

Моделирование приоритетного обслуживания

Лекция № 11

Доцент, к.т.н. Бабалова И.Ф.

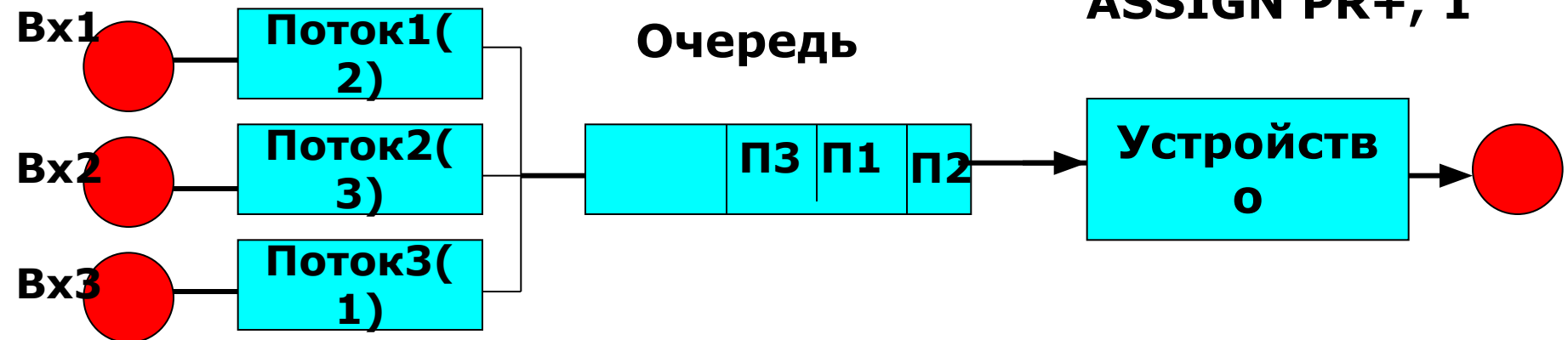
2016г.Маг_каз

Анализ приоритетов транзактов

В блоке **GENERATE** операнд **E** назначает приоритет планируемому транзакту. Блок **PRIORITY** назначает приоритет активному транзакту.

PRIORITY 2

ASSIGN PR+, 1



Приоритет в очереди. Для блока **SEIZE** приоритет транзакта учитывается только при формировании очереди в соответствии со значением приоритета транзакта в блоке **GENERATE** или назначением приоритета в блоке **PRIORITY**. Приоритет у активного транзакта хранится в параметре **PR**. Учтите, что формирование очереди всегда учитывает время прихода транзакта.

Прерывания для одноканальных устройств

Прерывание работы устройства транзактом с большим приоритетом может быть выполнено только в блоке PREEMPT (выгрузить, приобрести преимущество).

PREEMPT A [,B] [,C][,D][,E]

RETURN A

A - имя захватываемого устройства

B - приоритет {PR}

C - имя блока, к которому должен быть направлен прерванный транзакт

D – Номер параметра транзакта, в который будет записано время дообслуживания прерванного транзакта

