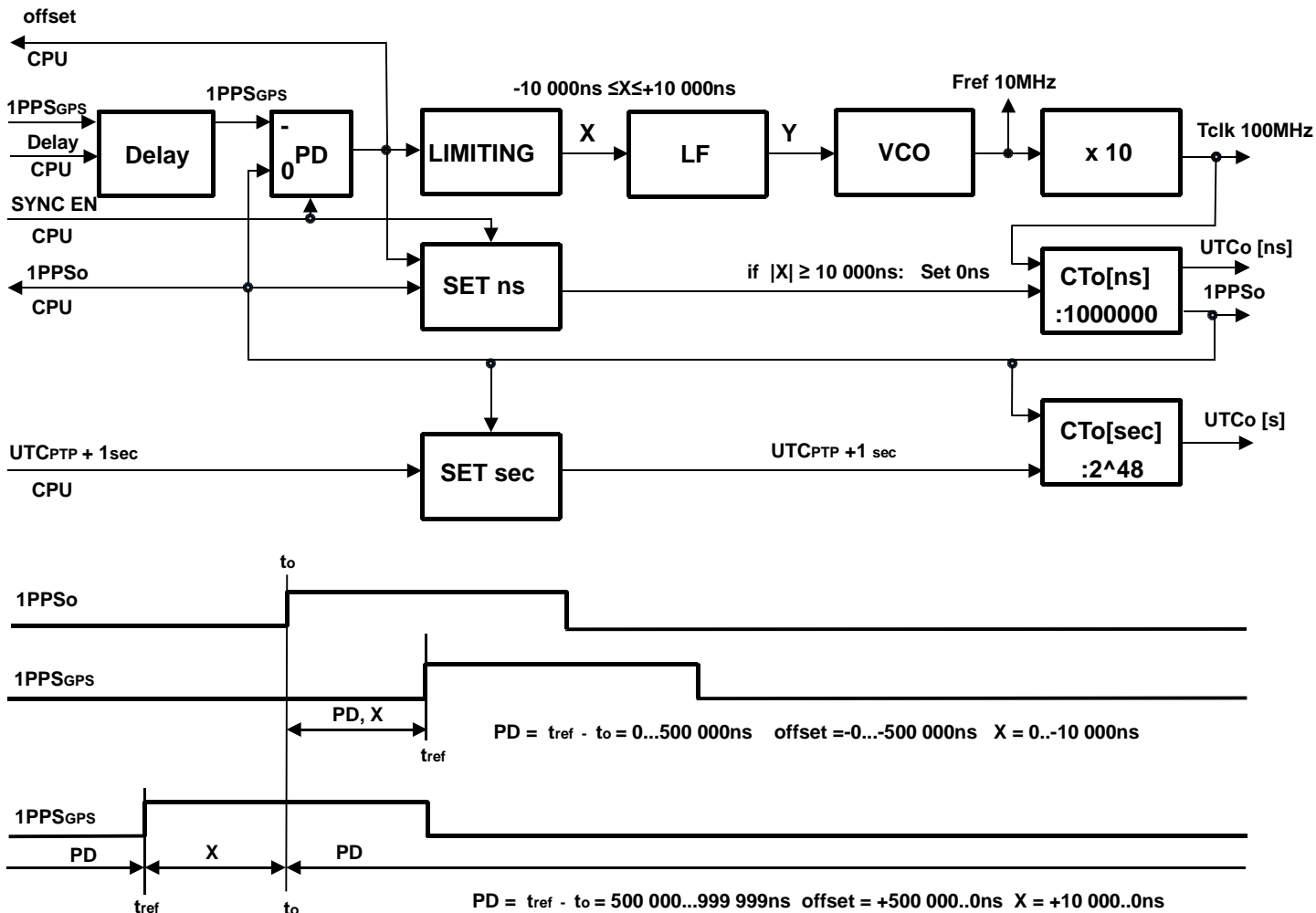


Проект M50 PTP

Встроенные часы PTP и генератор опорной частоты.

Сервера PTP Grand Master содержит встроенные PTP часы, которые должны быть синхронизированы со всемирным временем UTC (Universal Coordinated Time). Точность синхронизации должна быть не хуже ± 25 нс (требования стандарта ± 100 нс). Время PTP отличается от времени UTC. За начало шкалы берется 1 January 1970 TIA (International Atomic Time) и не учитываются добавленные секунды (leap-second). Структурная схема часов PTP приведена на рисунке.



