

## 实验 5 - RNG 随机数发生器实验

### 1. 实验目的

掌握 NRF24LE1 的 rng(Random number generator) 随机数发生器的使用。

### 2. 实验内容

利用 NRF24LE1 随机数发生器生成随机数，每隔 500ms 通过串口输出一次随机数数值。

### 3. 实验原理

#### 1. 随机数发生器特性

- 基于热噪声原理的非确定结构算法；
- 无需种子值；
- 非重复序列；
- 校正算法确保统计学的均匀分布；
- 最高 10KB/s 的速率；
- 处理器待机时仍可工作。

#### 2. 寄存器配置

随机数发生器通过两个寄存器 RNGCTL 和 RNGDAT 来控制 and 读取数据。

- RNGCTL: 包含控制位和状态位；
- RNGDAT: 包含产生的随机数据。

表 1: RNGCTL 寄存器

地址	位	名称	读写	功能	复位值
0xD6	7	powerUp	RW	随机数发生器上电。	0x00
	6	correctorEn	RW	使能校正算法。	
	5	resultReady	R	数据就绪标志。当新的随机数 RNGDAT 有效时置位，读出数据后或上电时清除该位。	
	4:0	—		未使用。	

表 2: RNGDAT 寄存器

地址	位	名称	读写	功能	复位值
0xD7	7:0	data	R	随机数据。	0x00

按照上述内容，使用随机数发生器时，可按照下述方式进行：

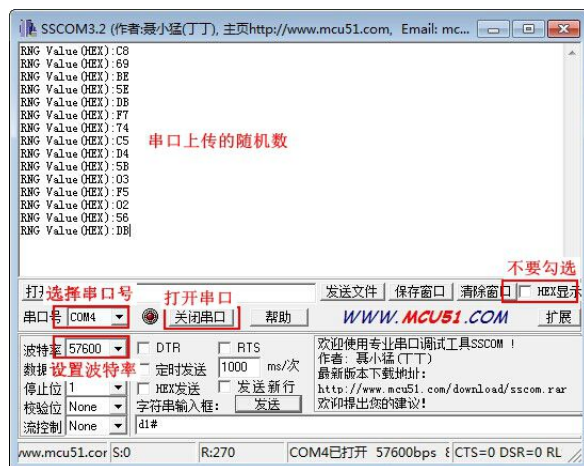
- ☐ 使能校正算法；
- ☐ RNG 上电；
- ☐ 查询数据标志是否置位，如标志置位，读取数据。

## 4. 实验步骤

- 在 Keil uVision4 中打开工程 “rng.uvproj” 工程；
- 编译工程，注意查看编译输出栏，观察编译的结果，如果有错误，修改程序，直到编译成功为止；

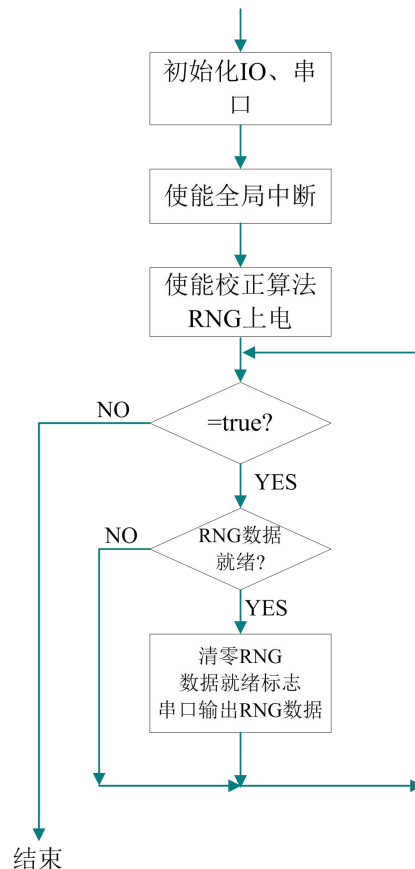


- 将编译生成的 HEX 文件 “rng.hex”（该文件位于工程目录下的 “Object” 文件夹中）通过编程器下载到开发板中运行。
- 打开串口调试助手，选择串口号，设置波特率为 57600，打开串口，注意不要勾选 “HEX 显示”。观察串口输出的随机数据。



## 5. 实验程序

### 5.1. 程序流程



## 5.2. 程序清单

```
#define D1    P00 //开发板上的指示灯 D1
```

```

/*****
*描 述：配置 IO P0.0 为输出，驱动 LED。P03 输出: UART TXD, P04:输入 UART RXD
*入 参：无
*返回值：无
*****/

```

```
void IO_Init(void)
```

```

{
    P0DIR &= ~0x01; //配置 P0.0 为输出
    P0DIR &= ~0x08; //P03:输出 UART TXD
    P0DIR |= 0x10;   //P04:输入 UART RXD
    D1 = 1;         //设置 D1 初始状态为熄灭
}

```

```

/*****
*描 述：串口打印字符串
*入 参：无
*返回值：无
*****/

```

```
void PutString(char *s)
{
    while(*s != 0)
        hal_uart_putchar(*s++);
}

/*****
*描 述：主函数
*入 参：无
*返回值：无
*****/
void main(void)
{
    uint8_t RngDat;

    IO_Init(); //初始化 IO
    hal_uart_init(UART_BAUD_57K6); // 初始化 UART，波特率 57600
    while(hal_clk_get_16m_source() != HAL_CLK_XOSC16M) // 等待时钟稳定
        ;
    EA = 1; // 开启全局中断

    hal_rng_bias_corr_enable(true); // 使能校正算法
    hal_rng_power_up(true); // RNG 上电

    while(1)
    {
        delay_ms(500); //延时 500ms
        D1 = ~D1; //D1 指示灯闪烁，指示系统工作正常

        if(hal_rng_data_ready()) //RNG 数据就绪?
        {
            RngDat = hal_rng_read(); //读取 RNG 数据
            PutString("RNG Value(HEX):"); //串口打印出 RNG 数据
            PutHexString(RngDat);
            PutString("\n");
        }
    }
}
```