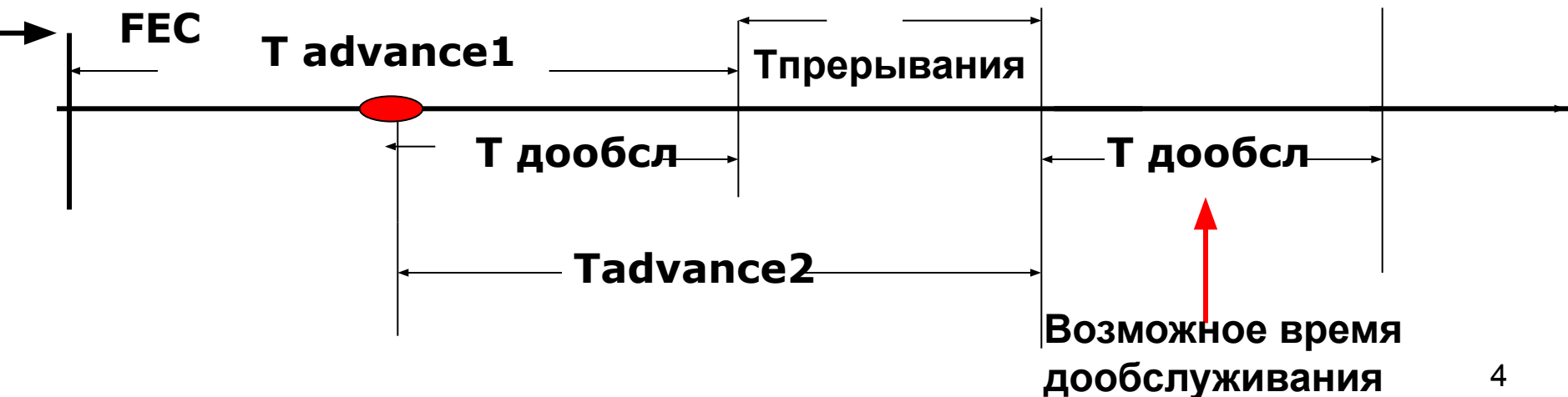
E – Режим удаления прерванного транзакта

Механизм одноуровневого прерывания:

1. Устройство свободно SEIZE, PREEMPT
2. Блок PREEMPT занят
3. Устройство SEIZE занято

В первом случае транзакт обычно занимает устройство
Во втором случае транзакт не может войти в устройство и помещается в список задержек.

В третьем случае, если транзакт был в FEC, то он удаляется из этого списка, снабжается индикатором прерываний и время его дообслуживания запоминается



Пример. Процессор решает большую фоновую задачу. Короткий запрос снимает фоновую задачу и возвращает процессор после ее завершения. Определить число прерываний фоновой задачи и среднее время ее решения.

```

GENERATE 200,50
  QUEUEQQEVM
  SeizeEVM
DEPART QQEVM
  SAVEVALUE ddd+,1
  MARK 8
ADVANCE 180,60
Release EVM
  TABULATE TAB8
TERMINATE
  
```

```

GENERATE 100,20,,,1
  Preempt EVM,PR
SAVEVALUE xxx+,1
Mark 10
  Advance 20,6
Return EVM
  TABULATE Tab10
  Terminate
  
```

```

  Tab10      Table MP10,10,10,10
  TAB8      Table MP8,200,50,10
    GENERATE 100000
    TERMINATE 1
  
```

Многоуровневые прерывания

PREEMPT A [,B] [,C][,D][,E]

Формат блока для организации многоуровневых прерываний

Операнды C, D, E блока PREEMPT реализуют режимы:

1. Режим без дообслуживания PREEMPT CPU, PR, MET,, RE

Транзакт с меньшим приоритетом отстраняется от дальнейшего обслуживания. Он может вернуться на обслуживание, но время его задержки в блоке ADVANCE будет полным, без учета предыдущей задержки.

2. Режим прерывания с дообслуживанием PREEMPT CPU, PR, MET,1

Время дообслуживания транзакта с меньшим приоритетом сохраняется в параметре транзакта P1. При освобождении ресурса транзакт входит в него со временем P1.

