

План поведения работ в Logisly на ФРКТ-3

1. Занятие 1. Комбинаторная логика.

1. Реализовать терм-конъюнктор $y = x'_1x_2x'_3x_4$, опознающий комбинацию (0101). Вывести его таблицу истинности. Преобразовать его в теом-дизъюнктор $y = x_1 + x'_2 + x_3 + x'_4$.

2. Реализовать сумматор по модулю два в нормальной конъюнктивной форме по схеме $z = x \oplus y = xy' + x'y$ и в нормальной дизъюнктивной форме по схеме $x = x \oplus y = (x + y)(x' + y')$

3. Реализовать его же на четырех вентилях NAND по схеме

$$z = x \oplus y = (((xy)'x)'((xy)'y)')'$$

Заменить вентили NAND на NOR

$$z = (((x + y)' + x)' + ((x + y)' + y)')'$$

Составить таблицу истинности того, что получилось.

4. Реализовать мажоритарный элемент

$$MAJ(x, y, z) = x'yz + xy'z + xyz' + xyz$$

в минимизированной форме ($MAJ(x, y, z) = xy + xz + yz$) и в форме с сумматором по модулю два ($MAJ(x, y, z) = (x \oplus y)z + xy$).

5. По модели COMB/dx изучить структуру и работу 1 и 2-разрядных дешифраторов. Взяв за основу 2-разрядный дешифратор (библиотечный элемент), реализуем а) 4-разрядный дешифратор и б) 4-входовый мультиплексор.

6. По модели Math/INC изучить работу и структуру инкременторов со сквозным и параллельным переносами и универсального инкрементора/декрементора на полусумматорах. Взяв за основу внутренность блока INC4 (Copy to Clipboard - Paste), превратить его в декриментор.

7. По модели Math/ADD1 изучите 1-разрядные сумматоры/вычитатели. Самостоятельно реализуйте двухрежимный сумматор/вычитатель, воспользовавшись блоком мультиплексора MUX2 из библиотеки. Изучите работу 4-разрядных Math/INC сумматора и вычитателя по модели Math/ADD4.

8. Изучите работу сумматора с ускоренным переносом - Math/FADD. По модели Math/ADD2n разберитесь с принципом реализации быстрых сумматоров по схеме удвоения. Реализуйте 8-разрядный сумматор из двух 4-разрядных (библиотечный блок FADD4).

2. Занятие 2. Триггеры.

1. По модели FFs/rs разберитесь с работой триггера на двух вентилях NAND. Самостоятельно реализуйте триггер на двух NOR. По модели FFs/rsen разберитесь со организацией RS-триггеров со входом разрешения en. Реализуйте аналогичную структуру на 4 вентилях NOR и разберитесь с ее работой.

2. По модели FFs/latch разберитесь с работой триггера-защелки latch. Попробуйте подать на d-вход защелки инверсный сигнал с выхода. Убедитесь в возникновении самовозбуждения при активном уровне en.

3. Изучите работу двухтактного RS-триггера (FFs/rs2t) и двухтактного d-триггера (FFs/d2t). Подав на d-вход инверсный сигнал q' с выхода организуйте работу в счетном режиме.

4. По модели FFs/dlt изучите устройство и работу однофазного d-триггера. По модели FFs/d2tclr изучите схему однофазного d-триггера со входом асинхронного сброса в ноль. Самостоятельно добавьте в нее вход установки в единицу.

3. Занятие 3. Регистровая логика.

1. По модели Regs/sr4 проанализируйте структуру сдвигового регистра. Самостоятельно измените связи с тем, чтобы изменить направление сдвига. Изучите схему и работу реверсивного регистра со режимом синхронной начальной установки (Regs/srlr). Превратите его в генератор псевдослучайной последовательности, подав на вход SLI сигнал $y = q_3 \oplus q_2$. Сумматор по модулю два взять в библиотеке.

2. По модели Regs/cnt ознакомьтесь с организацией инкрементирующего и декрементирующего счетчиков. Подумайте, как совместить оба эти режима в одной схеме и проверьте свои домыслы по модели Regs/cntud.

3. По модели Regs/tff изучите устройство и работу счетного Т-триггера. Изучите принципы организации счетчиков со сквозным и параллельным переносами на Т-триггерах. Взяв за основу внутренность счетчика со сквозным переносом, превратить его в вычитающий. Реализуйте реверсивный счетчик (нужный для этого мультиплексор MUX имеется в библиотеке).

4. По модели Regs/cntm уясните реализацию декадного счетчика (по модулю $M = 10$). Переделайте его в счетчик по модулю $M = 7$, $M = 13$.

5. По модели MATH/COMP уяснить, как 4-разрядный компаратор CMP4 построен из 4-х одноразрядных CMP1. Перенеся блок компаратора в модель Regs/cntm, построить 4-разрядный счетчик по произвольно задаваемому модулю M .