

Зс3Н

NTF-клиент высокой точности для ОС QNX4.25 (версия 2.1, октябрь 2016 года)

Описание применения

Назначение

NTF-клиент Зс3Н предназначен для синхронизации часов локальной ЭВМ (ЭВМ-клиента) с часами NTF-сервера по каналу связи, обеспечивающему взаимодействие по протоколам семейства TCP/IP. Синхронизация выполняется в соответствии с протоколом RFC-2030 без использования механизма аутентификации. После первичной грубой синхронизации дальнейшая подстройка локальных часов выполняется посредством астатического регулятора без рывков и обратного хода. Зс3Н предназначен для работы в среде операционной системы QNX4.25/КПДА.00002 . Достижимая точность синхронизации часов ЭВМ-клиента с часами NTF-сервера составляет 30..70 микросекунд.

Требования к аппаратному окружению

Обязательным требованием к вычислительным средствам целевой системы является **запрет** использования **SMI (System Mode Interrupt)** прерываний.

Требования к программному окружению

Перед установкой **Зс3Н** на ЭВМ должны быть установлены следующие программные продукты:

- 1) **Операционная система QNX4.25/КПДА.00002;**
- 2) **Библиотека TCP/IP RT;**
- 3) **Графическая оболочка Photon RT v1.14 (требуется только для обеспечения функций наблюдения за работой NTF-клиента в локальной сессии Photon).**

Описание функционирования

NTF-клиент Зс3Н может работать в одном из двух режимов:

- 1) **Режим ординарной точности.** В этом режиме NTF-клиент синхронизирует с осью NTF-сервера непосредственно системные часы. Организация системных часов в QNX-4.25 принципиально не позволяет синхронизировать его точнее величины `ticksizе`.
- 2) **Режим высокой точности.** В этом режиме NTF-клиент синхронизирует с часами NTF-сервера не системные часы локальной ЭВМ а специально разработанные для этой цели часы **Clocker**. Организация часов **Clocker** позволяет синхронизировать их с точностью до единиц микросекунд. Поскольку многие стандартные программы для получения отсчетов времени обращаются к системным часам, то для поддержки актуальности этих часов **Clocker** выполняет постройку системных часов под часы **Clocker** с той точностью, которую позволяет

выдержать системные часы. Реально достижимая точность синхронизации часов Cloccker с NTP-сервером по каналу Ethernet 100 Mbod при использовании NTP-клиента в режиме высокой точности составляет 30..70 мкс.

Для работы в режиме высокой точности NTP-клиент Zc3H опирается на механизм таймера высокого разрешения **Htimer** и на часы высокой точности **Cloccker**. Эти компоненты входят в состав дистрибутива Zc3H.

Описание компонента Htimer приведено в приложении 1.

Описание компонента Cloccker приведено в приложении 2.

Состав дистрибутива

NTP-клиент Zc3H, как самостоятельный коммерческий продукт, поставляется в виде дистрибутива, состоящего из трех файлов:

- 1) архивный файл дистрибутива **Zc3H_licen_nomer.tar**;
- 2) текстовый файл **Zc3H_licen_nomer.crc** с указанием типа лицензии, длины архивного файла и контрольной суммы архивного файла.
- 3) файл **Zc3H.pdf** , содержащий настоящий документ.

Имена архивного файла и файла контрольной суммы содержат поля с обозначением типа лицензии и уникального номера дистрибутива:

licen -поле типа лицензии; возможные значения:

lnode -лицензия на 1 узел;

unlim -неограниченная лицензия.

nomer -поле уникального номера дистрибутива.

Размер свободного дискового пространства для выполнения установки должен составлять не менее 2.5 от размера файла-дистрибутива **Zc3H_licen_nomer.tar**;

Процедура установки

Изложенная ниже процедура установки выполняется вручную без использования самораскрывающихся архивов. В ходе установки не выполняется каких-либо скрытых модификаций системных файлов и не выполняется записи какой-либо информации за пределы указанного каталога. Все действия обратимы.