Проверка возможности входа в блок PREEMPT выполняется блоком GATE

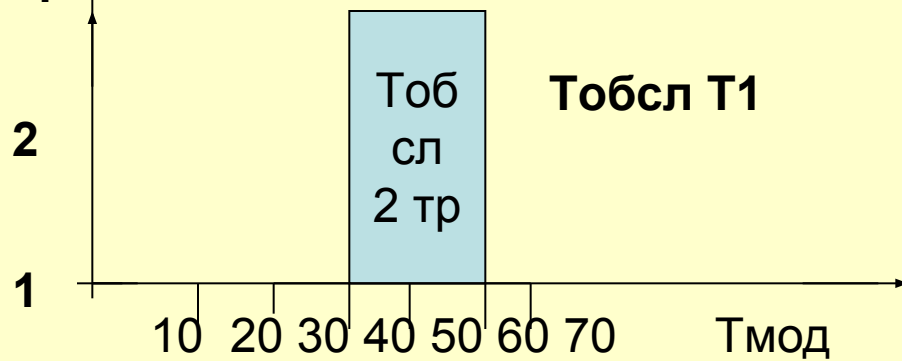
Рассмотрим различия в реализации этих режимов обработки.

Многоуровневые прерывания

PREEMPT A, B, C, D [,E]- полный формат блока

PREEMPT CPU, PR, MET,, RE

Прерывание без дообслуживания



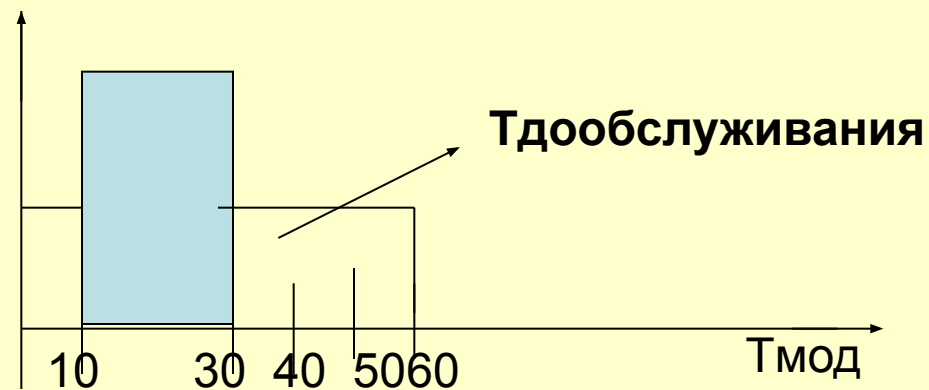
Тобсл1=50

```
Generate ,,1,1
PREEMPT CPU
mark 5
ADVANCE 40
return cpu
savevalue 1,mp5
TERMINATE
```

```
GENERATE ,,30,1,2
PREEMPT CPU,PR,out,,RE
mark 6
ADVANCE 20
return cpu
savevalue 2,mp6
out TERMINATE
```

PREEMPT CPU, PR, MET,1

Прерывание с сохранением времени дообслуживания

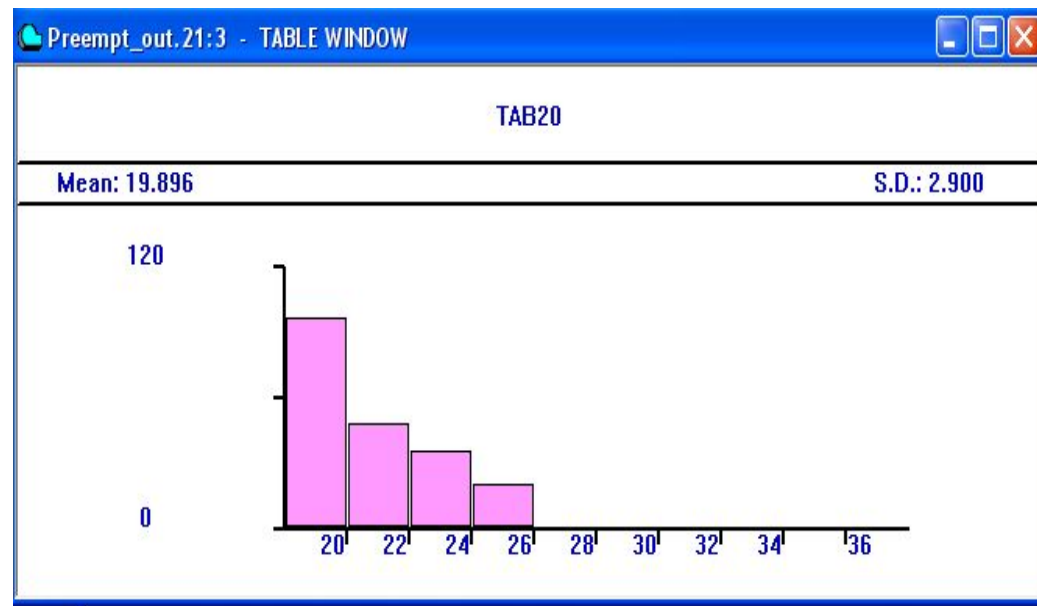


```
Generate ,,1,1      assign 10,40
PREEMPT CPU
met ADVANCE p10
return cpu          Параметр
TERMINATE           транзакта
GENERATE ,,10,1,2
PREEMPT CPU,PR,met,10
ADVANCE 20 return cpu 7
```

