ГЛАВА 6

СВЕРТКА

Свертка – это математический способ комбинирования двух сигналов для формирования третьего сигнала. Это один из самых важных методов ЦОС. Пользуясь стратегией импульсного разложения, системы описываются сигналом, называемым *импульсной характеристикой*. Свертка важна, так как она связывает три сигнала: входной сигнал, выходной сигнал и импульсную характеристику. Эта глава представляет свертку с двух разных точек зрения, названных алгоритмом входной стороны и алгоритмом выходной стороны. Свертка представлена исключительно в математических рамках для ЦОС.

Дельта-функция и импульсная характеристика

В предыдущей главе было описано, как сигнал может быть разложен на группу составляющих, называемых *импульсами*. Импульс – это сигнал, составленный из нулевых значений за исключением одной ненулевой точки. В результате, импульсное разложение предоставляет способ поточечного анализа сигнала. Предыдущая глава представила также основную концепцию ЦОС: входной сигнал раскладывается на простые слагаемые составляющие, каждая из этих составляющих пропускается через линейную систему, и результирующие выходные компоненты складываются. Сигнал, полученный таким способом, идентичен тому, который получается пропусканием исходного сигнала через линейную систему. В то время как существует множество методов разложения, основу обработки сигналов составляют только два: импульсное разложение и разложение Фурье. Когда используется импульсное разложение, процедура может быть описана математической операцией, называемой *свертка*. В этой главе (и большинстве следующих) мы будем иметь дело с *дискретными* сигналами. Свертка применяется также к *непрерывным* сигналам, хотя для этого требуется более сложная математика. Обработка непрерывных сигналов будет рассмотрена в Главе 13.

Рис.6-1 определяет два важных термина, используемых в ЦОС. Первый это *дельта-функция*, обозначаемая греческой буквой дельта, $\delta[n]$. Дельта-функция является *нормированным* импульсом, в котором нулевая выборка содержит единичное значение, а все другие нулевое. По этой причине, дельта-функция часто называется *единичным импульсом*.

Второй термин, определенный на Рис.6-1, *импульсная характеристика*. Как показывает название, импульсная характеристика является сигналом, который получается на выходе системы, если на ее входе поступает дельта-функция (единичный импульс). Если две системы отличны друг от друга, они будут иметь разные импульсные характеристики. Также как входному и выходному сигналам присваиваются символы $x[n]$ и $y[n]$, импульсная характеристика часто обозначается как h[n]. Конечно, это обозначение может быть заменено другим более наглядным, например, символ f[n] может использоваться для обозначения импульсной характеристики фильтра.

Любой импульс может быть представлен как *смещенная* и *промасштабированная* дельта-функция. Рассмотрим сигнал a[n], состоящий из выборок со значением ноль, за исключением восьмой выборки со значением –3. Этот сигнал является той же самой дельта-функцией, которая смещена вправо на 8 выборок и умножена на –3. В форме выражения это представляется как: $a[n] = -3\delta[n-8]$. Убедитесь, что вы поняли это выражение, поскольку оно используется почти во всех выражениях ЦОС.

Если на вход системы поступит импульс, такой как $-3\delta[n-8]$, что будет на выходе системы? Здесь используются свойства однородности и инвариантности относительно смещения. Масштабирование и смещение входного сигнала выражается в идентичном масштабировании и смещении выходного сигнала. Если $\delta[n]$ вызывает h[n], из этого следует, что $-3\delta[n-8]$ вызовет на выходе $-3\delta[h-8]$. Другими словами, выход является версией импульсной характеристики, промасштабированной и смещенной идентично дельта-функции на входе. Если известна импульсная характеристика системы, можно узнать, как она отреагирует на любой импульс.

Свертка

Давайте обобщим понятие, как система преобразует входной сигнал в выходной. Во-первых, входной сигнал может быть разложен на множество импульсов, каждый из которых может быть рассмотрен как масштабированная и смещенная дельта-функция. Во-вторых, выход, полученный от каждого импульса, является масштабированной и смещенной импульсной характеристикой. В-третьих, выходной сигнал может быть найден сложением этих масштабированных и смещенных импульсных характеристик. Другими словами, зная импульсную характеристику системы, можно всегда рассчитать выходной сигнал для любого входного сигнала. Это означает, что вы знаете *все* о системе. Это все, что можно сказать о характеристиках линейных систем. (Тем не менее, в последующих главах будет показано, как эта информация может быть представлена в других формах).

В некоторых приложениях импульсная характеристика может выступать под другим названием. Если система рассматривается как *фильтр*, импульсная характеристика называется *ядром фильтра*, *ядром свертки* или просто *ядром*. В приложениях обработки изображений импульсная характеристика называется *функцией рассеяния точки*. В то время как эти термины используются в немного разных методах, они означают одно и то же, сигнал, полученный на выходе системы, если на ее входе поступает дельта-функция.

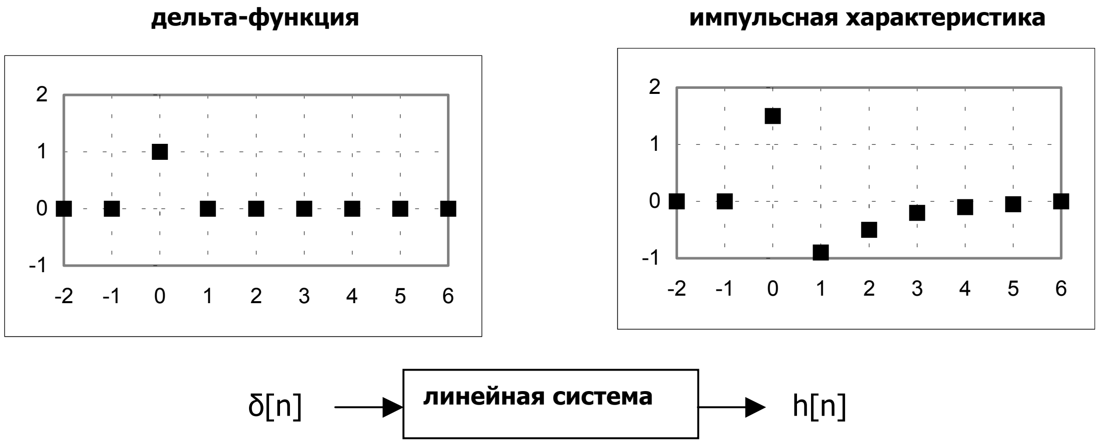


Рис.6-1. Определение дельта-функции и импульсной характеристики.

Свертка является формальной математической операцией, такой же, как умножение, сложение и интегрирование. Сложение берет два *числа* и производит третье *число*, в то время как свертка берет два *сигнала* и производит третий *сигнал*. Свертка используется во многих областях математики, например, статистике и вероятности. В линейных системах свертка применяется для описания соотношения между тремя сигналами: входным, выходным и импульсной характеристикой.

Рис.6-2 показывает обозначение свертки при использовании ее с линейными системами. Входной сигнал, x[n], поступает в линейную систему с импульсной характеристикой h[n], в результате чего порождается выходной сигнал, y[n]. В форме выражения это выглядит как: x[n]*h[n] = y[n]. Другими словами, свернутый в соответствии с импульсной характеристикой входной сигнал равен выходному сигналу. Также как сложение представляю знаком «+», умножение знаком «x», свертку представляют знаком «*». К сожалению, большинство языков программирования используют знак «*» для индикации операции умножения. В компьютерных программах этот знак обозначает умножение, в то время как в выражениях операцию свертки.

