

4. Tissot J. L., Rothan F., Vedel C. et al.//Proceedings of SPffi. 1998. V. 3379. P. 139.

5. Unewisse M. H., Possmore S. J., Liddiard K. C., Watson R. J.// Ibid. 1994. V. 2269. P. 43.

Статья поступила в редакцию 11 октября 2006 г.

## Investigation of film $\alpha$ -Si and application there in technology of matrix silicon MOS multiplexers and bolometers

V. K. Borisov, E. A. Klimanov, N. B. Kagan, I. U. Kjurbet, V. P. Liseykin  
Orion Research-and-Production Association, Moscow, Russia

*In this paper is adduced investigation on films  $\alpha$ -Si made by magnetron evaporation of silicon In vacuum and comparison with CVD polysiticon is given. It is shown, the  $\alpha$ -Si films after diffusion and treatments have best perfection of structure. Producing multiplexers 128×128 and 384×256 on basis MOS transistors with  $\alpha$ -Si gates have better issue. System V— $\alpha$ Si, made by magnetron evaporation is examined in the capacity of contacts to highly resistance films a-Si.*

УДК 537.533.3:681.7.015.2

## Способы и принципы построения алгоритмов ЦОС многоэлементного фотоприемного устройства ИК-диапазона на основе микросхемы программируемой логики

И. И. Кремис, Ю. Ф. Однолько

Конструкторско-технологический институт прикладной микроэлектроники СО РАН,  
г. Новосибирск, Россия

*Рассмотрены способы и принципы реализации алгоритмов обработки сигналов многоэлементного фотоприемного устройства ИК-диапазона посредством использования жесткой логики программируемой логической интегральной микросхемы (ПЛИС). Представлены схемотехнические решения построения алгоритмов. Приведены результаты практического использования реализованных алгоритмов. Алгоритмы цифровой обработки сигналов реализованы на микросхеме программируемой логики серии Cyclone фирмы Altera.*

Программируемая логическая интегральная микросхема высокого уровня интеграции позволяет создать систему, основные функциональные узлы которой выполнены на едином кристалле микросхемы [1—4]. Использование данного принципа построения системы позволяет повысить ее надежность, снизить ее массогабаритные показатели и показатели энергопотребления. В этом случае при построении системы обработки сигналов на ПЛИС возникают трудности создания узлов цифровой обработки сигналов (ЦОС) на элементах жесткой логики. В то же время ПЛИС высокого уровня интеграции позволяет реализовать любой программный алгоритм, обычно выполняемый DSP [1—4]. Ограничение на самостоятельную разработку ядра про-

цессора обработки сигналов для ПЛИС или ядра, вычисляющего сложный алгоритм обработки сигнала, нецелесообразно ввиду большой трудоемкости этих задач. Причем средства для их решения предоставляются самой фирмой-производителем в виде soft-ядер процессоров и соответствующего специализированного программного обеспечения [1, 3]. Использование готовых программных soft-ядер процессоров ограничивается необходимостью покупки соответствующего программного обеспечения и мегафункций soft-процессоров. Напротив, алгоритмы, не требующие сложного анализа данных, достаточно просто реализовать, используя жесткую логику ПЛИС с применением общедоступного программного обеспечения Quartus.

В данной работе рассматриваются способы и принципы построения алгоритмов ЦОС посредством жесткой логики, реализованные при разработке в Конструкторско-технологическом институте прикладной микроэлектроники СО РАН унифицированного электронного модуля (УЭМ). Разработанный электронный модуль является уникальным и использовался совместно с приемниками производства Sofradir 320×256 и ULIS 320×240. При использовании указанных приемников модуль не требует внесения изменений в его электрическую схему. В основу УЭМ положен принцип модульности и построения основных функциональных узлов системы обработки сигналов на кристалле ПЛИС типа FPGA. Все арифметические модули системы обработки сигналов были выполнены на основе ПЛИС Cyclone фирмы Altera с использованием программного обеспечения Quartus 4.0.

Построение арифметических модулей ЦОС основано на принципе составления арифметических узлов в соответствии с заданной последовательностью выполнения операций.

Рассмотрим арифметический модуль, вычисляющий функцию  $y = \frac{(a+500)}{b}$ . Соответствующий арифметический модуль представлен на рис. 1.

Здесь каждое арифметическое действие представлено соответствующей мегафункцией, размещенной в библиотеке стандартных функций программы Quartus. Выходы узлов с результатами промежуточных операций (узел суммирования  $a + 500$ ) поступают непосредственно на вход следующего арифметического узла (узел деления  $\frac{(a+500)}{b}$ ), тем самым обеспечивая минимальное время расчета функции.

Предварительно при выполнении вычислений необходимо обеспечить своевременное поступле-

ние данных на входы арифметических узлов. Возможно поступление данных на входы арифметического модуля как одновременно, так и в различные моменты времени. В первом случае правильные данные на выходе арифметического модуля появятся через промежуток времени, равный времени расчета функции арифметическим узлом  $t_{cal}$ . Во втором случае это время равно  $t_{cal} + t_{del}$ , где  $t_{del}$  — разница во времени между поступлением данных на входы 1 и 2.

Для фиксации данных на входах и выходах арифметических узлов используются буферы на D-триггерах (см. рис. 1). В случае, когда данные выставляются на время, превышающее время расчета арифметического узла (например, выходные данные с аналого-цифрового преобразователя (АЦП) при периоде тактовой частоты, превышающем время расчета арифметического узла), буфер не ставится. В данном случае важно точно рассчитать время, в которое необходимо зафиксировать выходные данные, так как длительность задержки данных от входа к выходу для различных арифметических устройств, реализованных посредством ПЛИС, различно и может составлять от десяти и более наносекунд. Причем существенно проявляется зависимость задержки данных и от внутренней архитектуры ПЛИС. Так, для ПЛИС Cyclone задержка данных на 14-разрядном сумматоре составляет 9,5 нс, для ПЛИС Asex, для того же сумматора, задержка составляет 11 нс.

