

АЛГОРИТМ ДЛЯ ОТЛАДКИ ПРОЦЕССОВ ВЗАИМНОЙ СИНХРОНИЗАЦИИ УЗЛОВ MESH-СЕТИ

А.С. Семенович, С.Н. Семенович, И.П. Стецко

Белорусский государственный университет, Минск, Беларусь

E-mail: semenoviAS1@bsu.by

В работе исследуется алгоритм взаимной синхронизации устройств mesh-сети, основанный на рекламных пакетах (Advertising PDU) в сочетании с функционалом исходного стека BLE.

Ключевые слова: Bluetooth Low Energy; время синхронизации; беспроводные сети; Интернета вещей; IEEE 802.1AS.

Синхронизация времени является ключевой функцией для беспроводных сетей из сенсорных устройств, которые применяются в медицинских и спортивных IoT-приложениях, часто подразумевающих использование большого количества автономных устройств, объединённых в одну беспроводную mesh-сеть, узлы которой взаимодействуют друг с другом, обмениваясь данными, и выполняют различные асинхронные задачи [1]. При использовании ячеистых сетей со сложной маршрутизацией данных требуется взаимная синхронизация внутренних таймеров всех устройств для обеспечения хронологического порядка при регистрации событий от встроенных датчиков, для координации этапов общего измерительного процесса между устройствами и синхронизации исполнительных устройств сети.

Хотя в сенсорных сетях с ячеистой топологией используются различные беспроводные протоколы, именно Bluetooth Low Energy в последние годы привлек внимание разработчиков благодаря аппаратной поддержке режимов коммуникации с низким энергопотреблением и доступности готовых решений для автономных датчиков.

В данной работе исследован алгоритм взаимной синхронизации устройств mesh-сети, основанный на рекламных пакетах (ADV PDU) в сочетании с функционалом исходного стека BLE. Продемонстрирован метод оценки смещения хронологии событий и дрейфа частоты встроенного таймера узлового устройства, а также исследовано влияние количества пакетов синхронизации на общую точность взаимной синхронизации узлов.

Предложенный алгоритм синхронизации основан на использовании уровня MD (media-dependent) стандарта IEEE 802.1AS для беспроводных систем связи. Стандарт определяет протокол и процедуры, используемые для обеспечения синхронизации времени с минимальным дрейфом синхронизации в течении суток для мостовых и виртуальных мостовых локальных сетей с критически важной хронологией событий. MD-сущность

отвечает за передачу временных меток и синхронизацию времени между устройствами по принципу «master-slave» – подчиненные узлы синхронизируются сами с часами главного узла на разных уровнях mesh-сети. Такой подход обеспечивает субмикросекундную точность синхронизации без потери пропускной способности радиоканала для приложений с асинхронными процессами [1].

Алгоритм взаимной синхронизации WPTP (Wireless precision time protocol) выполняет синхронизацию по уровням [2]. На первом этапе синхронизируются узлы одного уровня. Эти узлы становятся главными для узлов следующего уровня только после синхронизации узлов предыдущего уровня, и так далее, как показано на рис. 1.