Часы представляют собой счетчик CTo работающий от импульсов $F_{REF} 10\ MHz$ высокостабильного генератора управляемого напряжением - VCO . Для увеличения разрешающей способности часов такая частота работы счетчика увеличена до $100\ MHz$ при помощи PLL FPGA.

Разряды счетчика соответствуют формату времени PTP (timestamp): 48+32 бита. Старшие 48 представляю собой двоичный счетчик целого числа секунд, а младшие 32 бита соответствуют целому числу наносекунд (счетчик о модулю $1\ 000\ 000\ 000$ нс).

Синхронизация часов разрешается сигналом процессора $SYNC\ EN$ после захвата синхронизации от GPS (определяется процессом $ntpd$).

Синхронизация запрещается сразу же после пропадания сигнала GPS (определяется процессором по сообщениям NMEA) .

В первой версии прошивки FPGA синхронизация разрешена всегда. При этом расхождение импульса $1PPS_{So}$ и $1PPS_{GPS}$ никогда не превышает $\pm 10\ \mu s$. Проблема состоит в том, что после каждого захвата сигнала GPS взаимное положение импульса $1PPS_{So}$ и $1PPS_{GPS}$ произвольное. Процесс $ntpd$ который определяет захват синхронизации получает эту синхронизацию от импульса $1PPS_{So}$ и от сообщений NMEA которые привязаны к другому импульсу - $1PPS_{GPS}$. Процесс $ntpd$ такой разной привязки не понимает и не может захватить синхронизацию без перезапуска. При расхождении импульсов в пределах $10\ \mu s$ этой проблемы нет.

Установка счетчика наносекунд CTo[ns] в 0 в момент прихода импульса 1PPS-GPS (грубая подстройка) выполняется аппаратно каждый раз, если разность фаз больше или равна 10 000нс. Причем:

1. Разность фаз измеряется фазовым детектором PD от фронта импульса 1PPSo до фронта импульса 1PPS-GPS. Измеренное значение фиксируется по фронту следующего импульса 1PPSo. Измеренное значение разности фаз PD лежит в пределах 0...999 999 999 нс

2. Значение offset для отображения вычисляется по PD:

если $PD \leq 500\,000\,000\text{ ns}$ \rightarrow offset = - PD

если $PD > 500\,000\,000\text{ ns}$ \rightarrow offset = 1 000 000 000 - PD ns

offset = -500 000 000...0...+499 999 999нс.

3. Значение разности фаз X для управления VCO формируется из значения offset ограничителем:

если $-10\,000\text{ ns} \leq \text{offset} \leq +10\,000\text{ ns}$ \rightarrow x := offset

если $\text{offset} < -10\,000\text{ ns}$ \rightarrow x := -10 000нс

если $\text{offset} > +10\,000\text{ ns}$ \rightarrow x := +10 000нс

Ограничение устраняет выбросы сигнала разности фаз сразу после включения синхронизации, которые возможны до первой грубой подстройки.

4. Сброс счетчика CTo[ns] выполняется аппаратно по переходу из 0 в 1 импульса 1PPS-GPS, если текущее абсолютное значение разности фаз offset больше или равно 10 000нс.

Установка счетчика секунд CTo[sec] выполняется по команде процессора.

Причем:

1. Установка счетчика секунд процессором должна выполняться не ранее чем через 2сек после разрешения подстройки, когда импульсы 1PPS-GPS и 1PPS-PTP будут выравнены с ошибкой не более 10 000 нс аппаратно. грубой подстройкой.

2. Процессор, по прерыванию от 1PPSo, выдает на установочные входы время PTP + 1 сек.

3. Запись времени в счетчик CTo[sec] происходит по фронту следующего импульса 1PPSo.

4. Установка может выполняться не каждую секунду, но не реже одного раза за 64 сек .

Значение времени UTC для отображения через WEB-интерфейс считывается из счетчика CTo[sec] и, так как это время PTP, UTC пересчитывается по формуле:

$\text{NTP Seconds} = \text{PTP Seconds} + 2\,208\,988\,800 - \text{currentUtcOffset},$

где currentUtcOffset = +35 - количество leap-second вставленных за все время существования шкалы UTC.

Компенсация задержки в кабеле антенны Delay сдвигает импульс 1PPS назад. Значение Delay в пределах -2000...0 нс устанавливает CPU.

Для точной подстройки частоты генератора используется алгоритм PLL.

CPU, каждую секунду, по прерыванию от 1PPSo считывает значение PD по которому формируется значение X (текущая разность фаз) и вычисляет значение управляющего напряжения Y для VCO.

Значение Y загружается в ЦАП AD5683R по интерфейсу SPI реализованному программно.