Пример 1. Прерывание без дообслуживания

```
GENERATE 20,,,1
PREEMPT CPU
Mark 10
ADVANCE 40,10
REturn CPU
Tabulate Tab10
TRANsfer ,mmm
GENERATE 50,,10,,2
Preempt Cpu,PR,nnn,,Re
mark 20
Advance 20,5
Return Cpu
Tabulate Tab20
TRANsfer ,mmm
nnn savevalue xxx+,1
mmm Terminate
tab10 Table MP10,30,2,10
tab20 Table MP20,20,2,10
Generate 10000
Terminate 1
```

Пример 1. На входе процессора два типа задач с приоритетами 1 и 2. Определить количество решенных задач с каждым приоритетом и среднее время работы процессора.



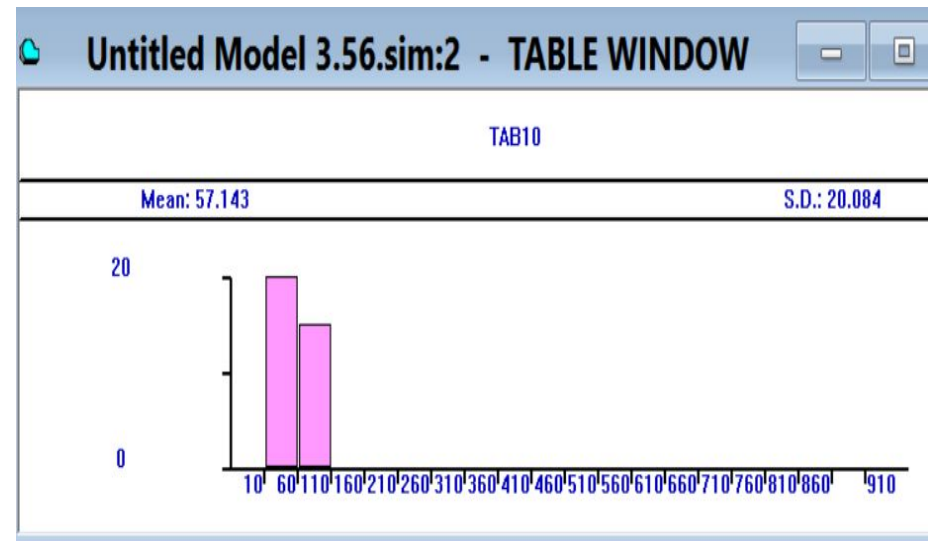
[Пример прерывания.doc](#)

Прерывание с дообслуживанием

```

GENERATE 20,10,,,1
  Assign 2,40
  Savevalue kkk+,1
qqq Queue  cpu
  PREEMPT CPU
  depart cpu
  Mark 10
  ADVANCE *2
  REturn CPU
  Tabulate Tab10
  TRANSfer ,mmm
  GENERATE 100,5,500,,2
  Assign 2,80
  Preempt Cpu,PR,Met,2
mark 20
  Advance *2
  Return Cpu
  Tabulate Tab20
  Transfer ,qqq
met return CPU
mmm Terminate
  
