

УДК 004.382

**С.А. Полищев**

Институт проблем искусственного интеллекта МОН Украины и НАН Украины,  
г. Донецк, Украина  
Украина, 83048, г. Донецк, ул. Артема, 118-б

## Адаптивная синхронизация в беспроводной сети

**S.A. Polivtsev**

*Institute of Artificial Intelligence MES of Ukraine and NAS of Ukraine, c. Donetsk, Ukraine  
Ukraine, 83048, c. Donetsk, Artema st., 118-b*

## *Adaptive Synchronisation in a Wireless Network*

**С.О. Поліщев**

Институт проблем штучного інтелекту МОН України і НАН України,  
м. Донецьк, Україна  
Україна, 83048, м. Донецьк, вул. Артема 118-б

## Адаптивна синхронізація у бездротовій мережі

В статье рассматриваются теоретические и практические вопросы синхронизации узлов беспроводной сети ZigBee диапазона 2,4 ГГц. Показано, что на неспециализированном микроконтроллере типа STM32F100RC можно достичь точности синхронизации  $\pm 0,5$  мС. Приводятся основные соотношения, позволяющие достичь заданной точности.

**Ключевые слова:** ZigBee, CC2520, беспроводная сеть, STM32F100RC, синхронизация.

In article theoretical and practical questions of synchronisation of knots of wireless network ZigBee of a range of 2,4 GHz are considered. It is shown that on the unspecialized microcontroller of type STM32F10x it is possible to reach accuracy of synchronisation  $\pm 0,5$  a mS. The main relations are resulted, allowing to reach the set accuracy.

**Key words:** ZigBee, CC2520, wireless network, STM32F100RC, synchronisation.

У статті розглядаються теоретичні і практичні питання синхронізації вузлів безпроводної мережі ZigBee діапазону 2,4 ГГц. Показано, що на неспеціалізованому мікроконтролері типу STM32F100RC можна досягти точності синхронізації  $\pm 0,5$  мС. Наводяться основні співвідношення, що дозволяють досягти заданої точності.

**Ключові слова:** ZigBee, CC2520, бездротова мережа, STM32F100RC, синхронізація.

## Введение

В последние годы заметно возросло внимание к такой области исследований, как сенсорные сети [1]. Сенсорная сеть состоит из большого числа сенсорных узлов, обычно оборудованных микроконтроллером, маломощным приёмопередатчиком, источником энергии, интерфейсом для связи с датчиками и приводами. Большой интерес к изучению сенсорных сетей обусловлен широкими возможностями их применения. Разнообразие доступных датчиков и исполнительных механизмов, простота установки и сравнительно низкая стоимость делают возможным применение таких сетей во многих отраслях жизни. Сенсорные сети используются для задач мониторинга и управления различными пространственно-распределёнными объектами. Области их применения включают мониторинг окружающей среды, сельское хозяйство, производство, медицину и т.п. В частности, есть задача обмена небольшим потоком информации ( $< 2$  Кбайт/сек) нескольких подвижных объектов (людей) и одним неподвижным модулем. Можно счи-

тать, что это классическая задача для системы типа ZigBee (IEEE 802.15.4) [2] в диапазоне 2,4 ГГц с топологией «звезда», в которой имеется в точности один узел – координатор и множество периферийных узлов.

Мощность передатчика на любом объекте ограничена до 5 мВт. Расстояние между объектами и координатором находится в пределах 2 – 20 м в помещении или вне его. Подвижные объекты имеют батарейные источники питания и одна из задач обеспечения работоспособности системы состоит в экономном расходовании заряда батареи на каждом объекте. С этой задачей непосредственно связана задача обмена данными со всеми подвижными объектами с заданным интервалом времени. Из этих двух задач следует, что система в целом должна иметь службу синхронизации во времени. Причем эта служба должна быть распределена между всеми узлами сети.

