

УДК 65.011.56:004.725.4

А.В. Тарасова, аспирант,

С.Н. Бобылев, ст. преподаватель

Севастопольский национальный технический университет

ул. Университетская, 33, г. Севастополь, Украина, 99053

E-mail: anna_tarasowa@rambler.ru

МОДЕЛЬ УЗЛА РАСПРЕДЕЛЕННОЙ АВТОМАТИЗИРОВАННОЙ СИСТЕМЫ УПРАВЛЕНИЯ ПРОИЗВОДСТВОМ

Предлагается имитационная модель узла для проектирования и анализа распределенной АСУП. Рассматривается представление узла в виде СМО. Приводится описание полной и упрощенной моделей, а также результаты моделирования при использовании конкретных интерфейсов.

Ключевые слова: *распределенная АСУП, узел сети, микроконтроллер, имитационное моделирование*

Введение. Основной целью построения распределенных автоматизированных систем управления производством (АСУП) является удешевление и упрощение технологий и менеджмента производства и эксплуатации конечной системы [1]. Частично это достигается за счет организации сквозного сетевого доступа: от мощных компьютеров и многофункциональных микроконтроллеров до периферийных устройств (исполнительные механизмы, регуляторы, датчики и т.д.). При этом на подобную связь налагаются все современные требования по надежности, открытости, функциональности. Технология единой информационной сети, объединяющей датчики, исполнительные механизмы и контроллеры, получила название fieldbus – полевая шина или промышленная сеть. Это не только физический способ объединения устройств, но и программно-логический протокол их взаимодействия. В условиях роста производства микропроцессорных устройств, все большее распространение получают цифровые промышленные сети (ЦПС), состоящие из большого числа узлов. Особенности ЦПС – это распределенный характер узлов сети и цифровой способ обмена информацией между узлами сети. ЦПС позволяют учитывать такие требования АСУП, как гибкость и модифицируемость комплекса при необходимости коррекции их состава, скорость обмена данными, надежность и т.д.

При объединении в одной ЦПС нескольких устройств возникают проблемы их взаимодействия. В случае если коммуникационная система гомогенна (однородна), проблемы аппаратно-программного характера, как правило, бывают решены. Но когда сеть гетерогенна (разнородна), неизбежно возникают вопросы их согласования. Зачастую, гетерогенность обусловлена тем, что используются устройства различных производителей.

Одним из способов устранения конфликтов аппаратно-программного вида является использование микроконтроллеров в качестве узлов сетей АСУП. При описании промышленных сетей, используемых в АСУП, как правило, достаточно нижних уровней модели OSI (ISO). К нижним уровням модели относят протоколы физического, канального, сетевого и транспортного уровней, такие, как: RS – 232, CAN, LIN, ASI, LON и т.д. [1]. На сегодняшний день разработчики микроконтроллеров, предвосхищая потребности в ближайшем будущем, интегрируют в микроконтроллеры широкий спектр интерфейсов. Так, микроконтроллер AT90CAN128 фирмы Atmel имеет интегрированные интерфейсы JTAG, CAN, UART/USART, SPI, TWI. Это позволяет реализовать на его основе узел для сетей различной топологии, взаимодействие в которых реализовано через разные интерфейсы.

Целью данной работы является рассмотрение особенностей узлов сетей АСУП на базе микроконтроллеров и описание разработанной имитационной модели для их проектирования и анализа.

Представление узла АСУП как системы массового обслуживания (СМО). Классическая СМО производит обслуживание поступающих в неё требований (заявок) на обслуживающих приборах и может содержать от одного до бесконечного числа приборов [2].

В терминах теории массового обслуживания элементы узла сети могут быть представлены следующим образом: 1) запросы на прерывание: заявки; 2) флаг прерывания: очередь единичной длины; 3) микроконтроллер: обслуживающий прибор.

Исходя из аппаратной реализации системы прерываний в микроконтроллере, СМО, соответствующая рассматриваемому узлу сети, представляет собой одноканальную СМО с потерями и дисциплиной обслуживания с относительными приоритетами [3].

Когда обслуживающий прибор свободен, следующая заявка на обслуживание выбирается из имеющихся в очередях в соответствии с приоритетами. Обслуживающий прибор задерживает каждую поступающую в него заявку на время, необходимое для обработки прерывания данного типа, и после освобождения готов принять следующую.