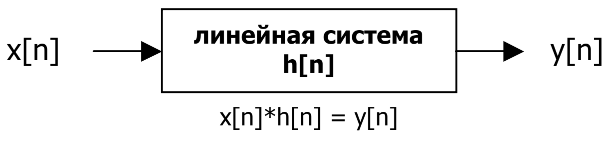
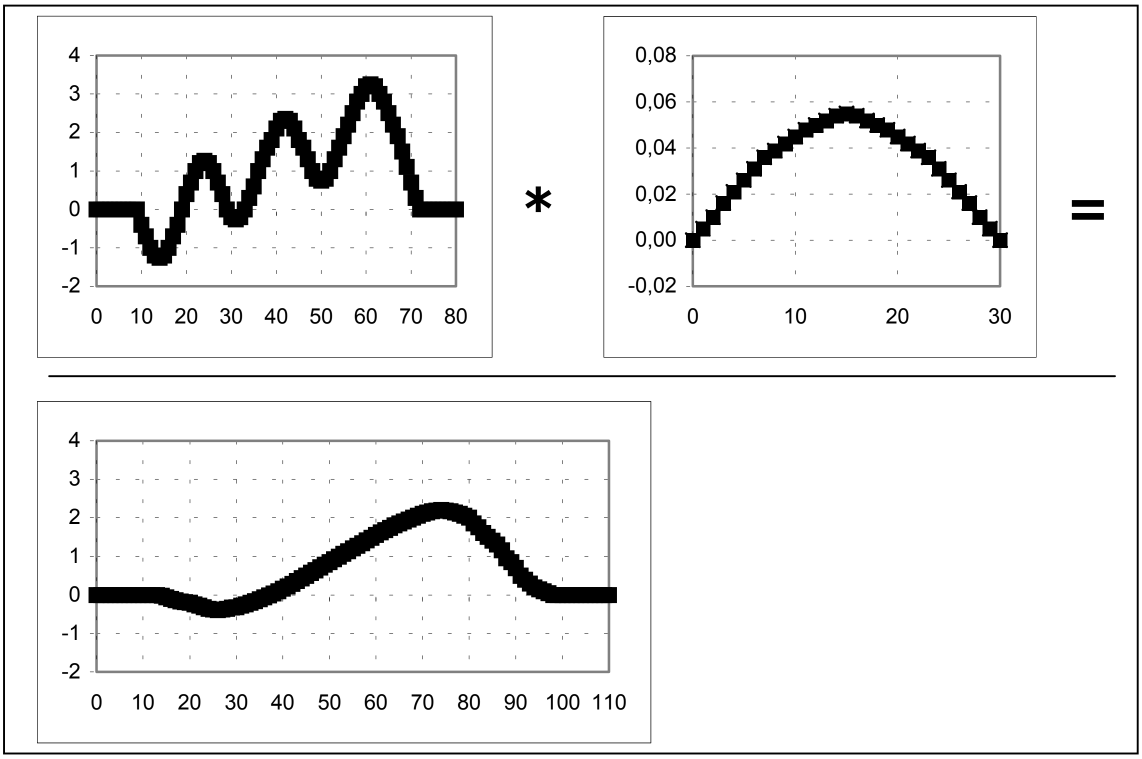


Рис.6-2. Использование операции свертки в ЦОС.

А. Фильтр нижних частот.



Б. Фильтр высоких частот.

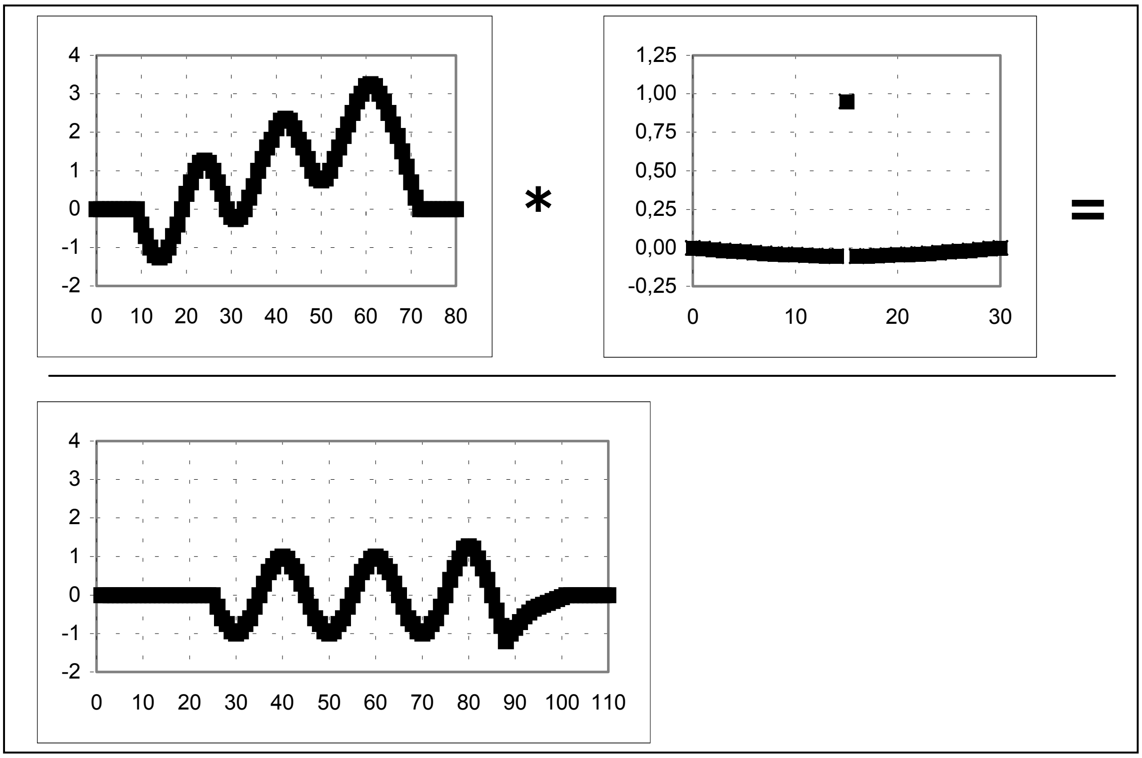


Рис.6-3. Пример фильтрации с помощью операции свертки.

Рис.6-3 показывает использование операции свертки для низкочастотной и высокочастотной фильтрации. Входной сигнал в примере является суммой двух составляющих: три периода синусоиды (представляющих высокую частоту) и медленно возрастающий уклон (состоящий из низких частот). На Рис.6-3а импульсная характеристика низкочастотного фильтра представляет собой гладкую дугу, пропускающую на выход только медленно изменяющиеся волны. В противоположность ему, высокочастотный фильтр на Рис.6-3б пропускает только быстро изменяющиеся волны.

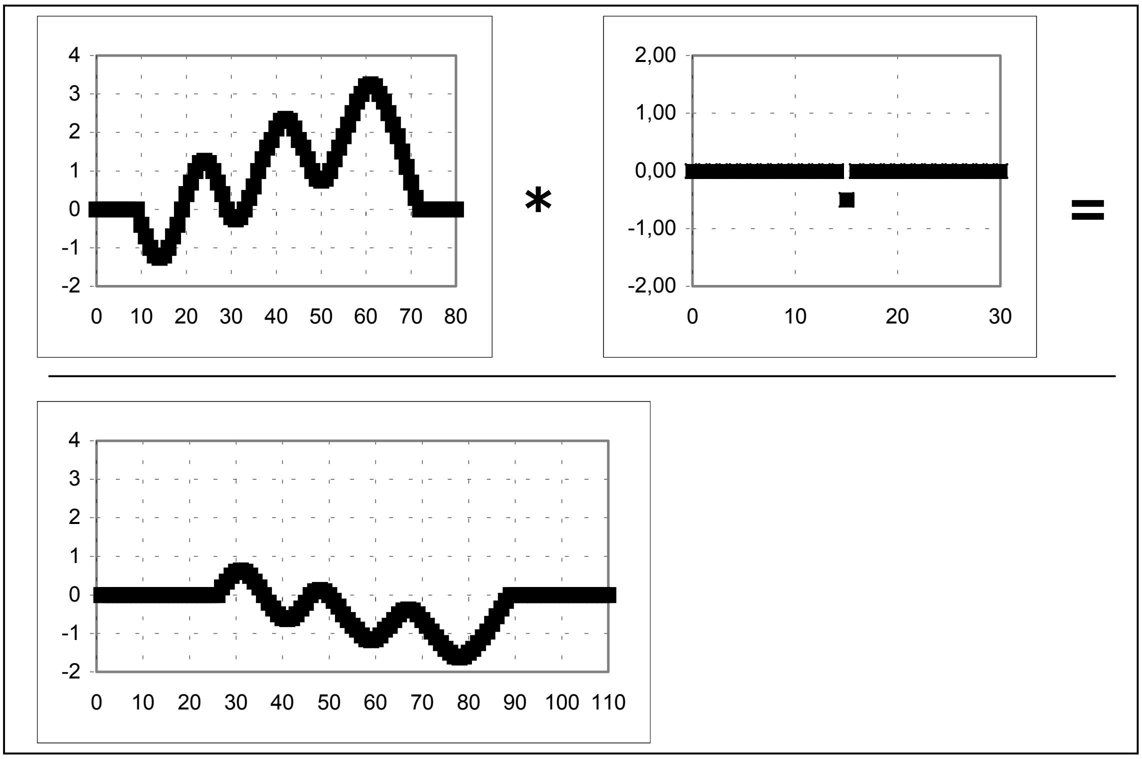
Рис.6-4 показывает два дополнительных примера применения свертки для обработки сигналов. Инвертирующий аттенюатор, Рис.6-4а, отражает сигнал сверху вниз и понижает его амплитуду. Дискретная производная (называемая еще первой разностью), показанная на Рис.6-4б, порождает выходной сигнал относительно *наклона* входного сигнала.