Задержка данных арифметического узла рассчитывается посредством использования программы Quartus. Для этого арифметический модуль компилируется как индивидуальный проект для типа микросхемы программируемой логики, используемой в проектируемой системе ЦОС. Далее проект симулируется. В результате симуляции получаются диаграммы выходных сигналов со всеми необходимыми значениями временных параметров.

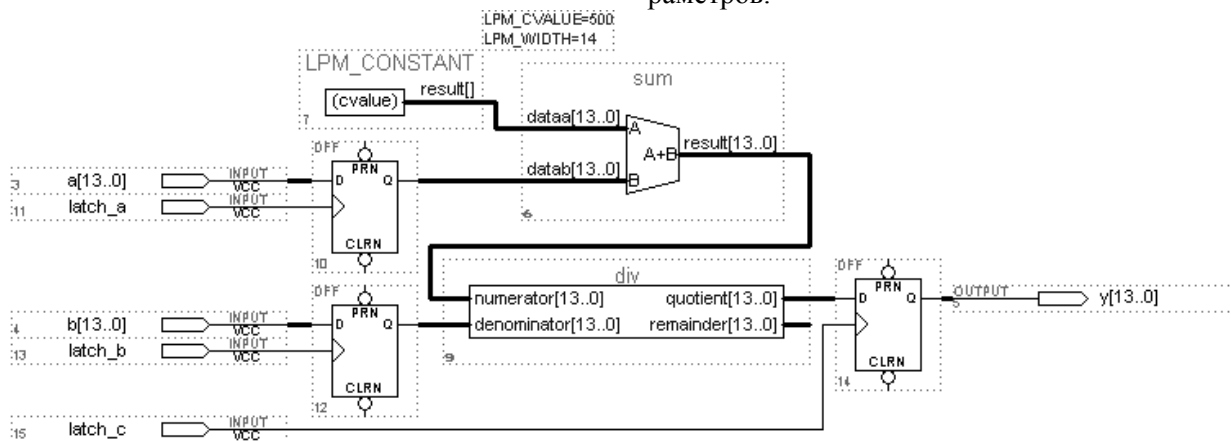


Рис. 1. Арифметический модуль, вычисляющий функцию  $y = \frac{(a+500)}{b}$

Для каждого арифметического узла выполняется свой генератор импульсов фиксации данных. Он представляет собой счетчик, работа которого разрешается при начале работы соответствующего алгоритма. Выходными сигналами данного узла являются импульсы, появляющиеся в требуемое время, когда нужно зафиксировать данные, поступающие на вход *D*-триггера (см. рис. 1).

Доступ к данным при работе модуля организован так, что все алгоритмы обработки данных используют один генератор импульсов для триггеров фиксации данных. Причем фиксация промежуточных данных на *D*-триггерах применена только в некоторых алгоритмах обработки сигнала (алгоритм расчета таблицы дефектных элементов матрицы и алгоритм суммирования сигналов элементов), во всех остальных случаях для фиксации данных достаточно использования банков памяти, АЦП и цифроаналогового преобразователя (ЦАП).

Функциональная схема, поясняющая синхронизацию потоков данных модуля, представлена на рис. 2, где *MC\_ACP* – тактовая частота АЦП; *DATA\_ACP* – выходные данные АЦП; *Read\_B1* – импульс чтения в банк № 1; *Read\_B2* – импульс чтения в банк № 2; *Write\_B3* – импульс записи в банк № 3; *Data\_B1* – шина данных банка № 1; *Data\_B2* – шина данных банка № 2; *Data\_B3* – шина данных банка № 3; *Y* – выходная шина данных. Здесь импульсом фиксации данных результата расчетов является импульс записи в память, а данные выставляются с АЦП на период такта частоты квантования, эта же частота подается на счетчик адресов памяти, тем самым обеспечивая синхронность работы потоков данных, т. е. данные появляются и изменяются на шине при окончании цикла вычисления. После этого адресный счетчик микросхем памяти переключается и для расчетов формируется следующее слово данных. Все это происходит

синхронно с тактовой частотой АЦП. Для уменьшения времени доступа к памяти, упрощения схем коммутации шин данных и обеспечения возможности одновременного доступа к банкам памяти применяется общая адресная шина, что дает возможность одновременного доступа к разным банкам.

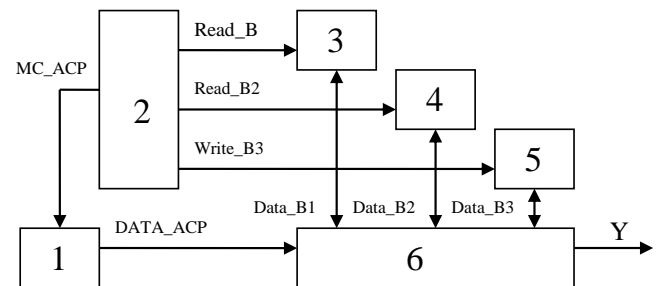


Рис. 2. Функциональная схема синхронизации данных электронного модуля:

1 — АЦП; 2 — генератор импульсов фиксации данных; 3 — банк памяти № 1; 4 — банк памяти № 2; 5 — банк памяти № 3; 6 — арифметико-логическое устройство

Выбор арифметического модуля для вычисления того или иного алгоритма осуществляется посредством мультиплексирования соответствующих шин данных (рис. 3). Здесь *TEC[13..0]* – шина данных выходного сигнала матрицы; *Bank\_Min[13..0]* – шина данных банка памяти "холодного" кадра; *Bank\_Max[13..0]* – шина данных банка памяти "горячего" кадра; *Bank\_k[13..0]* – шина данных банка памяти коэффициентов коррекции; *adres[1..0]* – шина адреса управления мультиплексором; *y[13..0]* – шина выходных данных.

