

*Ткаченко К.Е.,
Студент магистратуры 2 курса
Радиотехнический факультет
Томский государственный университет систем управления и
радиоэлектроники
Россия, г. Томск*

РАЗРАБОТКА СИМУЛЯТОРА СЕРВЕРНОЙ ЧАСТИ ПРИЛОЖЕНИЯ SMARTPTT НА ЯЗЫКЕ ПРОГРАММИРОВАНИЯ C#

***Аннотация:** в данной работе представлена разработка симулятора серверной части приложения SmartPTT на языке программирования C#. Разработка приложения проводилась в интегрированной среде разработки Visual Studio.*

***Ключевые слова:** WebSocket, C#, SmartPTT, WPF, UI, сетевое программирование.*

***Annotation:** this paper presents the development of a simulator for the server part of the SmartPTT application in the C# programming language. The application was developed in the Visual Studio integrated development environment.*

***Key words:** WebSocket, C#, SmartPTT, WPF, UI, NETWORK PROGRAMMING.*

Актуальность работы.

В настоящее время стремительно развиваются языки программирования. И многие языки программирования нужны для решения разных задач и внедрение в различные сферы жизни. В моем случае, разработка симулятора серверной части приложения написана на языке

программирования C# платформа .NET т.к. в дальнейшем будет применяется в продукте SmartPPT от компании ООО “Элком+”.

SmartPPT является программным комплексом, который предоставляет пользователям множество функциональных возможностей от использования системы MOTOTRBO. Диспетчерская консоль SmartPPT разрешает пользователям не только приобрести больше от стандартного функционала MOTOTRBO, но и предоставляет новые возможности, недоступные без использования специализированных аппаратных или программных решений:

- В режиме реального времени отслеживание местоположения абонентов
- Объединение радиосетей, в том числе аналоговых и цифровых
- Запись всех событий и переговоров, произошедших в системе
- Управление задачами абонентов и многое другое

Внедрение программного обеспечения SmartPPT позволяет выделить в структуре компании единый центр информационного контроля и оперативного реагирования.

Научное и техническое значение работы

Технология WebSocket нужна там, где используется постоянная двухсторонняя связь между клиентом и сервером. Работать с WebSocket довольно просто, не требует значимых усилий, что упрощает разработку на нем. К тому же WebSockets обладает рядом преимуществ, таких как: снижение используемого трафика, безопасность, минимальная задержка передачи данных. Все это улучшит ПО SmartPPT.

В начале работы разработан UI часть симулятора серверного приложения.

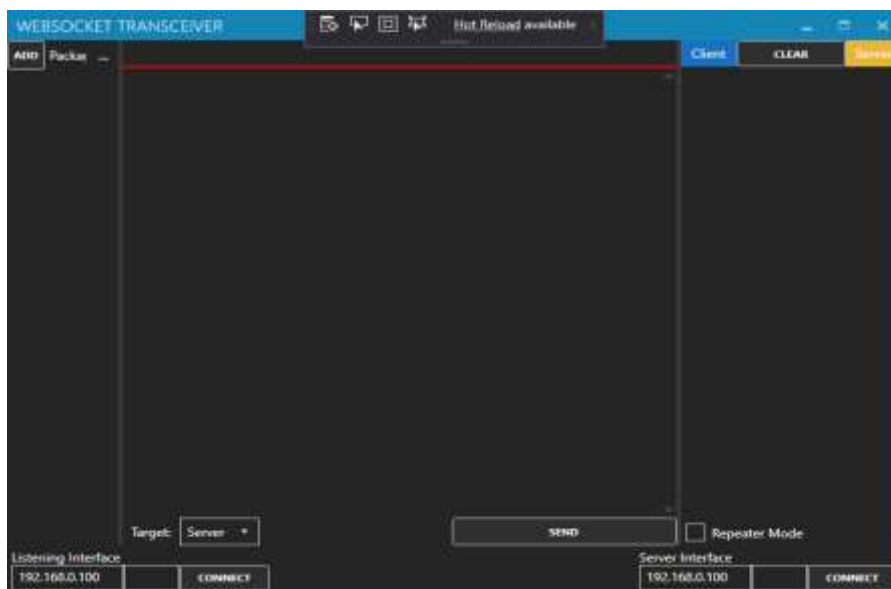


Рисунок 1 – UI часть приложения

Реализуется соединение с помощью технологии WebSocket. Проверяем работоспособность приложения.

