

## 7. Разработка программного кода модели.

### 7.1. Общее описание.

Программный код модели написан на языке описания аппаратуры VERILOG и отлажен в среде Active-HDL 6.1.

Модель цифровой части радиочастотной метки представляет собой основной блок DigitalPart), включающий в себя несколько подблоков (см. раздел 3).

Рассмотрим подробнее модель каждого из них.

#### 7.1.1. Блок тактирования и запуска инициализации (ClockBlock).

Входные сигналы:

- SubCarrier – поднесущая.

Выходные сигналы:

- Initialize – флаг запуска инициализации;
- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- Clk8 – поднесущая.

Регистры:

- ClkReg[2..0] - счётчик делителя частоты;
- InitReg[5..0] - счётчик для удержания флага инициализации;

Алгоритм работы модуля.

Clk1 – представляет собой старший бит регистра ClkReg.

После подачи питания на метку, начинается тактирование счётчика InitReg поднесущей. При этом флаг Initialize удерживается в положении «1». Это происходит до тех пор, пока InitReg не досчитает до 101010. Затем начинает работать счётчик ClkReg.

#### 7.1.2. Блок генератора случайных бит (RandomizeBlock).

Входные сигналы:

- RndStart - флаг запуска рандомизации;
- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- Clk8 – поднесущая;
- Initialize – флаг запуска инициализации;
- Initializing - флаг проведения инициализации;
- InitData - начальные данные для инициализации генератора;
- RndMix - случайные данные от аналоговой части, вмешиваемые в генератор.

Выходные сигналы:

- RndData - выходные случайные битовые данные.

Регистры:

- RndData - регистр с выходным битом;
- Rnd[15..0] - главный сдвиговый регистр.

#### Алгоритм работы модуля.

Во время запуска инициализации метки (флаг Initialize), регистр rnd заполняется начальными данными: АААА'16.

После того, как метка будет инициализирована, генератор случайных чисел будет работать следующим образом.

На отрицательном фронте сигнала Clk1 будет происходить сдвиг регистра rnd, при этом в его младший триггер будет записано значение согласно функции:  $\text{rnd}[0] = \text{rnd}[15] \wedge \text{rnd}[13] \wedge \text{rnd}[12] \wedge \text{rnd}[10] \wedge (\text{InitData} \wedge \text{Initializing} \wedge \text{RndMix})$ .

При положительном фронте сигнала Clk1, текущее значение триггера rnd[0] будет записано в регистр RndData.

### **7.1.3. Блок работы с замком (RWM\_Block).**

Входные сигналы:

- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- NewLockState - шина, передающая код нового состояния замка;
- ChangeLockState - флаг смены состояния замка.

Выходные сигналы:

- UnLocked - выходной бит состояния замка.

Регистры:

- State[3:0] - 4-разрядная ячейка памяти, хранящая код текущего состояния замка.

#### Алгоритм работы модуля.

Возможны два разрешённых состояния ячейки памяти:

А) 1010 – метка открыта;

Б) 0101 – метка закрыта.

При поднятом флаге ChangeLockState, по отрицательному перепаду сигнала Clk1 происходит смена состояния замка на состояние, закодированное на шине NewLockState.

### **7.1.4. Блок вычисления хеш-функции (HashBlock).**

Входные сигналы:

- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- GenHash - флаг запуска генерации хеш-значения;

- ResetHash - флаг приведения генератора в исходное положение;
- RndData - входной бит от генератора случайных чисел;
- KeyData - текущий бит ключа;
- MixHashRndKey - флаг разрешения замешивания случайного бита и бита ключа.

Выходные сигналы:

- HashData - выходной бит хеш-функции.

Регистры:

- HashData - регистр выходного значения;
- Hashreg[15:0] - главный регистр хеш-функции.

#### Алгоритм работы модуля.

Если поднят флаг GenHash, то при отрицательном перепаде на линии Clk1, при поднятом флаге ResetHash происходит приведение регистра hashreg к начальным установкам: {101010101010101, (RndData^KeyData) & MixHashRndKey}; если этот флаг опущен, то происходит сдвиг регистра hashreg влево и запись в его младший триггер следующего значения: hashreg[15]^hashreg[13]^hashreg[12]^hashreg[10]^((RndData^KeyData) & MixHashRndKey).

