

# T2MI PPS Generator для FPGA Lattice LFE5U-25F-6BG256C

## Описание проекта

Данный проект представляет собой готовое решение для генерации высокоточного сигнала PPS (Pulse Per Second) на основе временных штампов, извлекаемых из потока T2-MI стандарта DVB-T2. Проект разработан специально для FPGA Lattice LFE5U-25F-6BG256C и готов для открытия в среде разработки Diamond Lattice.

## Основные возможности

- **Парсинг T2-MI потока:** Автоматическое обнаружение и декодирование пакетов T2-MI
- **Извлечение временных штампов:** Обработка пакетов типа 0x20 с временными метками DVB-T2
- **Генерация PPS:** Формирование высокоточного сигнала 1 импульс в секунду
- **Синхронизация:** Автоматическая синхронизация с входящими временными штампами
- **Диагностика:** Встроенные средства мониторинга и отладки
- **Индикация состояния:** LED индикаторы для визуального контроля работы

## Архитектура системы

### Структура модулей

```
t2mi_pps_top (верхний уровень)
├── t2mi_packet_parser (парсер T2-MI пакетов)
├── timestamp_extractor (извлечение временных штампов)
├── pps_generator (генератор PPS)
└── sync_modules (модули синхронизации)
```

## Основные компоненты

1. **T2-MI Packet Parser** ( `t2mi_packet_parser.v` )
2. Синхронизация с потоком T2-MI

3. Извлечение отдельных пакетов
4. Определение типа пакетов
5. Контроль целостности данных
6. **Timestamp Extractor** ( `timestamp_extractor.v` )
7. Фильтрация пакетов типа 0x20
8. Декодирование структуры временных штампов
9. Извлечение секунд и долей секунды
10. Валидация данных
11. **PPS Generator** ( `pps_generator.v` )
12. Отслеживание текущего времени
13. Синхронизация с входящими штампами
14. Генерация точных PPS импульсов
15. Контроль дрейфа и ошибок
16. **Sync Modules** ( `sync_modules.v` )
17. Синхронизация тактовых доменов
18. Синхронизация сброса
19. Обработка метастабильности

## Технические характеристики

### Целевая FPGA

- **Модель:** Lattice LFE5U-25F-6BG256C
- **Семейство:** ECP5
- **Логические элементы:** 24,000 LUT
- **Встроенная память:** 1,032,192 бит
- **Количество I/O:** 197
- **Корпус:** CABGA-256

### Тактовые частоты

- **Системная частота:** 100 МГц
- **T2-MI частота:** 27 МГц (типовая)
- **PPS точность:**  $\pm 10$  нс (при стабильном источнике)

## Интерфейсы

### Входные сигналы

- `clk_100mhz` - системная тактовая частота 100 МГц
- `rst_n` - сигнал сброса (активный низкий уровень)
- `t2mi_clk` - тактовая частота T2-MI потока
- `t2mi_valid` - сигнал валидности данных T2-MI
- `t2mi_data[7:0]` - байт данных T2-MI
- `t2mi_sync` - сигнал синхронизации T2-MI

### Выходные сигналы

- `pps_out` - основной выход PPS (высокоточный)
- `timestamp_valid` - флаг валидности извлеченного штампа
- `sync_locked` - индикатор синхронизации с T2-MI
- `debug_status[7:0]` - регистр состояния для отладки
- `led_power` - индикатор питания
- `led_sync` - индикатор синхронизации
- `led_pps` - индикатор PPS импульсов
- `led_error` - индикатор ошибок

## Структура временного штампа DVB-T2

Согласно стандарту DVB-T2, временной штамп передается в пакетах типа 0x20 и имеет следующую структуру:

Поле	Размер	Описание
rfu	4 бита	Зарезервировано (должно быть 0)
bw	4 бита	Код ширины канала
utco	13 бит	Коррекция UTC (секунды)
seconds_since_2000	40 бит	Секунды с 1 января 2000 года
subseconds	32 бита	Доля секунды (высокая точность)

# Установка и использование

## Требования к системе

- Lattice Diamond 3.12 или новее
- ModelSim или Icarus Verilog (для симуляции)
- Make (для автоматизации сборки)

## Открытие проекта в Diamond Lattice

1. Запустите Lattice Diamond
2. Выберите "File" → "Open" → "Project"
3. Откройте файл `T2MI_PPS_Generator.ldf`
4. Проект автоматически загрузится со всеми исходными файлами

## Сборка проекта

### Через Diamond Lattice GUI:

1. В Project Navigator выберите Implementation "impl1"
2. Запустите "Process" → "Run All" для полной сборки
3. Результирующий bitstream будет в папке `impl1/`

### Через командную строку:

```
# Полная сборка
make all

# Только синтез
make synthesis

# Только трассировка
make place_route

# Генерация bitstream
make bitstream
```