Для упрощения схемы ЦОС все модули алгоритмов обработки выполняются отдельно (см. рис. 3). Причем некоторые модули, реализующие арифметические операции для различных алгоритмов, могут повторяться, что также позволяет упростить схему мультиплексирования потоков данных.

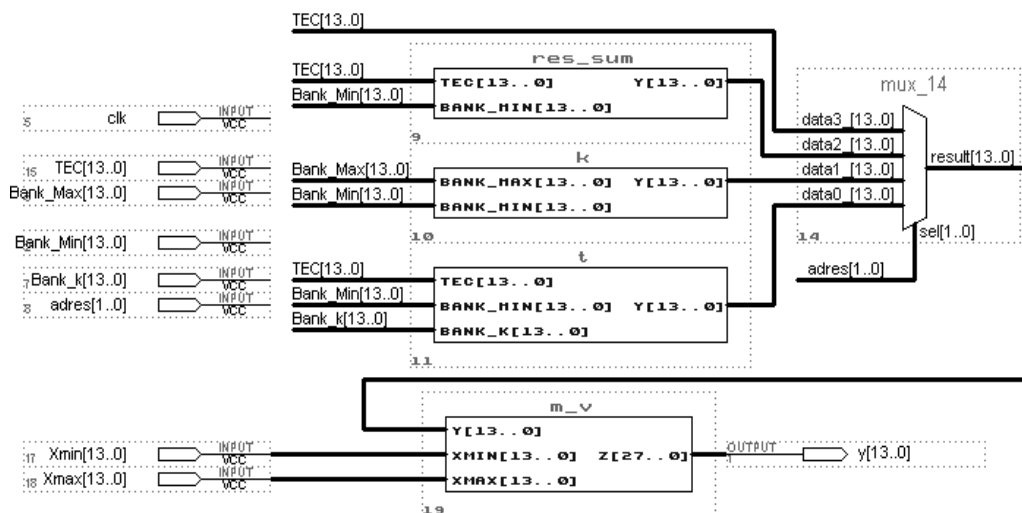


Рис. 3. Схема мультиплексирования потоков данных электронного модуля

### Алгоритмы обработки данных

На ПЛИС, используемой в УЭМ, реализованы следующие алгоритмы обработки данных.

**Алгоритм "вывод без обработки"**. Здесь 14-разрядный код сигнала ФПУ поступает непосредственно на модуль визуализации, где преобразуется в код, соответствующий разрядности ЦАП. Тепловизионное изображение, получаемое на экране монитора при работе алгоритма "вывод без обработки", представлено на рис. 4.

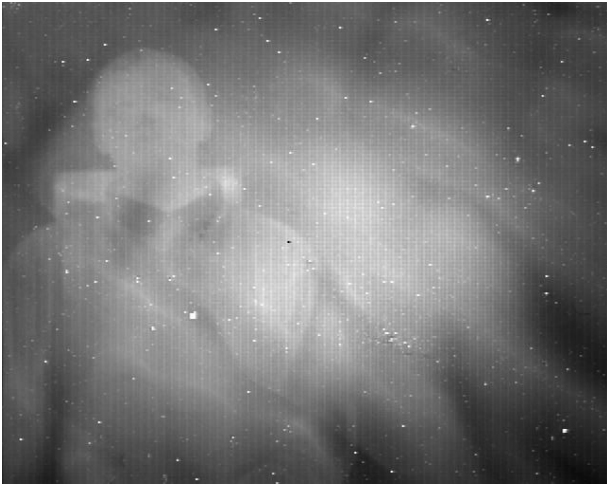


Рис. 4. Изображение при работе алгоритма "вывод без обработки"

**Алгоритм "вычитание"**. Здесь выполнение алгоритма осуществляется модулем, выполняющим операцию "вычитание". Модуль вычисляет функцию  $y[13..0] = \text{TEC}[13..0] - \text{Bank\_Min}[13..0]$ , т. е. осуществляет арифметическую операцию попиксельного вычитания из текущего 14-разрядного кода сигнала ФПУ соответствующего пиксельного значения "темнового" кадра (рис. 5). "Темновой" кадр представляет собой массив данных, записанный в банк памяти и содержащий оцифрованные сигналы элементов ФПУ, засвеченного равномерным полем выбранной температуры. Результат вычисления  $y[13..0]$  преобразуется в модуле визуализации в код, соответствующий разрядности ЦАП.

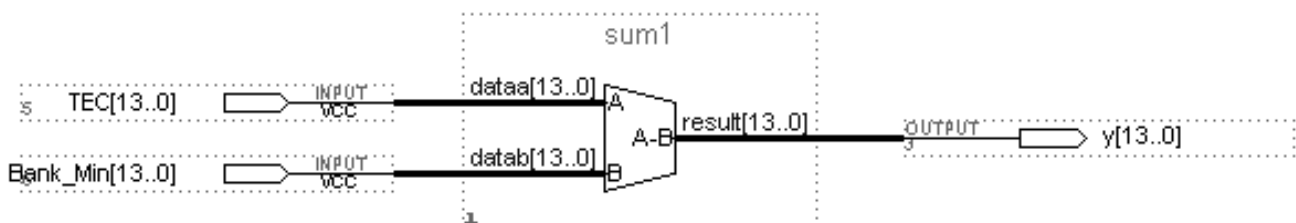


Рис. 5. Арифметический модуль, вычисляющий функцию  $y[13..0] = \text{TEC}[13..0] - \text{Bank\_Min}[13..0]$

Тепловизионное изображение, получаемое на экране монитора при работе алгоритма "вычитание", представлено на рис. 6. Здесь четко видна неравномерность чувствительности элементов матрицы в виде темного пятна в середине рисунка.