Поскольку не предполагается доступа к системе GPS, то абсолютное значение времени в каждом узле будет иметь свое значение, более того, уход времени в каждом узле будет иметь свое значение, зависящее от многих факторов. Из вышеизложенного следует актуальная задача – построить с заданной точностью службу синхронизации времени для ZigBee сети на относительно нестабильных опорных генераторах меток времени. В качестве приемопередатчиков узлов сети используются трансиверы типа CC2520 [3] (фирма TI, США), работающие в паре с 32 битным ARM микроконтроллером (МК), типа STM32F100RC [4] (фирма ST, США).

## Требования синхронизации в IEEE 802.15.4

Временное разделение ZigBee базируется на использовании режима синхронизации, при котором подчиненные сетевые узлы большую часть времени находятся в «спящем» состоянии, периодически «просыпаясь» для обмена данными с координатором или синхронизации с координатором. Координатор управляет обменом, выделяет частотные каналы и синхронизирует обмен с периодом от 15 мС до 252 С. Обмен данными определяет пропускную способность, время ожидания очереди доступа и размер временного интервала, на котором исключены коллизии в сети. Согласно стандарту, временной интервал доступа для каждого узла сети определяется либо координатором, либо посредством механизма CSMA-CA. Интервалы покоя (см. рис. 1) отображают либо реализацию энергосберегающего режима узла – координатора, либо время, отвещенное для не подключенных узлов.

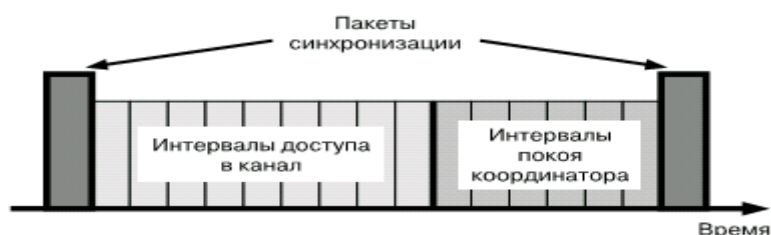


Рисунок 1 – Синхронизация доступа в сети ZigBee

Пакеты синхронизации должны следовать с определенными интервалами.

Синхронизированный доступ позволяет значительно упростить программную реализацию узла-координатора благодаря отсутствию случайных пересылок от оконечных узлов. Реализация беспроводной сети возможна и без использования стека ZigBee (под стеком понимается иерархический набор программ). Любой стек может использовать уровни MAC и PHY стандарта IEEE 802.15.4. На время написания статьи все производители сетевого оборудования предлагают и свой стек ZigBee, в виде объектных модулей. Стеки различных фирм не совместимы между собой, более того, их про-

токолы не раскрыты. Исключение составляет стек SimpliTI от фирмы TI, но он носит исключительно ознакомительный характер, позволяющий работать только двум устройствам. В этом стеке никак не затронуты вопросы синхронизации узлов. Возможно, со временем, появится открытый протокол синхронизации, но пока реалии жизни требуют создания собственного протокола, хотя бы в общих чертах отвечающего стандарту IEEE 802.15.4.

## Основные проблемы синхронизации

Фирма – производитель трансиверов типа CC2520, в [5] обосновала необходимость применения режима экономии электропитания для этих трансиверов. Из анализа потоков обмена следует, что каждому конечному узлу необходимо 7 мс времени на собственно прием – передачу, 3 мс – на переход трансивера из спящего режима в рабочий. Разумным будет отвести еще 2 мс на упреждение перехода трансивера в рабочий режим перед началом передачи узлом-координатором. Таким образом, общее время нахождения трансивера в рабочем состоянии будет равно 12 мс. Пусть каждый периферийный узел вызывается 1 раз в секунду. Тогда при выходной мощности передатчика 0 dBm, ток потребления составит 25 мА при 25°C (ток зависит от температуры). Двух батареек типа AA будет достаточно для питания в течение 2400 часов или 100 суток. Из этого вытекает, что часы системы должны быть синхронизированы с точностью не хуже, чем  $\pm 0,5$  мс.

Собственно трансивер типа CC2520 не обладает никакими ресурсами для реализации часов реального времени и, тем более, для их синхронизации с другими устройствами. Эта работа целиком ложится на управляющий МК.

