

УДК
ББК
Н62

Рецензент
канд. техн. наук, профессор **В.Б. Харитонов**

Никамин В.А.

Н62 Канальная модуляция в системах записи цифровых данных. – СПбГУКиТ, 2010. - с.

В книге обобщен материал, касающийся методов канальной модуляции и канального кодирования. Особое внимание уделено канальным кодам, используемым для записи информации на широко распространенные сейчас в мире носители – CD, DVD, DAT, BluRay и MiniDisc. Приведены основные характеристики кодов и методов модуляции. Издание предназначено для специалистов, работающих в области записи информации на носитель, а также для студентов, обучающихся по специальностям 210405 «Радиосвязь, радиовещание и телевидение» и 201400 «Аудиовизуальная техника».

Рекомендовано к изданию в качестве монографии кафедрой звукотехники СПбГУКиТ.
Протокол №9 от 24.12.2009 г.

Рекомендовано к изданию в качестве монографии Методическим советом ФАВТ
Протокол № от

© В.А. Никамин, 2010

ОГЛАВЛЕНИЕ

Введение.....	4
1. Основные определения.....	5
1.1. Самосинхронизация и окно детектирования.....	5
1.2. Последовательности с ограниченной длиной пробега. Канальное кодирование.....	9
1.3. Снижение уровня низкочастотных составляющих в спектре сигнала.....	12
1.4. Итоговый перечень основных параметров канальной модуляции.....	14
2. Методы канальной модуляции.....	16
2.1. Метод записи «с возвращением к нулю».....	16
2.2. Методы записи «без возвращения к нулю».....	17
2.3. Двухфазные методы модуляции.....	20
2.4. Модуляция задержкой.....	21
3. Канальные коды.....	24
3.1. Код 3PM.....	24
3.2. Код 4/5.....	26
3.3. Код 4/5 NRZI.....	29
3.4. Коды 2/3.....	31
3.4.1. Код 2/3 (I).....	32
3.4.2. Код 2/3 (II).....	34
3.5. Код 2/4.....	37
3.6. Код EFM.....	39
3.7. Код 8/10.....	50
3.8. Код EFM Plus.....	57
3.9. Код 17 PP.....	62
Заключение.....	64
Литература.....	65
Предметный указатель.....	68

Введение

Канальная модуляция является неизменным атрибутом процедуры формирования потока данных перед передачей его через канал связи. Под каналом в широком смысле этого слова может рассматриваться как эфирный или кабельный канал в радиосвязи и радиовещании, так и различные типы носителей информации: магнитная лента, оптический, магнитооптический, голографический или магнитный (жесткий) диск, используемый в компьютерах. Каждый из каналов обладает вполне определенными физическими свойствами и характеризуется своим набором искажающих воздействий на передаваемый (или записываемый на него) сигнал. Задача канальной модуляции состоит в том, чтобы преобразовать информацию к такому виду, который лучше всего подходит для ее передачи через данный тип канала, т.е. придать ей такие физические характеристики, которые оптимальным образом согласуются с физическими характеристиками канала и менее всего подвержены его искажающим воздействиям. Кроме того, путем правильного выбора метода канальной модуляции можно существенно увеличить количество информации, передаваемой через данный тип канала в единицу времени или записываемой на единице площади носителя. С учетом того, что информация, как правило, передается единым потоком или записывается на единственную дорожку носителя, с помощью канальной модуляции обеспечивается самосинхронизируемость передаваемого потока данных, т.е. возможность выделения из него тактовой частоты, равной скорости следования потока канальных битов.

В книге рассматриваются только те методы модуляции, которые используются при записи информации на различные типы носителей, в том числе определены международными стандартами на наиболее распространенные их типы: CD, DVD, MD, BD, DAT и пр.

Книга предназначена как для специалистов, так и для студентов вузов соответствующих специальностей.

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Канальная модуляция – это набор разнообразных методов представления цифровой информации в форме, обеспечивающей возможность записи наибольшего количества этой информации на единице площади или длины данного носителя и позволяющей использовать простые и надежные методы ее считывания.

