

```

1 void SystemClock_Config(void)
2 {
3
4     RCC_OscInitTypeDef RCC_OscInitStruct;
5     RCC_ClkInitTypeDef RCC_ClkInitStruct;
6     RCC_PeriphCLKInitTypeDef PeriphClkInitStruct;
7
8     /**Supply configuration update enable
9      */
10    MODIFY_REG(PWR->CR3, PWR_CR3_SCUEN, 0);
11
12    /**Configure the main internal regulator output voltage
13     */
14    __HAL_PWR_VOLTAGESCALING_CONFIG(PWR_REGULATOR_VOLTAGE_SCALE1);
15
16    while ((PWR->D3CR & (PWR_D3CR_VOSRDY)) != PWR_D3CR_VOSRDY)
17    {
18
19    }
20    /**Configure LSE Drive Capability
21     */
22    HAL_PWR_EnableBkUpAccess();
23
24    __HAL_RCC_LSEDRIVE_CONFIG(RCC_LSEDRIVE_MEDIUMLOW);
25
26    /**Initializes the CPU, AHB and APB busses clocks
27     */
28    RCC_OscInitStruct.OscillatorType = RCC_OSCILLATORTYPE_HSE|RCC_OSCILLATORTYPE_LSE;
29    RCC_OscInitStruct.HSEState = RCC_HSE_ON;
30    RCC_OscInitStruct.LSEState = RCC_LSE_ON;
31    RCC_OscInitStruct.PLL.PLLState = RCC_PLL_ON;
32    RCC_OscInitStruct.PLL.PLLSource = RCC_PLLSOURCE_HSE;
33    RCC_OscInitStruct.PLL.PLLM = 6;
34    RCC_OscInitStruct.PLL.PLLN = 400;
35    RCC_OscInitStruct.PLL.PLLP = 2;
36    RCC_OscInitStruct.PLL.PLLQ = 2;
37    RCC_OscInitStruct.PLL.PLLR = 2;
38    RCC_OscInitStruct.PLL.PLLRGE = RCC_PLL1VCIRANGE_1;
39    RCC_OscInitStruct.PLL.PLLVCOSEL = RCC_PLL1VCOWIDE;
40    RCC_OscInitStruct.PLL.PLLFRACN = 0;
41    if (HAL_RCC_OscConfig(&RCC_OscInitStruct) != HAL_OK)
42    {
43        _Error_Handler(__FILE__, __LINE__);
44    }
45
46    /**Initializes the CPU, AHB and APB busses clocks
47     */
48    RCC_ClkInitStruct.ClockType = RCC_CLOCKTYPE_HCLK|RCC_CLOCKTYPE_SYSCLK
49                                |RCC_CLOCKTYPE_PCLK1|RCC_CLOCKTYPE_PCLK2
50                                |RCC_CLOCKTYPE_D3PCLK1|RCC_CLOCKTYPE_D1PCLK1;
51    RCC_ClkInitStruct.SYSCLKSource = RCC_SYSCLKSOURCE_PLLCLK;
52    RCC_ClkInitStruct.SYSCLKDivider = RCC_SYSCLK_DIV1;
53    RCC_ClkInitStruct.AHBCLKDivider = RCC_HCLK_DIV2;
54    RCC_ClkInitStruct.APB3CLKDivider = RCC_APB3_DIV2;
55    RCC_ClkInitStruct.APB1CLKDivider = RCC_APB1_DIV2;
56    RCC_ClkInitStruct.APB2CLKDivider = RCC_APB2_DIV2;
57    RCC_ClkInitStruct.APB4CLKDivider = RCC_APB4_DIV2;
58
59    if (HAL_RCC_ClockConfig(&RCC_ClkInitStruct, FLASH_LATENCY_2) != HAL_OK)
60    {
61        _Error_Handler(__FILE__, __LINE__);
62    }
63
64    PeriphClkInitStruct.PeriphClockSelection = RCC_PERIPHCLK_RTC|RCC_PERIPHCLK_LPUART1
65                                |RCC_PERIPHCLK_UART4|RCC_PERIPHCLK_SPI4
66                                |RCC_PERIPHCLK_I2C1|RCC_PERIPHCLK_QSPI;
67    PeriphClkInitStruct.QspiClockSelection = RCC_QSPICLK_SOURCE_D1HCLK;
68    PeriphClkInitStruct.Spi45ClockSelection = RCC_SPI45CLK_SOURCE_D2PCLK1;
69    PeriphClkInitStruct.Usart234578ClockSelection = RCC_USART234578CLK_SOURCE_D2PCLK1;

```

```
70     PeriphClkInitStruct.I2c123ClockSelection = RCC_I2C123CLKSOURCE_D2PCLK1;
71     PeriphClkInitStruct.Lpuart1ClockSelection = RCC_LPUART1CLKSOURCE_D3PCLK1;
72     PeriphClkInitStruct.RTCClockSelection = RCC_RTCCLKSOURCE_LSE;
73     if (HAL_RCCEx_Pериф CLKConfig(&PeriphClkInitStruct) != HAL_OK)
74     {
75         _Error_Handler(__FILE__, __LINE__);
76     }
77
78     /**Configure the SysTick interrupt time
79     */
80     HAL_SYSTICK_Config(SystemCoreClock/1000);
81
82     /**Configure the SysTick
83     */
84     HAL_SYSTICK_CLKSourceConfig(SYSTICK_CLKSOURCE_HCLK);
85
86     /* SysTick_IRQn interrupt configuration */
87     HAL_NVIC_SetPriority(SysTick_IRQn, 15, 0);
88 }
89
```