Из спецификаций МК типа STM32F100RC известно, что он содержит часы реального времени (RTC), идущие от внешнего кварца частотой 32768 Гц, нескольких таймеров, тактируемых от системной шины, таймер типа WWDG («сторожевая собака»), тактируемый от системной шины, таймер типа IWDG («независимая сторожевая собака»), тактируемый от отдельного RC-генератора. Для данной задачи наиболее интересным представляется таймер IWDG. Из рис. 2 ниже (взят из [4]) видно, что таймер IWDG переведет МК в состояние системного сброса из любого текущего состояния. Это означает, что МК можно поставить в режим пониженного энергопотребления типа «Sleep mode», запустить IWDG. По прошествии заданного времени МК «проснется» и будет готов к работе.

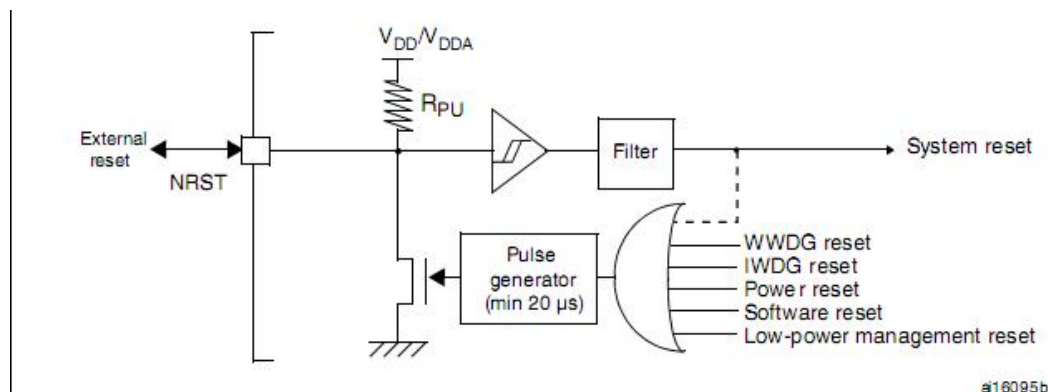


Рисунок 2 – Схема сброса МК типа STM32F100RC

Далее будут рассмотрены вопросы точности работы RTC, привязка времени срабатывания IWDG к RTC и способ коррекции связки IWDG и RTC как элементасинхронизация времени в узлах беспроводной сети.

Часы RTC работают от внешнего кварца, частотой 32768 Гц, имеющие низкую стабильность  $\pm 36$  ppm (1 ppm = 0,0001%) и энергопотребление около 1,4 мкА. На рис. 3 ниже показана типовая схема RTC (рис. 3 взят из [4]).

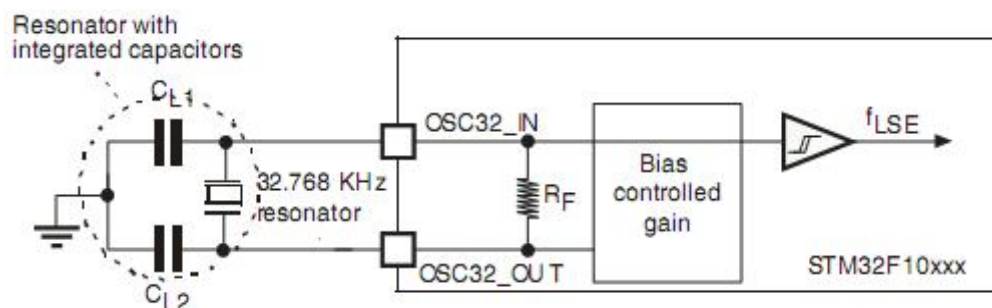


Рисунок 3 – Структурная схема генератора RTC

Структура RTC содержит механизм, позволяющий откалибровать частоту вручную или автоматически, с помощью внешнего эталонного генератора. Однако это не означает, что может быть сдвинута фазировка внешнего резонатора. Практически это означает, что все RTC всех узлов сети могут идти с высокой абсолютной точностью в пределах секундного интервала (т.е. их «уход» за сутки может быть минимальным), но фронты секундных импульсов будут начинаться в различное время (рис. 4).