1) Выполнить запуск на ЭВМ операционной системы QNX4.25; все дальнейшие действия, если не указано иное, следует выполнять с правами "root" .

2) Выбрать место расположения и название для корня каталога NTP-клиента **Zc3H**; создать пустой корневой каталог с желаемым названием для расположения пакета; для определенности при изложении инструкции считаем, что место расположения для пакета имеет абсолютный путь /Zc3H , то есть, корень пакета должен быть расположен в каталоге **/Zc3H/** .

3) Переписать в каталог /Zc3H/ любым доступным способом файлы Zc3H_licen_nomer.tar и Zc3H_licen_nomer.crc;

4) Перейти в каталог /Zc3H :

```
# cd /Zc3H
```

5) Проверить целостность дистрибутива:

```
# cksum Zc3H_licen_nomer.tar
```

- по завершении подсчета длины файла и контрольной суммы сравнить длину файла и контрольную сумму с данными в файле Zc3H_licen_nomer.crc; если вычисленная длина и контрольная сумма совпадают с указанными в файле, то можно выполнять следующие действия.

6) Выполнить распаковку архива:

```
# tar -x -v -f Zc3H_licen_nomer.tar
```

-в результате в каталоге /Zc3H будет создано поддерево каталогов и файлов:

```
/Zc3H/
|--/Htimer/      компоненты таймера высокого разрешения
|                (описание приведено в приложении 1)
|--/Clocker/    компоненты часов высокой точности
|                (описание приведено в приложении 2)
|--Zc3H.cfg      конфигурационный файл Ntp-клиента
|--Zc3H_Start    командный файл запуска Ntp-клиента
|--Zc3H_Stop     утилита для останова работы Ntp-клиента
|--Zc3H_main.exe основной процесс Ntp-клиента
|--Zc3H_Ini.exe  утилита начальной инициализации Ntp-клиента
|--Zc3H_load.exe утилита для загрузки конф.файла Ntp-клиента
|--Zc3H.exe      подчиненный процесс Ntp-клиента
|--Zc3H_Show     командный файл запуска граф.утилиты
|                мониторинга
|--Zc3H_Show_L.exe граф.утилита мониторинга для работы
|                с видеосистемой через драйвер int10
|                (без Photon RT).
|--Zc3H_Show_L.exe граф.утилита мониторинга для работы
|                через графический эмулятор терминала
|                QweRus в среде Photon RT.
|--Zc3H_vers     утилита для запроса версии
|--Zc3H.txt      Zc3. Описание применения
|                (текстовый файл в кодировке 866)
|--Zc3H_r1.bmp   снимок экрана утилиты Zc3H_Show
|                (рисунок 1 к тексту Zc3H.txt)
|--Zc3H_r2.bmp   снимок экрана утилиты Zc3_Show
|                (рисунок 2 к тексту Zc3H.txt)
|
|--Zc3H.pdf      Zc3H. Описание применения.
|
-/-Dht/         демонстрационная программа для иллюстрации
|                применения таймера высокого разрешения Htimer
|                (исходный текст программы и командный файл для
|                компиляции и построения).
-/-DClock/      демонстрационная программа для иллюстрации
|                применения часов Slocker Htimer
|                (исходный текст программы и командный файл для
|                компиляции и построения).
-/-Zc3/         дистрибутив ранней версии NTP-клиента для совместимости
|                с приложениями, разработанными с использованием
|                инструментального пакета Maestro 589.63074985.00001-01
|                (годы выпуска 2010..2014).
-/-QweRus/
|-QweRus        эмулятор терминала QweRus
|-pcterm14.phf шрифт для терминала
|-QweRus2_2.txt руководство по установке QweRus
|                (текст в кодировке 866)
|-QweRus2_2.rtf руководство по установке QweRus
```