Разрядность ЦАП - 16 бит. Сигналы SPI формируются программно выходами GPIO CPU:

SCLK - GPIO_96 (AE2) - RTS1 - UART4_RTS

SDI (MOSI) - GPIO_118 (B25). DTR0 - UART2_DTR

nSYNC (nCS) - GPIO_106 (AC6). DTR1 - DTR_1PPS_OUT_1 - FPGA - UART4 DTR

Ограничений по максимальной длительности импульсов SPI нет. Управляющее слово имеет длину 24 бита. Первым выводится старший бит.

Инициализация ЦАП. Требуется диапазон выходного напряжения 5V. При использовании внутреннего источника опорного напряжения 2.5V надо установить DB15.Gain Bit = 1 (см .AD5683R datasheet page 24, 23 Tab 9.).

Остальные управляющие биты по умолчанию 0.

Команда "Write control reg:

CCCC DDDD DDDD DDDD DDDD DDDD

3210 1111 1111 1100 0000 0000

9876 5432 1098 7654 3210

0100 0000 1000 0000 0000 0000 = 0x408000

Вывод отсчета d15..0 выполняется сразу в DAC reg, без использования буферного регистра Input reg. Используется команда .

Команда "Write DAC and Input reg:

CCCC DDDD DDDD DDDD DDDD DDDD

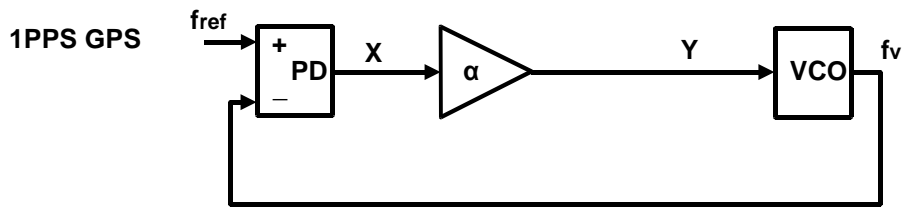
3210 1111 1111 1100 0000 0000

9876 5432 1098 7654 3210

0011 dddd dddd dddd dddd 0000 = 0x3----0
1111 1100 0000 0000
5432 1098 7654 3210

Модель PLL.

Пропорциональный регулятор (П-регулятор).



На рисунке изображен П-регулятор.

Опорная частота $f_{ref} = f_0(1 + \Delta f_{ref})$, где $f_0=1\text{Гц}$; Δf_{ref} - относительное отклонение частоты опорного сигнала от 1Гц.

Частота генератора VCO $f_{vco} = f_0(1 + \Delta f_{vco})$, Δf_{vco} - относительное отклонение частоты генератора от значения 1Гц вызванное управляющим сигналом Y.

Ошибка по фазе X определяется фазовым детектором как разность времени прихода фронта импульса опорной частоты f_{ref} и фронта импульса частоты f_{vco} от генератора VCO.

Сигнал ошибки X усиливается в α раз усилителем.

Пусть $f_{ref}=f_0$ и $f_{vco} = f_0(1 + \Delta f_{vco})$. Уход фазы импульсов генератора относительно фазы импульсов опорной частоты на величину x за время t , связан с относительным отклонением частоты генератора Δf_{vco} соотношением:

$$x[\text{сек}] = \Delta f_{vco} * t[\text{сек}] \quad (1)$$

Например, уход фазы на +1нс за 1сек соответствует отклонению частоты на $-1\text{e-}9$.

Модель нашей PLL работает дискретно. В конце каждой секунды измеряется разность фаз X, затем, в начале следующей секунды мгновенно (за время много меньшее 1 сек) сигналом Y изменяется частота генератора, и до начала новой секунды управляющее воздействие и, следовательно, частота генератора не меняются.

В генераторах VCO отклонение частоты Δf_{vco} имеет линейную зависимость от управляющего воздействия Y:

$$\Delta f_{vco} = k [1/\text{сек}] * Y[\text{сек}] \quad (2)$$

Для модели будем считать, что $k=1$.

Проверим, работу такой PLL.

Коэффициенте усиления $\alpha = 1$.

Исходное состояние: $f_{ref}=f_0$; $f_{vco} = f_0$; $X=0$ - фазы совпадают

Пусть в начале 1-ой секунды произошел скачок фазы опорного колебания на $+1\text{e-}9$ с (1 нс). Так как частоты равны, эта разность фаз будет оставаться все время постоянной.