При положительном перепаде на линии Clk1, значение триггера hashreg[0] запиывается в регистр HashData.

#### **7.1.5. Блок управления (ControlBlock).**

Входные сигналы:

- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- Clk8 – поднесущая;
- InReady - сигнал готовности входных данных от считывателя;
- InCode - код команды считывателя;
- InData - бит данных от считывателя;
- RndData - выходные случайные битовые данные;
- MemData - бит данных памяти;
- Initialize - флаг запуска инициализации;
- HashData - бит выходного значения хеш-функции;
- UnLocked - бит состояния замка.

Выходные сигналы:

- OutReady - флаг готовности выходных данных метки;
- OutData - выходные данные метки;
- RndStart - флаг запуска генератора случайных чисел;
- MemAddress - шина адреса памяти;
- Initializing - флаг проведения инициализации;

- GenHash - флаг запуска генерации хеш-значения;
- ResetHash - флаг переинициализации хеша;
- MixHashRndKey - флаг замешивания случайного бита и бита ключа в хеш;
- NewLockState - шина с новым кодом состояния замка;
- ChangeLockState - флаг смены состояния замка.

#### Регистры:

- Initializing - флаг проведения инициализации;
- Sleeping - регистр состояния "сон";
- Anticolliding - регистр состояния "антиколлизия";
- AnticollisionPaused - регистр состояния "пауза" при антиколлизии;
- ReceivingID - регистр состояния "приём ИН";
- ContinueChecking - регистр состояния "приём и проверка позиции паузы при антиколлизии";
- OperationCode[2..0] - код выполняемой над меткой операции;
- OutCounter[6..0] - счётчик выходных бит;
- MessagesCounter[8..0] - счётчик выходных посылок;
- NewLockState[3..0] - регистр с новым кодом состояния для замка;
- SentBit - регистр с последним отосланным битом;
- OutFromMem - флаг выдачи на выход бита памяти;
- GenHash - флаг запуска генерации хеш-значения;
- ResetHash - флаг переинициализации хеша;
- MixHashRndKey - флаг замешивания случайного бита и бита ключа в хеш;
- ChangeLockState - флаг смены состояния замка;
- MessagesCounterLoaded - признак загрузки счётчика выходных посылок;
- OutCounterLoaded - признак загрузки счётчика выходных бит;
- CorrectID - флаг корректности принимаемого от считывателя ИН при работе с замком;
- OutRd - дополнительный флаг готовности выходных данных;
- AnticollisionMatch - флаг корректности передаваемой считывателем позиции продолжения антиколлизии.

#### Алгоритм работы модуля.

Основные вычисления, производимые в данном модуле происходят при положительном перепаде на линии Clk1.

При поднятии флага `Initializing` происходит приведение основных регистров к начальным установкам:

```
OutCounter <= 64;
Anticolliding <= 0;
OutFromMem <= 1;
AnticollisionPaused <= 0;
MessagesCounterLoaded <= 0;
OutCounterLoaded <= 1;
Sleeping <= 0;
GenHash <= 0;
ResetHash <= 0;
MixHashRndKey <= 1;
NewLockState <= 0;
ChangeLockState <= 0;
ReceivingID <= 0;
CorrectID <= 1;
OutRd <= 1;
ContinueChecking <= 0;
AnticollisionMatch <= 1.
```

Если этот флаг не поднят, при поднятии флага `InReady` происходит приём данных из блока `InputBlock`, инициализация процесса их обработки и передачи на блок `OutputBlock` в соответствии с кодом полученной от считывателя команды.

Также, при опущенном флаге `Initializing`, но поднятом флаге `OutCounterLoaded` происходит промежуточная обработка и передача данных на `OutputBlock` в соответствии с кодом полученной от считывателя команды.

По окончании выполнения команды, когда нужные вычисления выполнены, нужные состояния переключены и нужные данные переданы на `OutputBlock`, флаг `OutCounterLoaded` опускается.