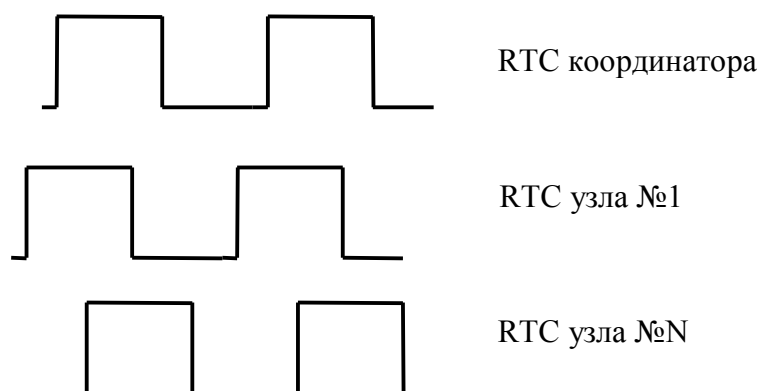


Рисунок 4 – Взаимная рассинхронизация генераторов RTC

Таймер типа IWDG («независимая сторожевая собака») тактируется от отдельного RC генератора, характеристики которого показаны в табл. 1 (взята из [4]).

Таблица 1 – Характеристики LSI генератора

Symbol	Parameter	Min	Typ	Max	Unit
$f_{LSI}$	Frequency	30	40	60	kHz
$\Delta f_{LSI(T)}$	Temperature-related frequency drift <sup>(2)</sup>	-9		9	%
$t_{su(LSI)}^{(3)}$	LSI oscillator startup time			85	$\mu$ s
$I_{DD(LSI)}^{(3)}$	LSI oscillator power consumption		0.65	1.2	$\mu$ A

Замечания 2 и 3 говорят о том, что эти данные не тестируются, а гарантируются конструкцией. Также оговаривается, что частота генератора зависит от напряжения

питания, но практически не зависит от температуры в диапазоне от  $-40$  до  $105$  °C. Частота генератора 30..60 кГц эквивалентна периодам времени 33..16 мкс. Блок-схема таймера IWDG показана на рис. 5 ниже.

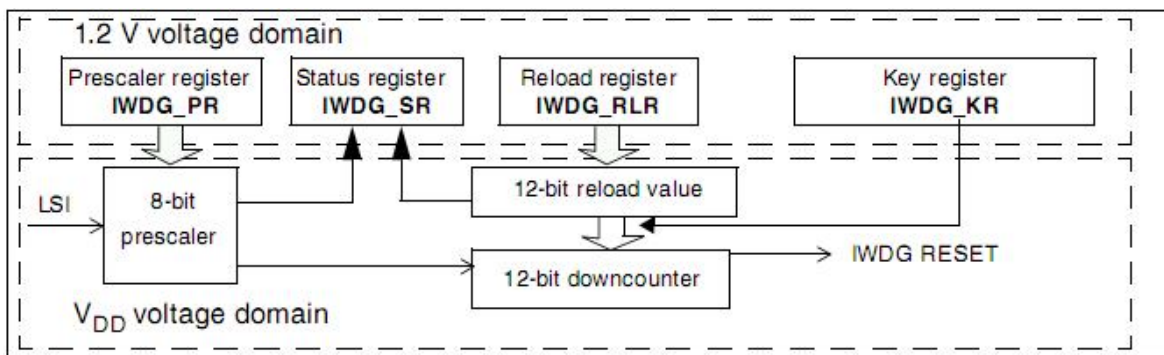


Рисунок 5 – Блок-схема таймера IWDG

Функция IWDG реализуется в области напряжения VDD, которое доступно также в режимах Stop и Standby.

Очевидно, что достаточно установить точное текущее значение LSI генератора, от которого тактируется таймер IWDG, чтобы определить с высокой точностью время «просыпания» узла  $N_x$  устройства совместно с работой RTC. Для этого необходимо знать сколько импульсов успевают отсчитать таймер IWDG за 1 Сек. Но здесь для МК семейства STM32F10x имеется ряд препятствий. В семействе нет доступа к «12 bit down-counter» (см. рис. 5), входящего в состав таймера IWDG. Это означает, что по прерываниям от секундных импульсов невозможно установить, сколько насчитал таймер IWDG. В семействе просто нет регистра, который позволяет определить, сколько тактов частоты 32768 кГц укладывается между двумя прерываниями от таймера IWDG. Поэтому задачу определения частоты LSI генератора приходится решать итерационным путем, суть которого сводится к следующему.