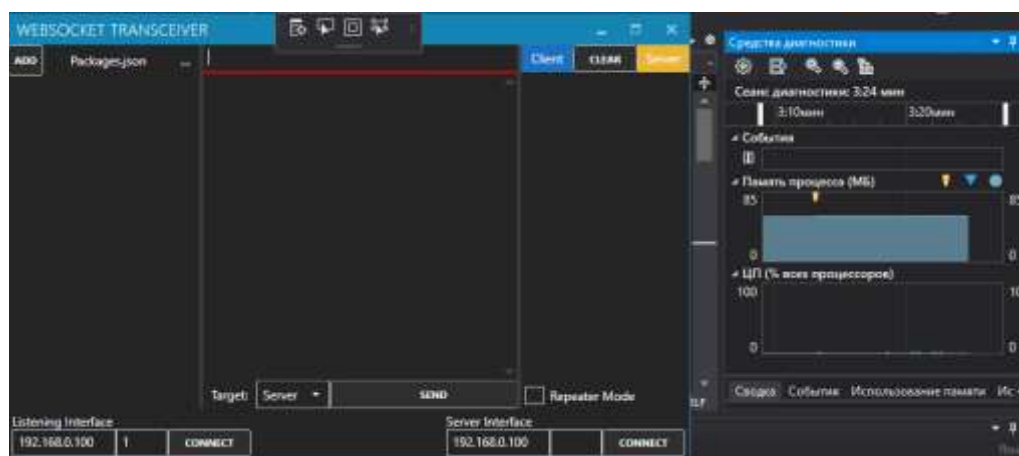


Рисунок 2 – Проверка работоспособности приложения

Теперь проводим испытания симулятора, для этого запускаем сервер SmartPPT, запускаем Web Client. Соединяем Web Client и сервер с помощью WebSocket Transceiver с включенной функцией Repeater Mode, которая позволяет в автономном режиме передавать пакеты данных между сервером и клиентом. Делаем вызовы с помощью радиостанции Motorola и передаем пакеты с данными. При этом смотрим на сколько симулятор грузит CPU и ОЗУ.

