

ПРОЕКТИРОВАНИЕ БЛОКА

«Потоковый арбитр с политикой Quality-of-Service (QoS)»

Выполнил

Ларченков К.Д.

Москва 2023

Описание разрабатываемого модуля

Модуль по нескольким параллельным потокам (количество потоков задается параметрически) принимает транзакции, а также значение приоритета каждого пакета транзакций. Из всех входных транзакций выбирается наиболее приоритетная, которая транслируется на выход модуля. Транзакции с равными значениями приоритета транслируются на выход по принципу Round-Robin. Значение приоритета транзакции, равное нулю, приравнивается к наибольшему значению приоритета. Арбитраж производится потранзакционно: переключение арбитра на следующий входной порт производится только после завершения передачи транзакции с текущего выбранного арбитром порта.

Описание интерфейса модуля

Помимо входа синхросигнала (clk) и синхронного сброса (rst_n), порты модуля разделяются на множество входных потоков и один выходной поток.

Каждый входной поток данных включает:

- s_data_i — шина данных транзакций;
- s_qos_i — шина значения приоритета текущего пакета транзакций;
- s_last_i — флаг завершения транзакции;
- s_valid_i — запрос от master-устройства на передачу 1 пакета данных;
- s_ready_o — сигнал о готовности принять 1 пакет данных, передаваемый master-устройству.

Выходной поток включает:

- m_data_o — шина данных транзакций;
- m_qos_o — шина значения приоритета текущего пакета транзакций;
- m_id_o — номер потока, с которого транслируется транзакция;
- m_last_o — флаг завершения транзакции;
- m_valid_o — запрос от модуля на передачу 1 пакета данных;
- m_ready_i — сигнал о готовности принять 1 пакет данных, передаваемый модулю.

Каждый писанный порт содержит в своем названии направление передаваемых данных (i — input, o — output).

Описание микроархитектуры модуля

Так как в данном модуле не предполагается буферизация данных, предлагается использовать комбинационную логику. Каждый узел предлагается реализовать в виде отдельного подмодуля, чтобы сохранить читаемость схемы и кода на выбранном языке описания аппаратуры. Помимо стандартных комбинационных узлов (в том числе и многоразрядных) необходимо описать узел сравнения приоритетов (на Рисунке 1 обозначен как СМР), экземпляры которого необходимо соединить последовательно в количестве $\text{STREAM_COUNT} - 1$.