Рис. 6. Изображение при работе алгоритма "вычитание"

**Алгоритм "коррекция по двум точкам"**. Данный алгоритм состоит из двух операций, реализуемых соответствующими модулями:

- алгоритм "расчет коэффициента  $k$ " — алгоритм, при котором арифметический модуль выполняет вычисление коэффициентов неравномерности чувствительности элементов матрицы. Выполнение алгоритма осуществляется модулем, аналогичном модулю, выполняющему алгоритм "вычитание". Модуль вычисляет функцию  $y[13..0] = \text{Bank\_Max}[13..0] - \text{Bank\_Min}[13..0]$ , где  $k[13..0] = y[13..0]$ , т. е. выполняет арифметическую операцию вычитания массива данных банка, соответствующего холодному фону из массива данных, соответствующих теплоте полю (рис. 7). Результат вычисления  $y[13..0]$  записывается в соответствующий банк микросхемы памяти и используется при выполнении алгоритма "коррекция по двум точкам";

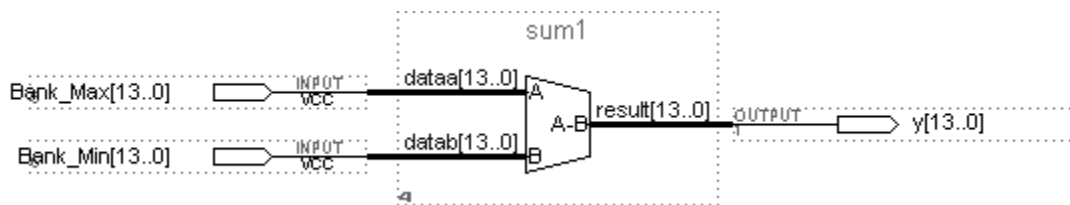


Рис. 7. Арифметический модуль, вычисляющий функцию  $y[13..0] = \text{Bank\_Max}[13..0] - \text{Bank\_Min}[13..0]$

• алгоритм "коррекция по двум точкам" — алгоритм выполняет коррекцию теплового изображения в соответствии с рассчитанными коэффициентами. Здесь выполнение алгоритма осуществляется модулем, вычисляющим функцию  $y[13..0] = \frac{\text{TEC}[13..0] - \text{Bank\_Min}[13..0]}{\text{Bank\_k}[13..0]}$ . Модуль осуществляет арифметические операции попиксельного вычитания из текущего 14-разрядного выходного кода сигнала ФПУ соответствующего пиксельного значения холодного кадра и деления

полученного промежуточного результата на соответствующее значение коэффициента "k" (рис. 8). Результат вычисления  $y[13..0]$  преобразуется в модуле визуализации в код, соответствующий разрядности ЦАП.

Тепловизионное изображение, получаемое на экране монитора при работе алгоритма "коррекция по двум точкам", представлено на рис. 9. Здесь устраняется неравномерность сигналов элементов матрицы. Благодаря этому становится четко видно изображение объектов в середине рис. 9.

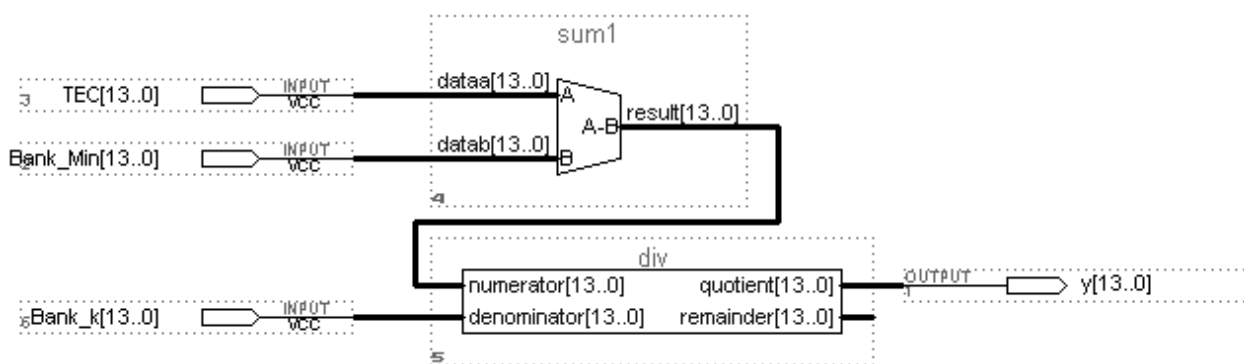


Рис. 8. Арифметический модуль, вычисляющий функцию  $y[13..0] = \frac{\text{TEC}[13..0] - \text{Bank\_Min}[13..0]}{\text{Bank\_k}[13..0]}$



Рис. 9. Изображение при работе алгоритма "коррекция по двум точкам"

**Алгоритм "суммирование сигналов элементов"** — это алгоритм, при котором ПЛИС для снижения шума вычисляет среднее значение сигнала каждого элемента ФПУ. Блок-схема арифметического модуля представлена на рис. 10. Здесь сигнал, поступающий с ФПУ  $\text{TEC}[13..0]$ , сравнивается с записанным в память сигналом  $\text{Bank\_Min}[13..0]$  (сигнал  $\text{Bank\_Min}[13..0]$  соответствует записанному в память "темновому" кадру). Если значение  $\text{TEC}[13..0]$  больше  $\text{Bank\_Min}[13..0]$ , то на выходе  $\text{alb}$  компаратора "b" появляется логическая единица. Сумматор  $\text{int\_sum}$  переходит в режим суммирования, в результате чего к значению сигнала  $\text{Bank\_Min}[13..0]$  прибавляется единица, и наоборот, из значения сигнала  $\text{Bank\_Min}[13..0]$  вычитается единица, если сигнал  $\text{TEC}[13..0]$  меньше сигнала  $\text{Bank\_Min}[13..0]$ .