Имитационная модель узла сети. В качестве моделируемого узла сети был выбран микроконтроллер AT90CAN128 [3]. Для построения имитационной модели использовалась среда моделирования AnyLogic 5.4.0 и встроенная библиотека элементов для создания сложных дискретно-событийных моделей Enter-prise Library [4].

Структура модели изображена на рисунке 1. Число типов генерируемых заявок и длина системы очередей определяется количеством поддерживаемых узлом прерываний и для построенной модели составляет 36. Такое достаточно большое число прерываний значительно увеличивает число функциональных блоков модели.

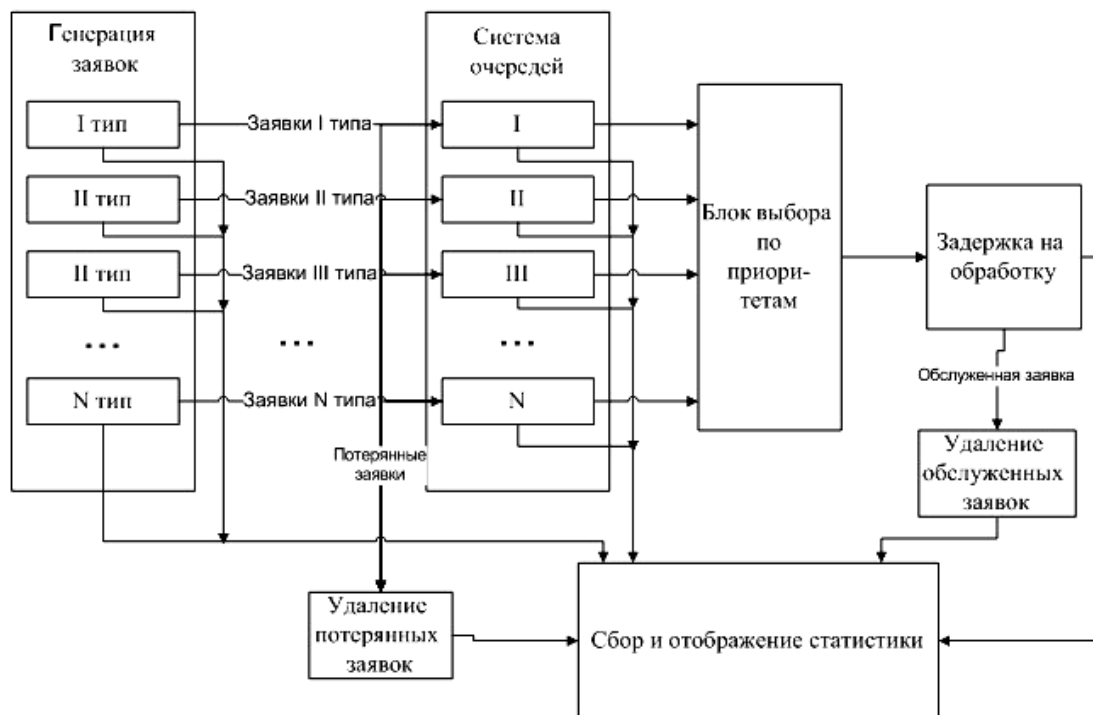


Рисунок 1 – Структура разработанной модели

С учетом аппаратных особенностей в среде AnyLogic модель включает в себя следующие элементы:

1. Блоки source_INT0 ... source_SPM_READY – являются источниками заявок и имитируют ситуацию поступления запроса на прерывание соответствующего типа в микроконтроллер. Частота генерации заявок может быть детерминированной, случайной по заданному закону распределения или согласно расписанию. Этот параметр задается пользователем и зависит, в том числе, от структуры АСУП и алгоритмов маршрутизации в ней.

2. Блоки Output1 ... Output33 – блоки выбора, может ли заявка стать в очередь на обслуживание или нет. В первом случае она должна быть отправлена в очередь на ожидание, во втором – на удаление из системы. Условием выбора является число свободных мест в очереди.

3. Блоки sink_INT0 ... sink_SPM_READY – блоки удаления заявки из системы, если она не может поступить в очередь на ожидание обслуживания. В реальном узле это соответствует отказу заявки на обслуживание и возврат её в сеть, где дальнейшие действия над заявкой определяются алгоритмами функционирования сети.