7) Выполнить, если это необходимо, корректировку параметров режима работы NTP-клиента в файле Zc3N.cfg. Кодировка символов в файле соответствует кодовой странице 866 (русская кодировка DOS). Каждый из параметров снабжен комментариями (комментарий предваряется символом "#"). Исходное содержание файла Zc3N.cfg приведено ниже:

```
#----- файл Zc3N.cfg -----
#
#   исходная конфигурация NTP-клиента
#-----
PORT_zNTP 123 "номер UDP-порта назначения запросов к серверу"
PORT_oNTP 6000 "номер UDP-порта для приема ответов от сервера"
#-----
SERV_NTP "ntp_server" # ip-имя NTP-сервера
KOM_SYNC "rtc -s at" # команда запомнить время в CMOS-таймере
# выполняется каждые 20 секунд если
# синхронизация в допуске
#-----
# Если сервер времени точный и доступ быстрый:
#
dTtxrxLim 0.6 # макс. допустимое время ответа [ms]
# (более поздние бракуются)
TsyUTC 1000 # период запросов [ms]
dUTCmin 0.001 # зона нечувствительности [ms]
dUTClim 0.5 # предел для объявления срыва синхронизации [ms]
dUTCmax 5.0 # предел для грубой синхронизации [ms]
#-----
# Если сервер времени не точный и доступ медленный:
#
#dTtxrxLim 5.0 # макс. допустимое время ответа
#TsyUTC 10000 # период запросов
#dUTCmin 10.0 # зона нечувствительности
#dUTClim 15.0 # предел для объявления срыва синхронизации
#dUTCmax 25.0 # предел для грубой синхронизации [ms]
#-----
```

8) Если предполагается наблюдение за работой NTP-клиента посредством сессии Photon RT (в окне эмулятора терминала), то следует выполнить процедуру установки эмулятора терминала **QweRus** в соответствии с документом QweRus2_2.rtf (либо QweRus2_2.txt).

9) Определить ip-имя NTP-сервера и собственное имя NTP-клиента в файле /etc/hosts , например:

```
#-----
# пример содержимого файла /etc/hosts :
.....
192.168.0.35 ntp_client # собственный ip-адрес ЭВМ клиента
192.168.0.71 ntp_server # ip-адрес доступного NTP-сервера
.....
#-----
```

10) В sysinit.xxx следует добавить последовательность команд для установки нулевого смещения локального времени относительно UTC, ticksize 1, запуска драйвера int10, запуска сетевой подсистемы Net, конкретного драйвера сетевой карты, последовательность команд запуска tcp/ip и запуска NTP-клиента, например:

```

#----фрагмент файла sysinit.kxx-----
#
.....
EXPORT TZ=UTC+0
.....
Net &          # запуск менеджера сети
Net.ether1000 & # запуск драйвера сетевой карты NE2000
#
# запуск tcp/ip :
#
Socket kwa &
ifconfig en1 ntp_client up
#-----
ticksize 1      # установить ticksize = 1 мс
int10 &         # запуск графического драйвера; требуется для работы
                # утилиты мониторинга на консоли без Photon RT;
                # наличие/отсутствие драйвера на на работу
                # собственно NTP-клиента не влияет.
#-----
# Запуск NTP-клиента в режиме высокой точности:
#
cd /Zc3H/Htimer
./Htimer_Start  # запуск таймера высокого разрешения
sleep 2
cd /Zc3H/Clocker
./Cllocker_Start # запуск точных часов
sleep 2
cd /Zc3H
./Zc3H_Start    # запуск NTP-клиента Zc3H
cd /
.....
#-----

```

11) После рестарта системы NTP-клиент Zc3H будет с указанным в конфигурационном файле периодом выполнять попытки синхронизировать часы ЭВМ-клиента с часами NTP-сервера.