Обратите внимание на размеры сигналов на Рис.6-3 и Рис.6-4. Длина входных сигналов составляет 81 выборку, в то время как каждая импульсная характеристика состоит из 31 выборки. В большинстве приложений ЦОС входной сигнал может состоять из тысяч или миллионов точек. Импульсная характеристика обычно намного короче, скажем от нескольких точек до нескольких сотен. Математика операции свертки не ограничивает длину используемых сигналов. Тем не менее, она определяет длину выходного сигнала, которая равна длине входного сигнала, плюс длине импульсной характеристики, минус один. Для выходных сигналов на Рис.6-3 и Рис.6-4 длина составляет 81+31-1 = 111 выборок.

Сейчас мы подошли к детальному рассмотрению математики операции свертки. Как принято в цифровой обработке сигналов, свертка может быть рассмотрена двумя способами. Первый взгляд на свертку *с точки зрения входного сигнала*. Это включает в себя анализ того, как каждая выборка во входном сигнале *вносит вклад* в выборки выходного сигнала. Второй способ: взглянуть на свертку *с точки зрения выходного сигнала*. Это рассмотрение, как каждая выборка в выходном сигнале *получает* информацию от выборок входного сигнала.

Помните, что эти два взгляда на свертку понимают под собой одну и ту же математическую операцию. Первая точка зрения важна, так как она обеспечивает *концептуальное* понимание того, как свертка относится к ЦОС. Вторая точка зрения описывает *математику* операции свертки. Она олицетворяет одну из наиболее сложных задач, с которой вы столкнетесь в ЦОС: приспособление вашего концептуального понимания к математике, реализующей идеи.

А. Инвертирующий аттенюатор.



Б. Дискретная производная.

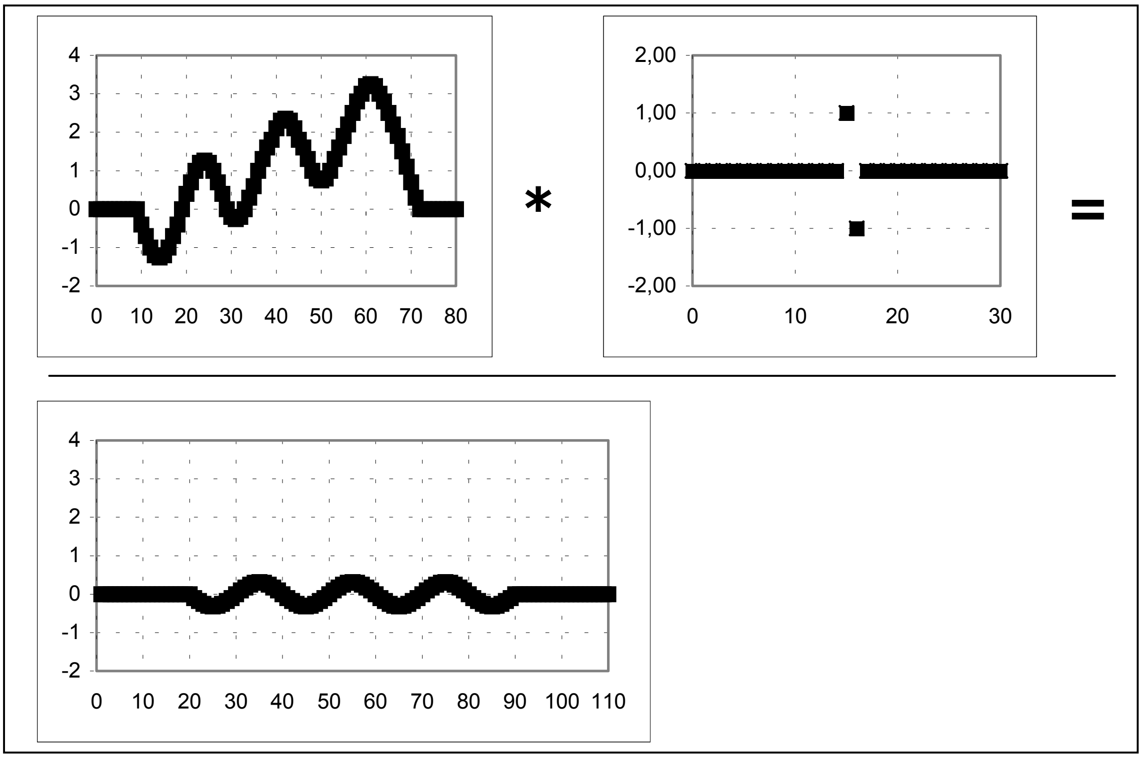


Рис.6-4. Пример сигналов, обработанных с использованием свертки.

Алгоритм входной стороны

На Рис.6-5 показана простая задача свертки: 9-точечный входной сигнал, x[n], пропускается через систему с 4-точечной импульсной характеристикой, h[n], и порождает выходной сигнал из 9+4-1=12 выборок, y[n]. На языке математики, x[n] свертываясь с h[n] производит y[n]. Это первая точка зрения на операцию свертки, основанная на фундаментальной концепции ЦОС: разделить входной сигнал, пропустить составляющие через систему и синтезировать выходной сигнал. В приведенном примере, каждая из девяти выборок входного сигнала будет вносить в выходной сигнал масштабированную и смещенную версию импульсной характеристики. Эти девять сигналов показаны на Рис.6-6. При сложении этих сигналов получается выходной сигнал, y[n].

Давайте рассмотрим эти девять сигналов более детально. Начнем с четвертой выборки входного сигнала. Она носит индекс x[4] и содержит значение 1.4. При разложении сигнала он представляется в виде импульса 1.4δ[n-4]. После прохождения через систему, порождаемая выходная составляющая будет иметь вид 1.4h[n-4]. Этот сигнал показан в центральной рамке на Рис.6-6. Заметьте, что это импульсная характеристика h[n], умноженная на 1.4 и смещенная вправо на четыре выборки. Нулевые значения были присвоены выборкам 0-3 и 8-11. Чтобы сделать Рис.6-6 более понятным, для представления точек данных используются *квадраты*, а для добавленных нулей *ромбы*.

Теперь рассмотрим последнюю точку в выходном сигнале. Он имеет индекс x[8] и содержит значение -0.5. Как показано на нижнем правом графике Рис.6-6, x[8] выражается в импульсной характеристике, смещенной на восемь выборок и умноженной на -0.5. Выборкам 0-7 присваивается нулевое значение. Напоследок, рассмотрим выборки x[0] и x[7]. Они содержат нулевое значение, и поэтому порождают выходные составляющие, содержащие только нулевые значения.