4. Блоки INT0 ... SPM_READY – блоки единичной очереди для задержки заявки на время ожидания до обслуживания. В реальном микроконтроллере такая очередь соответствует флагу запроса на прерывание и сохранению данного запроса до обработки. Все последующие пришедшие запросы того же типа в то время, пока флаг взведен, теряются.

5. Блоки Priority1 ... Priority36 – соответствуют реализации выбора прерывания на обработку в соответствии с приоритетом. Наибольшим приоритетом обладают запросы с меньшими номерами. Если в момент освобождения микроконтроллера более приоритетных прерываний нет, на обслуживание поступают менее приоритетные. Если все очереди пусты, на обработку пойдет первое пришедшее прерывание без ожидания в очереди.

6. Блок delay_CPU – блок имитации работы микроконтроллера. Обеспечивает задержку заявки на заданное время. Возможно установление разных времен задержки для разных типов заявок, в том числе и

основываясь на длине пакета данных, поступивших на обработку. По истечению заданного времени заявка отправляется на удаление из системы.

7. Блок sinkAll – удаление обслуженных заявок из системы.

Число блоков типов 1...4 соответствует числу элементов вектора прерываний микроконтроллера, без учета прерывания нулевого приоритета RESET (данный тип прерывания имеет абсолютный приоритет и соответствует отключению микроконтроллера, что является исключительной ситуацией и в дальнейшем не рассматривается).

Кроме основных блоков в модели существует еще ряд переменных, который позволяет производить сбор статистики по работе системы.

В процессе моделирования работы системы в среде AnyLogic вся разработанная модель находится в оперативной памяти. Поэтому с целью экономии оперативной памяти, и, как следствие, уменьшения времени моделирования при проведении экспериментов и увеличению скорости работы модели, был произведен переход к упрощенной модели. Эксперименты проводились для прерываний по INT1 (имитация локальной нагрузки на узел) и по UARTRX (имитация сетевой нагрузки), поэтому в упрощенной модели были оставлены только они. Упрощенная модель узла АСУП изображена на рисунке 2.

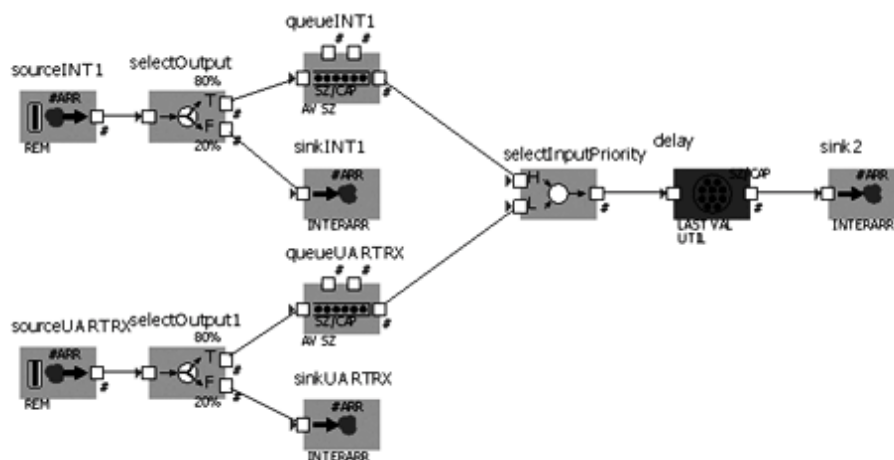


Рисунок 2 – Упрощенная модель узла АСУП в AnyLogic 5.4.0

В упрощенной модели были оставлены следующие элементы: источники заявок sourceINT1 и sourceUARTRX (имитируют поступление запросов на прерывание от внешнего источника прерываний INT1 и по интерфейсу UART), блоки выбора выходного порта selectOutput и selectOutput1, блоки удаления заявки из системы sinkINT1 и sinkUARTRX, очереди queueINT1 и queueUARTRX, блок приоритета SelectInputPriority, МК, блок удаления обслуженных заявок sinkAll, а также ряд переменных для сбора статистики по заявкам. Также для упрощенной модели для удобства работы с ней был разработан анимированный интерфейс пользователя, который в режиме реального времени отображает изменение основных параметров системы по каждому виду заявок.