## Симуляция

```
# С ModelSim
make simulate
```

```
# C Icarus Verilog
make sim_icarus
```

## Назначение выводов FPGA

### Основные сигналы

- **P3:** clk\_100mhz (системная частота)
- **P4:** t2mi\_clk (частота T2-MI)
- **T1:** rst\_n (сброс)
- **E1:** pps\_out (выход PPS) ★

### T2-MI интерфейс

- **R1-H1:** t2mi\_data[7:0] (данные)
- **G1:** t2mi\_valid (валидность)
- **F1:** t2mi\_sync (синхронизация)

### Индикаторы состояния

- **K2:** led\_power (питание)
- **J2:** led\_sync (синхронизация)
- **H2:** led\_pps (PPS активность)
- **G2:** led\_error (ошибки)

### Отладочные сигналы

- **D1:** timestamp\_valid
- **C1:** sync\_locked
- **B1-L2:** debug\_status[7:0]

## Принцип работы

### Алгоритм обработки

1. Инициализация
2. Сброс всех модулей
3. Ожидание стабилизации тактовых сигналов
4. Синхронизация с T2-MI

5. Поиск синхробайта 0x47
6. Установление синхронизации потока
7. Переход в режим парсинга пакетов
8. **Парсинг пакетов**
9. Извлечение заголовка пакета
10. Определение типа и длины
11. Буферизация данных пакета
12. **Обработка временных штампов**
13. Фильтрация пакетов типа 0x20
14. Декодирование полей штампа
15. Валидация корректности данных
16. **Генерация PPS**
17. Отслеживание текущего времени
18. Синхронизация с входящими штампами
19. Генерация импульсов на границах секунд

## Обработка ошибок

Система включает несколько уровней контроля ошибок:

- **Синхронизация потока:** Контроль наличия синхробайтов
- **Валидация пакетов:** Проверка длины и структуры
- **Проверка штампов:** Валидация полей временных меток
- **Контроль дрейфа:** Мониторинг точности PPS

## Настройка и калибровка

### Параметры конфигурации

В файле `pps_generator.v` можно настроить следующие параметры:

```
parameter CLK_FREQ_HZ = 100_000_000;    // Частота системной
тактовой
parameter PPS_PULSE_WIDTH = 1000;        // Длительность PPS
импульса (10 мкс)
```

```
parameter SYNC_THRESHOLD = 1000;           // Порог синхронизации
parameter MAX_DRIFT = 10000;               // Максимальный дрейф
```

## Калибровка точности

1. Подключите высокоточный частотомер к выходу PPS
2. Сравните с эталонным источником времени
3. При необходимости скорректируйте `subsec_increment`

## Диагностика и отладка

### Регистр состояния `debug_status[7:0]`

Бит	Назначение
7	pps_error - ошибка генерации PPS
6	extractor_error - ошибка извлечения штампов
5	parser_error - ошибка парсинга пакетов
4	timestamp_ready - готовность штампа
3	timestamp_valid - валидность штампа
2	packet_valid - валидность пакета
1	sync_locked - синхронизация установлена
0	rst_n_sync - состояние сброса

## Типичные проблемы и решения

**Проблема:** LED sync не загорается - **Причина:** Отсутствие синхронизации T2-MI -

**Решение:** Проверьте подключение `t2mi_clk` и `t2mi_sync`

**Проблема:** LED error горит постоянно - **Причина:** Ошибки в данных или конфигурации - **Решение:** Проверьте `debug_status` для определения источника

**Проблема:** PPS не генерируется - **Причина:** Отсутствие валидных временных штампов - **Решение:** Убедитесь в наличии пакетов типа 0x20 в потоке

# Файловая структура проекта

```
t2mi_pps_project/
├── T2MI_PPS_Generator.ldf      # Основной файл проекта Diamond
├── T2MI_PPS_Generator1.sty    # Файл стратегии синтеза
├── Makefile                    # Автоматизация сборки
├── README.md                  # Данная документация
├── src/                        # Исходные файлы HDL
│   ├── t2mi_pps_top.v         # Модуль верхнего уровня
│   ├── t2mi_packet_parser.v   # Парсер T2-MI пакетов
│   ├── timestamp_extractor.v  # Извлечение временных штампов
│   ├── pps_generator.v        # Генератор PPS
│   └── sync_modules.v         # Модули синхронизации
├── constraints/               # Файлы ограничений
│   └── t2mi_pps.lpf           # Назначение выводов и тайминги
├── sim/                       # Файлы симуляции
│   └── t2mi_pps_tb.v          # Testbench
└── doc/                       # Дополнительная документация
```

## Лицензия и авторские права

Проект разработан Manus AI для использования с FPGA Lattice LFE5U-25F-6BG256C. Код предоставляется "как есть" для образовательных и исследовательских целей.

## Поддержка и контакты

При возникновении вопросов или проблем с проектом:

1. Проверьте раздел "Диагностика и отладка"
2. Убедитесь в корректности подключения сигналов
3. Проверьте совместимость версии Diamond Lattice

---

**Дата создания:** 6 июня 2025

**Версия проекта:** 1.0

**Автор:** Manus AI