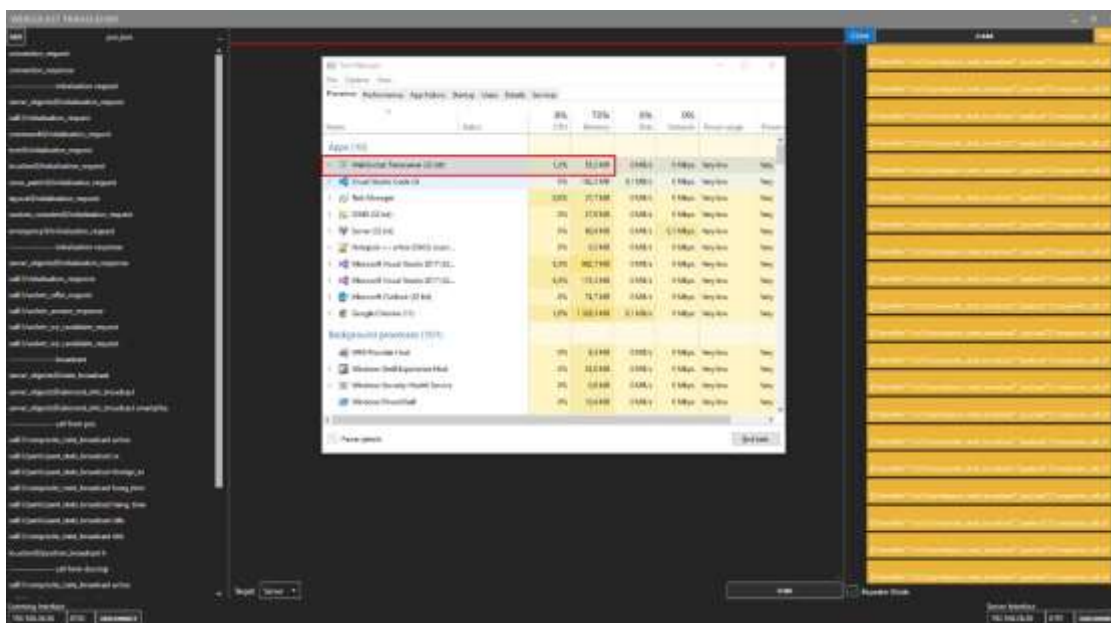


Рисунок 3 – Нагрузка на CPU и ОЗУ

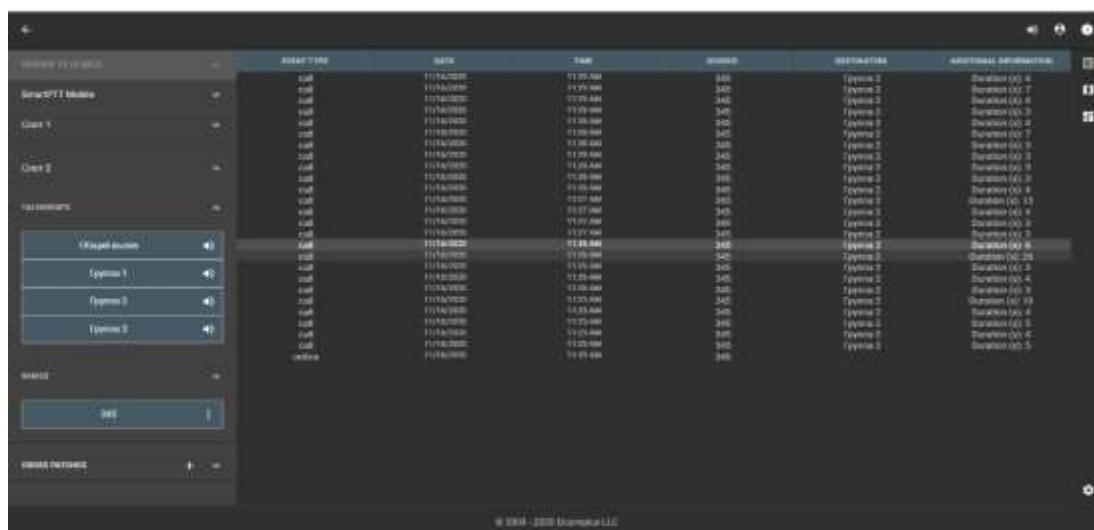


Рисунок 4 – Тестовые вызовы

Из результатов проведенной работы можно сделать вывод о том, что при использовании симулятора, он выполняет функции сервера, а именно: принятие пакетов при взаимодействии с радиоабонентом и отправки обратно. Это позволяет тестировать радиостанции виртуально, меняя данные в полученном пакете. В дальнейшем наработки данной магистерской работы будут использоваться при разработке продукта SmartPPT от компании ООО «Элком+».

1. Симулятор позволяет разрабатывать продукт без использования реального оборудования, что сокращает загруженность оборудования

2. Симулятор сокращается время на реализацию, т.к. настройка оборудования может занять много времени, а используя симулятор нужно только запустить его.

Использованные источники:

1. Рихтер Д. CLR via C# - 2018. – 896 с.
2. Троелсен Эндрю. Язык программирования C# 7 и платформы .NET и NET Core. – 2018. – 1316 с.
3. Роберт Сесил Мартин. Clean Code – 2008. – 368 с.