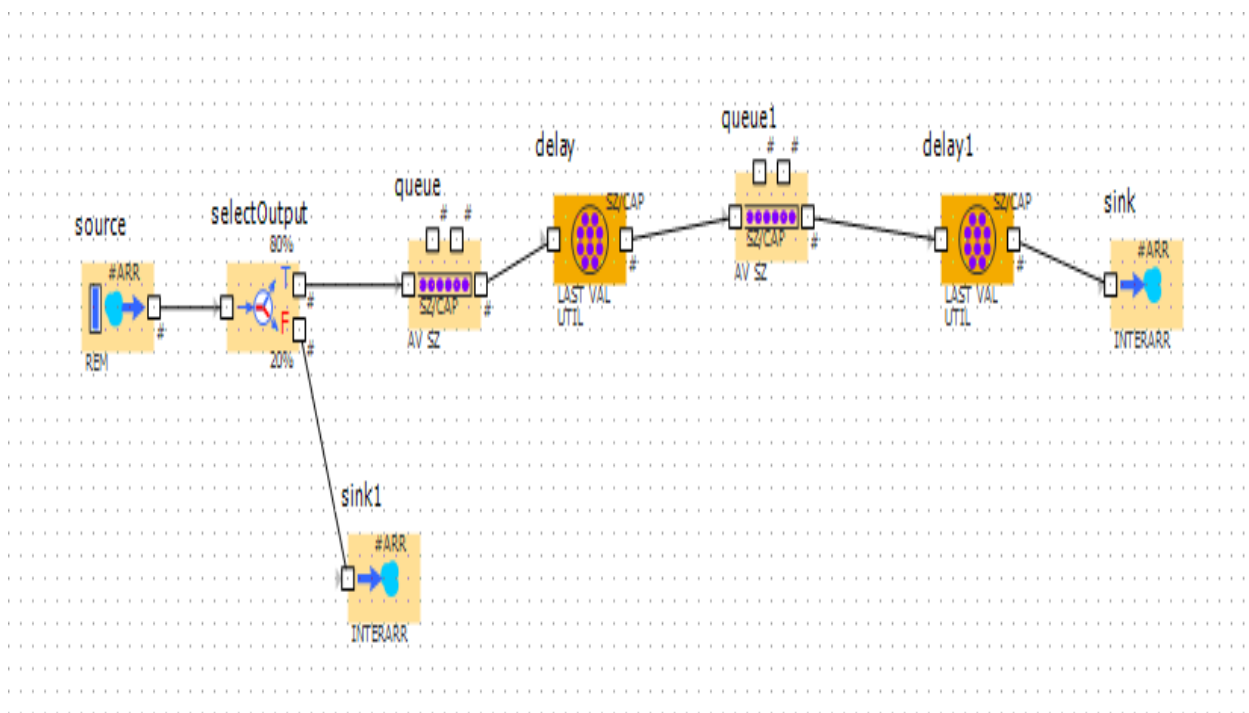


Задача 1.5. Измените условия задачи 1.4. Время ожидания распределено так: 20% заявок ждут 4 минуты, 30% - 5 минут и оставшиеся 50% - 6 минут.