Для передачи данных на блок `OutputBlock`, блоком управления на один такт `Clk1` на линию `OutData` подаётся бит выходных данных и поднимается флаг `OutReady`.

При положительном перепаде на линии `Clk8` при поднятом флаге `Initialize`, поднимается флаг `Initializing` и остаётся поднятым вплоть до завершения процесса инициализации.

При поднятом флаге `OutReady`, при отрицательном перепаде на линии `Clk1`, происходит запись передаваемого бита данных с линии `OutData` в регистр `SentBit`, применяемый для хранения последнего отправленного бита данных.

### **7.1.6. Блок преобразования входной последовательности (InputBlock).**

Входные сигналы:

- InStream - поток входных данных;
- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- DbgInReady - отладочный сигнал готовности входных данных;
- DbgInCode - отладочная шина с кодом команды считывателя;
- DbgInData - отладочный сигнал с битом входных данных от считывателя;

Выходные сигналы:

- InReady - сигнал готовности входных данных;
- InCode - шина с кодом команды считывателя;
- InData - сигнал с битом входных данных от считывателя.

#### Алгоритм работы модуля.

Данный модуль производит извлечение кода команды и входного бита данных их входного потока демодулированных данных от считывателя, а так же, в зависимости от алгоритма передачи, может выполнять проверку их корректности.

### **7.1.7. Блок генерации выходной последовательности (OutputBlock).**

Входные сигналы:

- Initializing - признак проведения инициализации;
- Clk1 - тактирование вычислений - в 8 раз меньше частоты поднесущей;
- Clk8 – поднесущая;
- OutReady - признак готовности выходных данных;
- OutData - шина с выходным битом данных.

Выходные сигналы:

- dbgOutReady - отладочный признак готовности выходных данных;
- dbgOutData - отладочная шина с выходным битом данных;
- OutStream - выходной поток информации.

Регистры:

- RgOut - регистр с выходным битом;
- SndOut - флаг, открывающий выходной канал.

### Алгоритм работы модуля.

В зависимости от того, поднят ли флаг Initializing, работа данного модуля тактируется либо сигналом Clk8(поднят), либо Clk1.

При отрицательном перепаде тактового импульса, если поднят флаг Initializing, то происходит инициализация регистра SndOut значением «0», в противном случае происходит запись в этот регистр значения флага OutReady, а в регистр RgOut – бита выходных данных OutData.

При поднятом флаге SndOut, в зависимости от значения регистра RgOut, на линию OutStream подаётся либо логическое перемножение сигналов Clk1 и Clk8, либо !Clk1 и Clk8.

Если флаг SndOut опущен, то OutStream замкнут на землю.

