i=0;
uint16_t table[]=
{3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200,1000,400,20
00,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200,1000,
400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,1000,200
,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,100,10
00,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,100,500,
100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,500,10
0,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,500,100,
500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,200,50
0,100,500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400,1000,
200,500,100,500,100,500,100,1000,200,1000,400,2000,600,3000,1000,2000,600,1000,400
,1000,200,500,100,500,100,500,100,1000,200,1000,400,2000,600};

int delay(uint64_t k)
{
        k=k*1000*2.4;
        while(k!=0)
        {
                k--;
        }
        return 0;
}

#ifdef FB_RUN
void LED(void)
{
        uint8_t j=0;
        for(j=0;j<3;j++)
        {
                LED_RED_ON();
                delay(table[i]);
                i++;
                LED_RED_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
                for(j=0;j<3;j++)
        {
                LED_GREEN_ON();
                delay(table[i]);
                i++;
                LED_GREEN_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
        for(j=0;j<3;j++)
        {
                LED_BLUE_ON();
                delay(table[i]);
                i++;
                LED_BLUE_OFF();
                delay(table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }

}
#endif

#ifdef FB_DEBUG
void LED(void)
{
        uint8_t j=0;
        uint64_t waittime=0;
        for(j=0;j<3;j++)
        {
                LED_RED_ON();
                PRINTF("\nLED RED ON");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
                LED_RED_OFF();
                PRINTF("\nLED RED OFF");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
        for(j=0;j<3;j++)
        {
                LED_GREEN_ON();
                PRINTF("\nLED GREEN ON");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
                LED_GREEN_OFF();
                PRINTF("\nLED GREEN OFF");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }
        for(j=0;j<3;j++)
        {
                LED_BLUE_ON();
                PRINTF("\nLED BLUE ON");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
                LED_BLUE_OFF();
                PRINTF("\nLED BLUE OFF");
                delay(table[i]);
                PRINTF("\t%d\n",table[i]);
                i++;
        }
        if(i==200)
        {
                LED_RED_OFF();
                LED_GREEN_OFF();
                LED_BLUE_OFF();
        }

}
#endif
```