При использовании канальной модуляции поток данных приобретает следующие свойства:

- обеспечивается возможность выделения из него синхронизирующей информации, т.е. формирования импульсов тактовой частоты, скорость следования которых изменяется синхронно с изменениями скорости входных данных;
- в спектре считанного сигнала ограничивается содержание низкочастотных и высокочастотных составляющих.

1.1. Самосинхронизация и окно детектирования

Функционирование любой цифровой системы обеспечивается наличием в ней специального синхронизирующего сигнала, с помощью которого производятся все операции внутри системы – переключение триггеров, перемещение битов по ячейкам памяти регистров, запись в оперативную память и т.д. Этот сигнал формируется задающим генератором и представляет собой последовательность импульсов, как правило, со скважностью 2 (длительность импульса равна длительности промежутка между импульсами). Чем выше частота задающего генератора, тем более высоким быстродействием обладает система. Этот сигнал называется **тактовой частотой** (в англоязычной литературе он, чаще всего, обозначается словом **clock** – часы). Разумеется, длительность битов данных, с которыми оперирует система, равна периоду T тактовой частоты (рис. 1.1).

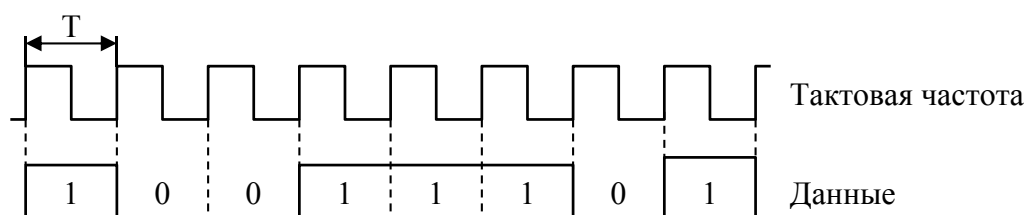


Рис. 1.1. К понятию тактовой частоты

В современных системах записи информации на носитель (имеются в виду системы записи на движущийся носитель – диск или ленту) запись данных осуществляется на одну дорожку. Сигнал сопровождения, необходимый для безошибочного воспроизведения информации, т.е. тактовая частота, отсутствует. При этом предполагается, что в воспроизводящем устройстве имеется своя собственная система, способная синтезировать синхронизирующий сигнал, т.е. сформировать последовательность импульсов, скорость следования которых равна

скорости следования битов считываемой информации. Такая система называется **системой тактовой синхронизации**.

При воспроизведении информации с носителя скорость считываемых данных все время меняется, поскольку скорость перемещения движущегося носителя невозможно сделать строго постоянной. Поэтому формирование внутренней тактовой частоты с помощью стабильного генератора невозможно. Следовательно, источником тактового синхросигнала может быть только перестраиваемый генератор, частота которого должна изменяться в соответствии с изменениями скорости перемещения носителя. Более того, между битами считываемой информации и импульсами тактовой частоты должно поддерживаться определенное фазовое соотношение, поскольку идентификация битов данных и запись их в приемный регистр производится одним из фронтов импульса тактовой частоты. На рис. 1.1 начало бита совпадает с передним фронтом импульса тактовой частоты (если предположить, что это данные воспроизведены с носителя и тактовая частота сформирована внутренним генератором). При таком фазовом соотношении идентификацию битов и запись их в приемный регистр производят задним фронтом (срезом) импульса тактовой частоты, поскольку он находится в центре бита, т. е. в позиции, где отношение сигнал/шум наиболее велико. Однако такое фазовое соотношение может существовать только в те короткие моменты времени, когда скорость следования битов информации в точности совпадает со скоростью следования синхроимпульсов. В реальном же процессе скорость потока входных данных постоянно меняется вследствие неравномерности перемещения носителя, а тактовая частота «пытается догнать» ее, то, слегка отставая от нее по фазе, когда скорость потока данных увеличивается, то, слегка опережая, когда скорость потока данных уменьшается. Такое отставание или опережение может происходить только в пределах половины периода тактовой частоты в одну и в другую сторону от точки оптимального фазового соотношения (рис. 1.2), т.е. $\pm T/2$, где T – период канальной тактовой частоты. Если рассогласование будет больше, то произойдет срыв синхронизации и последующая информация будет искажена. Промежуток времени, в пределах которого допустимы фазовые сдвиги между потоком данных и тактовой частотой, называется **окном детектирования** t_w . Окно детектирования всегда равно периоду тактовой частоты T (или $\pm T/2$). Такое его название обусловлено тем, что идентификация (детектирование) данных производится фронтом сигнала тактовой частоты, расположенным при оптимальном фазовом соотношении в центре тактового интервала. Если фазовый сдвиг станет больше, чем $T/2$, то детектирующий фронт тактовой частоты попадет на следующий (или предыдущий) бит данных и данные будут идентифицированы неверно - будет потерян (или дважды идентифицирован) один бит и вся последовательность сдвинется на один тактовый интервал, что приведет к полному искажению всей информации – до тех пор, пока состояние синхронизма не будет вновь восстановлено.