Наблюдение за работой NTP-клиента

Наблюдение за работой NTP-клиента возможно с помощью графических утилит Zc3H_Show тремя разными способами:

- 1) На локальной консоли ЭВМ-клиента в графическом режиме без использования Photon RT; обязательным условием является совместимость драйвера int10 с видеосистемой ЭВМ-клиента;
- 2) В локальной сессии графической оболочки Photon RT в окне эмулятора терминала QweRus; для обеспечения такой возможности на ЭВМ-клиенте должен быть установлен Photon RT v1.14 и эмулятор терминала QweRus (дистрибутив QweRus и документация к нему находятся в каталоге Zc3H/QweRus);
- 3) Удаленно с помощью ЭВМ оператора, имеющей контакт с ЭВМ-клиентом по сети (Telnet-сессия); на ЭВМ оператора должна быть установлена операционная система QNX4.25, библиотека TCP/IP RT, графическая оболочка Photon RT и эмулятор терминала QweRus.

Zc3H_Show это командный файл, который автоматически, в зависимости от типа сессии, запускает одну из двух утилит:

-если Zc3H_Show вызван с локальной текстовой консоли, то фактически выполняется запуск утилиты Zc3H_Show_L.exe; эта утилита

переводит консоль в графический режим 640x480 16 цветов и отображает видеоформу с информацией о работе NTP-клиента; по завершении работы утилиты (F10) консоль возвращается в текстовый режим.

-если Zc3N_Show вызван из окна QweRus с локальной либо удаленной ЭВМ, то фактически выполняется запуск утилиты Zc3N_Show_R.exe; эта утилита выполняет передачу на терминал команд векторной графики, эмулятор терминала переходит в графический режим 640x480 16 цветов и отображает видеоформу с информацией о работе NTP-клиента; по завершении работы утилиты (F10) эмулятор терминала возвращается в текстовый режим.

Вид экрана/терминального окна Zc3N_Show при работе Zc3N в режиме 1 (ординарная точность) приведен на рисунке 1 (файл Zc3N_r1.bmp). Вид экрана/терминального окна Zc3N_Show при работе Zc3N в режиме 2 (высокая точность) приведен на рисунке 2 (файл Zc3N_r2.bmp).

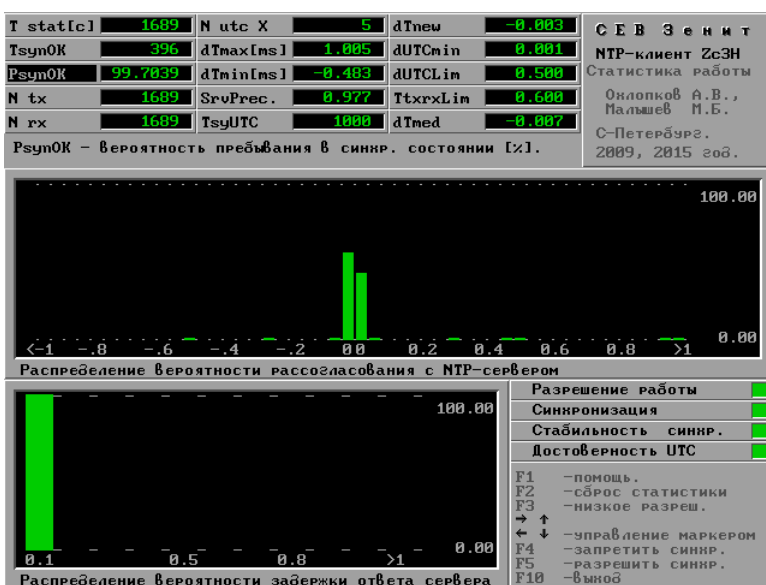


Рис.1. Работа NTP-клиента Zc3N в режиме ординарной точности.