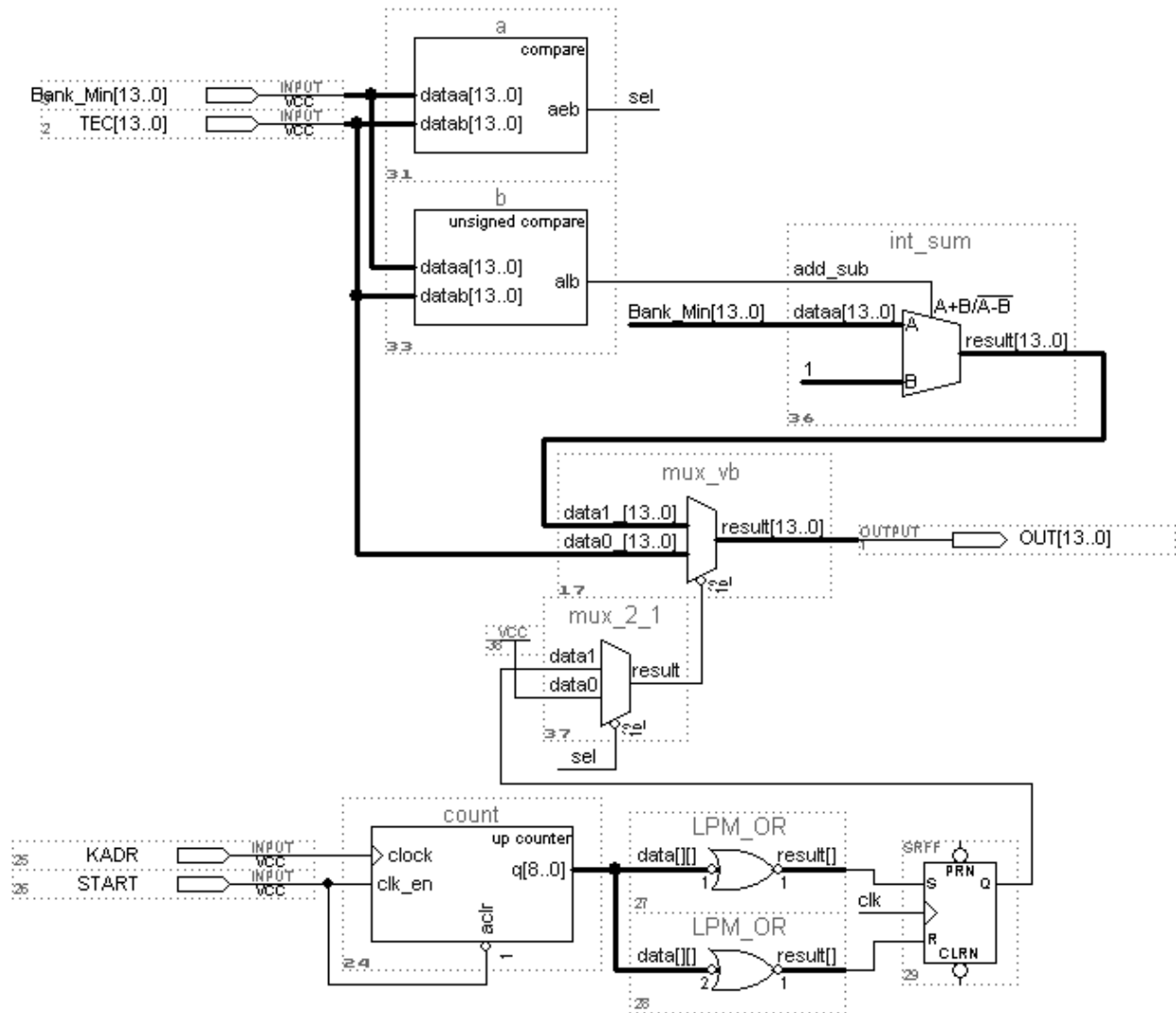


Рис. 10. Арифметический модуль, вычисляющий функцию  $y[13..0] = \frac{TEC[13..0] - Bank\_Min[13..0]}{Bank\_k[13..0]}$

В то же время при равенстве сигналов  $TEC[13..0]$  и  $Bank\_Min[13..0]$  на выходе  $aeb$  компаратора "a" появляется логическая единица. Мультиплексор  $mux\_vb$  коммутирует поток данных  $TEC[13..0]$  на выход  $OUT[13..0]$  модуля, тем самым исключая увеличение или уменьшение на единицу сигнала  $Bank\_Min[13..0]$ .

Для записи в память первого "темнового" кадра служит счетчик  $count$  с соответствующими цепями дешифрования. Счетчик запускается сигналом начала выполнения алгоритма  $START$  и считает количество кадровых синхроимпульсов. При первом и втором кадровом синхроимпульсе триггер  $srff$  находится в высоком логическом состоянии, что обеспечивает прохождение сигнала  $TEC[13..0]$  на выход  $OUT[13..0]$ . При появлении на счетчике заданного числа выполнение алгоритма прекращается (соответствующие цепи на блок-схеме не показаны).

В результате работы алгоритма входной сигнал  $TEC[13..0]$  фильтруется от импульсных помех, тем самым обеспечивая снижение шума каждого элемента при записи "темновых" кадров.

#### Алгоритмы определения и замещения дефектных элементов

В настоящее время существующие технологии не позволяют получать выход 100 % ФПУ с годными элементами 100 %, поэтому для исправления на изображении дефектных элементов используются различные алгоритмы, которые в соответствии с заданным критерием определяют дефектные элементы, а затем заменяют их полноценными элементами.

**Алгоритм "расчет таблицы дефектных элементов"**

В результате работы алгоритма формируется таблица дефектных элементов ФПУ. Таблица записывается в банк памяти в виде массива кодов, где дефектному элементу соответствует код "16383", а полноценному элементу — код "0". В начале работы алгоритма записывается таблица 100 % бездефектных элементов, т. е. всем элементам таблицы присваивается значение "0". Далее в процессе выполнения алгоритма по адресу, соответствующему определенному дефектному элементу, в таблицу записывается код "16383". Выполнение алгоритма происходит заданное число раз, при этом полученная ранее таблица дополняется новыми дефектными элементами, определенными на текущем этапе работы алгоритма. Таким образом, увеличивается вероятность определения дефектных элементов ФПУ. УЭМ позволяет определять дефектные элементы ФПУ по следующим критериям:

критерий среднеквадратичных ошибок (СКО) шума элементов ФПУ — критерий позволяет определить дефектные элементы, превышающие СКО-шума полноценных элементов ФПУ;

критерий превышения порога сигналов элементов ФПУ — критерий позволяет определить элементы, значение сигнала которых постоянно и превышает значения сигналов всех остальных элементов.