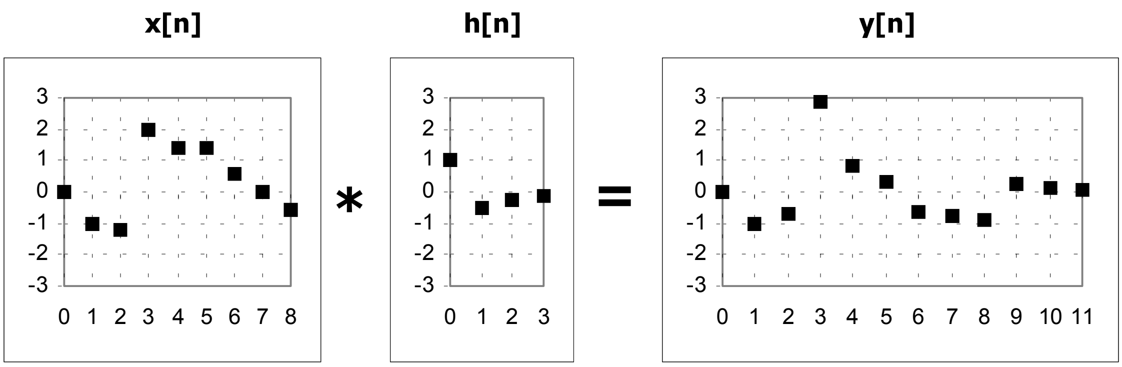


Рис.6-5. Пример операции свертки.

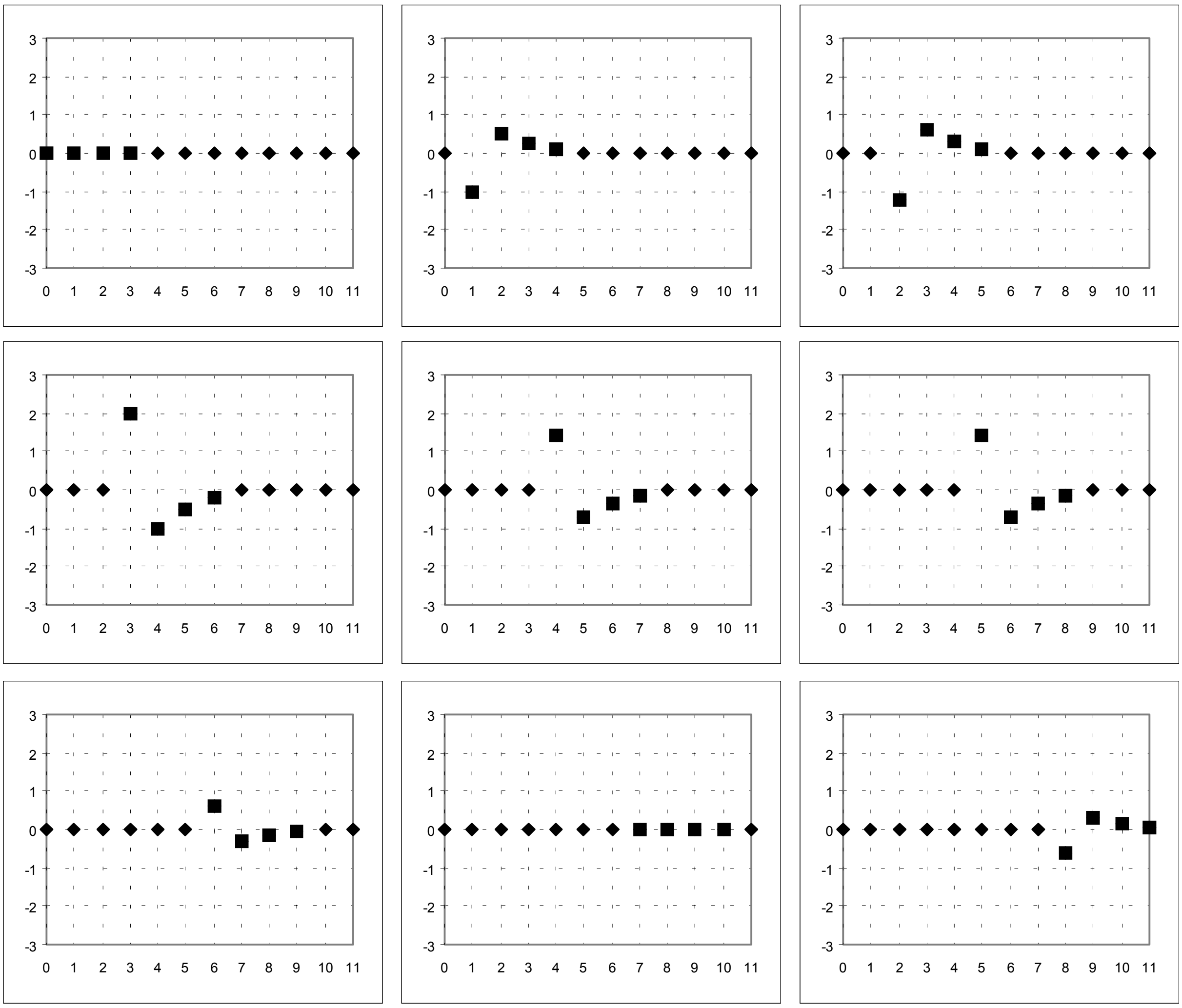


Рис.6-6. Выходные составляющие сигнала для свертки на Рис.6-5.

В этом примере, x[n] состоит из девяти точек, а h[n] из четырех. В примере, показанном на Рис.6-7 мы поменяем местами сигналы, сделав x[n], состоящим из четырех точек, а h[n] из девяти. Используются те же сигналы, которые были просто поменяны местами. Как показано на Рис.6-7, четыре выборки входного сигнала x[n] порождают четыре выходных составляющих, являющихся масштабированными и сдвинутыми версиями 9-точечной импульсной характеристики. Также как и в предыдущем примере, некоторым выборкам были добавлены нулевые значения.

Внимание! Выходной сигнал на Рис.6-7 *идентичен* выходному сигналу на Рис.6-5. Это не является ошибкой, а важным свойством. Сверка *коммутативна*: a[n]*b[n] = b[n]*a[n], т.е. важно не какой сигнал является входным, а какой импульсной характеристикой, а только то, что *два сигнала свернуты друг с другом*. С точки зрения математики смена мест двух сигналов не имеет физического смысла. Хотя входной сигнал и импульсная характеристика это два разных понятия, их замена не оказывает никакого влияния на результат. То, что предоставляет свойство коммутативности, является *математическим инструментом* для манипулирования выражениями с целью получения различных результатов.

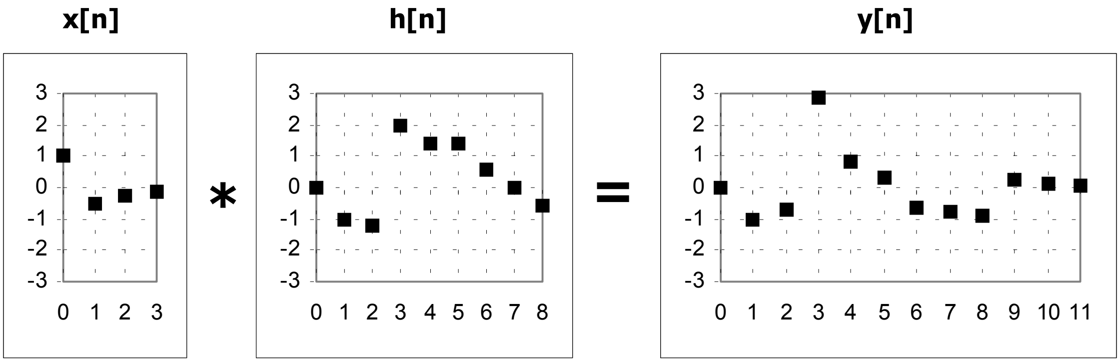
Программа расчета свертки, используемая в алгоритме входной стороны, представлена в Таблице 6-1. Помните, что программы в этой книге передают алгоритмы в простейшей форме, местами игнорируя хороший стиль программирования. Например, входы и выходы обрабатываются в *фиктивных* подпрограммах (строки 160 и 280), подразумевая, что нам неважно как определены эти операции. Не пропускайте программы в этой книге, они являются важной частью материала, и их необходимо их понять в деталях.

