Представление чисел с фиксированной точкой

Числа в цифровых устройствах представляются с фиксированной или плавающей точкой. Настоящая лекция посвящена представлению чисел с фиксированной точкой, плавающая точка рассматривается в следующей лекции.

Запись числа с фиксированной точкой

При представлении с фиксированной точкой число характеризуется размером слова (сколько бит оно занимает), положением точки/запятой и наличием знака. Дополнительно могут применяться смещение числа и масштабирование.

В двоичном виде с фиксированной точкой знаковое или беззнаковое число представляется в виде на рис. 1. Для записи числа отводится N битов, самый левый разряд – старший (most significant bit, MSB), самый правый – младший (least significant bit, LSB). Точкой (или запятой) разделяется целая и дробная части числа. Для записи знаковых чисел может применяться прямой, обратный или дополнительный коды (см. лекцию 4). Повсеместно используется дополнительный код, в данной лекции подразумевается, что все знаковые числа представлены в дополнительном коде.

b_{N-1}	b_{N-2}	b_{N-3}	…	b_5	b_4	,	b_3	b_2	b_1	b_0
MSB										**LSB**

Рис. 1. Представление числа с фиксированной точкой

Ключевым моментом при рассмотрении числа с фиксированной точкой является положение точки – разделителя целой и дробной частей числа. С помощью точки выполняется масштабирование числа. Другими словами, в одном и том же количестве разрядов можно записывать или большие (точка смещается вправо), или маленькие числа (точка смещается влево).

Суть идеи масштабирования заключается в следующем. Имея N-разрядное представление числа X (т.е. некоторую последовательность из N нулей и единиц), и зная, что точка размещена между F и $F + 1$ битами в записи числа, считая с единицы справа налево (от LSB к MSB), число X можно записать в следующем виде:

$$X = Q \cdot 2^{-F}. \quad (1)$$

В (1) Q – это целое число, которое записано в N-разрядном представлении числа X как будто точка отсутствует, а степень двойки, $-F$, называется экспонентой (или порядком) числа с фиксированной точкой. Таким образом, число X – это искомое, представляемое число, а Q – это его запись, машинное представление. Получается, что при записи числа с фиксированной точкой непосредственно в N-разрядное слово записывается только некоторое целое число, а масштаб, т.е. смещение точки не записывается. Более того, положение точки ничем не ограничивается, она может находиться далеко слева от MSB или справа от LSB: разрядность в представлении числа никак не связана с размером дробной части F. Он может быть больше N, лежать в диапазоне от 0 до N, а может быть даже отрицательным.

Например, пусть для записи беззнакового числа используется четыре разряда и в этих разрядах записаны биты b_3, b_2, b_1 и b_0. Тогда в зависимости от значения экспоненты запись $b_3b_2b_1b_0$ может соответствовать разным числам X:

- целое число
$$X = Q = 2^3b_3 + 2^2b_2 + 2^1b_1 + 2^0b_0,\ F = 0;$$
- целое число, умноженное на степень двойки, например, на 8:
$$X = 2^3Q = 2^6b_3 + 2^5b_2 + 2^4b_1 + 2^3b_0,\ F = -3;$$
- дробное число, с числом дробных разрядов, например, 2:

$$X = 2^{-2}Q = 2^1 b_3 + 2^0 b_2 + 2^{-1} b_1 + 2^{-2} b_0,\ F = 2;$$

– дробное число, с числом дробных разрядов больше четырех, например, 6:

$$X = 2^{-6}Q = 2^{-3} b_3 + 2^{-4} b_2 + 2^{-5} b_1 + 2^{-6} b_0,\ F = 6.$$

Во втором случае, можно было бы записать $\boldsymbol{b_3 b_2 b_1 b_0 000.}$, а в последнем – $\boldsymbol{0.00 b_3 b_2 b_1 b_0}$. Тем не менее, дополнительные нули никогда не изменятся на единицы, если только не будет явного преобразования записи двоичного числа. Поэтому при записи числа эти нули не указываются.

Рассмотрим еще один пример. Пусть есть восьмиразрядное знаковое двоичное число 0011 1100, а число дробных разрядов $F = 12$. Тогда Q = 60 и искомое число X в соответствии с (1) равно $60 \cdot 2^{-12} = 0{,}0146484375$.