Блок-схема арифметического модуля представлена на рис. 11. Сигнал, поступающий с ФПУ TEC[13..0], вычитается из записанного в память сигнала Bank\_Min[13..0]. Если значение TEC[13..0] меньше Bank\_Min[13..0], то на выходе sum\_tv сумматора result[13..0] (см. рис. 8) появляется положительный код разности сигналов. И наоборот, если значение TEC[13..0] больше Bank\_Min[13..0], то на выходе sum\_tv сумматора result[13..0] появляется отрицательный код разности сигналов. Для выполнения дальнейших операций код с выхода result[13..0] подается на вход мегафункции abs. Здесь из сигнала, поступающего на вход data[13..0], выделяется его абсолютное значение.

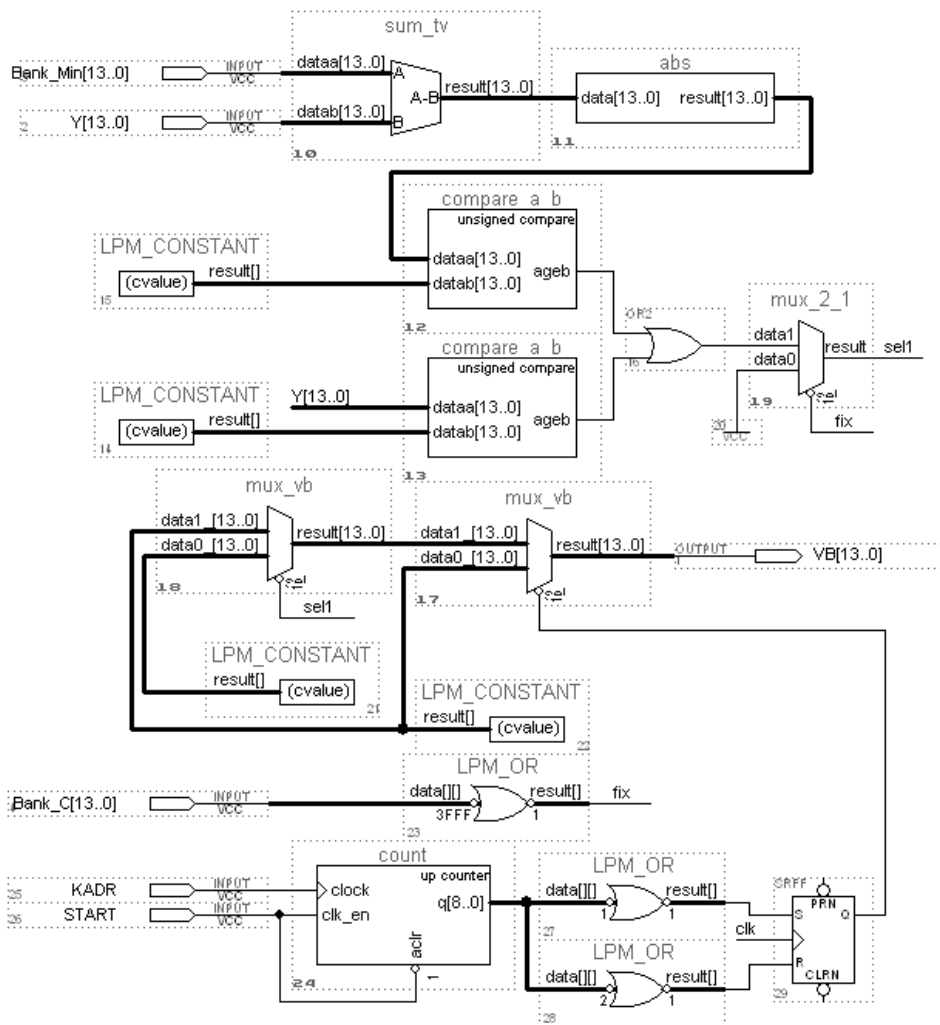


Рис. 11. Упрощенная схема модуля расчета таблицы дефектных элементов ФПУ





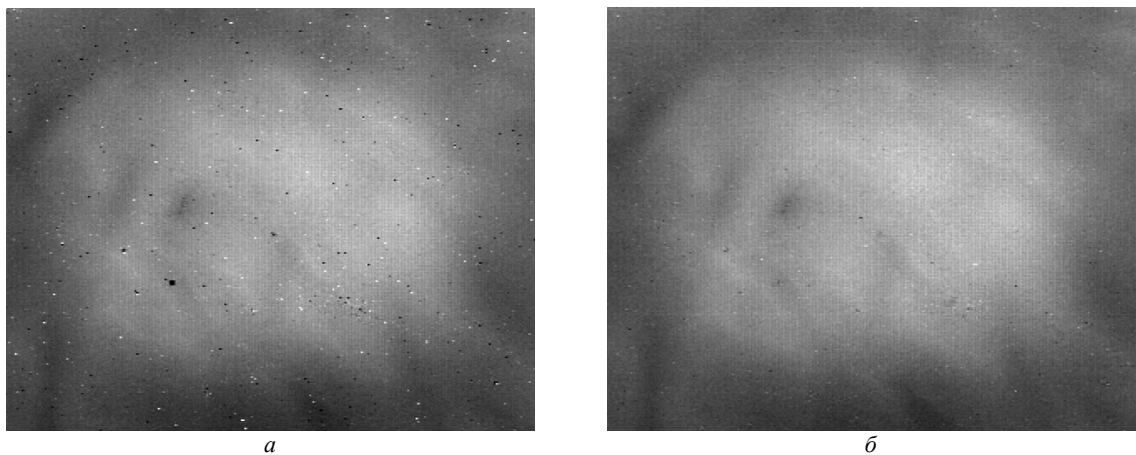


Рис. 13. Изображение:

*a* — с выключенной функцией "замещение дефектных элементов"; *б* — с включенной функцией "замещение дефектных элементов"

### Алгоритм "визуализация"

Задача визуализации связана с улучшением согласования динамического диапазона обработанного сигнала изображения и экрана ТВ-монитора, на котором выполняется визуализация. Если для цифрового представления каждого отсчета телевизионного изображения отводится 14 бит запоминающего устройства, то входной сигнал может принимать одно из 16384 значений. Для вывода обработанного сигнала на монитор производится преобразование выходного цифрового видеосигнала в аналоговый сигнал посредством использования ЦАП. Разрядность ЦАП определяется требованиями к количеству градаций серого, необходимых для качественного воспроизведения изображения на экране монитора. В системе ЦОС УЭМ используется 12-разрядный ЦАП, тем самым определяя рабочий диапазон 0—4095, при этом значение 0 соответствует при визуализации уровню черного, а значение 4095 — уровню белого.

Для согласования 14-разрядного динамического диапазона обработанного сигнала с 12-разрядным входом ЦАП используется модуль (рис. 14), попиксельно вычисляющий функцию

$$Z[27..0] = \frac{Y[13..0] - X_{\min}[13..0]}{X_{\max}[13..0] - X_{\min}[13..0]} \times 4095.$$

Здесь  $Z[27..0]$  — выходной сигнал модуля визуализации. Используются младшие 12 разрядов;  $X_{\min}[13..0]$ ,  $X_{\max}[13..0]$  — минимальная и максимальная яркости исходного изображения;  $Y[13..0]$  — входной 14-разрядный сигнал.

Изменение  $X_{\min}[13..0]$ ,  $X_{\max}[13..0]$  позволяет улучшать визуальное качество изображения как в ручном, так и в автоматическом режиме. В ручном режиме на входы  $X_{\min}[13..0]$  и  $X_{\max}[13..0]$  подаются коды, соответствующие желаемому диапазону исходного сигнала. Модуль `max_min` служит

для ограничения входного сигнала  $Y[13..0]$  в соответствии с условиями:

$$\text{if } (IN[13..0] > MIN[13..0] \text{ and } IN[13..0] < < MAX[13..0]) \text{ then out}[13..0] = IN[13..0].$$

Это позволяет, задавая диапазон сигнала  $MAX[13..0]$  и  $MIN[13..0]$ , изменять яркость и контраст изображения. Изменение  $MAX[13..0]$  и  $MIN[13..0]$  позволяет также выбрать желаемый участок гистограммы изображения и повысить контраст соответствующих объектов (рис. 15).

Так, на рис. 15, *a* показано изображение "холодного" фона при "горячем" объекте, а на рис. 15, *б*, *в* показан "горячий" объект при "холодном" фоне. Участки изображения, значение температуры которых расположено между значениями температуры "холодного" фона и "горячего" объекта, показаны на рис. 15, *г*.

В автоматическом режиме на входы  $X_{\min}[13..0]$  и  $X_{\max}[13..0]$  подается код, вычисленный по какому-либо алгоритму, автоматически определяющему диапазон визуализации. Таким алгоритмом может являться алгоритм, определяющий минимальное  $X_{\min}[13..0]$  и максимальное  $X_{\max}[13..0]$  значения сигнала в кадре. Алгоритм неудобен тем, что при появлении в кадре "горячего" объекта изображение "заплывает", тем самым резко снижается восприятие изображения. Более эффективным в большинстве случаев является алгоритм, при котором вычисляется среднее значение  $X_{avr}[13..0]$  по кадру с последующим вычислением

$$X_{\min}[13..0] = X_{avr} - \frac{D}{2} \text{ и } X_{\max}[13..0] =$$

$$= X_{avr} + \frac{D}{2}, \text{ где } D \text{ — желаемый диапазон визуализации.}$$

Здесь при появлении в кадре "горячего" объекта общая сумма значений сигналов элементов ФПУ изменяется незначительно, благодаря чему изображение сохраняет свою стабильность.