Обозначим через  $F_{LSE}$  частоту генератора для RTC, через  $F_{LSI}$  – частоту генератора для IWDG, через  $RTC\_PRL$  – период, загружаемый в таймер RTC для отсчета 1 Сек, через  $IWDG\_RLR$  – период, загружаемый в таймер IWDG. Если бы выполнялось равенство  $F_{LSE} = F_{LSI}$ , то для отсчета 1 Сек в  $RTC\_PRL$  и  $IWDG\_RLR$  можно было загружать одинаковые значения или иначе выполнялось бы соотношение:

$$F_{LSI} = F_{LSE} * \frac{RTC\_PRL}{IWDG\_RLR} \quad (1)$$

Из (1) следует, что искомая частота генератора таймера IWDG:

$$F_{LSI} = F_{LSE} * \frac{RTC\_PRL}{IWDG\_RLR} \quad (2)$$

В практической программе (2) реализуется изменением величины  $IWDG\_RLR$  до тех пор, пока секундное прерывание от RTC не станет наиболее близким к прерыванию от IWDG. Полученное значение  $F_{LSI}$  сохраняется в энергонезависимой памяти МК и периодически корректируется (практически корректировку достаточно делать 1 раз в час).

## Пакет адаптивной синхронизации

Пусть при включении нового узла  $N_x$  в сеть, для него в пакете синхронизации передается полное, 32 битовое значение RTC узла – координатора, полученное значе-

ние RTC заменяет текущее значение RTC узла  $N_X$ , согласно (2) вычисляется значение  $F_{LSI}$  для узла  $N_X$  и узел  $N_X$  остается в активном режиме. В трансивере типа CC2520 имеется сигнал SFD, который на стороне передатчика устанавливается в «0» по завершении передачи, а на стороне приемника устанавливается в «0» по завершении «правильного приема» (т.е. совпадают адреса, КС). На временной диаграмме ниже показаны графики исследуемых сигналов.

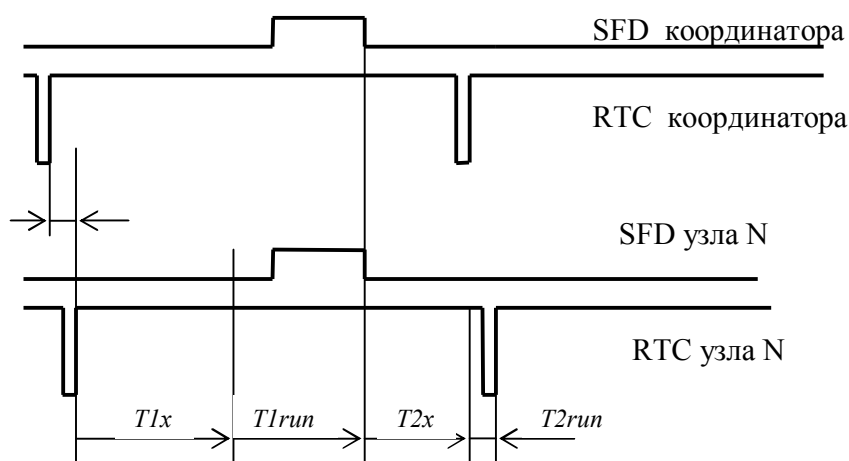


Рисунок 6 – Графики сигналов

При получении второго пакета синхронизации узлом  $N_X$  определяются время  $T1x$ , в течение которого узел  $N_X$  может находиться в спящем режиме, без риска пропустить сеанс связи с координатором. Также определяется время рабочего режима  $T1runx$  и время  $T2x$ , в течение которого узел  $N_X$  может находиться в спящем режиме без риска пропустить секундный импульс RTC узла  $N_X$ . Время активного режима  $T2runx$  определяет время активного режима – ожидания секундного импульса RTC узла  $N_X$ .