Программа свертывает входной сигнал из 81 выборки, содержащийся в массиве X[], с импульсной характеристикой из 31 точки, содержащейся в массиве H[], и синтезирует выходной сигнал из 111 точек, записываемый в массив Y[]. Используются те же размерности сигналов, что и на Рис.6-3 и Рис.6-4. Заметьте, что имена массивов записаны в верхнем регистре, хотя в предыдущих главах обсуждалось, что заглавные буквы зарезервированы для имен сигналов в частотной области. К сожалению, язык BASIC, используемый в этой книге, не поддерживает имена переменных в нижнем регистре. Заметьте также, что в строке 240 используется знак «*» для обозначения операции *умножения*.

Фиктивная подпрограмма в строке 160 записывает данные в массивы X[] и H[]. Строки 180-200 обнуляют значения в массиве Y[]. Это необходимо, так как Y[] используется как аккумулятор рассчитываемых выходных компонент. Оператор FOR в строке 220 управляет циклом, который проходит по всем точкам входного сигнала, X[]. Для каждой выборки входного сигнала, внутренний цикл (строки 230-250) рассчитывает масштабированную и смещенную версию импульсной характеристики и добавляет ее в массив, аккумулирующий выходной сигнал, Y[].

```
100 'РАСЧЕТ СВЕРТКИ, ИСПОЛЬЗУЮЩИЙ АЛГОРИТМ ВХОДНОЙ СТОРОНЫ
110 '
120 DIM X[80]               'входной сигнал, 81 точка
130 DIM H[30]               'импульсная характеристика, 31 точка
140 DIM Y[110]              'выходной сигнал, 111 точка
150 '
160 GOSUB XXXX              'фиктивная подпрограмма для загрузки X[] и H[]
170 '
180 FOR I% = 0 TO 110       'обнуление выходного массива
190     Y(I%) = 0
200 NEXT I%
210 '
220 FOR I% = 0 TO 80        'цикл для каждой точки в X[]
230     FOR J% = 0 TO 30    'цикл для каждой точки в H[]
240         Y[I%+J%] = Y[I%+J%] + X[I%] * H[J%]
250     NEXT J%
260 NEXT I% '(помните, что * это операция умножения!)
270 '
280 GOSUB XXXX              'фиктивная подпрограмма для сохранения Y[]
290 '
300 END
```

Таблица 6-1.



Выходные составляющие сигнала

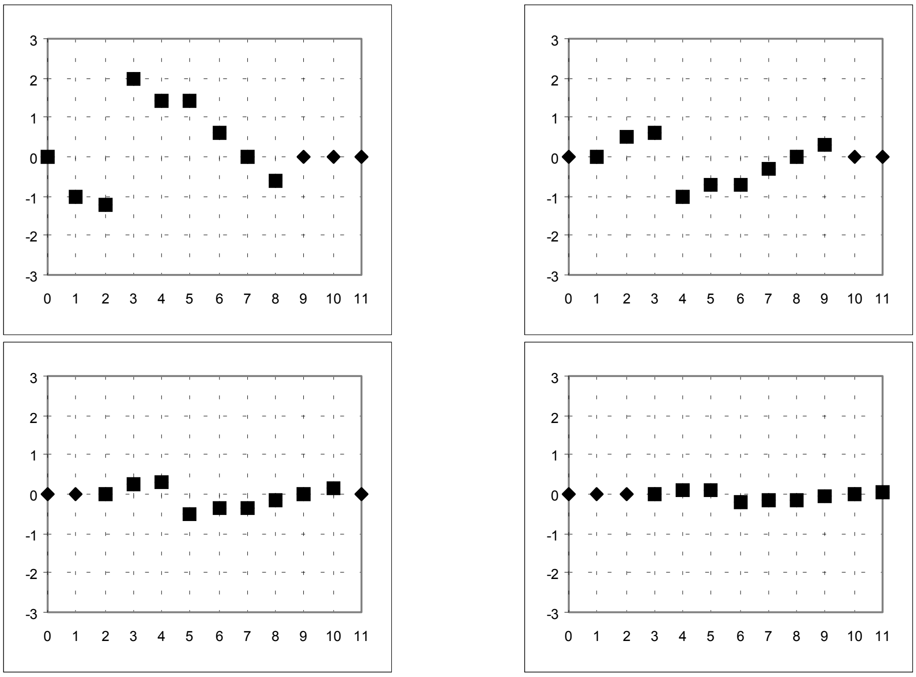


Рис.6-7. Второй пример операции свертки.

Прямая индексация в строке 240 может свести с ума! Давайте скажем, что мы находимся на полпути выполнения программы, так что идет работа с 40-ой выборкой, X[40], т.е. I%=40. Внутренний цикл, проходя через каждую точку импульсной характеристики, делает три действия. Первое, импульсная характеристика *масштабируется* умножением на величину входного сигнала. Если бы это было единственным действием, то строка 240 могла бы быть записана в виде: Y[J%]=X[40]*H[J%]. Второе, масштабированный импульс сдвигают на 40 точек вправо, суммируя это число с индексом, используемым в выходном сигнале. Это действие изменит строку 240 на: Y[40+J%]=X[40]*H[J%]. Третье, массив Y[] должен аккумулировать сигналы, полученные от каждой выборки. Поэтому новая информация должна складываться с уже существующей. Это приводит к окончательной команде: Y[40+J%]= Y[40+J%]+X[40]*H[J%]. Это очень запутанная команда, изучите ее тщательно!

Алгоритм выходной стороны

Первая точка зрения на операцию свертки рассматривал в виде анализа, как каждая выборка входного сигнала воздействует на выборки выходного сигнала. Вторая точка зрения отказывается от определения вклада, вносимого отдельными выборками входного сигнала в выборки выходного сигнала. Это важно как с математической, так и с практической стороны. Предположим, что имеется некоторый входной сигнал и некая импульсная характеристика, и необходимо найти сигнал их свертки. Наиболее простой метод состоял бы в том, чтобы написать программу, которая в цикле определяла бы каждую точку для *выходного сигнала*. Выражение для такого расчета выглядело бы приблизительно так: y[n] = *некоторая комбинация других переменных*. То есть, каждая выборка выходного сигнала должна быть вычислена независимо от других выборок. Это может реализовать алгоритм выходной стороны.

Давайте рассмотрим пример, как одна точка выходного сигнала связана с несколькими выборками во входном сигнале. Возьмем точку y[6] из примера на Рис.6-5. Значение в этой точке рассчитывается в виде суммы шести точек девяти выходных составляющих на Рис.6-6. Теперь внимательно рассмотрим эти девять составляющих и оценим, как они влияют на значение y[6]. Пять из выходных компонент *вносят* на шестой выборке нулевое значение (обозначены ромбами), и поэтому могут игнорироваться. Только четыре выходные составляющие имеют в шестой точке ненулевое значение. Эти составляющие были генерированы входными выборками: x[3], x[4], x[5] и x[6]. Сложение шестых выборок каждой из этих составляющих определяет y[6] как: y[6] = x[3] h[3]+ x[4] h[4] + x[5] h[5] + + x[6] h[0]. Это сумма четырех выборок входного сигнала, умноженных на четыре выборки импульсной характеристики.