```

Пример 2. На входе процессора два типа задач с приоритетами 1 и 2. Определить количество решенных задач с каждым приоритетом и среднее время работы процессора.



```

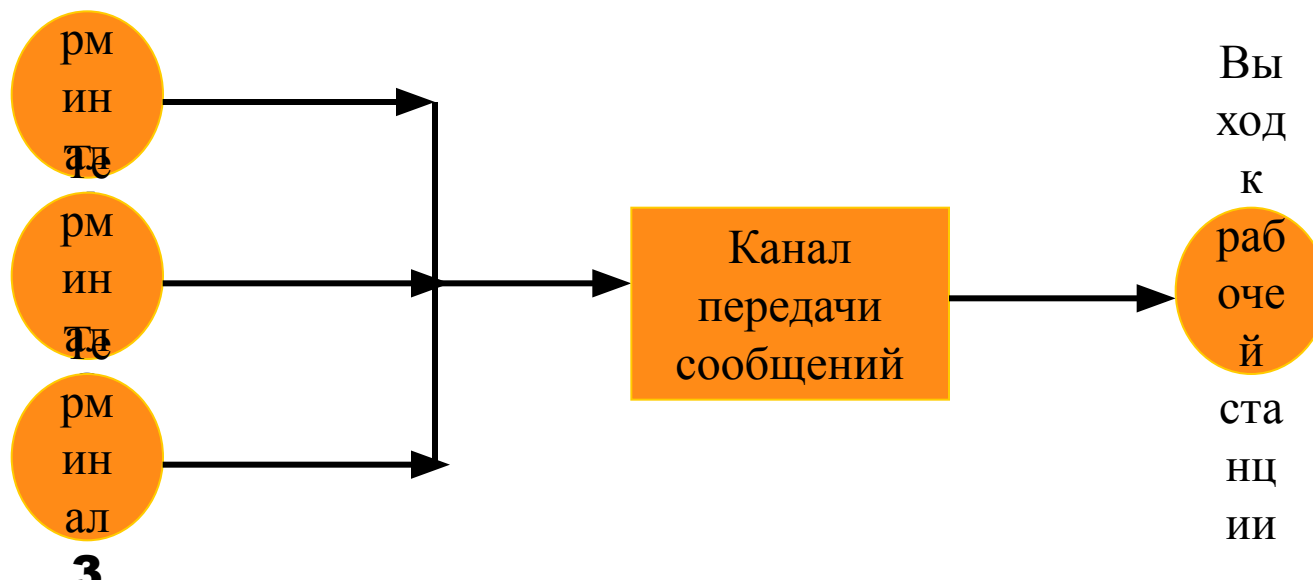
tab10 Table MP10,10,50,20
tab20 Table MP20,500,10,10
Generate 10000
Terminate 1
  
```

Пример использования приоритетов заявок

Задача. На рабочую станцию поступают сообщения с трех терминалов. Каждый терминал имеет приоритет, соответствующий номеру терминала. Поток сообщений описывается экспоненциальным законом. Сообщения приходят трех типов. Вероятность появления событий соответствующего времени обработки представлена в таблице:

0.2	0.25	0.55	Вероятность
32	64	128	Время обработки

Определить среднее время прохождения сообщений по каналу передачи сообщений, количество заявок, обработанных станцией от каждого терминала.