Очень важно понимать, что сама точка и ее положение существуют только в голове разработчика, устройство ничего не знает о точке и само никак не принимает в расчет, где она находится. При выполнении базовых математических операций, таких как сложение/вычитание или умножение, используются одни и те же схемы независимо от положения точки. Более того, устройство, схемы, реализующие математические операции, ничего не знают даже о знаке числа.

Важным отличием представления числа с фиксированной точкой от представления числа с плавающей точкой является тот факт, что в первом отсутствует указание значения экспоненты, поэтому числа с фиксированной точкой не ограничены, ограничен только диапазон возможных значений.

Для краткой записи представления чисел с фиксированной точкой используют обозначения вида (sgn,N,F), в котором sgn описывает наличие знака в представлении (0 – беззнаковое, 1 – знаковое), N – разрядность слов, F – количество разрядов после запятой. Например, запись (0,8,0) означает 8-разрядное целое беззнаковое число, а (1,16,15) – 16-разрядное, знаковое число с позицией точки сразу после знакового разряда.

Масштабирование в записи чисел с фиксированной точкой

Динамический диапазон чисел с фиксированной точкой значительно меньше, чем динамический диапазон чисел в плавающей точке, при одинаковой разрядности. Для снижения ошибок квантования и предотвращения переполнений числа с фиксированной точкой масштабируют. Простое масштабирование положением точки было рассмотрено в предыдущем пункте. В общем случае может применяться масштабирование не по степеням двойки в сочетании с ненулевым постоянным смещением:

$$X = Q \cdot (S \cdot 2^{-F}) + O. \qquad (2)$$

В (2) S – дополнительный масштабирующий коэффициент, $1 \le S < 2$, F, как и в (1), определяет положение точки, O – задает постоянное смещение.

Также как и в предыдущем пункте при записи числа X в соответствии с (2) непосредственно в цифровом устройстве хранится только N разрядов числа Q. Все остальные параметры числа существуют только в голове разработчика.

Точность представления и диапазон чисел при представлении с фиксированной точкой

Для записи числа с фиксированной точкой используется N разрядов. При такой разрядности можно записать 2^N разных чисел. С учетом масштабирования (2) и выбранной разрядности N диапазон возможных для записи чисел будет следующим (рис. 2):

для беззнаковых, т.е. неотрицательных, чисел – от O до $S \cdot 2^{-F} \cdot (2^N - 1) + O$;

для знаковых чисел в дополнительном коде – от $S \cdot 2^{-F} \cdot (-2^{N-1}) + O$ до $S \cdot 2^{-F} \cdot (2^{N-1} - 1) + O$.

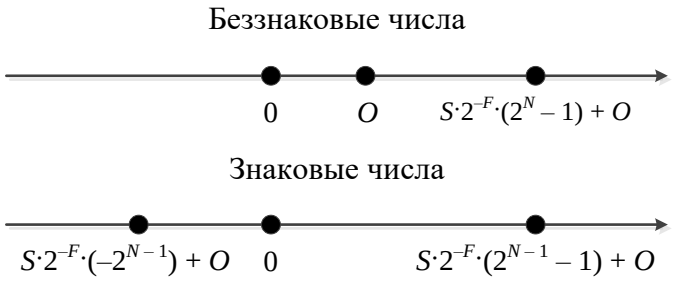


Рис. 2. Диапазоны чисел с фиксированной точкой

Например, для беззнаковых целых чисел с $S = 1$, $F = 0$ и $O = 0$ минимальное представимое число 0, а максимальное – $2^N – 1$. Обратим внимание, что максимальное число на 1 меньше 2^N, т.к. для представления нуля необходимо выделить одну возможную комбинацию. Для знаковых чисел одна половина диапазона выделяется для представления отрицательных чисел, а вторая – для представления неотрицательных чисел, включая ноль. Например, для знаковых целых чисел с $S = 1$, $F = 0$ и $O = 0$ наибольшее число равно $2^{N-1} – 1$, а наименьшее отрицательное – $(–2^{N-1})$. Это значит, что модуль максимального положительного числа на 1 меньше модуля минимального отрицательного. Поэтому инверсию знаковых чисел с фиксированной точкой необходимо выполнять аккуратно: инвертировать минимальное отрицательное число без изменения разрядности нельзя.