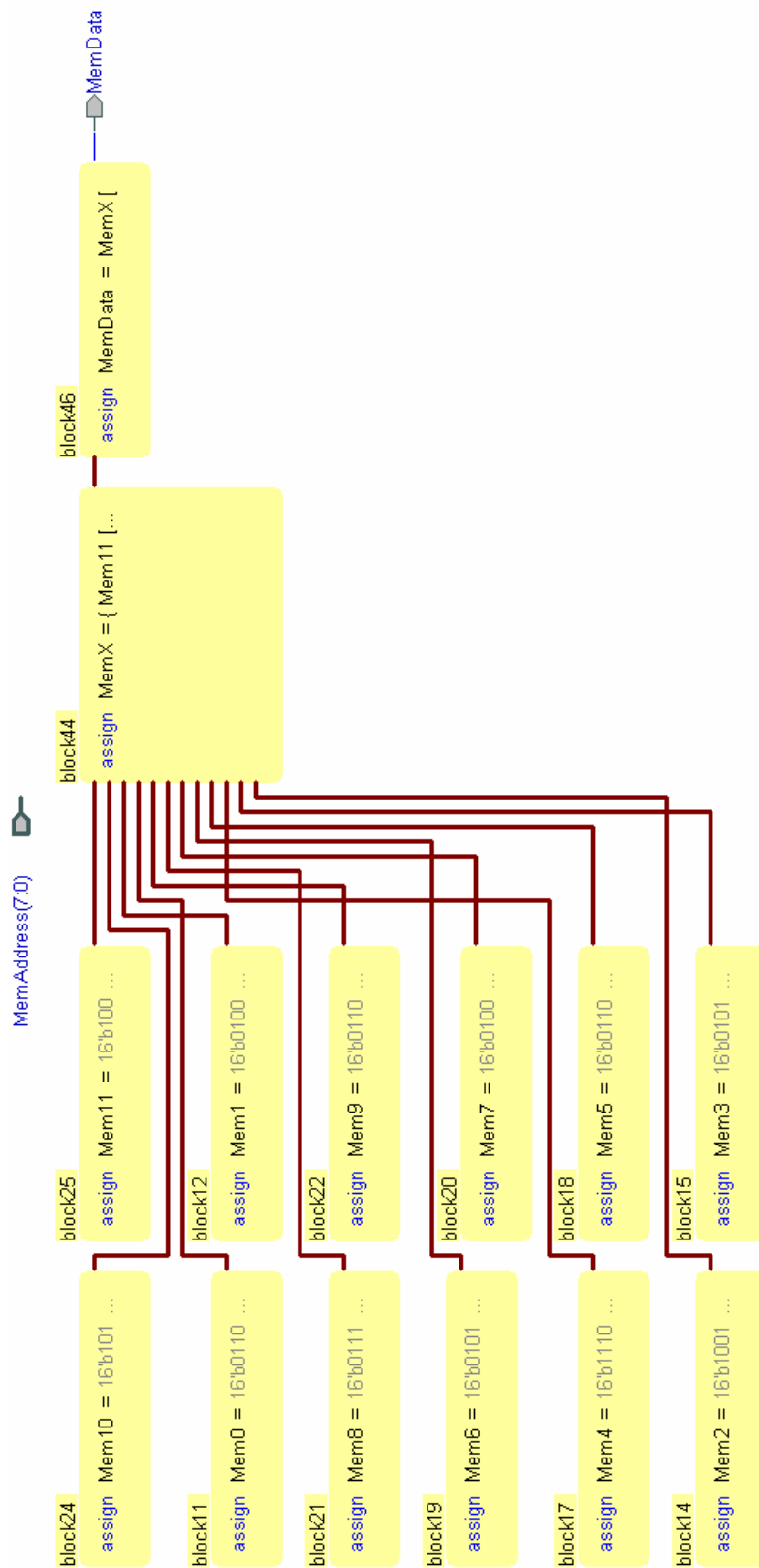
## **7.2. Блок-схемы основного модуля и подмодулей.**

Данные блок-схемы получены при помощи средства “Code2Graphics Conversion Wizard” среды Active-HDL 6.1.



