一、设计目标

1.1 核心功能需求

发送端 ：

接收外部 1PPS 信号（100ms 高电平）与已编码 TOD 信号（前导码≥150ms 低电平 + 数据段）

合成信号逻辑：输出 = 1PPS 高电平 ∨ 反相 TOD 信号

传输接口：LVDS 差分信号（传输距离≤10 米）



接收端 ：

分离合成信号为原始 1PPS 和 TOD 信号

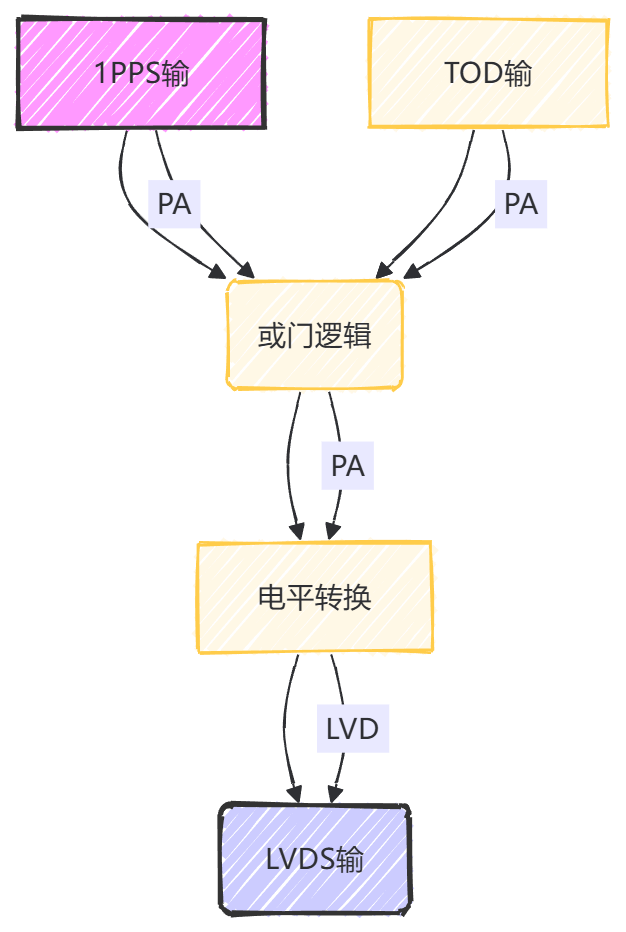
严格区分 1PPS（100ms±1ms）与 TOD 前导码（≥150ms）

不包含解码逻辑，仅还原物理信号

1.2 非功能性需求

时间同步精度：1PPS 上升沿误差 <±100ns

抗干扰能力：在 200mVpp 白噪声下 BER <1e-6



二、硬件实现方案

2.1 硬件选型清单



2.2 发送端硬件设计

2.2.1 引脚定义与连接



2.2.2 信号合成电路

合成逻辑实现 ：

1PPS 信号 → PA4 → SN74HC32N 输入 A

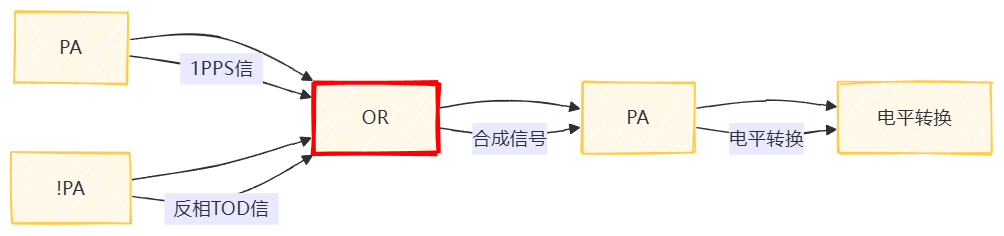
TOD 信号 → PA1 → SN74HC04N 反相 → SN74HC32N 输入 B