При последующих сеансах связи координатора с узлом  $N_X$ , для узла  $N_X$  проводится минимизация времен  $T1runx$  и  $T2runx$ .

Практическая программная реализация показала, что возможна синхронизация процессов в сети и с меньшей погрешностью, чем  $\pm 0,5$  мС, но это справедливо только для устойчивой связи. При появлении сбоев связи координатору и узлам, с которыми потеряна связь, придется увеличить времена активных режимов, увеличить число пакетов связи и, возможно, заново произвести синхронизацию в сети.

## Выводы

1 Предложен подход к реализации синхронизации в беспроводной сети, типа ZigBee (IEEE 802.15.4) [2], в диапазоне 2,4 ГГц с топологией «звезда», в которой имеется в точности один узел – координатор и множество периферийных узлов.

2 Реализация не требует применения дополнительных микросхем и на стандартных таймерах микроконтроллера типа STM32F100RC позволяет достичь точности синхронизации  $\pm 0,5$  мС.

3 Реализация синхронизации позволяет работать периферийному узлу беспроводной сети до 100 дней от 2 батареек типа АА.

4 Предложенная реализация синхронизации работает полностью в автоматическом режиме как при настройке, так и при эксплуатации беспроводной сети.

## Литература

1. Mottola L. Programming wireless sensor networks: Fundamental concepts and state of the art / L. Mottola, G.P. Picco // ACM Computing Surveys. – 2011. – № 3. – P. 51
2. Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). – IEEE 802.15.4 2006 Specification.
3. SWRS068. CC2520 DATASHEET – 2.4 GHZ IEEE 802.15.4/ZIGBEE® RF TRANSCEIVER // [Электронный ресурс]. – Режим доступа : [www.ti.com/wireless](http://www.ti.com/wireless).
4. RM0041. Reference manual. STM32F100xx advanced ARM-based 32-bit MCUs // [Электронный ресурс]. – Режим доступа : [www.st.com](http://www.st.com).
5. Application Note AN061. Optimizing Power Consumption with the CC2520DK // [Электронный ресурс]. – Режим доступа : [www.ti.com/wireless](http://www.ti.com/wireless).

## Literatura

1. Mottola L. Programming wireless sensor networks: Fundamental concepts and state of the art / L. Mottola, G.P. Picco // ACM Computing Surveys. – 2011. – № 3. – P. 51
2. Part 15.4 : Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications for Low-Rate Wireless Personal Area Networks (LR-WPANs). – IEEE 802.15.4 2006 Specification.
3. SWRS068. CC2520 DATASHEET – 2.4 GHZ IEEE 802.15.4/ZIGBEE® RF TRANSCEIVER // [www.ti.com/wireless](http://www.ti.com/wireless).
4. RM0041. Reference manual. STM32F100xx advanced ARM-based 32-bit MCUs // [www.st.com](http://www.st.com).
5. Application Note AN061. Optimizing Power Consumption with the CC2520DK // [www.ti.com/wireless](http://www.ti.com/wireless).

### RESUME

*S.A. Polivtsev*

### *Adaptive Synchronisation in a Wireless Network*

In article theoretical and practical questions of synchronisation of knots of wireless network ZigBee of a range of 2,4 GHz are considered. It is shown that on the unspecialized microcontroller of type STM32F10x it is possible to reach accuracy of synchronisation  $\pm 0,5$  a mS. The main relations are resulted, allowing to reach the set accuracy.

The approach to synchronisation implementation in a wireless network of type ZigBee (IEEE 802.15.4) [2] in a range of 2,4 GHz with topology «star» in which there is in accuracy of one knot – the co-ordinator and set of peripheral knots is offered.

Implementation does not demand application of additional chips and on standard timers of the microcontroller of type STM32F100RC allows to reach accuracy of synchronisation  $\pm 0,5$  a msec. Synchronisation implementation allows to work to peripheral knot of a wireless network till 100 days from 2 batteries of type AA. The offered implementation of synchronisation works completely in an automatic mode both at customisation and at maintenance of a wireless network.

*Статья поступила в редакцию 03.07.2013.*