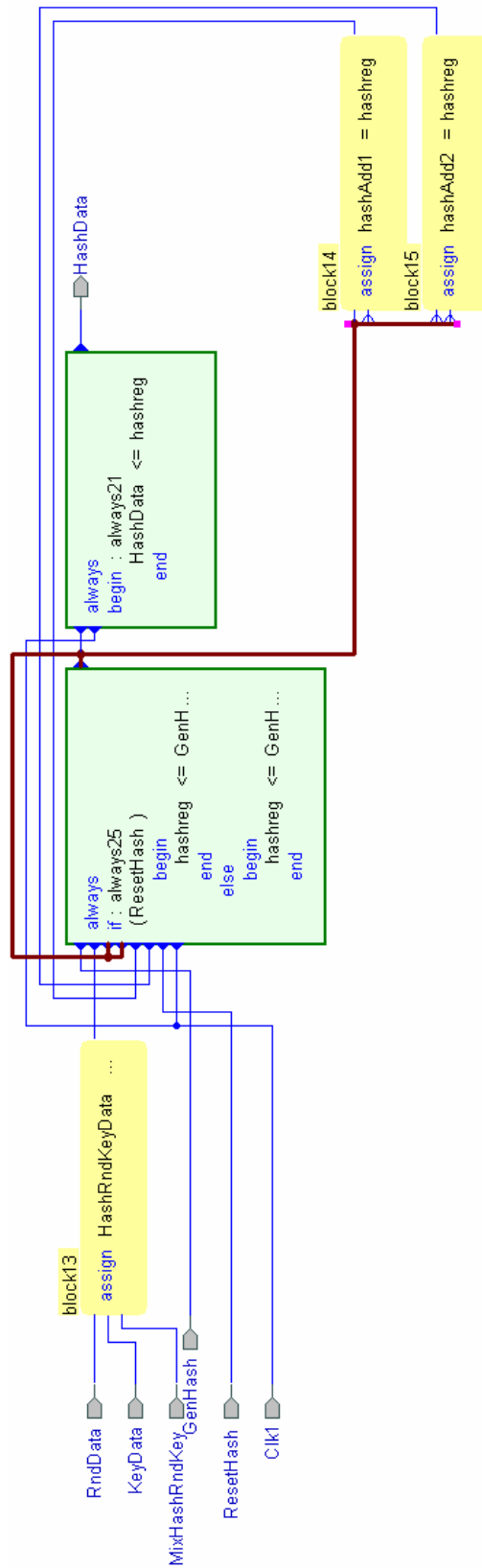




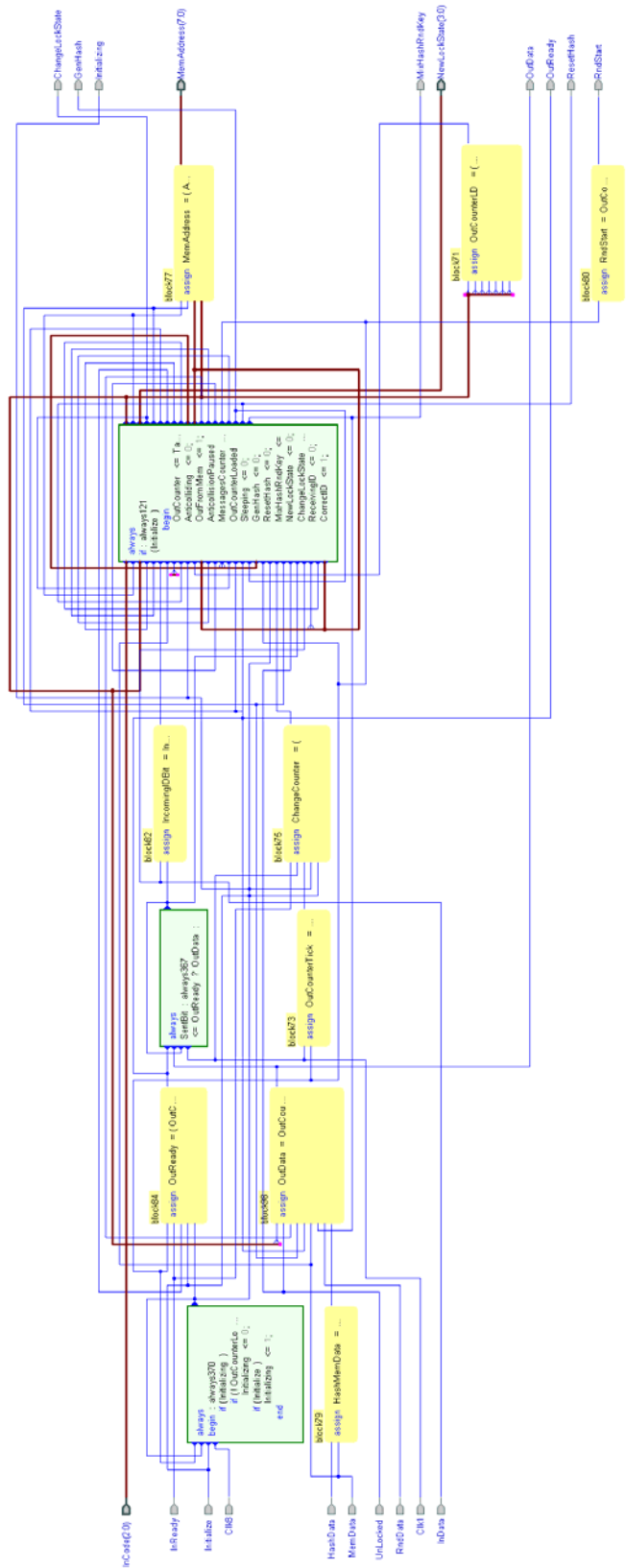


Изм.	Лист	№ докум.	Подп.	Дата	Модуль ROM_Block. Блочная диаграмма.	Лит.	Масса	Масштаб
Разраб.		Лайков Ю.М.						
Пров.		Широ Г.Э.						
						Лист 1	Листов 1	