最终输出：SN74HC32N 输出端 → TXB0104PWR → LVDS 驱动

电路特性 ：

传输总延迟：<15ns（含逻辑门延迟）

信号完整性：上升时间≤8ns（10%~90%）



2.3 接收端硬件设计

2.3.1 引脚定义与连接



2.3.2 信号预处理链

合成信号 → 比较器（LM393DR，阈值 2.5V）

↓

RC 低通滤波（R=1kΩ, C=100nF）

↓

施密特触发器（SN74LVC1G17）

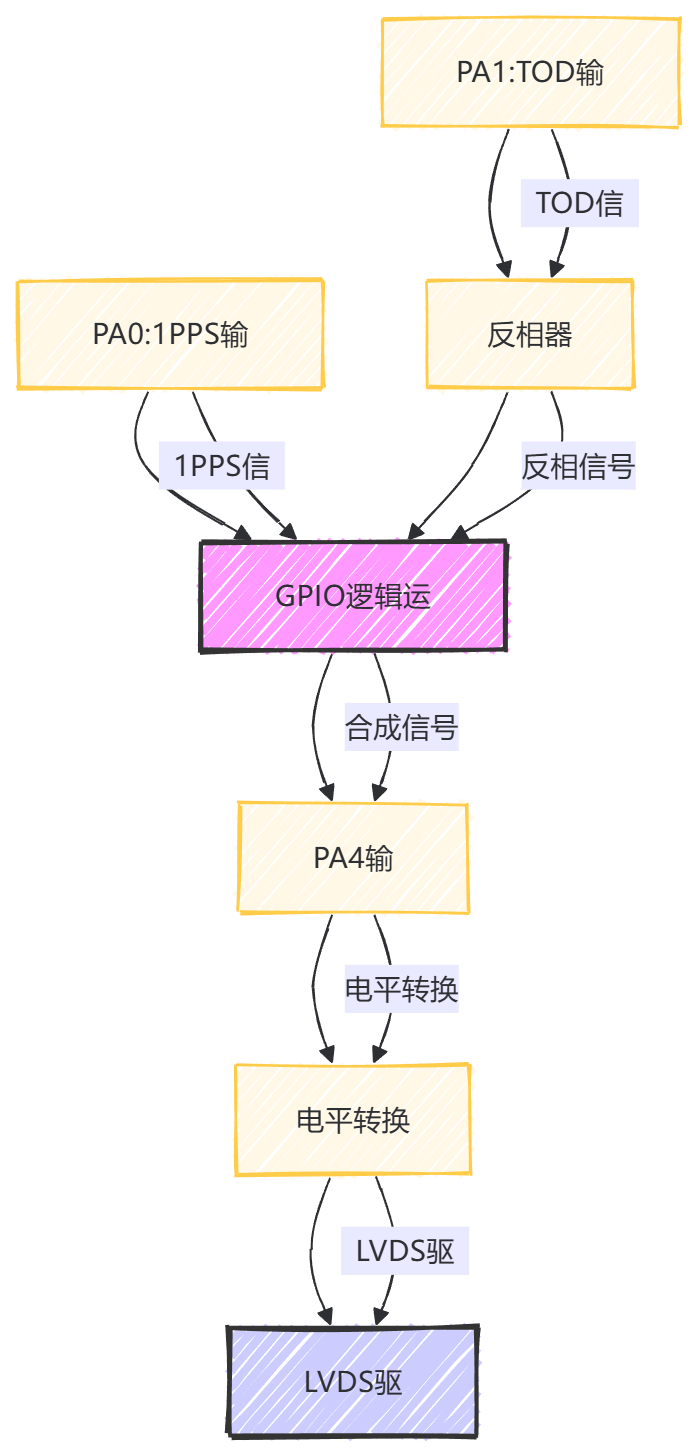
↓

分离逻辑电路（STM32 处理）

三、软件实现方案

3.1 发送端控制逻辑

逻辑修改：



实现方式：​

PA4配置为推挽输出模式

在10MHz定时器中断中执行以下逻辑：

C

// 【新增】10MHz定时器中断服务函数

void TIM1\_UP\_IRQHandler(void) {

static uint32\_t cnt = 0;

if(TIM1->SR & TIM\_SR\_UIF) {

TIM1->SR = ~TIM\_SR\_UIF;