Рис. 1.2. К понятию окна детектирования

Для того чтобы обеспечить формирование внутренней тактовой частоты и поддерживать заданное фазовое соотношение между битами данных и тактовыми импульсами, используют схемы **фазовой автоподстройки частоты (ФАПЧ)**. Источником тактовой частоты здесь является генератор, управляемый напряжением (ГУН). Управляющее напряжение вырабатывается фазовым дискриминатором, который производит оценку разности между моментами появления перепадов уровня в потоке данных (0-1 или 1-0) и моментами формирования одного из фронтов импульса тактовой частоты. От величины этой разности зависит величина напряжения, которое подается на управляющий вход ГУН и изменяет его частоту в направлении уменьшения фазовой ошибки. Примерная схема системы тактовой синхронизации с петлей ФАПЧ, выполняющей роль формирователя тактовой частоты и устройства идентификации (стробирования) считанных данных, показана на рис.1.3.

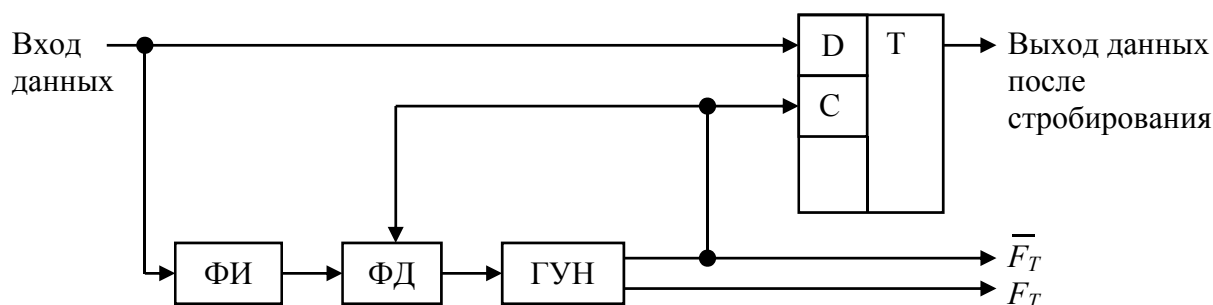


Рис.1.3. Система тактовой синхронизации с петлей ФАПЧ.
 ФИ – формирователь импульсов; ФД – фазовый дискриминатор;
 ГУН – генератор, управляемый напряжением

Роль анализатора ошибки синхронизации (фазового сдвига) и формирователя управляющего напряжения здесь выполняет фазовый

дискриминатор ФД. На один из его входов поступает тактовая частота F_T , вырабатываемая генератором, управляемым напряжением (ГУН) (на схеме – ее инверсия $\overline{F_T}$). На другой вход ФД поступают короткие импульсы, сформированные по фронтам воспроизведенного сигнала. В соответствии с величиной и знаком фазовой ошибки, ФД вырабатывает управляющее напряжение, которое вынуждает ГУН изменить свою частоту так, чтобы ее период приблизился к длительности канального бита воспроизведенного сигнала. Чтобы устранить возможные искажения фронтов сигнала, он идентифицируется (стробируется) в середине тактового интервала с помощью обычного D-триггера. Затем полученный сигнал направляется для дальнейшей обработки.