Рис.6-8 иллюстрирует алгоритм выходной стороны, как *механизм свертки*, схему последовательных действий операции свертки. Представьте входной сигнал, x[n], и выходной сигнал, y[n], как показано на рисунке. Механизм свертки находится внутри пунктирной линии и свободно перемещается влево и вправо. Он позиционируется так, чтобы его выход был выровнен с выходной выборкой, которая должна быть вычислена. Четыре выборки входного сигнала попадают на входы механизма свертки. Эти величины умножаются на соответствующие значения точек импульсной характеристики, и произведения складываются. Таким образом получают значение выходного сигнала в определенной точке. Например, y[6], как показано, вычисляется из четырех входных выборок: x[3], x[4], x[5] и x[6].

Чтобы вычислить значение y[7], механизм свертки перемещаются на одну выборку вправо. Это приводит к тому, что в механизм входят другие четыре выборки от x[4] до x[7]. Этот процесс повторяется для всех точек выходного сигнала, которые требуют вычисления.

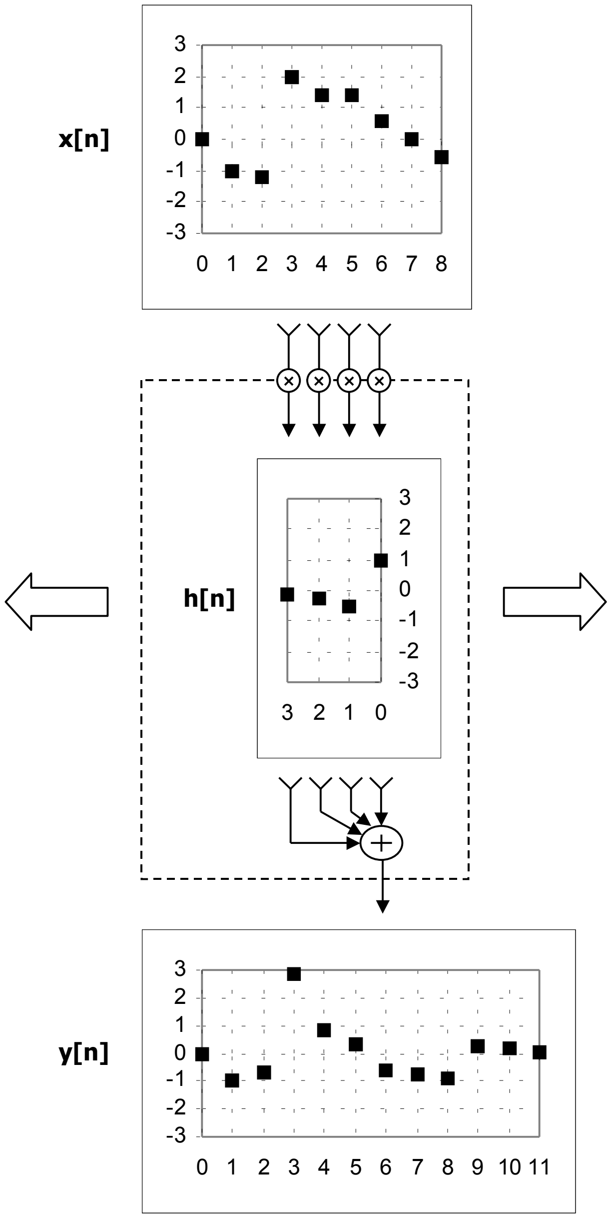
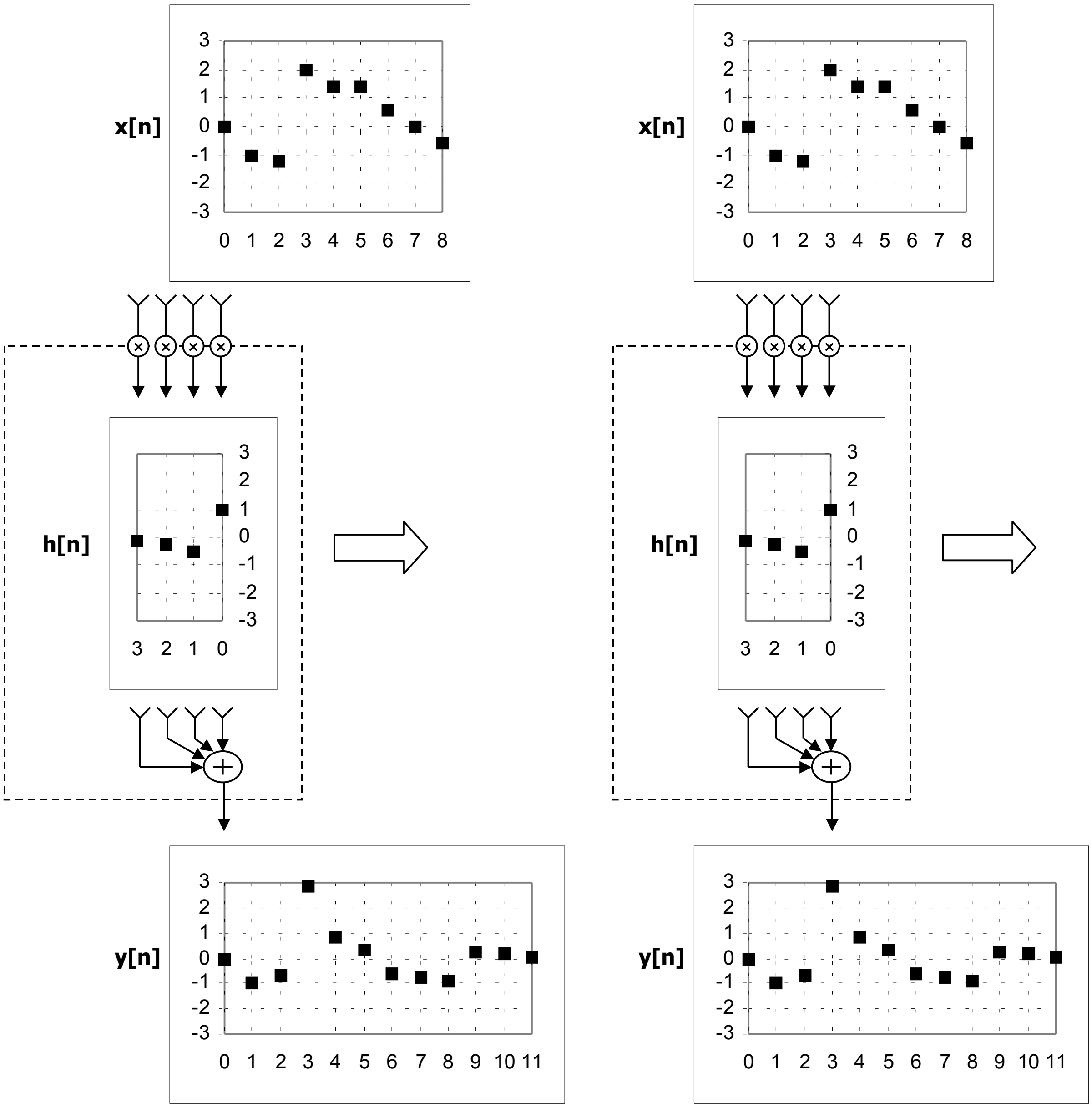


Рис.6-8. Механизм свертки.