Разность двух последовательных чисел с фиксированной точкой называется точностью представления и определяется весом младшего разряда. Другими словами, точность представления – это минимальное число, которое необходимо добавить к какому-либо числу, чтобы получить другое число при выбранных параметрах представления с фиксированной точкой. Точность представления с фиксированной точкой равна $S·2^{–F}$.

В таблице 1 приведены примеры точности представления и диапазоны 8-разрядных знаковых и беззнаковых чисел для различных значений параметров F при $S = 1$ и $O = 0$.

Таблица 1. Точности и диапазоны 8-разрядных чисел при F =1,0,…(–8), S = 1 и O = 0

$2^{–F}$	Точность	Диапазон значений	
		Беззнаковые	Знаковые
2^1	2	0…510	–256…254
2^0	1	0…255	–128…127
$2^{–1}$	0,5	0…127,5	–64…63,5
$2^{–2}$	0,25	0…63,75	–32…31,75
$2^{–3}$	0,125	0…31,875	–16…15,875
$2^{–4}$	0,0625	0…15,9375	–8…7,9375
$2^{–5}$	0,03125	0…7,96875	–4…3,96875
$2^{–6}$	0,015625	0…3,984375	–2…1,984375
$2^{–7}$	0,0078125	0…1,9921875	–1…0,9921875
$2^{–8}$	0,00390625	0…0,99609375	–0,5…0,49609375

Переполнение и округление чисел с фиксированной точкой

При выполнении различных операций над числами с фиксированной точкой может меняться их разрядность. Например, в общем случае при сложении двух чисел в одинаковой фиксированной точке для записи результата необходимо на один разряд больше, чем у слагаемых. Для записи результата перемножения двух чисел требуется число разрядов, равное сумме разрядностей операндов. Если не выполнять увеличения разрядностей, может произойти ошибка.

Соответственно, при последовательном выполнении сотен различных математических операций разрядности чисел должны расти и, в конце концов, становиться огромными. Поэтому при

работе с фиксированной точкой разрядность чисел ограничивают. В общем случае при выполнении операции разряды могут добавляться как слева (со стороны MSB), так и справа (со стороны LSB). В первом случае увеличивается диапазон чисел (например, сложили целые числа), а во втором – точность представления (например, перемножили числа типа (1,16,15), получили результат в (1,32,30). Соответственно и сокращать разрядность можно со стороны старших или младших разрядов.

Ограничение разрядности со стороны младших разрядов числа называется округлением. Простейшим округлением является отбрасывание лишних разрядов. Например, после перемножения чисел типа (1,16,15) получили число типа (1,32,30) и отбросили последние 15 разрядов результата, получив число (1,17,15). Недостатком такого метода округления является то, что исходное число всегда округляется в меньшую сторону. Т.е. ошибка округления (разность между исходным и округленным числом) систематическая и всегда имеет один знак. Это значит, что в спектре округленного сигнала появляется дискретная составляющая. Для устранения этого недостатка при отбрасывании разрядов к результирующему числу добавляют старший отброшенный разряд. В этом случае систематическая ошибка округления отсутствует, но вычислительная сложность округления выше.

Пусть исходное беззнаковое число $222{,}67578125_{dec}$ = 1101 1110, 1010 1101. Округлим его до четырех знаков после запятой двумя рассмотренными методами.

1. Отбрасываем младшие 4 разряда: 1101 1110, 1010 = $222{,}625_{dec}$.
Ошибка округления 222,67578125 – 222,625 = 0,05078125.

2. Отбрасываем младшие 4 разряда с добавлением старшего отброшенного к результату: 1101 1110, 1010 + 1 = 1101 1110, 1011 = $222{,}6875_{dec}$.
Ошибка округления 222,67578125 – 222,6875 = –0,01171875.