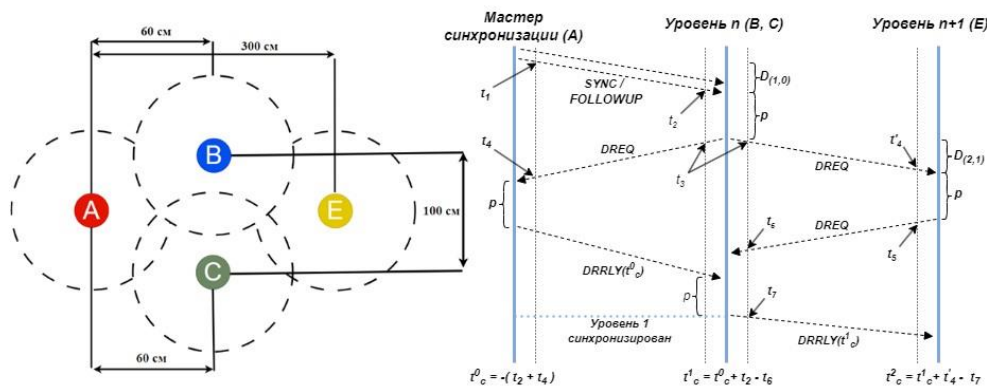


Рис. 1. Принцип работы алгоритма взаимной синхронизации узлов mesh-сети

Таким образом, ведомые узлы могут быть соединены с ведущим двумя способами: *напрямую* – узлы, которые находятся в пределах диапазона передачи главного узла т.е. узлы уровня n (узлы B и C); *косвенно* – узлы, которые не находятся в пределах диапазона передачи главного узла (узел E). Для этого используются промежуточные узлы в качестве переходов для соединения с ведущим (т.е. узлы уровня n).

Синхронизация узлов n уровня. Формула для вычисления смещения выглядит следующим образом:

$$\Delta = \frac{(t_2 + t_3) - (t_4 + t_1)}{2} \quad (1)$$

где t_2, t_3 относятся к шкале времени ведомого устройства, временные метки t_1, t_4 относятся к шкале времени мастер-устройства, следовательно они могут быть рассчитаны на этом уровне и отправлены в виде единой консолидированной временной метки t_c^0 для последующего использования ее на уровне n, где время смещения события будет рассчитываться по формуле (2):

$$\Delta_{(1,0)} = \frac{(t_2 + t_3) + t_c^0}{2} \quad (2)$$

Следовательно, алгоритм синхронизации требует только три радиопакета для вычисления и коррекции смещения события. Для учета задержки передачи сообщения используется второй набор сообщений: *SYNC*-маяк используется для инициации синхронизации и *FOLLOWUP*-маяк, используемый для поддержания синхронизации.

Синхронизация узлов n+1 уровня. Взаимная синхронизация узлов осуществляется по Bluetooth в ISM-диапазоне 2,4 ГГц (полоса частот которого разделена на 40 физических каналов, только три из которых (№ 37, 38 и 39) отведены под рекламные сообщения), путем обмена сообщениями синхронизации (маяками) в несколько этапов [3].

Использования ADV-каналов связи в беспроводных сетях подразумевает, что каждое сообщение-маяк будет передаваться как широкополосное. Следовательно, *DREQ*-маяк запрос данных у рекламирующего устройства, отправленное узлами уровня n узлам предыдущего уровня n-1 может быть использовано как *SYNC*-маяк для узлов следующего уровня n+1. Тогда смещение событий для узлов уровня n+1 вычисляется по формуле:

$$\Delta_{(2,0)} = t'_4 - D_{(2,1)} - p - D_{(1,0)} - t_1 \quad (3)$$

где $D_{(2,1)}$ – задержка передачи между уровнем n+1 и уровнем n; p – время на прием и обработку сообщения узлом, предполагается одинаковым для всех узлов; $\Delta_{(2,0)}$ – смещение часов между узлами уровня n+1 и уровня 0. Используя терминологию временных меток (рис. 1) и формулу (3) для расчета задержки передачи сообщения между уровнями, можно вычислить временную метку, вносимую на каждом уровне:

$$\Delta_{(2,0)} = \underbrace{(t'_4 + t_5)}_{\text{уровень } n+1} + \underbrace{(t_2 - t_6)}_{\text{уровень } n} + \underbrace{(t_2 - t_6)}_{\text{уровень } 0} + \overbrace{t_c^1} \quad (4)$$