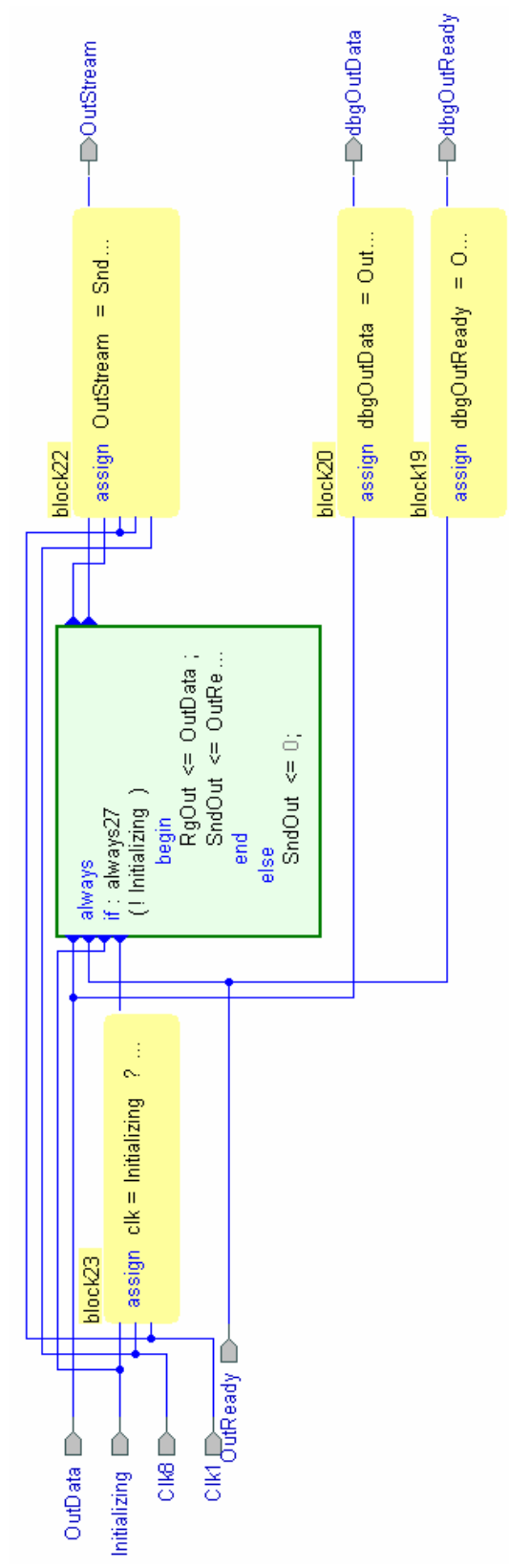
Изм.	Лист	№ докум.	Подп.	Дата	Модуль HashBlock. Блочная диаграмма.	Лит.	Масса	Масштаб
Разраб.		Лайков Ю.М.						
Пров.		Широ Г.Э.						
						Лист 1	Листов 1	



