ИУ5-61

Белоусов Евгений

РК по ОПЖЦАСОИУ №1

Задание:

*На ПОСП описать процесс функционирования следующей системы:*

Сеть передачи данных включает 2 узла коммутации и 1 дуплексный канал между ними. Поток пакетов на каждый из узлов имеет интенсивность L1 и L2 соответственно. Узлы обмениваются между собой пакетами с посылкой подтверждений получения. Новый пакет не отсылается, пока не получено подтверждение о приёме предыдущего пакета. Подтверждение при получении пакета отсылается в первую очередь. Канал периодически ломается, время безотказной работы канала равно В, время ремонта равно R. Пакет или подтверждение, попавшие на «поломку», пропадают. Остальные – ждут устранения неполадки. При этом задано значение таймаута для каждого переданного пакета равное P, по истечении которого при отсутствии подтверждений производится повторная передача пакета. Величины В, R - случайные.

# Модель процесса обмена пакетами сети состоящей из двух узлов и одного дуплексного канала между ними.

Дуплексный канал связи состоит из двух физических каналов, рассмотрим каждый из них по отдельности.

Так как в задании не указаны значения величин, подставим вместо них случайные числа, указав в комментарии отношение к букве величины из задания.

Ввиду отсутствия стандартных блоков (контроллера, агрегата и процессора) в программном решении Visio, для определения типа блока на схеме будем использовать букву в левом верхнем углу блока.



Рисунок . Блочная схема.

Описание алгоритма на ПОСП:

Блок-контроллер Генератор\_поломок

описание

Поломка, Тс - скаляр;

все описание;

алгоритм

НАЧ:

Поломка := "сломан";

Тс := ВРЕМЯ + (RAND \* 10); //Пусть R принадлежит [0, 10]

ждать ВРЕМЯ = Тс;

Поломка := "не сломан"

Тс := ВРЕМЯ + (RAND \* 1000); // Пусть B принадлежит [0, 1000]

ждать ВРЕМЯ = Тс

направить ИНИЦИАТОР на метку НАЧ;

все алгоритм;

все блок.

// дуплексный канал состоит из двух физических каналов, будем рассматривать каждый из них по отдельности

Блок-процессор Физический\_канал\_1

описание

Исходящее\_2 - скаляр в блоке Узел\_2;

Входящее\_1 - скаляр в блоке Узел\_1;

Тс - скаляр;

сообщение - скаляр;

Поломка - скаляр в блоке Генератор\_поломок;

все описание;

алгоритм

НАЧ:

сообщение := Исходящее\_2; // оно точно есть

Исходящее\_2 := NULL;

Тс := ВРЕМЯ + (RAND \* 5); // предположим время прохода прохода сообщения принадлежит [0, 5]

ждать ВРЕМЯ = Тс;

если Поломка = "сломан" то напраить ИНИЦИАТОР на метку КОН;

Входящее\_1 := сообщение;

КОН:

уничтожить ИНИЦИАТОР;

все алгоритм;

все блок.

Блок-процессор Физический\_канал\_2

описание

Исходящее\_1 - скаляр в блоке Узел\_1;

Входящее\_2 - скаляр в блоке Узел\_2;

Тс - скаляр;

сообщение - скаляр;

Поломка - скаляр в блоке Генератор\_поломок;

все описание;

алгоритм

НАЧ:

сообщение := Исходящее\_1; // оно точно есть

Исходящее\_1 := NULL;

Тс := ВРЕМЯ + (RAND \* 5); // предположим время прохода прохода сообщения принадлежит [0, 5]

ждать ВРЕМЯ = Тс;

если Поломка = "сломан" то напраить ИНИЦИАТОР на метку КОН;

Входящее\_2 := сообщение;

КОН:

уничтожить ИНИЦИАТОР;

все алгоритм;

все блок.