Так как уровень 0 уже был рассчитан, следовательно можно передавать единую консолидированную временную метку $t_c^1 = (t_2 - t_6) + t_c^0$ до уровня n+1. Результирующая формула для вычисления смещения на уровне n+1 будет выглядеть как:

$$\Delta_{(n,0)} = \frac{1}{2}(t_{r(n-1)}^n + t_{s(n-1)}^n + t_c^{n-1}) \quad (5)$$

где t_c^{n-1} может быть рассчитан как:

$$t_c^{n-1} = \begin{cases} t_c^{n-1} + t_r^n - t_{r(n+1)}^n, & \text{если } n > 1 \\ -(t_4 + t_1), & n = 0 \end{cases} \quad (6)$$

Таким образом, алгоритм взаимной синхронизации узлов mesh-сети построен на передачи консолидированного значения временной метки узлам уровня $n-1$, используя *DRPLY*-маяк, в то время как маяки *SYNC* и *DREQ* не содержат временных меток и используются только для инициирования взаимной синхронизации. Такой механизм значительно сокращает общее время согласования временных шкал и количество пакетов синхронизации без ущерба для точности синхронизации и пропускной способности канала связи.

На рис. 2 (слева) представлены осциллограммы взаимной синхронизации трех устройств разных уровней в режиме свободно работающего таймера – сигналы синхронизации не передаются; синим цветом показан мастер синхронизации; красным – узлы 1 уровня; зеленым – узлы 2 уровня. В течение 1 часа максимальный разброс значений для смещения события 3 уровня составил ~ 700 нс. Справа на рис. 2 – результат эксперимента с использованием сигналов синхронизации, в этом случае наибольший разброс составил ~ 100 нс.

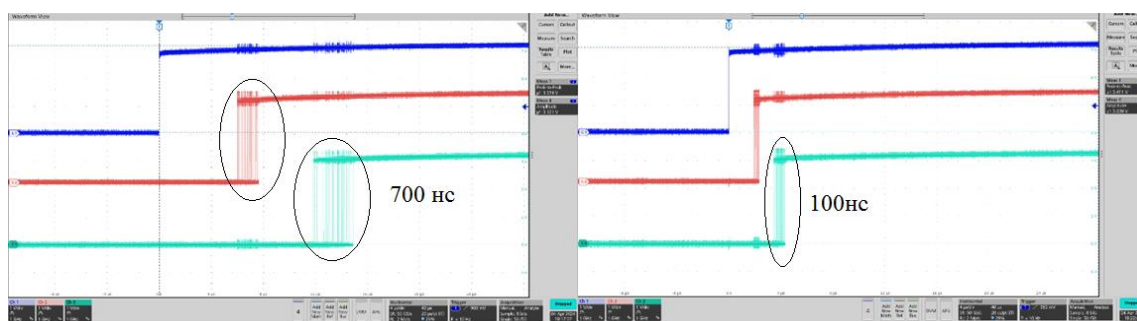


Рис. 2. Осциллограммы смещений событий синхронизации для узлов 3-х уровней

ВЫВОД

Была исследована задача отладки процессов взаимной синхронизации узлов BLE mesh-сети на основе протокола WPTP. Проведен эксперимент, подтверждающий высокую точность взаимной синхронизации и возможность применения этого алгоритма в IoT-приложениях для реализации измерительного процесса и синхронизации исполнительных устройств сети в режиме реального времени.

БИБЛИОГРАФИЧЕСКИЕ ССЫЛКИ

1. Семенович А.С. Масштабируемый IoT тренажер спортсмена. Сборник научных работ студентов Республики Беларусь «НИРС 2020». 2021. С. 143-144.
2. IEEE 1588 for Clock Synchronization in Industrial IoT and Related Application: A Review on Contributing Technologies, Protocols and Enhancement Methodologies / Idrees, Z., Granados, J., Sun, Y. [et al.] // IEEE Access. 2020.
3. Iannizzotto, G.; Milici, M.; Nucita, A.; Lo Bello, L. A Perspective on Passive Human Sensing with Bluetooth. Sensors. 2022.