Система тактовой синхронизации должна обладать определенной инерционностью, поскольку информация о фазовом соотношении извлекается из перепадов уровня входного сигнала, т.е. при смене «нуля» на «единицу» и наоборот, а такие перепады не могут возникать в каждом такте – в потоке информации могут быть достаточно длительные последовательности одного уровня. С другой стороны, инерционность не может быть слишком велика, поскольку при срыве синхронизации из-за искажений, система должна обладать способностью к быстрому восстановлению состояния синхронизма. Но в реальном потоке данных число подряд следующих «нулей» и «единиц» может быть очень большим и, если не принимать никаких мер, нарушения синхронизации будут неизбежны, а следовательно, информация неизбежно будет искажена.

Выход из этой ситуации состоит в том, чтобы попытаться преобразовать данные в такую форму, которая предполагала бы достаточно частое формирование перепадов уровня в полученной последовательности – независимо от того, насколько велико число подряд следующих «нулей» и «единиц» в исходных данных. Последовательности такого вида называются **самосинхронизирующимися**.

Данные в цифровых системах, как правило, представляются в виде двухуровневого сигнала: логической «единице» соответствует высокий уровень, а логическому «нулю» – низкий – как показано на рис. 1.1. Такое представление данных называется NRZ-представлением (сокращение английского выражения Non Return to Zero – без возврата к нулю). Чтобы обеспечить наличие по крайней мере одного изменения уровня на каждый бит информации (независимо от того, «нуль» это или «единица») можно воспользоваться несколько иным представлением «единиц» и «нулей». Одним из таких представлений является **частотная модуляция (ЧМ)**, которая предполагает обязательное изменение уровня в начале каждого тактового интервала и, кроме того, изменение уровня в центре тактового интервала, если данный бит является «единицей» (рис. 1.4). Такой способ модуляции является идеальным с точки зрения возможности выделения из сформированной последовательности синхронизирующей информации, но имеет один очень серьезный недостаток.

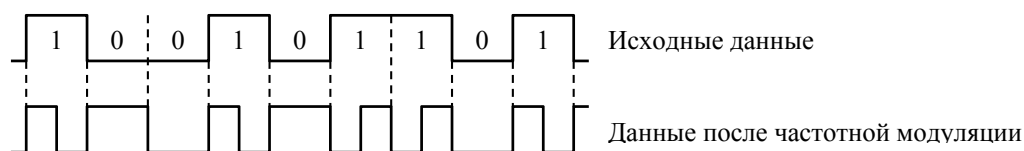


Рис. 1.4. Частотная модуляция

В любой системе записи информации полоса пропускания канала записи/воспроизведения всегда ограничена: в системе магнитной записи – частотной характеристикой пары головка-носитель, в оптической и магнитооптической записи – длиной волны считывающего лазера и числовой апертурой объектива. Стремление получить максимально возможную плотность записи предполагает использование **минимальной длины волны записи λ_{min}** (соответствующей **максимальной частоте f_{max} полосы пропускания канала**) для передачи двух битов информации – отрицательной полуволны для передачи логического «нуля» и положительной полуволны - для передачи логической «единицы» (рис. 1.5). При использовании частотной модуляции минимальная длина волны записи будет соответствовать только одному биту исходных данных – логической «единице». Следовательно, плотность записи уменьшится ровно вдвое и на один и тот же носитель при прочих равных условиях можно будет записать в два раза меньше информации. Поэтому, несмотря на хорошую самосинхронизацию, использование частотной модуляции для записи с высокой плотностью является далеко не лучшим вариантом.

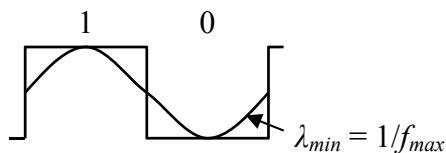


Рис. 1.5. Минимальная длина волны записи

1.2. Последовательности с ограниченной длиной пробега. Канальное кодирование

Для того чтобы придать потоку данных, предназначенных для записи на носитель, свойство самосинхронизации и одновременно обеспечить возможность увеличения плотности записи, используют технологию его преобразования в так называемые **последовательности с ограниченной длиной пробега**, или, в общепринятой международной терминологии - **RLL-последовательности** (RLL – сокращение англоязычного словосочетания Run Length Limited). Такие последовательности характеризуются наличием ограничений, накладываемых на промежутки между двумя соседними перепадами уровня как сверху, так и снизу. Другими словами, последовательность одного уровня может быть не более чем T_{max} периодов тактовой частоты (тактовых интервалов) и не менее чем T_{min} периодов [1].

