

СИСТЕМИ ЄДИНОГО ЧАСУ НА БАЗІ ПЛАТФОРМИ ARDUINO

Орієнтування в часі, а також його точність, завжди були дуже важливими для людей. В сучасному темпі життя, мати точний час майже життєво необхідно.

На даному етапі розвитку інформаційно-комп'ютерних технологій, комп'ютери та інші подібні системи застосовуються майже у всіх сферах людського життя.

Використовувати час та часові мітки є характерним для комп'ютерних технологій. Timestamp являє собою послідовність символів або закодованої інформації, що показує, коли відбулася певна подія. Зазвичай показує дату і час (іноді з точністю до часток секунди).

При проведенні різних операцій над файлами (створення, редагування, зберігання), даному файлу присвоюється часова мітка проведення тієї чи іншої операції. Це дозволяє систематизувати файли в комп'ютерній системі, що робить зручнішим їх пошук та операції з ними.

Часові мітки отримують більше значення, коли комп'ютер підключається до мережі і починає обмінюватися даними з іншими елементами мережі. Тоді виникає необхідність єдиного часу в мережі, щоб не виникло такої ситуації, коли відповідь на ваш запит фактично прийшла раніше, ніж ви його відправили.

Електронна відмітка – це спосіб достовірно стежити за часом створення і модифікації документа. «Достовірно» тут означає, що ніхто, навіть власник цього документа, не в змозі змінити створену одного разу інформацію так, щоб її цілісність не порушилася. Адміністративна сторона включає прозору збірку управління відмітками часу, їх створення і оновлення.

Захищена позначка часу – це позначка, видана при свідках. Trusted third party (ТТР) веде себе як timestamping authority (TSA). Це використовується для підтвердження існування певних даних до певного моменту часу (контракти, дані дослідження, медичні записи і т. ін.) без можливості дописування заднім числом.

Складні TSA можуть використовуватися для підвищення надійності і зменшення уразливості. При цифровому підписі – timestamp посилається на time code або digitally signed timestamp, які призначені для підтвердження існування певного документа в певний час. Ця техніка заснована на цифрових підписах і хеш-функції.

Для вирішення цієї проблеми, необхідно призначити один з комп'ютерів «головним годинником», який буде повідомляти точний час в мережі. Таке рішення підходить для локальних мереж, але не для Інтернету, де затримки передачі даних дуже важко передбачити і залежать від багатьох факторів.

Особливо важлива система синхронізації часу на великих промислових підприємствах, де затримка має бути не більше однієї або декількох мікросекунд (наприклад, енергетичні підприємства).

Ефективним варіантом є побудова на великих підприємствах мережі за допомогою технології Ethernet. Проте використання цієї технології призводить до деяких незручностей:

- всі вузли в мережі мають однаковий мережевий доступ в будь-який момент часу, а для систем автоматизації на підприємствах одні вузли є більш важливими ніж інші;
- Ethernet по суті є «статичним» – неможливо передбачити, коли поставлений в чергу запит досягне своєї цілі; системи автоматизації вимагають точних значень;
- Ethernet передає невеликі пакети даних.

Протокол мережевого часу NTP та його більш проста версія SNTP – найбільш поширений метод синхронізації часу в мережі. В його основі лежать складні механізми доступу до центрального серверу часу, що дозволяють через різні канали зв'язку синхронізувати час в підмережах, а також узгодити локальний сервер часу в кожному вузлі.

Ці протоколи використовуються для синхронізації часу в локальних та глобальних мережах, де допустима точність до мілісекунд.

Проте, за допомогою налаштованого протоколу SNTP та стандартних комутаторів мережі Ethernet можна досягти точності до 25 мікросекунд.

Для цього необхідно проставляти мітки часу на рівні переривань зі сторони клієнта та зі сторони сервера часу.

Крім цього, можна проводити синхронізацію, де часові мітки передачі даних проставляє клієнт, а часові мітки отримання даних проставляє сервер.

Побічною дією використання цього методу є відсутність механізмів автоматичного калібрування часу запізнення, тому калібрування затримки передачі з порту сервера і порту клієнта і мінімальну затримку на комутаторі доводиться виконувати вручну.