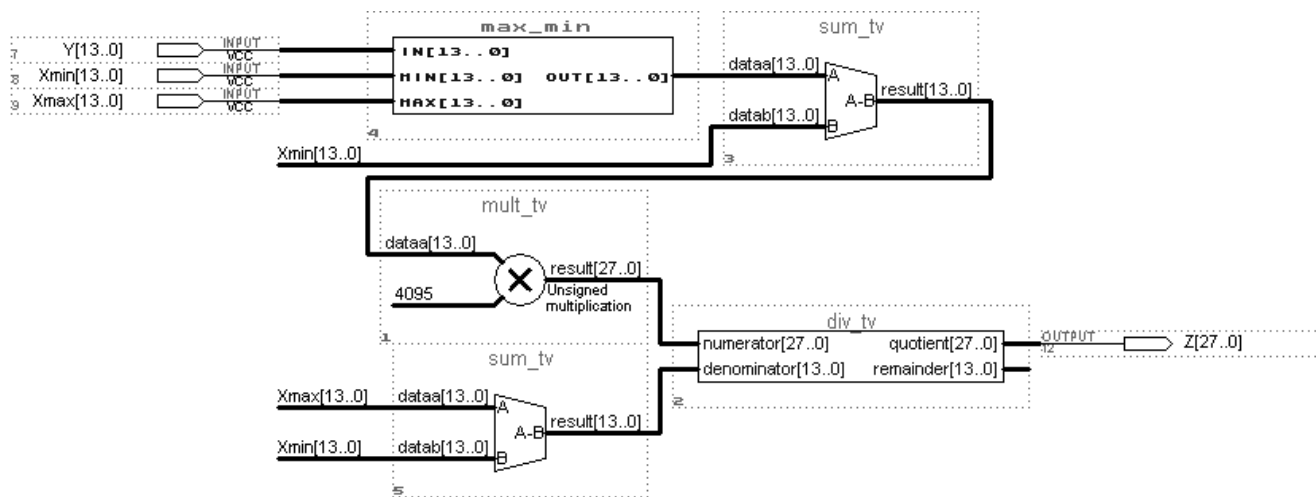


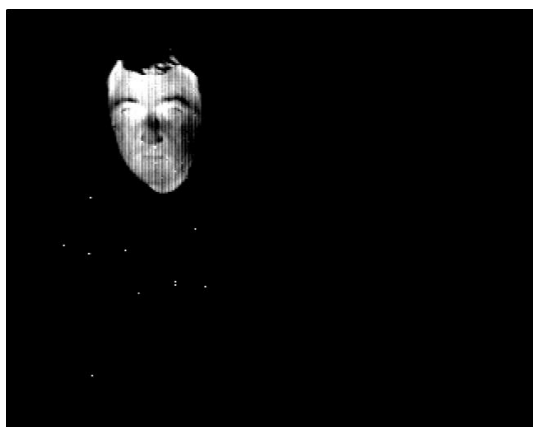
Рис. 14. Арифметический модуль, вычисляющий функцию  $Z[27..0] = \frac{Y[13..0] - Xmin[13..0]}{Xmax[13..0] - Xmin[13..0]} \times 4095$



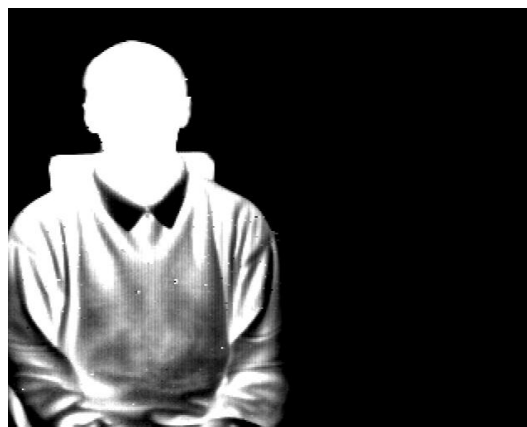
a



b



в



г

Рис. 15. Изображения:

a — "холодного" фона при "горячем" объекте; б, в — "горячего" объекта при "холодном" фоне; г — участки изображения, значение температуры которых расположено между значениями температуры "холодного" фона и "горячего" объекта

### З а к л ю ч е н и е

Рассмотрены способы и принципы реализации алгоритмов ЦОС посредством жесткой логики ПЛИС. Даны схемотехнические решения построения алгоритмов. Приведены результаты практического использования алгоритмов. Предложенные решения создания узлов цифровой обработки сигналов позволяют создавать системы на кристалле ПЛИС, используя доступное программное обеспечение Quartus.

### Л и т е р а т у р а

1. Грувицкий Р. И., Мурсаев А. Х., Угрюмов Е. П. Проектирование систем на микросхемах программируемой логики. — СПб.: БХВ-Петербург, 2002. — 608 с.
2. Прохоренко А. ПЛИС как DSP. Электронные компоненты, 1999. № 5. [http://www.elcp.ru/index.php?state=izd&i\\_izd=elcomp&i\\_num=1999\\_05&i\\_art=11](http://www.elcp.ru/index.php?state=izd&i_izd=elcomp&i_num=1999_05&i_art=11)
3. <http://www.altera.com>
4. Губанов Д., Стещенко В., Шипулин С. Современные алгоритмы ЦОС: перспективы реализации // Электроника НТБ, 2005. <http://www.electronics.ru/790.html#>

Статья поступила в редакцию 11 октября 2006 г.

## Methods and principles of construction of algorithms for processing the signals of the IR multielement photodetective assembly on the basis of a chip of programmed logic

*I. I. Kremis, Yu. F. Odnolko*

Engineering and Design Institute for Applied Microelectronics, Novosibirsk, Russia

*Considered are schematic decisions for construction of algorithms. Results of practical use of the realized algorithms are given. Algorithms of digital processing the signal are realized on a chip of programmed logic (Cyclone series, Altera Co.).*

УДК 621. 315. 592

## Защитная оксидная пленка на эпитаксиальных слоях тройных твердых растворов кадмий–ртуть–теллур и марганец–ртуть–теллур

*В. Н. Рыжков, Н. В. Алеева, В. А. Андреев, И. М. Несмелова*

ФГУП «НПО "Государственный институт прикладной оптики"», г. Казань, Россия

*Г. Г. Гумаров, В. Ю. Петухов*

Казанский физико-технический институт КазНЦ РАН, г. Казань, Россия

*Данная статья является продолжением серии работ ФГУП «НПО "ГИПО"» и КФТИ КНЦ РАН по исследованию электрофизических свойств полумажнитного твердого раствора  $MnHgTe$  и сравнению его электрофизических свойств со свойствами твердого раствора  $CdHgTe$ . Изучена защитная оксидная пленка на эпитаксиальных слоях  $CdHgTe$  и  $MnHgTe$ , полученная анодированием поверхности в водном растворе лимонно-кислого натрия. Показано, что оксидная пленка на слоях  $MnHgTe$  имеет большую химико-механическую устойчивость по сравнению с устойчивостью пленок на слоях  $CdHgTe$ .*

Известно, что поверхность фоточувствительных элементов для оптоэлектроники, изготовленных на основе тройных твердых растворов  $Cd_xHg_{1-x}Te$  (КРТ), необходимо пассивировать, что обеспечивает герметизацию полупроводника, ста-

билизацию его химических и механических свойств, при этом пассивирующее покрытие действует как антиотражающее.

Традиционным методом пассивации поверхности КРТ является анодное окисление поверхности