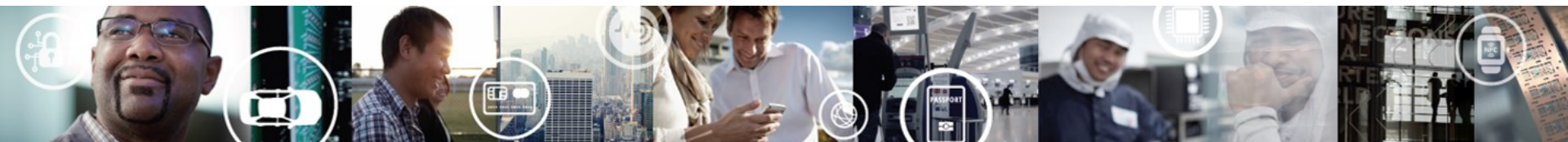


LPC82X 培训资料

循环冗余检查CRC引擎 动手实验

MAY, 2016



EXTERNAL USE



SECURE CONNECTIONS
FOR A SMARTER WORLD

内容

- 实验简介（目的，内容，结果）
- 软/硬件环境搭建
- 实验步骤
- 相关底层驱动APIs说明

实验简介

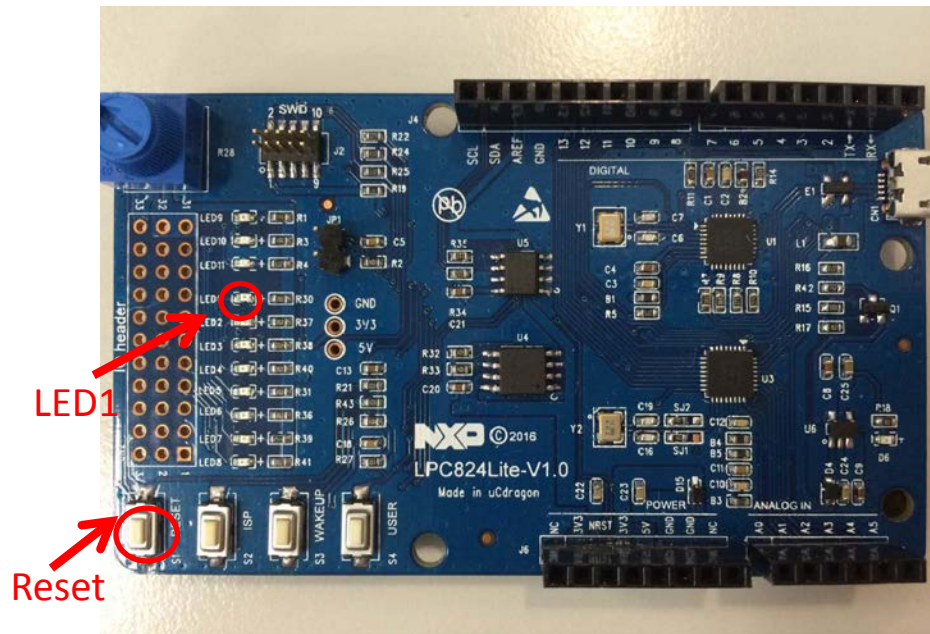
- **目的**：通过本实验，理解和掌握LPC82x CRC的过程
- **描述**：使用CRC引擎计算8位、16位和32位的CRC计算。CRC引擎持续的进行CRC计算并且校验CRC的结果。system tick产生一个10Hz的时钟，每隔一个时钟，把CRC的结果result[2]进行减1(即修改为错误的值)。每个循环计算的result[0] ~ result[4]的值再进行CRC32计算
- **结果**:计算出的结果gencrc与期盼的结果进行比较，正确LED1灭，不正确LED1亮

软/硬件环境搭建

- 硬件
 - 评估板：LPC824Lite-V1.0
- 工程位置
 - ..\peri_example\crc\crc_example\project_crc_example.uvprojx

硬件配置

- 无需特别的硬件配置
 - 使用LED D1来指示CRC的计算结果正确与否
 - 使用复位按钮来开始整个动手实验



动手实验步骤

- 第一步 - 编译下载CRC的工程生成的可执行文件
- 第二步 - 给板上电，按复位按钮开始动手实验
- 第三步 - LED1 不停的闪烁，表明LPC824开始工作，LED1亮时表明CRC的结果错误（程序中人为修改为错误的值），LED1灭时表明CRC的结果正确

相关底层驱动APIs说明

1.初始化CRC。使能CRC的时钟

void Chip_CRC_Init(void)

2. 进行CCITT多项式CRC计算

*uint32_t Chip_CRC_CRC8(const uint8_t *data, uint32_t bytes)*

3.进行CRC-16多项式CRC计算

*uint32_t Chip_CRC_CRC16(const uint16_t *data, uint32_t hwords)*

4.进行CRC-32多项式CRC计算

*uint32_t Chip_CRC_CRC32(const uint32_t *data, uint32_t words)*



SECURE CONNECTIONS
FOR A SMARTER WORLD