Для достижения этой цели используется следующий алгоритм. Прежде всего последовательность исходных информационных битов делится на группы по n бит в каждой. Затем каждая такая группа заменяется (кодируется) другой группой из m бит, где $m > n$ (рис. 1.6). Поскольку множество $A_m = 2^m$ m -разрядных групп (**канальных символов**) всегда больше множества $A_n = 2^n$ n -разрядных групп (**информационных символов**), то появляется возможность выбрать из него A_n таких канальных символов, у которых число «нулей» между двумя соседними «единицами» ограничено сверху величиной k и снизу величиной d . После чего полученная последовательность преобразуется из формы NRZ в форму NRZI (Non Return to Zero Inverted – без возвращения к нулю инвертированная), где «единице» соответствует изменение уровня в начале тактового интервала, а «нулю» – отсутствие такого изменения, т.е. сохранение прежнего уровня (рис. 1.7). При этом минимальное расстояние T_{min} между двумя соседними перепадами уровня, как видно из рис. 1.7, оказывается равным $d + 1$, а максимальное расстояние $T_{max} = k + 1$. Для того чтобы при каскадном соединении канальных символов заданные ограничения не нарушались, предусматриваются особые правила выполнения таких соединений. Например, замена n -разрядного информационного символа m -разрядным канальным может быть неоднозначной, т.е. одному и тому же n -разрядному символу могут соответствовать два или более альтернативных варианта m -разрядных символов, из которых при выполнении кодирования подбирается наиболее подходящий. В другом случае могут быть предусмотрены один или более дополнительных разрядов, которые вставляются между смежными канальными символами и служат исключительно для того, чтобы обеспечить требуемое расстояние между соседними перепадами уровня.

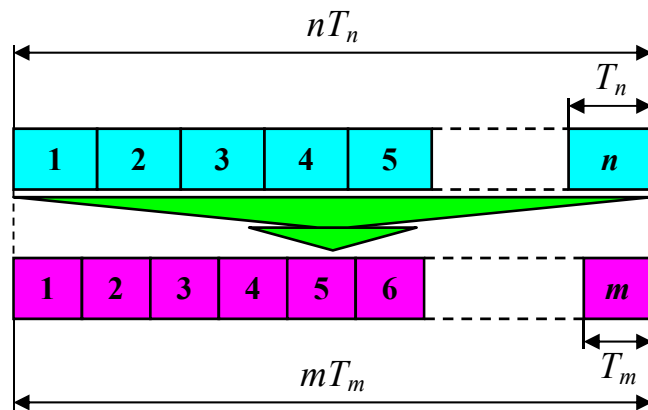


Рис.1.6. Преобразование n бит в m

Таким образом, в формируемой последовательности канальных символов будет присутствовать, по крайней мере, один перепад на $k+1$ тактовых интервалов (5 в примере на рис.1.7), что обеспечит хорошую самосинхронизируемость последовательности. С другой стороны, ограничение длины пробега снизу величиной T_{min} обеспечит возможность передачи при минимальной длине волны записи не 2 бит исходной

информации, а $2(d+1)$ канальных бит. Отношение $2(d+1)/2 = d+1$ показывает, что при использовании данного канального кода на одном и том же участке носителя можно записать в $d+1$ раз больше канальных бит, чем исходных информационных. Но оценка изменения плотности записи подразумевает сравнение относительно информационных символов (до кодирования и после него). Поскольку m канальных бит эквивалентны n информационным, то увеличение объема исходной информации будет в m/n раз меньше, чем $d+1$. Величина $K = (d+1) \cdot \frac{n}{m}$ называется **коэффициентом повышения плотности записи**.