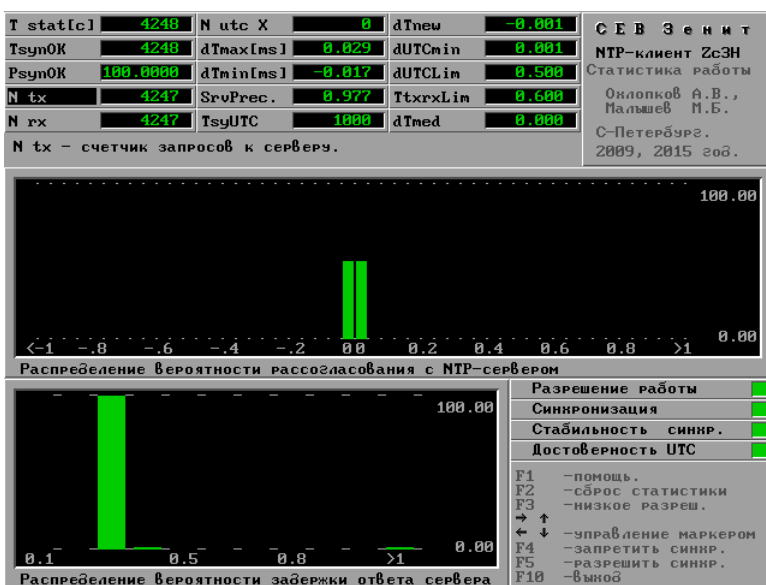


Рис.2. Работа NTP-клиента Zc3 в режиме высокой точности.

Условия и порядок распространения программы Zc3H.

Условия и порядок распространения программы Zc3H приведены в приложении А к настоящему документу. Права конечного пользователя зависят от типа лицензии на программу. Предусмотрено два типа лицензий:

- 1) Лицензия на один узел "1 node license"**
- 2) Неограниченная лицензия "unlimited license"**

Фактический тип лицензии указан в поле `licen` имени архивного файла дистрибутива и файла контрольной суммы.

Если программа Zc3H уже работает, то для определения версии и типа лицензии следует вызвать утилиту `Zc3H_vers`, в ответ будет выведены строки с информацией о версии и типе лицензии.

Дополнительная информация о программе

доступна на сайте www.polsta.com

Htimer

Таймер высокого разрешения.

Инструментальный комплект Htimer это набор исполняемых модулей и библиотек, предназначенный для организации таймера высокого разрешения, доступного для всех процессов (потоков) в пределах ЭВМ.

Процесс Htimer

Процесс Htimer предназначен для организации и поддержания программно-аппаратной оси времени с разрешением 1 микросекунда. В зависимости от параметров запуска Htimer будет опираться на счетчик тиков процессора либо на внешний аппаратный счетчик в составе модуля micro-PC 5300 (Octagon System). Htimer обеспечивает возможность доступа к таймеру для всех процессов, работающих на этой же ЭВМ. Доступ к таймеру обеспечивается средствами библиотеки **Htim.o**. Прототипы функций для доступа определены в файле **Htim.h**.

Htimer_Start -командный файл запуска Htimer
Htimer_Stop -командный файл корректного останова Htimer

Ограничения механизма Htimer

Перед запуском Htimer и в процессе дальнейшей работы период диспетчеризации `ticksize` должен быть установлен равным 1 мс (999847 наносекунд).

Механизм Htimer корректно работает на одноядерных процессорах при отсутствии длительных интервалов работы в режиме SMM (System Monitoring Mode). Режим SMM возможен на некоторых вычислительных системах при обработке SMI-прерываний.

ВНИМАНИЕ! Вычислительный модуль `src150` (Fastwel) при запрещенных SMI-прерываниях имеет нестандартную частоту внутреннего генератора. При задании `ticksize` 999847 наносекунд фактический (физический) период прерываний `irq0` будет равняться 999000 наносекунд. Эта особенность будет учитываться в Htimer при условии запуска с ключем `src150` (см.комментарии в командном файле `Htimer_Start`).

Для обеспечения возможности работы всех проектов с таймером высокого разрешения рекомендуется выполнить запуск Htimer единожды при старте системы (в файле `sysinit`). Пример использования механизма Htimer в прикладных задачах приведен в демонстрационном проекте **DHt**.

Clocker

Инструментальный комплект для организации точных часов UTC

Clocker это набор утилит и объектная библиотека для организации точных часов UTC с разрешением 1 микросекунда. Часы Clocker являются альтернативой системным часам.