Если для записи результата операции используется недостаточная разрядность, то может произойти переполнение. Например, если при сложении двух 8-разрядных знаковых целых чисел записывать результат тоже в 8-разрядное слово, то, сложив числа 100 ($0110\ 0100_{bin}$) и 90 ($0101\ 1010_{bin}$), получим в двоичном представлении 10111110. Т.к. число знаковое, то старший разряд интерпретируется как знак, т.е. имеем число –66, а не 190. Изменение знака числа при округлении – это серьезная ошибка, которая может приводить к неожиданному поведению алгоритмов.

Таким образом, ограничивать разрядность со стороны старших разрядов необходимо очень аккуратно, иначе можно столкнуться с эффектом переполнения. Для этого перед отбрасыванием старших разрядов необходимо проверить, не отбрасываются ли значащие разряды, т.е. незнаковые разряды отличные от 0 у неотрицательного числа, и разряды, отличные от 1 у отрицательного. Причем в эту проверку должен входить и самый старший разряд, который остается.

Пусть исходное знаковое число $+222{,}67578125_{dec}$ = 0 <u>110***1***</u> 1110, 1010 1101. Ограничим его разрядность со стороны MSB так, чтобы в полученном числе осталось 13 разрядов. Это значит, что необходимо убрать старшие четыре разряда. Если их просто отбросить, то будут потеряны не только значащие разряды, но еще изменится и знак числа, т.к. в старшем разряде результата появится выделенная курсивом единица. Для корректного сокращения разрядности необходимо выполнить следующие действия.

Проверить, есть ли единицы в подчеркнутых разрядах. Если нет, то их можно просто отбросить, и слева дописать нулевой знаковый разряд. В нашем примере в этих разрядах есть единицы, значит ближайшее к исходному числу число в новой разрядности – это максимально возможное число: 0 1111, 1111 1111 = $15{,}99609375_{dec}$.

Скользящее среднее

Определение

Пусть есть некая последовательность комплексных чисел x_k, k = …, –2, –1, 0, 1, 2, …: $\{x_k\}_{-\infty}^{\infty}$. Скользящим средним с окном шириной N последовательности $\{x_k\}_{-\infty}^{\infty}$ называется следующая величина:

$$m_k = \sum_{i=0}^{N-1} \alpha_i x_{k-i} \,, \tag{3}$$

где α_i – взвешивающие коэффициенты (отсчеты взвешивающей функции). В случае простого скользящего среднего $\alpha_i = 1/N$.

При расчете простого скользящего среднего для каждого k вычисляется среднее арифметическое последних N отсчетов входной последовательности. Таким образом, скользящее среднее также представляет из себя последовательность чисел. В случаях, когда последовательность чисел $\{x_k\}$ конечна, недостающие элементы при расчете принимаются равными 0. В табл. 2 приведен пример расчета скользящего среднего окном шириной 4 отсчета для первых 15 положительных четных чисел.

Таблица 2. Пример расчета скользящего среднего

k	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
$\{x_k\}$	0	2	4	6	8	10	12	14	16	18	20	22	24	26	28
$\{m_k\}$	0	0,5	1,5	3	5	7	9	11	13	15	17	19	21	23	25

В общем случае скользящее среднее – это такая функция, значение которой в каждой точке определения равно среднему значению исходной функции за предыдущий период заданной величины. Усреднение необязательно арифметическое, может применяться более сложная взвешивающая функция, например, линейная, экспоненциальная. Взвешивающие функции полезны, когда некоторые значения исходной функции необходимо сделать более важными, чем другие. Например, при расчете бегущего среднего временных рядов последние (самые свежие) данные часто являются более значимыми, чем старые данные. При расчете временных рядов биржевых цен необходимо учитывать не только момент сделки, но и ее объем.

Отметим, что скользящее среднее можно строить не только на основе арифметического усреднения, но и других типов средних (среднее степенное, геометрическое и т.п.).

Далее ограничимся рассмотрением только простого скользящего среднего с арифметическим усреднением.