**FF1 Function RN1,D3
0.2,38/0.45,72/1.0,128**

Generate (Exponential(2,70,10))

Savevalue 10,c1

SAVEVALUE 10-,X20

SAVEVALUE 20,c1

TABULATE ttExp

ASSIGN 5,Fn\$FF1

QUEUE Qcan

Seize Can

DEPART Qcan

Mark 7

Advance p5

RELEASE Can

TABULATE ttcan

TERMINATE

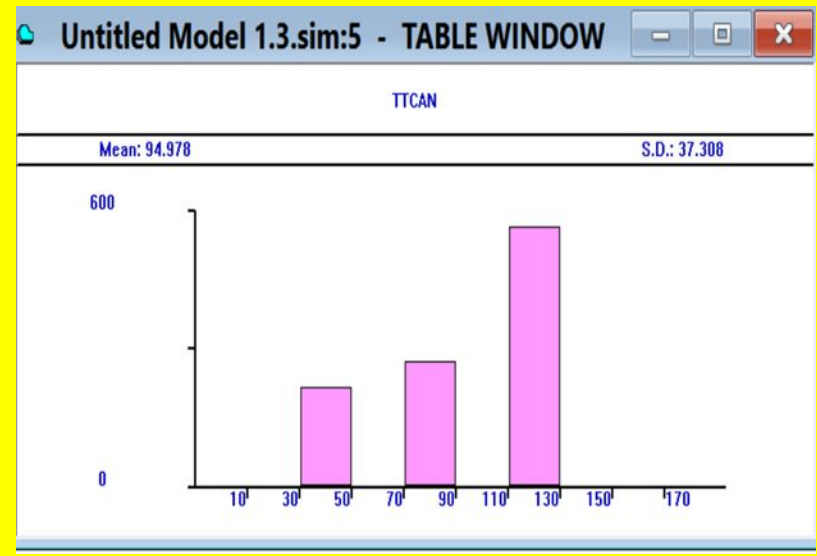
ttcan Table mp7,10,20,10

ttExp Table X10, 50,100,10

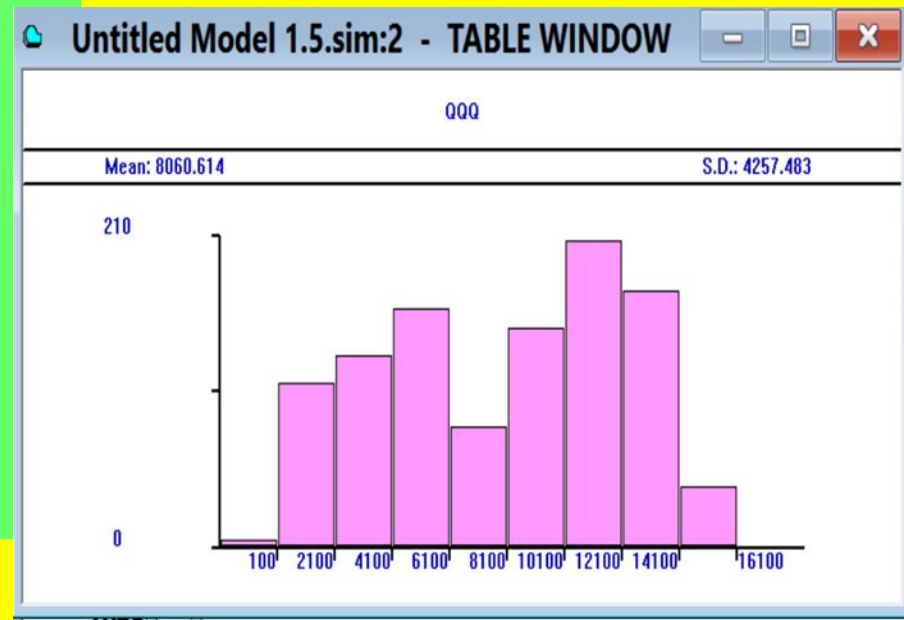
QQQ Qtable Qcan,500,1000,20

GENERATE 10000

TERMINATE 1



Времена передачи сообщений



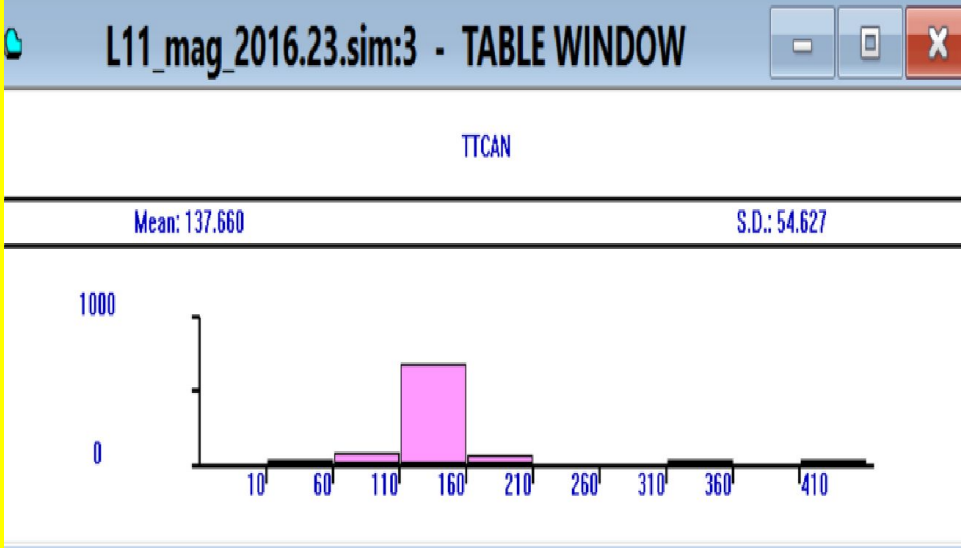
Формирование очереди

Модель станции

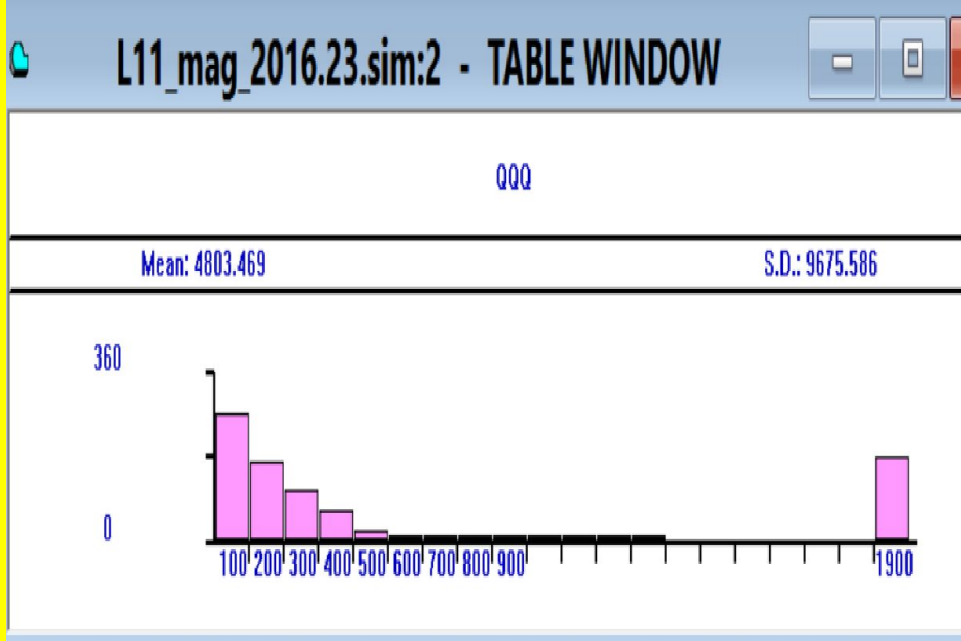
Пример

```

FF1 Function RN1,D3
0.2,38/0.45,72/1.0,128
Generate (Exponential(2,70,10))
    Savevalue 10,c1
    SAVEVALUE 10-,X20
    SAVEVALUE 20,c1
    TABULATE ttExp
        ASSIGN 5,Fn$FF1
    TEST E *5,38,met2
    PRIORITY 1
    savevalue xx1+,1
    TRANSFER ,met1
met2    TEST E *5,72,met3
    PRIORITY 2
    savevalue xx2+,1
    TRANSFER ,met1
met3    TEST E *5,128,met1
    PRIORITY 3
    savevalue xx3+,1
met1    QUEUE Qcan
    preempt Can,pr
    DEPART Qcan
    Mark 7
    Advance p5
    Return Can
    TABULATE ttcan
    TERMINATE
ttcan Table mp7,10,10,20
ttExp Table X10,60,5,20
QQQ Qtable Qcan,100,2000,10
GENERATE 100000 TERMINATE 1
    
```