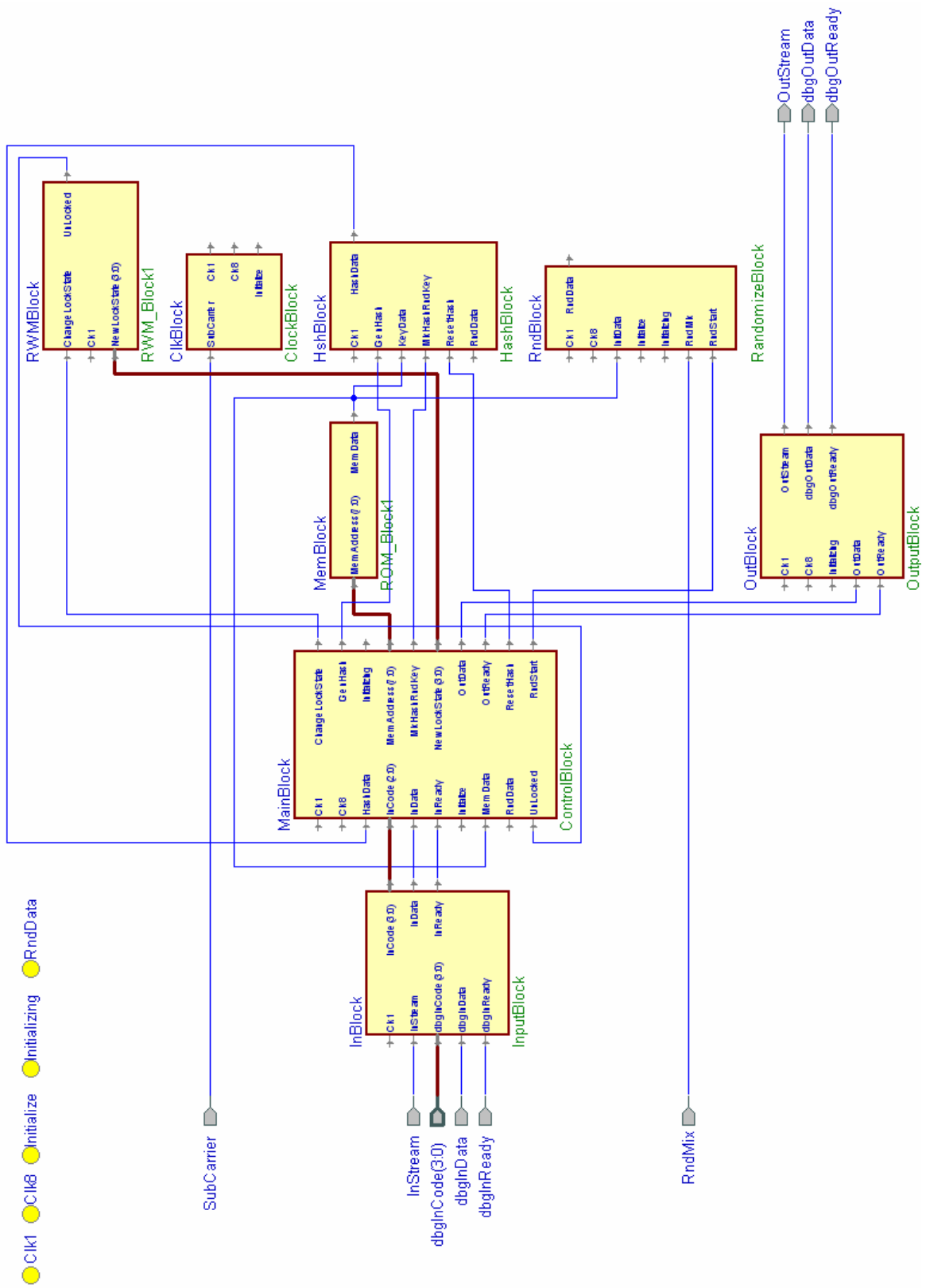
Изм.	Лист	№ докум.	Подп.	Дата
Разраб.	Лайков Ю.М.			
Пров.	Широ Г.Э.			

Модуль ControlBlock.  
Блочная диаграмма.

Лит.	Масса	Масштаб
Лист 1	Листов 1	



Изм.	Лист	№ докум.	Подп.	Дата	Модуль Output.Block Блочная диаграмма.	Лит.	Масса	Масштаб
Разраб.		Лайков Ю.М.						
Пров.		Широ Г.Э.						
						Лист 1	Листов 1	



Изм.	Лист	№ докум.	Подп.	Дата	Модуль DigitalPart. Блочная диаграмма.	Лит.	Масса	Масштаб
Разраб.	Лайков Ю.М.							
Пров.	Широ Г.Э.							
						Лист 1	Листов 1	