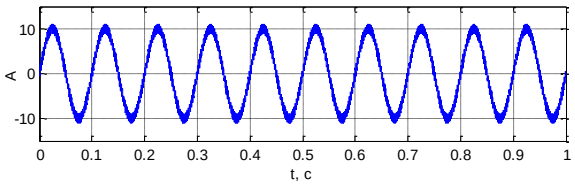
Скользящее среднее как фильтр нижних частот

При малом размере окна сигнал на выходе скользящего среднего практически полностью повторяет поведение исходной функции. При большом размере окна на выходе скользящего среднего “быстрые” флуктуации входного сигала исчезают за счет усреднения, остается только “общее” поведение входного сигнала. Это типичный пример работы фильтра нижних частот: скользящее среднее выполняет фильтрацию входных данных, не реагируя на их быстрые изменения, но оставляя общую форму. Чем больше окно скользящего среднего, тем более плавным получается сигнал на выходе.

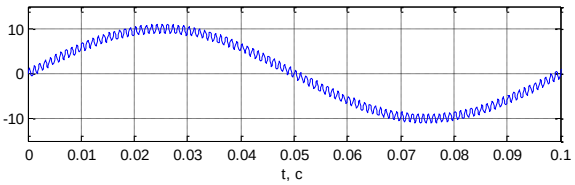
На рис. 3 показан пример[1] фильтрации синусоидального сигнала скользящим средним с разным размером окна: 4 и 8. На вход скользящего среднего поступает сумма двух гармонических

[1] Исходный код Matlab рассмотренного примера представлен в конце описания этой лабораторной работы.

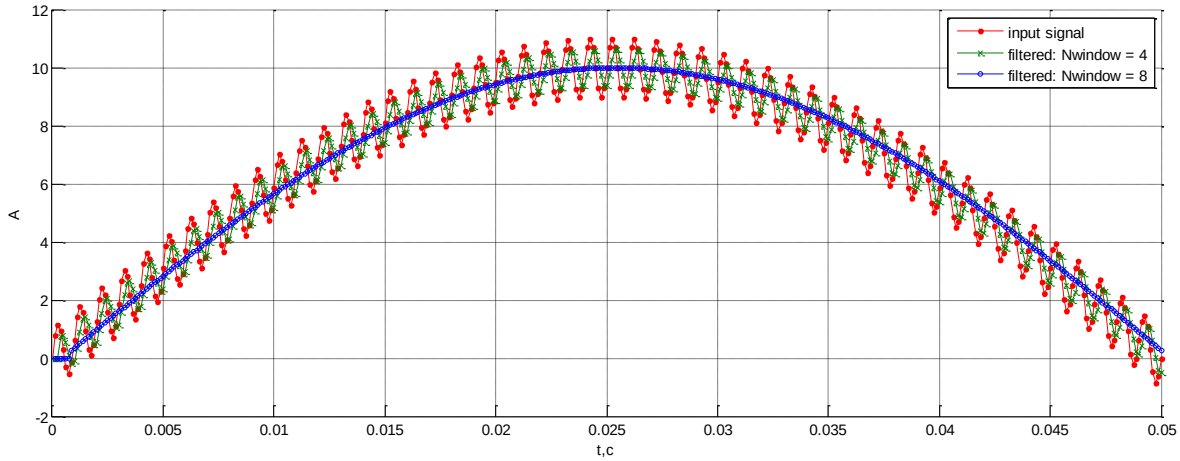
сигналов: полезного (частота 10 Гц, амплитуда 10) и мешающего (частота 1000 Гц, амплитуда 1). Частота дискретизации 8000 Гц. Из рис. 2,а-б видно, что полезный сигнал фактически промодулирован по амплитуде мешающей синусоидой, частота которой велика, по сравнению с частотой полезного сигнала. После пропускания сигналов через ФНЧ на основе бегущего среднего получаем следующую картину: при малом размере окна фильтра помеха незначительно снижается, тогда как при большом размере окна – полностью подавляется.