В конце первой секунды будет зафиксировано значение $X = +1\text{e-}9$ сек и в начале следующей секунды подано управляющее воздействие:

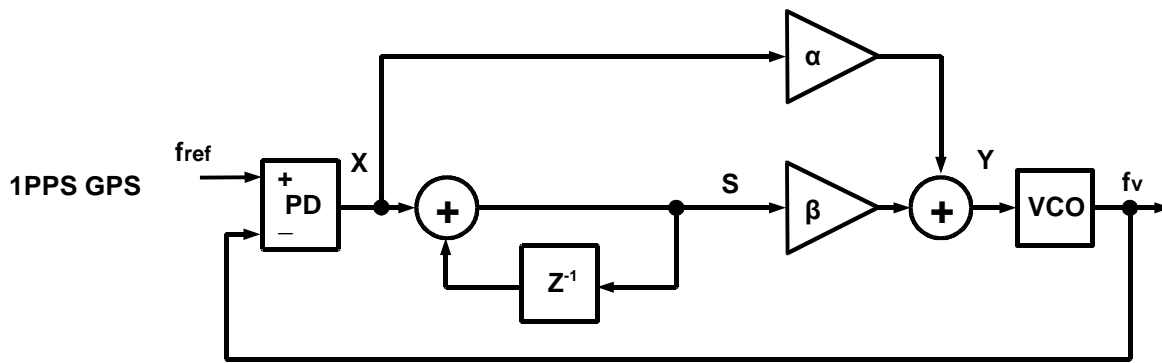
$$Y = \alpha * X = X = +1\text{e-}9 \text{ сек},$$

Частота генератора f_{vco} изменится на величину $\Delta f_{vco} = Y = +1\text{e-}9$, что к концу 2-ой секунды даст набег фазы импульсов от генератора:

$$x = \Delta f_{vco} * t = 1\text{e-}9 * 1 \text{ сек} = 1\text{e-}9 \text{ сек}.$$

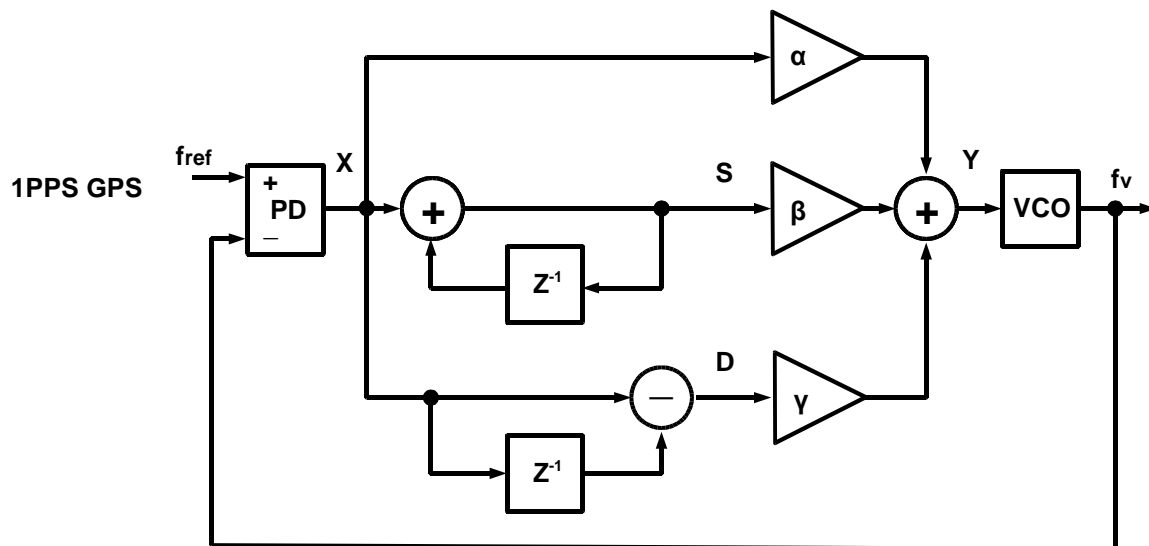
Фазы выравнялись.

ПИ-регулятор.



На рисунке изображен ПИ-регулятор. По теории обеспечивает нулевую ошибку по частоте и конечную по фазе.

ПИД-регулятор.



На рисунке изображен ПИД-регулятор. По теории обеспечивает нулевую ошибку по частоте, конечную по фазе и меньшее запаздывание.