Состав:

- Clocker.exe** процесс поддержания оси времени UTC с опорой на таймер высокого разрешения **Htimer**. Clocker.exe должен запускаться после запуска таймера высокого разрешения.
- Clocker_StaRt** командный файл запуска Clocker.exe с комментариями
- Clocker_Stop** утилита для корректного завершения Clocker.exe
- Clocker_Stat_L.exe** графическая утилита для наблюдения динамики рассогласования системных часов и часов Clocker; вариант утилиты для работы на собственной консоли через драйвер int10.
- Clocker_Stat_R.exe** графическая утилита для наблюдения динамики рассогласования системных часов и часов Clocker; вариант утилиты для работы на графическом эмуляторе терминала (QweRus/QWinT) в локальной сессии Photon либо удаленно (по протоколу TELNET);
- Clocker_Stat** командный файл запуска графической утилиты Clocker_Stat_L.exe / Clocker_Stat_R.exe в зависимости от фактического типа терминальной сессии;
- cdate.exe** утилита для установки времени альтернативных часов Clocker - упрощенный аналог утилиты date; используется вместо утилиты date в случае, когда часы Clocker являются основными, а системные часы подчиненными.
- Set_UTC_ok.exe** отладочная утилита для принудительной установки/сброса бита достоверности UTC.
- Clock.o, common.o** библиотека функций для доступа к часам Clocker из прикладных задач; определения и прототипы функций приведены в файле **Clock.h**.

Описание механизма Clocker

Системный механизм времени QNX 4.25 , к сожалению, обладает двумя существенными недостатками:

- 1) Вызов функции `clock_gettime(...)` из обработчика прерывания запрещен;
- 2) Результат двух вызовов `clock_gettime(...)`, выполненных сколь угодно близко по времени могут отличаться на величину `ticksizes`. То есть, при величине `ticksizes 1 ms` привязка события к астрономическому времени в общем случае может давать ошибку до 1000 микросекунд.

Для компенсации этих недостатков разработан механизм поддержания оси астрономического времени Clocker.

Процесс Clocker.exe

При запуске Clocker.exe следует задать один параметр - период подстройки оси времени в миллисекундах. Сразу после запуска процесс Clocker.exe подключается к общей области Ntim и получает доступ к счетчику микросекунд. После этого создается общая область памяти под названием Clock15, туда записывается текущее системное время (секунды и наносекунды от 1 января 1970 года), параметры динамической подстройки часов, слово состояния системы единого времени (СЕВ), резервируется память для накопления статистики и память для динамических параметров экстраполяции. С этого момента, наряду с системной осью астрономического времени, будет существовать альтернативная системной ось астрономического времени Clocker. После этого процесс Clocker.exe начинает выполнять циклическую подстройку оси времени в режиме, который задан параметром при запуске Clocker.exe .

Предусмотрены 3 режима подстройки:

- 1) Отсутствие подстройки. В этом случае ось времени Clocker постепенно уходит от системной оси из-за ошибки измерения скорости процессора и ошибок округления при расчетах.
- 2) Подстройка оси Clocker под системную ось. В этом случае в качестве главной (истинной) оси времени принимается системная ось времени. Ось времени Clocker плавно подстраивается под системную ось.
- 3) Подстройка системной оси времени под ось Clocker. В этом случае в качестве главной (истинной) оси времени принимается ось времени Clocker. Системная ось времени плавно подстраивается под ось времени Clocker.

Режимы 2 и 3 позволяют разным процессам пользоваться как классическими обращениями к системным часам так и функциями доступа к часам Clocker без существенного расхождения результатов. Рекомендуемые значения периода подстройки от 200 до 500 миллисекунд, что не слишком загружает процессор и обеспечивает

приемлемую точность подстройки. Если период подстройки задан =0, то подстройка не выполняется. Если значение периода положительное, то истинной осью времени считается ось времени Clocker, системная ось подстраивается. Если значение периода отрицательное, то истинной осью времени считается системная ось, ось времени Clocker подстраивается под системную ось.

Утилита `cdate.exe`

Утилита `cdate.exe` является аналогом системной утилиты `date` применительно к механизму Clocker.