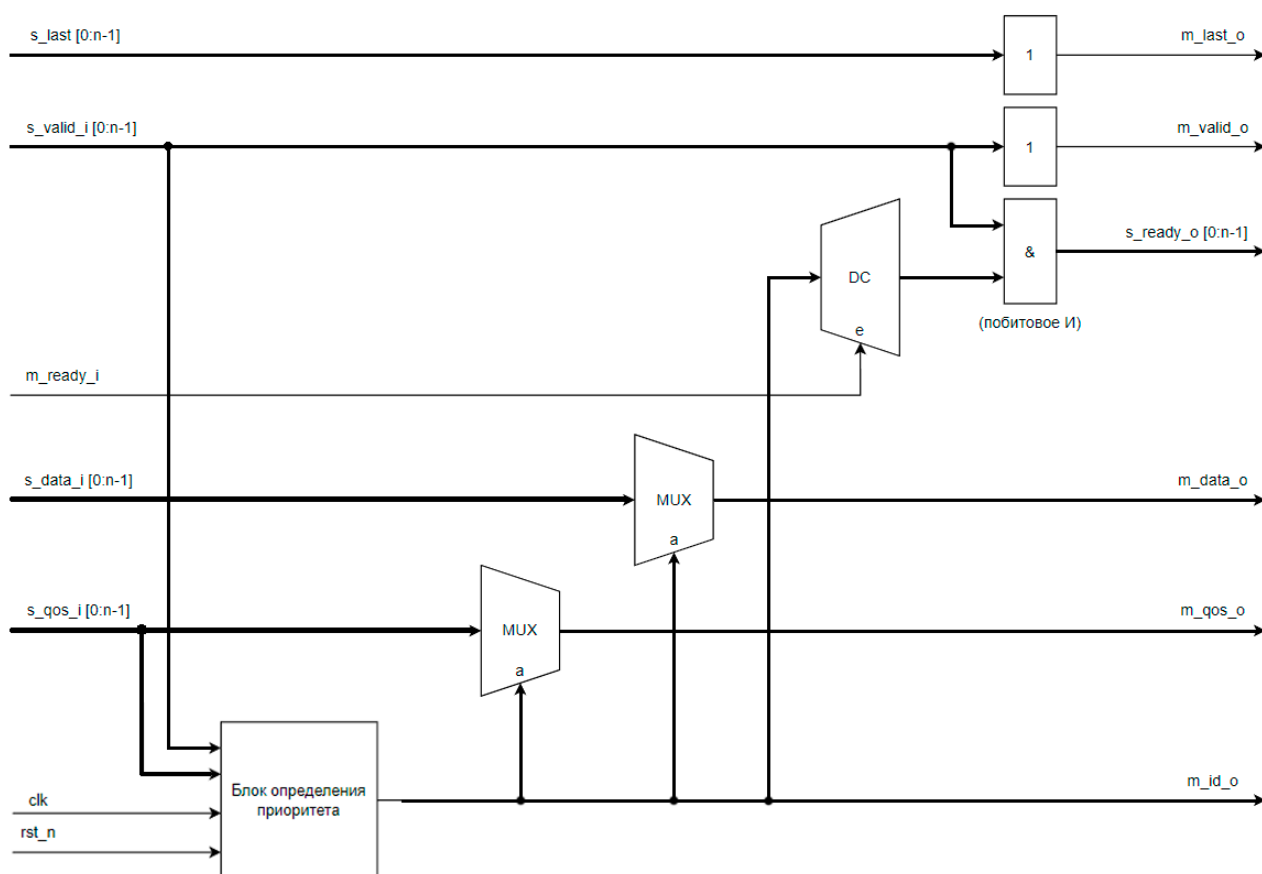


Рисунок 1 — Структурная диаграмма модуля

Остановимся подробнее на каждом блоке модуля.

1. Блок выбора потока

Данный блок представляет собой компаратор, сравнивающий `STREAM_COUNT` операндов и выдающий на выходе номер первого наибольшего (или равного нулю) операнда при активном соответствующему ему сигнале `valid`. Проще всего реализовать данный подмодуль, используя цикл с параметром.

На вход данного подмодуля необходимо подать значения `QoS` и `valid` со всех входных потоков, синхросигнал и сигнал синхронного сброса, а выходное значение будет соответствовать номеру приоритетного потока (`m_id_o`).

2. Блок выбора значения приоритета

Данный блок предлагается реализовать при помощи параметрического мультиплексора с многоразрядными входами данных. На входы данных подаются значения приоритетов со всех входных потоков, а на адресный вход — номер приоритетного потока.

3. Блок выбора данных

Данный блок предлагается реализовать по аналогии с предыдущим, но на входы данных мультиплексора необходимо подать данные транзакций со всех входных потоков.

4. Блок определения сигнала готовности принять пакет

В данной реализации предлагается реализовать блок определения сигнала готовности принять пакет данных при помощи дешифратора с количеством выходов, равному `STREAM_COUNT`, и узла побитовой конъюнкции. На информационный вход дешифратора подается значение `m_id_o`, а на разрешающий вход — `m_ready_i`. Значения с выходов дешифратора подаются на один из входов узла побитовой конъюнкции, а на второй вход подается значение `s_valid_i`. Таким образом, если `master`-устройство готово принять пакет данных,

а slave-устройство готово его предоставить, значение `s_ready_o` выбранного потока становится равным 1.

5. Блок определения сигнала готовности передать пакет

В данной реализации модуля предполагается, что устройство готово передать пакет в тот момент, когда slave-устройство готово передать его на один из входных потоков. Таким образом, сигнал `m_valid_o` формируется как результат дизъюнкции всех битов `s_valid_i`.

6. Блок определения флага завершения транзакции

В данной реализации модуля предполагается, что флаг завершения транзакции на выходе устройства поднимается в момент поднятия аналогичного флага на одном из входных потоков. Таким образом, сигнал `m_last_o` формируется как результат дизъюнкции всех битов `s_valid_i`.