Из данной структуры получается алгоритм вычислений:

$$\begin{aligned} s(n) &= s(n-1) + x(n) \quad \text{- интегратор} \\ d(n) &= x(n) - x(n-1) \quad \text{- дифференциатор} \\ y(n) &= \alpha * x(n) + \beta * s(n) + \gamma * d(n) \quad (3) \end{aligned}$$

или в рекуррентной форме [10]

$$y(n) = (\alpha + \beta + \gamma) * x(n) - (\alpha + 2\gamma)x(n-1) + \gamma * x(n-2) + y(n-1) \quad (4)$$

Переход от модели к реальной схеме.

Основные соотношения.

1. Если фазовый детектор выдает значение X в долях секунды x_s , для пересчета должен быть использован соответствующий коэффициент единиц измерения k_t

$$X[\text{сек}] = k_t * X[x_s], \quad \text{где } k_t = 1 / x_s$$

Например, если время измеряется в нс, $x_s = 1e-9$ сек, $k_t = 1e+9$

2. В нашей модели VCO, управляющее воздействие $Y = 1$ сек, вызывает относительное отклонение частоты $\Delta f_{vco} = 1$. Частота реального генератора управляется напряжением, которое пересчитывается:

$$Y_v[B] = k [1/\text{сек}] * Y[\text{сек}] / k_f [1/B], \quad \text{где}$$

$k = 1 [1/\text{сек}]$ - нужен только для приведения единиц измерения

$k_f [1/B]$ - коэффициентом преобразования напряжения в частоту генератора.

Для генератора АОСЧУ5 - $k_f = 2e-7 [1/B]$ (относительное отклонение частоты от $-0.5e-6 \dots +0.5e-6$ при изменении напряжения от 0 до 5В).

Кроме того, отклонение равно нулю соответствует напряжению смещения Y_o равному примерно 2.5В.

3. Для получения напряжения Y_v и смещения Y_o на выходе ЦАП на его вход надо подать целое положительное число (двоичный код) Y_{vd} и Y_{od} .

Если разрядность ЦАП равна N , а опорное напряжение равно V_{ref} , то выходное напряжение ЦАП управляющее генератором определяется:

$$Y_{dac} [B] = Y_v + Y_o [B] = V_{ref} [B] * (Y_{vd} + Y_{od}) / 2^N = k_d * (Y_{vd} + Y_{od})$$

$$k_d = V_{ref} [B] / 2^N$$

Например, для 16-битного ЦАП с опорным напряжением 5В:

$$k_d = 5/65536 = 0,0000763$$

Тогда, коды на входе ЦАП:

$$Y_{vd} = Y_v[V] / k_d$$

$$Y_{od} = Y_o[V] / k_d$$

Перерасчет от модели в схему.

Пусть на модели определена функция управления (алгоритм регулирования) по фазовой ошибке:

$$Y [\text{сек}] = F (X[\text{сек}])$$

Так как эта функция линейна, то для перерасчета результата в реальную схему все коэффициенты могут быть вынесены за пределы функции:

$$Y_{vd} = k_t / (k_f * k_d) * F (X[x_s])$$

$$Y_{od} = 1 / k_d * V_o [B]$$

Например, в нашей реализации:

$$k_t = 1e+9, \quad X [\text{нс}];$$

$$k_f = 2e-7 [1/B];$$

$$k_d = 0,0000763$$

$$V_o = 2.5B$$

$$Y_{vd} = k_t / (k_f * k_d) * F (X[x_s]) \quad Y_{vd} = 65.5 * F (X[\text{нс}])$$

$$Y_{od} = 1 / k_d * V_o [B] = 32\,765$$

Ссылки

- [1] <http://ru.wikipedia.org/wiki/ФАПЧ>
- [2] file:///Server0kb/archive/zinfo/book/schemТитце_У_Шенк_К_Полупроводниковая_схемотехника1982.djvu
- [3] <\\Server0kb\\archive\\zinfo\\pll\\base\\AnalysisAndPerformancePassivFilterDesignTechnique.pdf>
- [4] <http://videos.analog.com/video/products/rf-ics/756428873001/ADIsimPLL-Development-Software/>
- [5] <file:///Server0kb/archive/zinfo/pll/base/DesignProcedureAll-DigitalPLL.pdf>
- [6] <\\Server0kb\\archive\\zinfo\\pll\\base\\Fractional-Integer-N-PLLBasics.pdf>
- [7] \\Server0kb\\archive\\zinfo\\pll\\base\\PLL_Performance_Simulation_Design.pdf
- [8] <file:///Server0kb/archive/zinfo/pll/base/DesignPLLwhenProgN.pdf>
- [9] \\Server0kb\\archive\\zinfo\\pll\\base\\IDT_82V3391-WANPLL_PRB_20111206.pdf
- [10] Андрей.