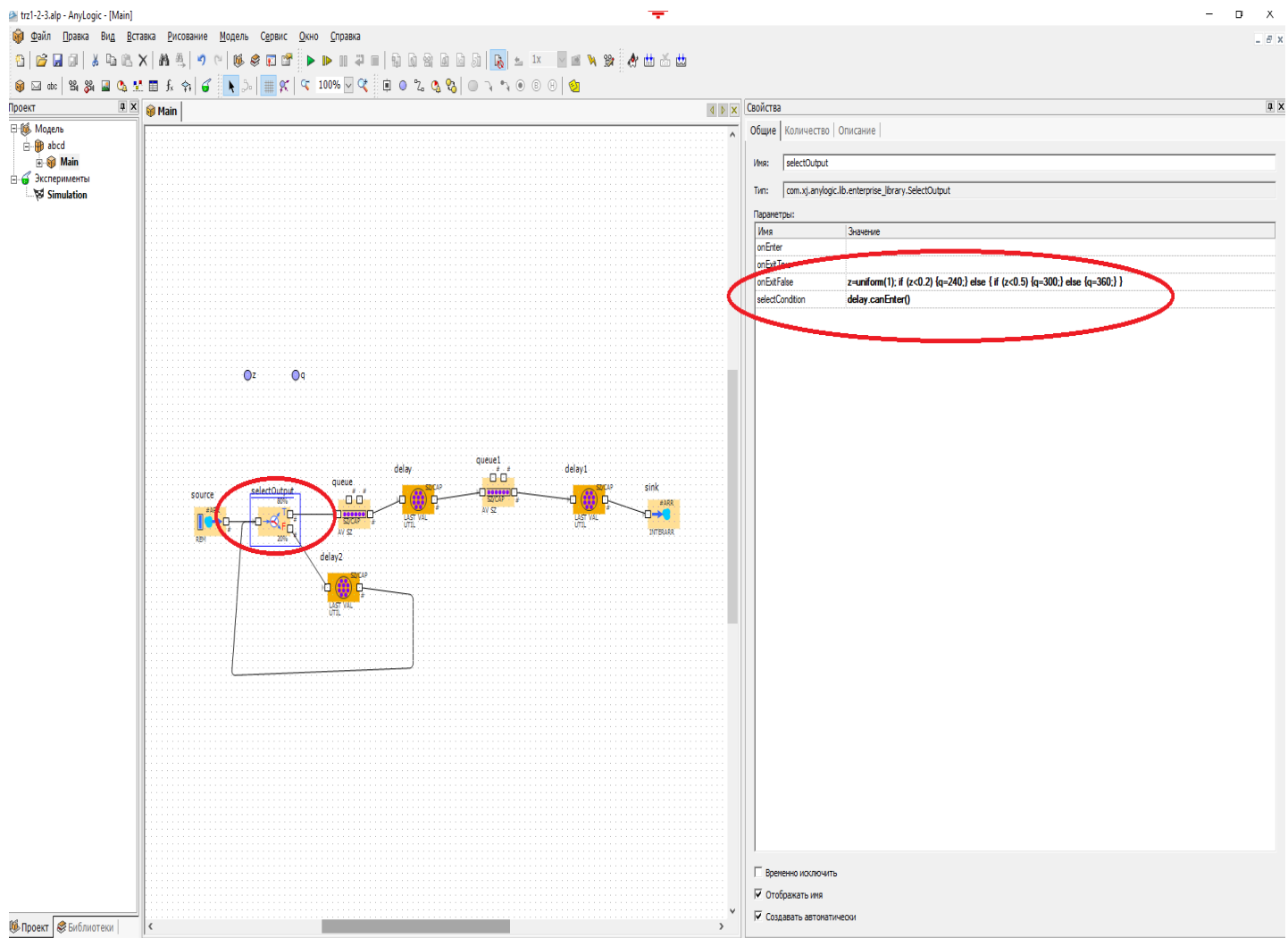
Схема остаётся как задаче 1.3, только теперь если первый прибор занят – заявка ждёт определённое время в зависимости в какой процентный промежуток она попадает, а не выводится из системы.

В Anylogic работа с процентами напрямую не удобна, поэтому воспользуемся ещё одним приёмом. Проценты очень похожи на другую математическую абстракцию – вероятности. 100% - это вероятность равная 1. 20% - вероятность в 0,2 и т.д.



Теперь ставим блок задержки заявки на определённое время (это блок delay) вместо sink, а так же нам понадобятся 2 новые переменные для процентных промежутков (переменная z) и для значения времени задержки (переменная q). А старые переменные можно удалить (но это не обязательно) вместе с выражениями, где они учувствуют, в том числе убрать Стоп по времени и дополнительное условие остановки в свойствах эксперимента.

В блоке selectOutput в поле selectCondition вставляем уже известное нам условие проверки доступности первого прибора. Если первый прибор готов принять заявку, то она пойдёт по верхнему выходу True.



А вот если прибор занят, то заявка идёт по нижнему выходу False в блок задержки delay2, и в поле onExitFalse мы должны вставить следующий код:

***$z=uniform(1); \text{if } (z<0.2) \{q=240;\} \text{ else } \{ \text{if } (z<0.5) \{q=300;\} \text{ else } \{q=360;\} \}$***

Сначала мы присваиваем переменной  $z$  результат выполнения функции `uniform(1)`, которая случайным образом выбирает число от 0 до 1. Это и есть имитация процентов с помощью вероятности. Далее ставим условие – если выпавшее нам число, записанное в  $z$ , меньше 0,2 (т.е. попадает в первые 20%), то мы присваиваем переменной  $q$  значение равное 240 (это 4 минуты в секундах). Иначе мы проверяем следующие 30%, т.е.  $z<0.5$  (0,5 взялось так – мы уже проверили 20%, это 0,2 плюс текущие 30% это 0,3). Если заявка попала в этот процентный промежуток, то время задержки мы ставим 300 секунд, иначе 360 секунд.

Осталось только поставить переменную  $q$  в поле `delayTime` блока `delay2`, так как именно в эту переменную мы записывали количество секунд, на которое нужно задержать заявку до того как она снова попытается занять прибор.

ts1-2-3alp - AnyLogic - [Main]

Файл Правка Вид Вставка Рисование Модель Сервис Окно Справка

Проект: Model  
Main  
Эксперименты  
Simulation

source selectOutput queue delay queue1 delay1 sink

delay2

Свойства

Общие	Количество	Описание
Имя:	delay2	
Тип:	com.anylogic.lib.enterprise_library.Delay	
Параметры:		
Имя	Значение	
onEnter		
onExit		
delayTime	q	
capacity		
statusEnabled	false	
animationShape		
animationType	AUTO	
animationForward	true	
schedule	without_schedule	

☐ Временно исключать  
☒ Отображать имя