Времена передачи сообщений



Формирование очереди

Условие задачи.

Базовая схема для моделирования состоит из N терминалов пользователя, одного канала передачи данных к серверу и сервера. Заявки на выполнение заданий поступают в интервале $[a, b]$ с указанным в варианте законом распределения.

На терминале заявки формируются за время $[k, v]$.

Заявки со всех терминалов поступают в канал передачи данных единичной ёмкости, но обрабатывающий приоритет заявок.

Если канал занят и очередь к каналу превышает L единиц, то заявки возвращаются на соответствующий терминал и пытаются снова пройти канал.

Время передачи заявок по каналу находится в диапазоне $[e, f]$.

Время обработки заданий на сервере определено в интервале $[c, d]$. Ёмкость сервера $2 * N$. Закон времён обработки сообщений задан в варианте задания.

После обработки заявки на сервере должно быть отправлено сообщение на соответствующий терминал. Приоритет ответа должен быть самым высоким из всех заявок. Определить длины очередей к каналу и серверу, среднее время обработки заявок, количество попыток повторной передачи заявок, количество необработанных заявок.

Условие

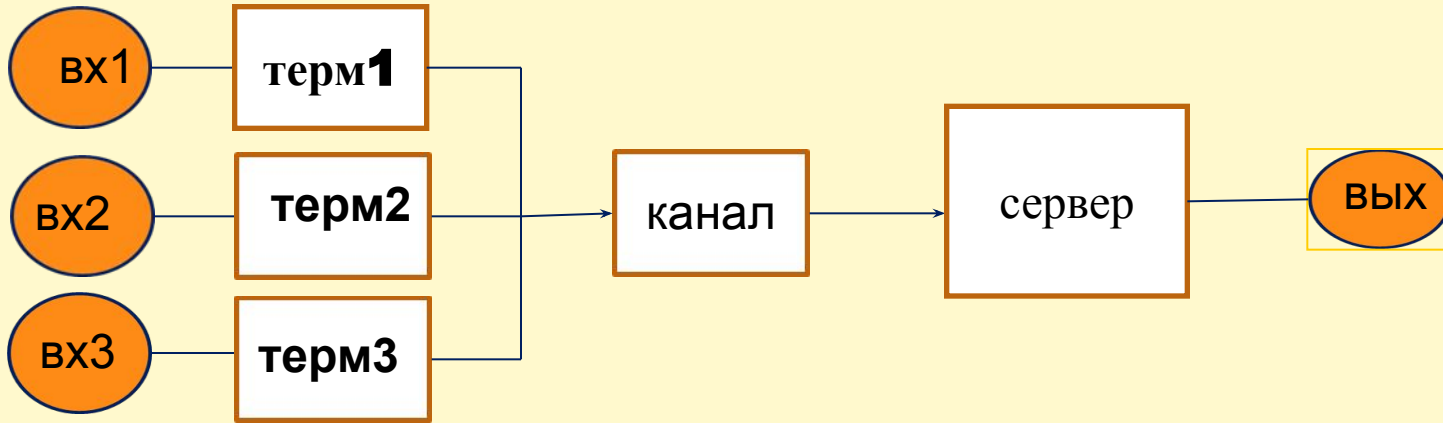


Схема модели