Определение пропускной способности узла сети. Эффективность работы узла сети зависит от большого числа параметров, потому исследование временных характеристик играет немаловажную роль для дальнейшей оптимизации структуры сети в целом. Изучению могут подлежать многие характеристики, такие как: задержка заявки заданного типа на обработку, время нахождения заявки в системе, интенсивности поступления заявок заданного типа в систему, число обслуживаемых заявок в единицу времени (пропускная способность узла) и т.д. Также немаловажен учет возможных потерь пакетов данных в сети из-за несопоставимых нагрузок на узел и пропускной способности узла.

Как уже отмечалось ранее, моделирование проводилось для двух видов прерываний: внешнее прерывание по INT1 и прерывание по UART/RX. В качестве исследуемого параметра рассматривались различные интенсивности поступления запросов на прерывания и возникающие при этом потери пакетов данных в узле. В соответствии с аппаратной реализацией, прерывания по INT1 являются более приоритетными, чем прерывания по UART/RX. Задержка на обработку одного прерывания вычисляется по формуле

$$t_{пер} = \frac{T_{пер}}{\nu_{ОП}},$$

где $t_{пер}$, $T_{пер}$, $\nu_{ОП}$ – время обработки одного прерывания, длительность одного прерывания в тактах выполнения и частота работы обслуживающего прибора соответственно. Моделирование проводилось

для $T_{пер} = 5$ тактам, $\nu_{оп} = 20833$ тактов/с и битрейта (англ. bitrate – буквально, скорость прохождения битов информации) UART 2400, 4800, 9600, 19200 и 38400 бод.

Результаты экспериментов. В ходе проведения экспериментов было получено подтверждение соотношений локальной и сетевой нагрузок, рассчитанных исходя из теоретических данных, результаты изображены в виде графиков на рисунках 3 и 4.

Модель показала высокую степень чувствительности к изменяемым параметрам экспериментов. Модель дала реакцию на изменение законов распределения интенсивностей поступающих заявок в соответствии с изменениями, характерными для реальных сетей. Так, при переходе от детерминированных интенсивностей входных потоков заявок к распределенным по случайным законам, наблюдается смещение краевых границ (полное прохождение локального трафика и полные потери сетевого трафика) в сторону их расширения.

Закключение. Построенная имитационная модель предназначена для анализа возможностей и характеристик узлов сетей АСУП. На базе разработанной модели проведено исследование работы узла для конкретных интерфейсов. Результаты моделирования показали возможность использования модели при проектировании и эксплуатации сетей АСУП. При проектировании сетей использование модели позволяет оптимизировать структуру сети с целью минимизации ее стоимости. При эксплуатации использование модели упрощает реконфигурацию при добавлении новых узлов и/или замене существующих.

Модель показала большую степень гибкости и возможна ее адаптация под другие виды узлов с наименьшими затратами на перестроение.