После запуска `Clocker.exe` в режиме 2 установка системного времени с помощью функции `clock_settime(...)` приводит к кратковременному скачку системного времени с последующим возвратом к оси Clocker; утилита `date` для установки времени не пригодна, следует использовать утилиту `cdate.exe`. Командная строка установки времени:

`cdate.exe` - display or set the date and time for Clocker

`cdate.exe` [<date>]

The following format is used for a <date> when setting the date/time:

```
[[[CC]YY]MM]DD]hhmm[.SS]
```

Пример (показать текущую дату и время):

```
$ cdate.exe
```

```
Wed May 25 13:43:24 2016
```

Пример (установить дату и время 25 мая 2016 года 21:10:33 UTC):

```
# cdate.exe 201605252110.33
```

Утилита `Set.UTC.ok.exe`

При отладке распределенной системы единого времени в отсутствие источника достоверного UTC можно "обмануть" СЕВ, установив на сервере времени бит достоверности UTC принудительно:

Пример (установить признак UTC достоверно):

```
# Set.UTC.ok.exe 1
```

Пример (снять признак достоверности UTC)

```
# Set.UTC.ok.exe 0
```

Утилита Clocker_Stat

Графическая утилита Clocker_Stat предназначена для наблюдения за поведением системной оси времени/оси Clocker при ее подстройке. Если запущен процесс Clocker.exe, то можно воспользоваться утилитой Clocker_Stat для наблюдения за отклонением системной оси времени от оси Clocker. Утилита может работать на собственной консоли компьютера (локальный режим работы) при условии поддержки графики через механизм int10. Утилита может работать в окне псевдо-терминала QweRus (в среде Photon-Rt) как локально, так и удаленно (через Telnet-сессию). Внешний вид экрана утилиты Clocker_Stat показан на рисунке 1.

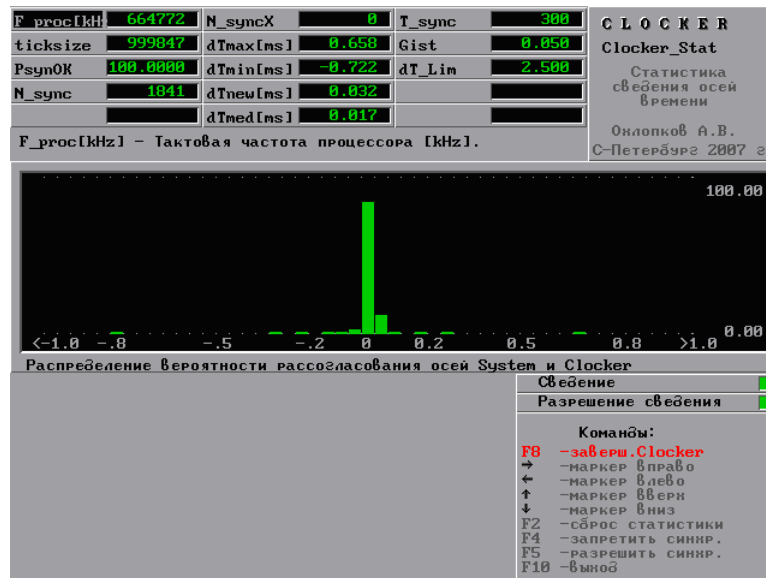


Рис.1. Вид экрана графической утилиты Clocker_Stat

Библиотека Clock.o

Библиотека **Clock.o** содержит функции, аналогичные системным, которые служат для доступа к часам Clocker. Прототипы функций находятся в файле **Clock.h**. Функции библиотеки Clock аналогичны системным и позволяют получить время часов Clocker, установить время часов Clocker, замедлить/ускорить ход часов Clocker. В отличие от стандартных системных функций, функции библиотеки Clock.o требуют системных привилегий для выполнения, могут быть вызваны из обработчика прерывания и позволяют управлять битами слова состояния системы единого времени (СЕВ); то есть, библиотека Clock.o позволяет создать процесс для управления ходом часов Clocker и синхронизации часов Clocker с некоторым источником сигналов точного времени. Таким источником может быть спутниковый приемник ГЛОНАСС/NAVSTAR либо внешний сервер точного времени, доступный по протоколу NTP. Такому процессу, как правило, вменяется обязанность управлять битами слова состояния СЕВ, размещенными в общей области памяти Clock15. Биты состояния СЕВ доступны прикладным задачам на чтение, что позволяет кроме отсчета времени UTC получить сведения о достоверности этого времени и точности синхронизации. Пример применения библиотеки Clock приведен в демо-проекте **DClock**.