Блок-контроллер Узел\_2

описание

сообщение\_было\_отправлено - скаляр;

таймер - скаляр;//при инициализации 0

Исходящее\_2 - скаляр;

Входящее\_2 - скаляр;

сообщение - скаляр;

а - скаляр;

Тс - скаляр;

нужно\_новое\_сообщение - скаляр;//при инициализации - "да"

все описание;

алгоритм

НАЧ:

если не (нужно\_новое\_сообщение = "да") то направить инициатор на метку ОТПР;

сообщение := RAND \* 255;// допустиим, отправляем char :)

сообщение\_было\_отправлено := "нет";

нужно\_новое\_сообщение = "нет";

Тс := ВРЕМЯ + 8; //L2

ОТПР:

ждать 0.0001; // ждем короткий промежуток времени, обеспечивающий движение системы во времени

если ВРЕМЯ > таймер то сообщение\_было\_отправлено := нет;

если (не (Входящее\_2 = NULL) или сообщение\_было\_отправлено = "да" )то направить ИНИЦИАТОР на метку ПРИЕМ;

создать Исходящее\_2 типа ссылка;

Исходящее\_2 := ссылка на скаляр сообщение;

таймер := ВРЕМЯ + 30; // Пусть Р=30

активизировать инициатор из Исходящее\_2 в блок Физический\_канал\_1 на метку НАЧ;

сообщение\_было\_отправлено := "да";

ПРИЕМ:

если Входящее\_2 = "уведомление" то нужно\_новое\_сообщение = "ожидание";

если (Входящее\_2 = NULL) то направиь ИНИЦИАТОР на метку НОВ;

а := "уведомление";

создать Исходящее\_2 типа ссылка;

Исходящее\_2 := ссылка на скаляр а;

активизировать инициатор из С2 в блок физический\_канал\_1 на метку НАЧ;

направить ИНИЦИАТОР на метку НАЧ;

НОВ:

если (не (нужно\_новое\_сообщение = "ожидание") и (ВРЕМЯ > Тс)) то направить ИНИЦИАТОР на метку НАЧ;

нужно\_новое\_сообщение := "да";

направить инициатор на метку НАЧ;

все алгоритм;

все блок.

Блок-контроллер Узел\_1

описание

сообщение\_было\_отправлено - скаляр;

таймер - скаляр;//при инициализации 0

Исходящее\_1 - скаляр;

Входящее\_1 - скаляр;

сообщение - скаляр;

а - скаляр;

Тс - скаляр;

нужно\_новое\_сообщение - скаляр;//при инициализации - "да"

все описание;

алгоритм

НАЧ:

если не (нужно\_новое\_сообщение = "да") то направить инициатор на метку ОТПР;

сообщение := RAND \* 255;// допустиим, отправляем char :)

сообщение\_было\_отправлено := "нет";

нужно\_новое\_сообщение = "нет";

Тс := ВРЕМЯ + 11; //L1

ОТПР:

ждать 0.0001; // ждем короткий промежуток времени, обеспечивающий движение системы во времени

если ВРЕМЯ > таймер то сообщение\_было\_отправлено := нет;

если (не (Входящее\_1 = NULL) или сообщение\_было\_отправлено = "да" )то направить ИНИЦИАТОР на метку ПРИЕМ;

создать Исходящее\_1 типа ссылка;

Исходящее\_1 := ссылка на скаляр сообщение;

таймер := ВРЕМЯ + 30; // Пусть Р=30

активизировать инициатор из Исходящее\_1 в блок Физический\_канал\_2 на метку НАЧ;

сообщение\_было\_отправлено := "да";

ПРИЕМ:

если Входящее\_1 = "уведомление" то нужно\_новое\_сообщение = "ожидание";

если (Входящее\_1 = NULL) то направиь ИНИЦИАТОР на метку НОВ;

а := "уведомление";

создать Исходящее\_1 типа ссылка;

Исходящее\_1 := ссылка на скаляр а;

активизировать инициатор из С2 в блок физический\_канал\_2 на метку НАЧ;

направить ИНИЦИАТОР на метку НАЧ;

НОВ:

если (не (нужно\_новое\_сообщение = "ожидание") и (ВРЕМЯ > Тс)) то направить ИНИЦИАТОР на метку НАЧ;

нужно\_новое\_сообщение := "да";

направить инициатор на метку НАЧ;

все алгоритм;

все блок.