Следует отметить, что замена n информационных битов на m канальных приводит к соответствующему изменению тактовой частоты. Канальная тактовая частота F_{TK} (чаще всего обозначается F_K или f_k), очевидно, будет в m/n раз выше, чем тактовая частота F_T исходной информационной последовательности. Следовательно, изменится и абсолютная величина окна детектирования t_w . Поскольку величина окна детектирования всегда равна периоду тактовой частоты T (или $\pm T/2$), то увеличение тактовой частоты в m/n раз повлечет за собой уменьшение во столько же раз ее периода и окна детектирования. Таким образом, окно детектирования кодированной последовательности будет равно $t_w = T_K = \pm \frac{T_K}{2} = \frac{n}{m} T = \pm \frac{n}{m} \cdot \frac{T}{2}$.

Проиллюстрируем вышесказанное конкретным примером. Пусть задана исходная последовательность 4-разрядных информационных символов ($n = 4$), которую требуется преобразовать в последовательность 6-разрядных канальных ($m = 6$) символов (рис. 1.7) при заданных ограничениях на длину пробега $d=1$ и $k=4$. Для простоты условимся, что для соединения смежных символов предусмотрен алгоритм альтернативной замены одного канального символа другим – так, чтобы заданные ограничения на длину пробега при этом не нарушались.

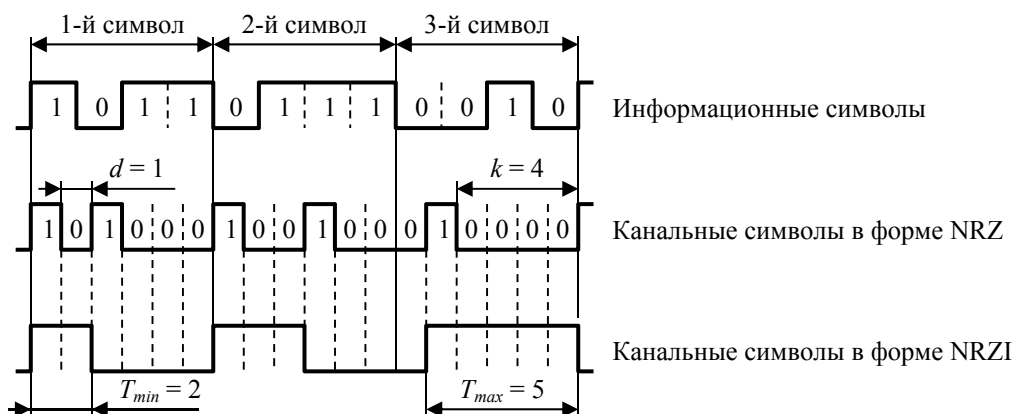


Рис. 1.7. Кодирование 4-6 при $k = 4$ и $d = 1$

После замены исходных 4-разрядных символов 6-разрядными канальными символами и модуляции полученной последовательности по

методу NRZI получим новую последовательность с минимальной длиной пробег $T_{min} = d + 1 = 2$. Повышение плотности записи при этом составит

$$K = (1 + 1) \cdot \frac{4}{6} = \frac{4}{3} = 1,33.$$

Другими словами, количество информации, которую требуется записать на носитель, за счет использования данного канального кода увеличивается на треть. Это не так мало.

Окно детектирования t_w кодированной последовательности будет равно $2/3 T$, где T – период исходной тактовой частоты (длительность исходного бита данных).

Процедура описанного преобразования последовательности n -разрядных информационных символов в последовательность m -разрядных канальных символов называется **канальным кодированием**.

Величина $R = n/m$ называется **скоростью канального кода**.