Ограничения механизма Clocker

Применение механизма Clocker обусловлено рядом ограничений:

1) Для нормальной работы Clocker.exe требуется предварительный запуск таймера высокого разрешения Htimer .

2) После запуска Htimer и Clocker.exe не следует изменять величину ticksize во избежание срыва поддержки оси времени;

3) После запуска Clocker.exe в режиме 2 установка системного времени с помощью функции clock_gettime(...) приводит к кратковременному скачку системного времени с последующим возвратом к оси Clocker; утилита date для установки времени не пригодна, следует использовать утилиту cdate.exe .

4) Механизм Clocker подразумевает ведение часов UTC, переменная окружения, определяющая смещение часового пояса должна быть установлена TZ=0 , часы при настройке SETUP вычислительного модуля должны быть установлены на время UTC . Преобразование времени UTC в локальное время, если таковое необходимо, должно выполняться на прикладном уровне.

**Условия и порядок распространения программы для ЭВМ
"Zc3H. NTP-клиент высокой точности для ОС QNX4.25"**

Программа Zc3H является коммерческим продуктом. Программа Zc3H является одним из компонентов программы для ЭВМ "Система единого времени Зенит" Все права на Программу для ЭВМ "Система единого времени Зенит" защищены Законом Российской Федерации (свидетельство об официальной регистрации программ для ЭВМ № 2008613388 Федеральной службы по интеллектуальной собственности, патентам и товарным знакам) .

Автором программы Zc3H является Охлопков Андрей Витальевич.

Поставка программы Zc3H лицу-покупателю (далее Покупатель) осуществляется на основании прејскуранта на продукты и услуги Поставщика. Порядок и форма оплаты определяются Поставщиком.

Программа Zc3H позиционируется как средство для высокоточной синхронизации вычислительных средств, функционирующих под управлением операционной системы ОС QNX 4.25 (КПДА.00002), с сервером времени по протоколу NTP(Net Time Protocol).

Лицензионное соглашение

1. Покупатель получает неисключительное право на использование программы.

2. Покупатель берет на себя обязательства по защите приобретенных экземпляров программы от незаконного копирования и использования.

3. Ограничения на использование программы Zc3H Покупателем и границы ответственности Поставщика и Автора:

3.1. Автор через Поставщика передает Покупателю программу Zc3H в состоянии "как есть".

3.2. Автор и Поставщик не несут ответственности за ущерб, нанесенный кому-либо вследствие нерасчетной работы программы.

3.3. Покупатель вправе использовать программу с лицензией на 1 узел (1 node license) в составе одной и только одной ЭВМ.

3.4. Покупатель, который приобрел программу Zc3H с неограниченной лицензией (unlimited license), вправе использовать программу без ограничения количества копий.

3.5. Покупатель вправе передавать третьим лицам и организациям программу Zc3H в составе своего вновь разработанного программного обеспечения в порядке, соответствующем типу лицензии.

3.6. Неограниченная лицензия не дает право Покупателю тиражировать программу для передачи третьим лицам кроме случаев передачи в составе программного обеспечения собственной разработки.

3.7. В случае намерения Покупателя применить программу в системе с высоким уровнем ответственности Автор может предоставить исходные тексты программы Покупателю для выполнения процедуры верификации/сертификации. Условия такой передачи должны регулироваться отдельным договором.

3.8. Фактическое использование Покупателем программы Zc3H расценивается как согласие с условиями настоящего документа.