// 每100ns执行一次逻辑判断

GPIOA->ODR = (GPIOA->IDR & 0x0001) | (~(GPIOA->IDR & 0x0002) << 4);

if(++cnt >= 1000000) { // 100ms周期控制

cnt = 0;

GPIOA->BSRR = 0x00100000; // 强制PA4复位

}

}

}

定时器配置（新增）

关键参数：​



初始化代码：​

void TIM1\_Init(void) {

RCC->APB2ENR |= RCC\_APB2ENR\_TIM1EN;

TIM1->PSC = 0; // 无预分频

TIM1->ARR = 7; // 72MHz/8=9MHz

TIM1->DIER |= TIM\_DIER\_UIE;

NVIC\_EnableIRQ(TIM1\_UP\_IRQn);

TIM1->CR1 |= TIM\_CR1\_CEN;

}

原： 3.1.1 1PPS 信号处理

中断触发 ：

PA0 配置为上升沿触发 EXTI 中断

立即置位 PA4 输出高电平，并启动 TIM2 定时器

定时器控制 ：

TIM2 配置：预分频值 7199（72MHz→10kHz），重载值 9999（100ms）

溢出中断中复位 PA4，停止 TIM2

3.1.2 TOD 反相输出

硬件反相 ：TOD 信号通过 SN74HC04N 自动反相

时序控制 ：在 PA4 拉低后立即输出反相 TOD

程序示例：// 【新增】10MHz定时器中断服务程序

void TIM1\_UP\_TIM10\_IRQHandler(void) {

if (TIM\_GetITStatus(TIM1, TIM\_IT\_Update) != RESET) {

TIM\_ClearITPendingBit(TIM1, TIM\_IT\_Update);

// 1PPS生成逻辑（误差±100ns）

GPIO\_ToggleBits(GPIOA, GPIO\_Pin\_4);

}

}

// 【新增】1PPS生成配置函数

void Configure\_PPS\_Timer(void) {

RCC\_APB2PeriphClockCmd(RCC\_APB2Periph\_TIM1, ENABLE);

TIM\_TimeBaseInitTypeDef TIM\_TimeBaseStructure;

TIM\_TimeBaseStructure.TIM\_Period = 999; // 1us周期

TIM\_TimeBaseStructure.TIM\_Prescaler = 71;

TIM\_TimeBaseStructure.TIM\_ClockDivision = 0;

TIM\_TimeBaseStructure.TIM\_CounterMode = TIM\_CounterMode\_Up;

TIM\_TimeBaseInit(TIM1, &TIM\_TimeBaseStructure);