а) Исходный сигнал



б) Исходный сигнал, увеличенный масштаб



в) Сигналы: исходный(красный), отфильтрованный скользящим средним с окном N = 4 (зеленый), отфильтрованный скользящим средним с окном N = 8 (синий). Увеличенный масштаб

Рис. 3. Пример фильтрации сигнала скользящим средним

Скользящее среднее как простейший ФНЧ применяется при обработке сигналов, когда необходимо “сгладить” данные от каких-либо датчиков, в петлях цифровых ФАПЧ, в так называемых CIC-фильтрах (Cascaded Integrator Comb, разновидность вычислительно простых ФНЧ, не требующих умножителей для своей реализации). Различные варианты скользящего среднего в качестве технического индикатора широко применяются при игре на финансовой бирже. Также скользящее среднее используется для построения автокорреляторов (блоков, рассчитывающих коэффициент корреляции между двумя участками сигнала, используется для обнаружения сигналов).

Безусловным достоинством ФНЧ на основе скользящего среднего является его низкая вычислительная сложность (см. ниже).

Архитектура усредняющего фильтра

При прямой реализации выражения (3) в FPGA необходимо на каждом интервале дискретизации (на каждый входной отсчет) выполнить $(N-1)$ операций суммирования. При этом необходимо хранить в памяти $(N-1)$ предыдущих элементов последовательности. Выполнение $(N-1)$ операций сложения при большом N ($N > 10$) неудобно: необходимо организовывать дерево сумматоров при этом либо будет невысокое общее быстродействие, либо потребуется большое количество триггеров и появится дополнительная задержка вывода данных. Тем не менее, для случая простого скользящего среднего возможна реализация, в которой требуется выполнять только две операции сложения на интервале дискретизации.

Запишем еще раз выражение для скользящего среднего:

$$m_k = \frac{1}{N}\sum_{i=0}^{N-1} x_{k-i} = \frac{1}{N}\left(x_k + x_{k-1} + x_{k-2} + ...x_{k-(N-2)} + x_{k-(N-1)}\right). \quad (4)$$

Для предыдущего, (k – 1)-го символа, выражение (2) примет следующий вид:

$$m_{k-1} = \frac{1}{N}\sum_{i=0}^{N-1} x_{k-1-i} = \frac{1}{N}\left(x_{k-1} + x_{k-2} + x_{k-3} + ...x_{k-(N-1)} + x_{k-N}\right). \quad (5)$$

Сравнивая (4) и (5) несложно заметить, что они различаются только тем, что по сравнению с (5) в (4) добавилось новое слагаемое x_k/N и исчезло слагаемое x_{k-N}/N. Таким образом, зная предыдущее значение скользящего среднего, можно получить текущее по следующей рекуррентной формуле:

$$m_k = m_{k-1} - \frac{x_{k-N}}{N} + \frac{x_k}{N}. \quad (6)$$

Т.е. для построения ФНЧ на основе скользящего среднего достаточно поставить линию задержки входного сигнала на N тактов и трехвходовый аккумулятор на ее выходе. Структурная схема скользящего среднего, соответствующая выражению (6), представлена на рис. 4. На рис. 4 не показана операция деления $1/N$.

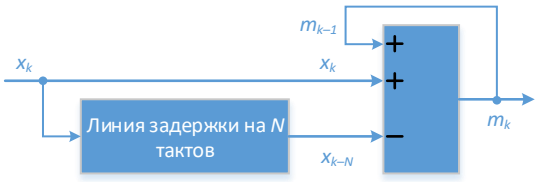


Рис. 4. Структурная схема скользящего среднего на основе линии задержки и аккумулятора (без операции 1/N)

Важным моментом, на который необходимо обратить внимание, является реализация формулы (5) в арифметике с фиксированной точкой. Возможны два подхода: сначала выполнить деление на N, а потом операции сложения и вычитания, и наоборот – сначала сложение и вычитание, а в последний момент – деление. Второй подход иллюстрируется выражением (7).

$$\begin{aligned} \left(Nm_k\right) &= \left(Nm_{k-1}\right) - x_{k-N} + x_k \\ m_k &= \left(Nm_k\right)/N \end{aligned}. \quad (7)$$

Второй подход потребует больше ресурсов (т.к. разрядности обрабатываемых чисел больше), но результат получится точнее. Последнее связано с тем, что при выполнении деления младшие разряды числа с фиксированной точкой так или иначе теряются, т.е. происходит ошибка округления. При выполнении (6) ошибка округления накапливается постоянно, а при выполнении (7) – возникает только в последний момент.