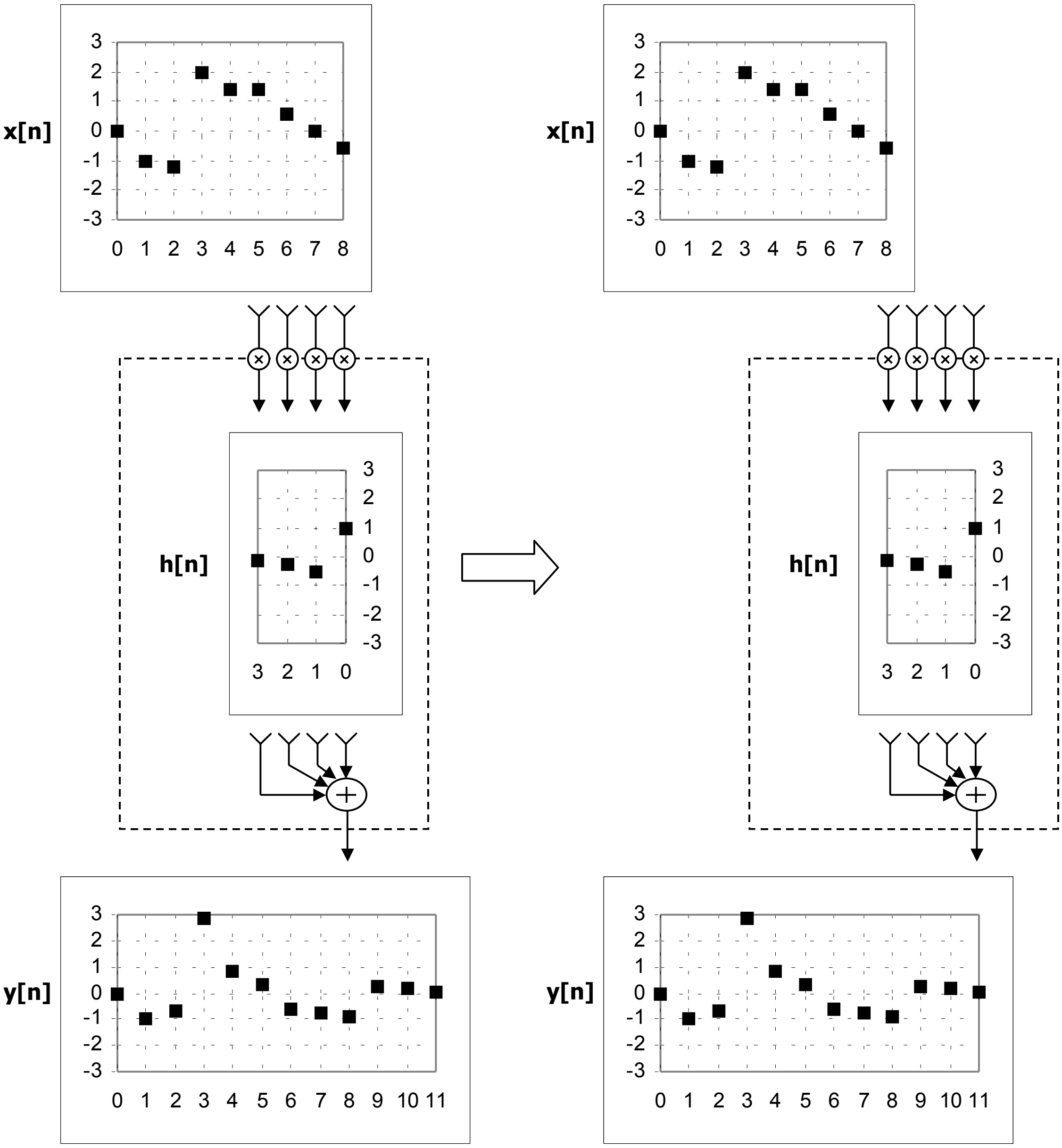
Расположение импульсной характеристики внутри механизма свертки очень важно. Импульсная характеристика *переворачивается слева направо*. Это приводит к размещению нулевой выборки справа. Сравните это с импульсной характеристикой на Рис.6-5. Зачем требуется это переворачивание? Это следует из математики. Импульсная характеристика описывает, каждая точка входного сигнала влияет на выходной сигнал. Это выражается в том, что каждая точка выходного сигнала зависит от точек входного сигнала, взвешенных перевернутой импульсной характеристикой.



А. установлено для расчета x[0]   **Б. установлено для расчета x[3]**

Рис.6-9. Механизм свертки в работе.

Рис.6-9 показывает, как используется механизм свертки для вычисления нескольких выборок выходного сигнала. Диаграмма показывает существующую неприятность при свертке. На Рис.6-9а механизм свертки расположен в крайнем левом положении, с выходом на y[0]. В этой позиции он пытается получить на входы значения с выборок x[-3], x[-2], x[-1] и x[0]. Проблема состоит в том, что три из них не существуют! Такая же дилемма существует, когда механизм свертки находится в крайне правом положении (Рис.6-9г) и пытается получить значения с выборок x[9], x[10] и x[11].



В. установлено для расчета x[8]

Г. установлено для расчета x[11]

Рис.6-9. Механизм свертки в работе (продолжение).

Единственным способом решить эту проблему является *внедрение* несуществующих выборок. Это реализуется добавлением дополнительных в конец сигнала выборок с *нулевым значением*. Это называется *заполнением* сигнала нулями. Вместо попытки получить несуществующие выборки, механизм свертки будет обращаться к выборкам, которые содержат нулевые значения. Поскольку этот ноль участвует в умножении, математически результат будет таким же, как при игнорировании несуществующих выборок.

Важно, что самые левые и самые правые выборки выходного сигнала базируются на *неполной* информации. На языке ЦОС, *импульсная характеристика не полностью погружена во входной сигнал*. Если импульсная характеристика имеет M точек, первые и последние M-1 выборок в выходном сигнале базируются на меньшей информации, чем выборки между ними. Это аналогично действию электронной цепи, которой требуется некоторое время для стабилизации после включения питания. Разница состоит лишь в том, что это время мало и игнорируется в электронике, но очень заметно в ЦОС.

Рис.6-10 показывает пример, какие проблемы может вызвать этот эффект. Входной сигнал является синусоидой с постоянной составляющей. Требуется убрать постоянную составляющую, сохранив при этом форму синусоиды. Для этого применяется фильтр высоких частот с импульсной характеристикой, показанной на диаграмме. Проблема состоит в том, что первые и последние 30 точек недостоверны! Форма этих оконечных регионов может быть понята, если представить входной сигнал, заполненный слева (x[-1] до x[-30]) и справа (x[81] до x[110]) тридцатью нулевыми выборками. Выходной сигнал может быть рассмотрен как отфильтрованная версия этого удлиненной волны. Этот «концевой эффект» является распространенной проблемой в ЦОС. Как основное правило, следует ожидать, что первые и последние выборки при обработке сигналов окажутся непригодными.

Теперь перейдем к математике. Принимая за основу механизм свертки, можно написать стандартное выражение для расчета свертки. Если x[n] является N-точечным сигналом с индексом от 0 до N-1, и h[n] является M-точечным сигналом с индексом от 0 до M-1, то свертка этих сигналов: y[n] = x[n]*h[n] будет N+M-1 точечным сигналом с индексом от 0 до N+M-2, рассчитываемый как:

$$y[i] = \sum_{j=0}^{M-1} h[j]\,x[i-j]$$

Уравнение 6-1. Расчет свертки.

Это выражение называется *суммой свертки*. Оно позволяет вычислить каждую точку выходного сигнала независимо от других. Индекс, i, определяет какая выборка выходного сигнала будет вычислена, и какой этому соответствует сдвиг позиции механизма свертки. В компьютерных программах, выполняющих операцию свертки, цикл использует этот индекс для прохождения каждой выборки выходного сигнала. Для вычисления одной из точек выходного сигнала внутри цикла механизма свертки используется индекс j. Поскольку j изменяется от 0 до M-1, каждая выборка в импульсной характеристике, h[j], умножается на соответствующую выборку входного сигнала, x[i-j]. Все произведения складываются для получения значения выборки выходного сигнала. Изучите выражение для расчета свертки. Многие операции ЦОС основаны на нем. Пусть вас не смущает символ n. Он показывает, что *некоторая* переменная является индексом в массиве. Иногда выражения записываются в форме: y[] = x[]*h[], чтобы избежать использования бессмысленного символа).