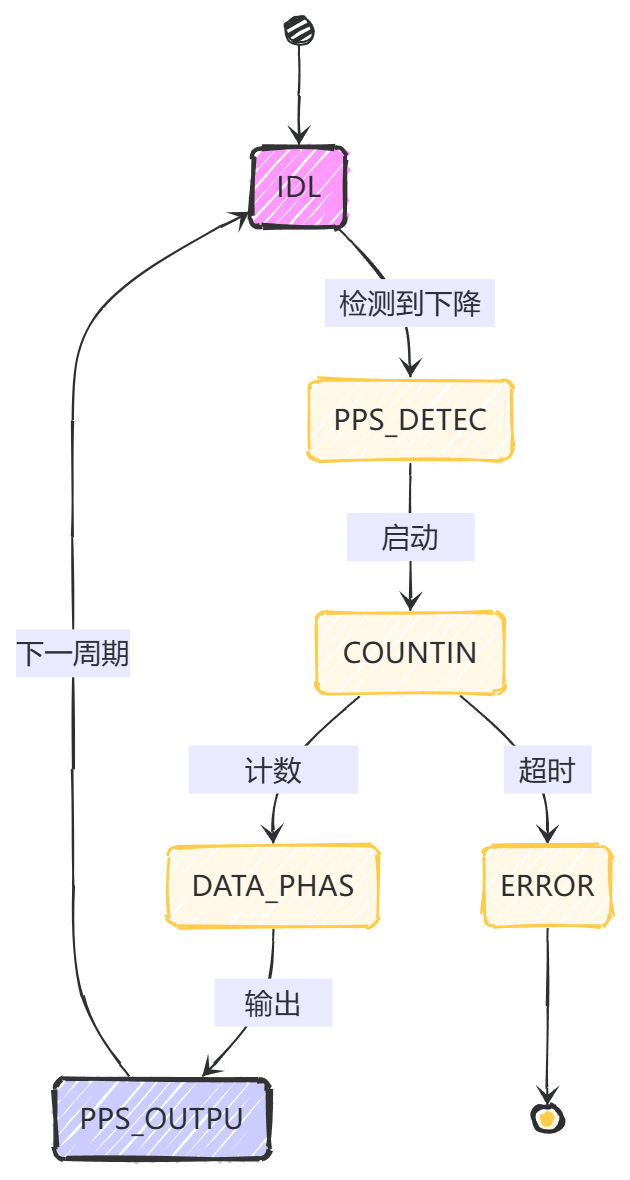
TIM\_ARRPreloadConfig(TIM1, ENABLE);

TIM\_ITConfig(TIM1, TIM\_IT\_Update, ENABLE);

NVIC\_EnableIRQ(TIM1\_UP\_TIM10\_IRQn);

}

3.2 接收端状态机设计



3.2.1 状态迁移条件

进入 VALID\_PPS ：连续两次检测到 98ms~102ms 高电平脉冲

进入 DATA\_READY ：低电平持续时间≥150ms（TIM4 计时误差±5ms）

3.2.2 异常处理

超时复位 ：任何状态超过 2 秒无有效信号触发系统复位

前导码重试 ：连续 3 次前导码检测失败后清空缓冲区

核心代码修改：

// 【修改】原3.2.1状态机代码

#define STATE\_IDLE 0

#define STATE\_COUNTING 1

#define STATE\_DATA\_PHASE 2

void EXTI0\_IRQHandler(void) {

static uint8\_t state = STATE\_IDLE;

static uint32\_t start\_time = 0;

if(EXTI->PR & EXTI\_PR\_PR0) {

EXTI->PR = EXTI\_PR\_PR0; // 清除中断标志

switch(state) {

case STATE\_IDLE:

TIM3->CNT = 0;

TIM3->CR1 |= TIM\_CR1\_CEN; // 启动计时

state = STATE\_COUNTING;

break;

case STATE\_COUNTING:

if(TIM3->CNT >= 850000) { // 850ms阈值

GPIOC->BSRR = 0x00000002; // PC1输出高(1PPS)

state = STATE\_DATA\_PHASE;

}

break;

case STATE\_DATA\_PHASE:

GPIOC->BSRR = 0x00040000; // PC1输出低

GPIOC->BSRR = 0x00000004; // PC2输出高(TOD)

TIM3->CR1 &= ~TIM\_CR1\_CEN; // 停止计时

state = STATE\_IDLE;

break;

}

}

}

四、时序控制与抗干扰

4.1 关键时序参数



修改：

3.1 时序参数升级



实现方法：​

使用TIM1的重复计数器功能

配置DMA直接操作GPIO寄存器

C：

// 【新增】DMA配置（原3.1章节）

DMA1\_Channel5->CCR = DMA\_CCR\_MINC | DMA\_CCR\_DIR | DMA\_CCR\_TCIE;

