

# MCX A153 ADC DMA 定时器采样应用

## 1. 概述

该演示程序在NXP MCXA153微控制器开发板上实现了使用DMA和硬件定时器触发的高速ADC采样。程序展示了LPADC、EDMA和LPTMR等外设的高级协调配置，实现连续数据采集功能。

## 2. 系统配置

系统使用外部20MHz晶振，并通过FRO自动校准实现高精度时钟源：

- 内核时钟:** 96MHz
- 定时器时钟:** 96MHz，用于高精度计时
- 串口波特率: 115200

## 3. 开发环境

- IDE:** Keil MDK
- RTOS:** RT-Thread

## 4. DMA+ADC+定时器采样模块

### 4.1 硬件配置

- ADC控制器:** ADC0
- ADC通道:** 通道0 (引脚P2\_0)
- DMA控制器:** EDMA
- DMA通道:** 通道0
- 定时器:** LPTMR0 (低功耗定时器)
- 触发链路:** LPTMR0 → INPUTMUX → ADC0

### 4.2 核心功能

#### 4.2.1 1. 可配置采样率

- 默认值:** 6.4kHz采样率
- 接口:** `ADC_DMA_SetSampleRate(uint32_t sampleRateHz)`

#### 4.2.2 2. 连续DMA缓冲

- 缓冲区大小:** 128个采样点 (ADC\_DMA\_BUFFER\_LENGTH)
- 数据类型:** 32位对齐，优化DMA性能
- 内存区域:** 非缓存区段

## 4.3 工作流程

### 1. 初始化阶段

- 配置ADC0: 12位分辨率, 硬件触发模式
- 配置EDMA: 循环传输模式, 中断使能
- 配置LPTMR0: 定时触发ADC转换
- 配置CTIMER0: 高精度时间戳

### 2. 采样阶段

- LPTMR0定时触发ADC转换
- ADC完成转换后触发DMA传输
- DMA将数据写入缓冲区
- DMA中断处理: 记录时间戳, 更新计数

### 3. 数据处理

- 通过 `ADC_DMA_GetResults()` 获取最新数据
- 数据包含ADC值、时间戳和采样计数
- 支持实时数据分析和处理

## 4.4 API接口

```
1 // 初始化ADC DMA定时器系统
2 status_t ADC_DMA_Init(void);
3
4 // 启动采样
5 status_t ADC_DMA_Start(void);
6
7 // 停止采样
8 status_t ADC_DMA_Stop(void);
9
10 // 设置采样率
11 status_t ADC_DMA_SetSampleRate(uint32_t sampleRateHz);
12
13 // 获取采样结果
14 adc_dma_timer_result_t ADC_DMA_GetResults(void);
15
16 // 反初始化
17 void ADC_DMA_Deinit(void);
```

## 4.5 注意事项

1. **DMA缓冲区**: 使用非缓存内存区域，确保数据一致性
2. **中断优先级**: DMA中断优先级需合理配置，避免数据丢失
3. **采样率限制**: 受ADC转换时间和DMA传输速度限制
4. **电源管理**: LPTMR0支持低功耗模式，适合电池供电应用

## 5. 编译和运行

---

1. 在Keil MDK中打开工程
2. 编译并下载到FRDM-MCXA153开发板
3. 通过串口或调试器查看采样数据
4. 可通过修改 `g_currentSampleRate` 调整采样率