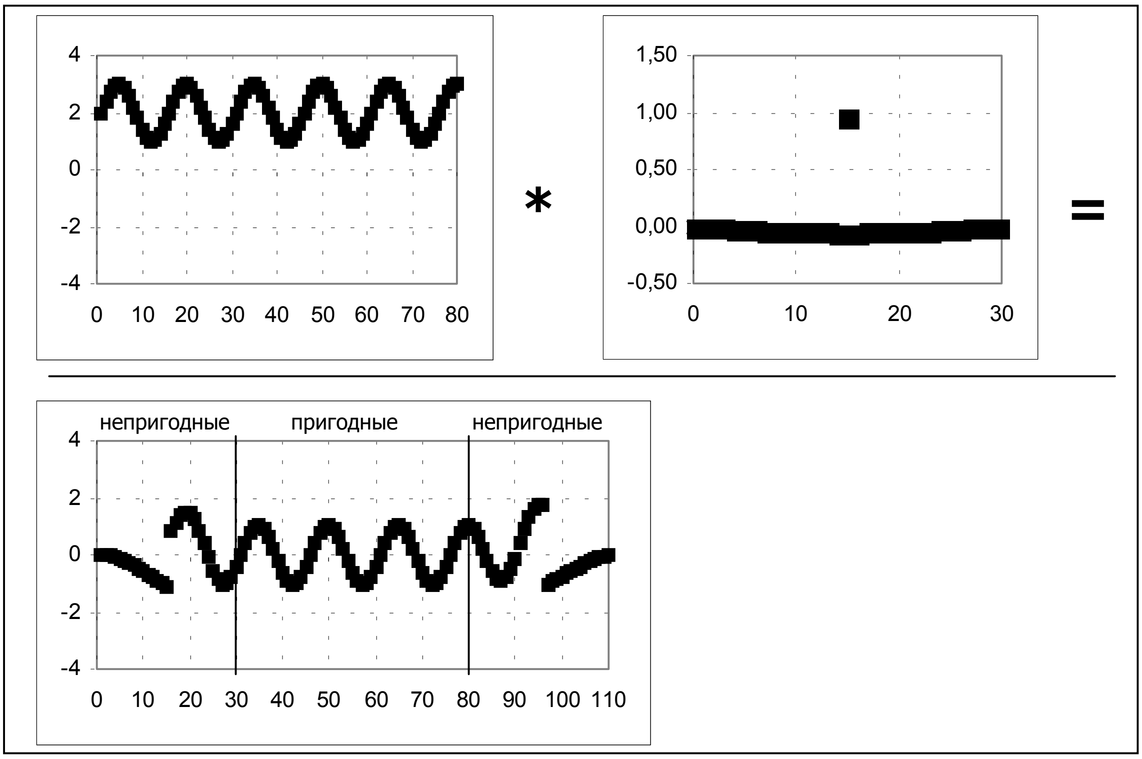


Рис.6-10. «Концевой эффект» при свертке.

Таблица 6-2 показывает программу для расчета свертки, построенную по алгоритму выходной стороны и прямо использующей Уравнение 6-1. Эта программа производит такой же выходной сигнал, как и приведенная выше программа, построенная по алгоритму входной стороны. Основное отличие между программами состоит в том, что цикл в алгоритме входной стороны проходит каждую выборку *входного сигнала* (строка 220 в Таблице 6-1), в то время как цикл в алгоритме выходной стороны проходит каждую выборку *выходного сигнала* (строка 180 в Таблице 6-2).

Рассмотрим работу программу детально. Цикл FOR…NEXT в строках 180-250 проходит каждую выборку выходного сигнала, используя индекс I%. Внутренний цикл, построенный в строках 200-240, вычисляет значение выходной выборки, Y[I%]. Обнуление значения Y[I%] в строке 190 позволяет накапливать ее значение внутри механизма свертки. Цикл FOR…NEXT в строках 200-240 производит прямой расчет по Уравнению 6-1. Индекс J% проходит по каждой выборке в импульсной характеристике. В строка 230 производится умножение каждой выборки импульсной характеристики, H[J%], на соответствующую выборку входного сигнала, X[I%-J%], и добавление результата в аккумулятор.

В строке 230 используется значение выборка входного сигнала X[I%-J%]. Строки 210 и 220 предотвращают возможный выход за пределы массива, X[0]-X[80]. Другими словами, программа контролирует неопределенные выборки входного сигнала, *игнорируя* их. В другом варианте, можно было определить область входного сигнала от X[-30] до X[110], заполнив тридцать оконечных выборок нулевыми значениями. Наконец, можно было изменить строку 180, указав циклу FOR…NEXT работать в пределах от 30 до 80, а не от 0 до 110. В этом случае, программа считала бы только те выборки, в которые *полностью погружается импульсная характеристика*. Самое главное, что *необходимо* использовать один из этих трех методов, иначе произойдет аварийное завершение программы при попытке прочитать данные за границами массива.

```
100 ' РАСЧЕТ СВЕРТКИ, ИСПОЛЬЗУЮЩИЙ АЛГОРИТМ ВЫХОДНОЙ СТОРОНЫ
110 '
120 DIM X[80]              'входной сигнал, 81 точка
130 DIM H[30]              'импульсная характеристика, 31 точка
140 DIM Y[110]             'выходной сигнал, 111 точка
150 '
160 GOSUB XXXX             'фиктивная подпрограмма для загрузки X[ ] и H[ ]
170 '
180 FOR I% = 0 TO 110      'цикл по каждой выборке в Y[ ]
190     Y[I%] = 0          'обнуление значения выборки выходного массива
200     FOR J% = 0 TO 30   'цикл по каждой выборке в H[ ]
210         IF (I%-J% < 0) THEN GOTO 240
220         IF (I%-J% > 80) THEN GOTO 240
230         Y(I%) = Y(I%) + H(J%) t X(I%-J%)
240     NEXT J%
250 NEXT I%
260 '
270 GOSUB XXXX             'фиктивная подпрограмма для сохранения Y[ ]
280 '
290 END
```

Таблица 6-2.

Сумма взвешенных входов

Характеристики линейной системы полностью описываются ее импульсной характеристикой. Это является основой алгоритма входной стороны: каждая точка входного сигнала вносит в выходной сигнал масштабированную и смещенную версию импульсной характеристики. Математическая последовательность этого приводит к алгоритму выходной стороны: каждая точка выходного сигнала принимает вклад нескольких точек входного сигнала, умноженных на *перевернутую* импульсную характеристику. Несмотря на то, что все правильно, это не дает полной картины того, почему свертка важна в ЦОС.

Вернемся к механизму свертки, представленному на Рис.6-8, и проигнорируем, что сигнал внутри пунктира это *импульсная характеристика*. Представим его как набор *взвешенных коэффициентов*, который встроен в схему последовательности операций. С этой точки зрения, каждая выборка выходного сигнала является суммой взвешенных входов. Каждая выборка выходного сигнала находится под влиянием области выборок входного сигнала, определяемым выбранными взвешенными коэффициентами. Например, представьте десять взвешенных коэффициентов, значение каждого из которых одна десятая. Это приведет к тому, что каждая выборка выходного сигнала составит *среднее* от десяти выборок входного сигнала.

Принимая это, взвешенные коэффициенты могут не ограничиваться по левой стороне при расчете выходной выборки. Например, Рис.6-8 показывает вычисление y[6] из выборок: x[3], x[4], x[5] и x[6]. Рассматривая механизм свертки как сумму взвешенных входов, взвешенные коэффициенты могут быть выбраны симметрично около выходной выборки. Например, y[6] может принимать вклад от входных выборок: x[4], x[5], x[6], x[7] и x[8]. Используя такое же обозначение, как на Рис.6-8, взвешенные коэффициенты для этих входов содержатся в h[2], h[1], h[0], h[-1] и h[-2]. Другими словами, импульсная характеристика, которая соответствует нашему выбору симметричных взвешенных коэффициентов, требует использования отрицательных индексов. Мы вернемся к этому вопросу в следующей главе.

Математически, здесь есть одна концепция: свертка определяется по выражению 6-1. Тем не менее, научные и инженерные задачи приближаются к этой концепции с двух разных направлений. Иногда можно представить систему в терминах, как выглядит ее импульсная характеристика. В другой раз систему можно представить в виде набор взвешенных коэффициентов.