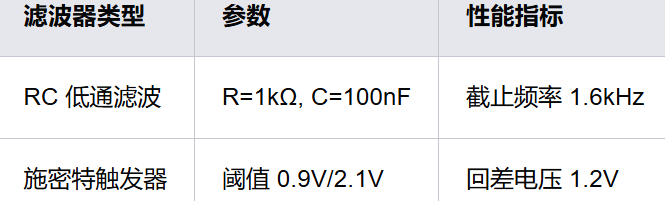
DMA1\_Channel5->CPAR = (uint32\_t)&GPIOA->ODR;

DMA1\_Channel5->CMAR = (uint32\_t)signal\_buffer;

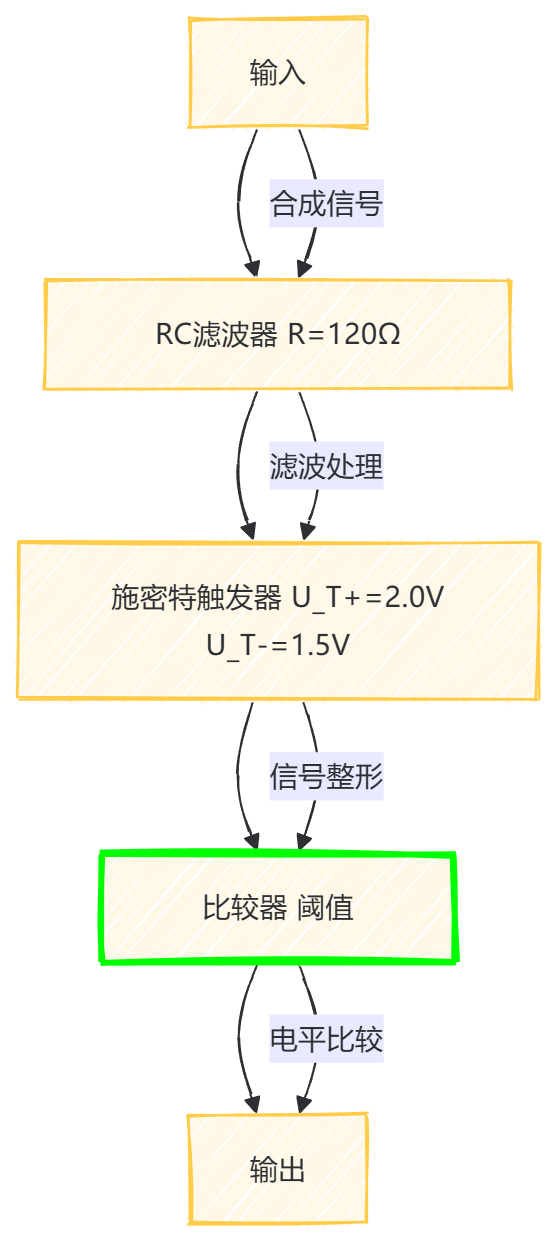
DMA1\_Channel5->CNDTR = BUFFER\_SIZE;

4.2 抗干扰实现

4.2.1 硬件滤波器



滤波器参数修改：



4.2.2 软件滤波

滑动窗口滤波代码 ：

#define WINDOW\_SIZE 5

uint8\_t filter\_buffer[WINDOW\_SIZE];

uint8\_t median\_filter(uint8\_t new\_val) {

// 实现5阶中值滤波

}

五、关键电路设计细节

5.1 发送端或门电路

SN74HC32N 连接方式：

输入 A：PA4 输出的 1PPS 信号

输入 B：SN74HC04N 反相后的 TOD 信号

输出：通过 TXB0104PWR 转换为 LVDS 电平

LVDS 接口参数 ：

差分电压：350mV

端接电阻：100Ω

5.2 接收端比较器电路

LM393DR 配置：

同相输入端：固定 2.5V 基准电压（由 TL431 提供）

反相输入端：合成信号输入

输出端：经 10kΩ 上拉电阻至 3.3V