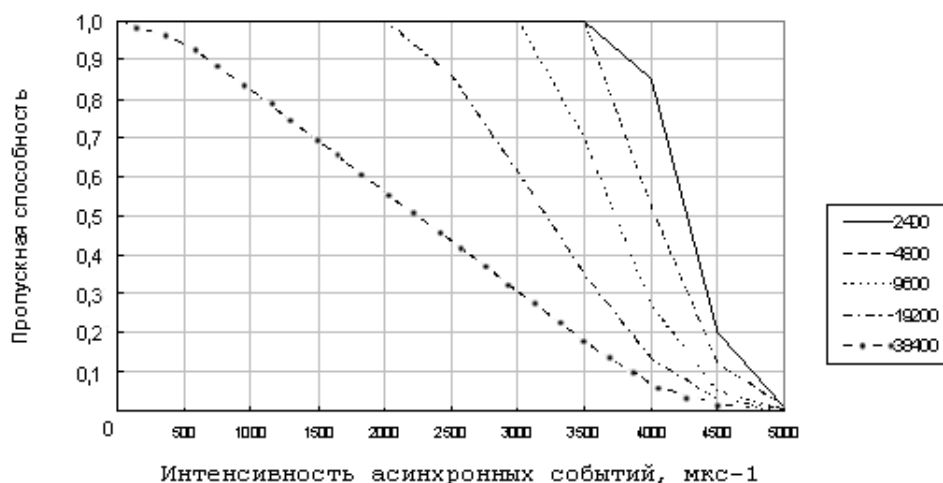


Рисунок 3 – Полученные результаты имитационного моделирования

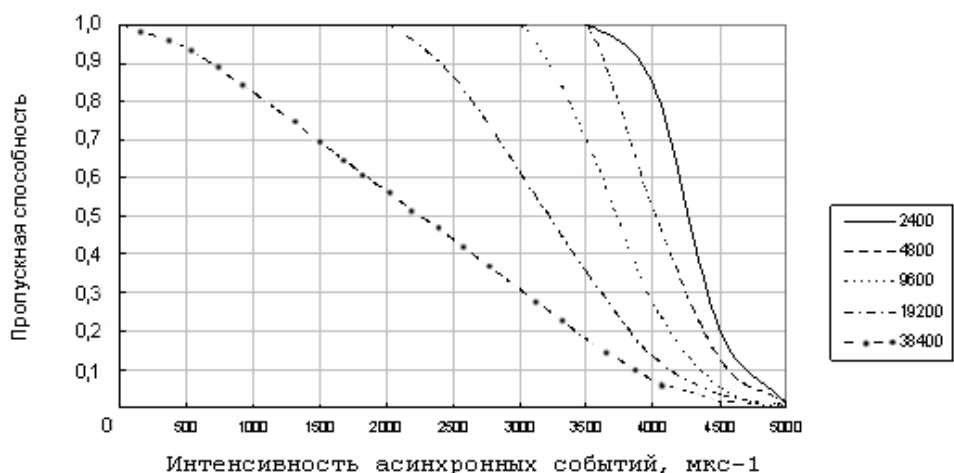


Рисунок 4 – Сглаженные графики результатов имитационного моделирования

Модель предназначена для построения и исследования гетерогенных сетей. На ее базе возможно проектирование ЦПС АСУП с использованием различных типов интерфейсов с добавлением широкого спектра периферии.

Полученные в ходе проведенных экспериментов данные использованы для анализа распределения локальной и сетевой нагрузки от двух любых типов интерфейсов. Предполагается использование модели для определения пропускной способности узлов с большим количеством интерфейсов.

Библиографический список использованной литературы

1. АСУТП.ру - Промышленные сети [Электронный ресурс]. — Электрон. текстовые данные (282 887 bytes). — Режим доступа: <http://www.asutp.ru/?p=600085>, Tuesday, 13 September 2011, 15:09:02.
2. Клейнрок Л. Теория массового обслуживания: пер. с англ. / И.И. Грушко; под ред. В.И. Неймана. — М.: Машиностроение, 1979. — 432 с.
3. 8-bit Microcontroller with 128K Bytes of ISP Flash and CAN Controller AT90CAN128 [Электронный ресурс]. — Электрон. текстовые данные (5 742 631 bytes). — Режим доступа: http://www.atmel.com/dyn/resources/prod_documents/doc7522.pdf Tuesday, 13 September 2011, 12:06:48.
4. AnyLogic. Справочное руководство по Enterprise Library [Электронный ресурс]. — Электрон. текстовые данные (2 108 648 bytes). — Режим доступа: <http://www.xjtek.com/support/download/documentation/> Monday, 5 September 2011, 12:53:16.

Поступила в редакцию 15.11.2011 г.

Тарасова А.В., Бобылев С.Н. Модель вузла розподіленої автоматизованої системи управління підприємством

Запропоновано імітаційну модель вузла для проектування та аналізу розподіленої АСУП. Розглядається подання вузла у вигляді СМО. У статті наводиться опис повної та спрощеної моделей та результати моделювання при застосуванні конкретних інтерфейсів.

Ключові слова: розподіленна АСУП, вузол мережі, мікроконтролер, імітаційне моделювання.

Tarasova A.V., Bobylev S.N. The model of host distributed computer-integrated production

A simulation model of a host for the design and analysis of computer-integrated production is suggested. Representation of the host is considered as a queuing system. The full and simplified models and simulation results using specific interfaces are described in the article.

Keywords: distributed computer-integrated manufacturing, host, microcontroller, simulation modeling.