1.3. Снижение уровня низкочастотных составляющих в спектре сигнала

Теперь рассмотрим один важный аспект процедуры считывания информации с носителя. Данные, записанные на носитель, имеют вид достаточно узкой дорожки конечной ширины. Информация на дорожке представлена в виде изменений одного из параметров носителя: характера намагниченности рабочего слоя магнитного носителя, интенсивности отраженного света на оптическом носителе или угла поворота плоскости поляризации на магнитооптическом носителе. Для того чтобы считывание такой информации было возможно, считывающий элемент воспроизводящего устройства (магнитная или оптическая головка) должен следовать как можно более точно вдоль осевой линии дорожки. При считывании оптических и магнитооптических дисков, кроме того, должно поддерживаться заданное расстояние между дорожкой и оптической головкой. Обеспечивают процесс считывания системы автоматического регулирования (сервосистемы). Информация о степени точности слежения за дорожкой записи (или о степени точности фокусировки) извлекается ими из самого информационного сигнала – других источников нет. Ошибка слежения возникает из-за неравномерности вращения или перемещения носителя, а также из-за вертикальных биений дискового носителя, поэтому сигнал ошибки располагается в области очень низких частот – порядка нескольких десятков герц, и проявляется, как правило, в виде изменения уровня огибающей информационного сигнала. Кроме того, при использовании для воспроизведения магнитной сигналограммы индукционной головки низкие частоты воспроизводятся очень плохо. Вследствие этого крайне важно, чтобы в спектре информационного сигнала низкочастотных составляющих либо не было вообще, либо их уровень был пренебрежимо мал. В противном случае сервосистемы будут воспринимать изменения в низкочастотной части информационного сигнала как ошибку

слежения, а это может привести к нарушению процесса считывания и появлению большого числа ошибок в считанных данных. Поэтому для записи информации на носитель используются такие методы канальной модуляции и канальные коды, которые имеют в своем спектре как можно меньший уровень низкочастотных компонент, либо предусматривают механизм, обеспечивающий контроль и подавление их уровня до приемлемого значения.

Канальное кодирование с образованием последовательностей ограниченной длины достаточно хорошо формирует спектр записываемого сигнала, поскольку основная энергия сигнала здесь будет сосредоточена в полосе частот между $f_{max} = 1/\lambda_{min} = 1/2T_{min}$ и $f_{min} = 1/\lambda_{max} = 1/2T_{max}$. На рис. 1.8 показан спектр описанного выше кода 4/6, энергия которого сосредоточена в полосе частот от $1/4T_k$ до $1/10T_k$, где T_k – длительность канального бита.

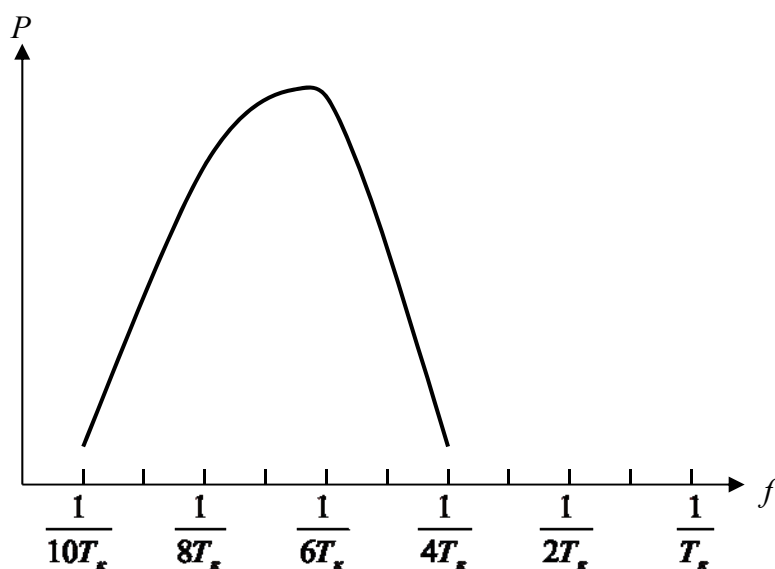


Рис. 1.8. Спектр канального кода 4/6

Если не обращать внимание на снижение плотности записи, то с точки зрения малого уровня низкочастотных составляющих для записи на носитель также хорошо подходит и частотная модуляция. Здесь вся энергия сосредоточена всего на двух частотах $f_{min} = 1/T$ и $f_{max} = 2/T$, где T – длительность исходного бита данных. А вот модуляция по способу NRZI для записи на носитель совершенно не годится – здесь спектр простирается от $f_{max} = 1/T$ до 0 Гц, поскольку информационный сигнал может содержать сколь угодно длинные последовательности «нулей» (рис. 1.9).

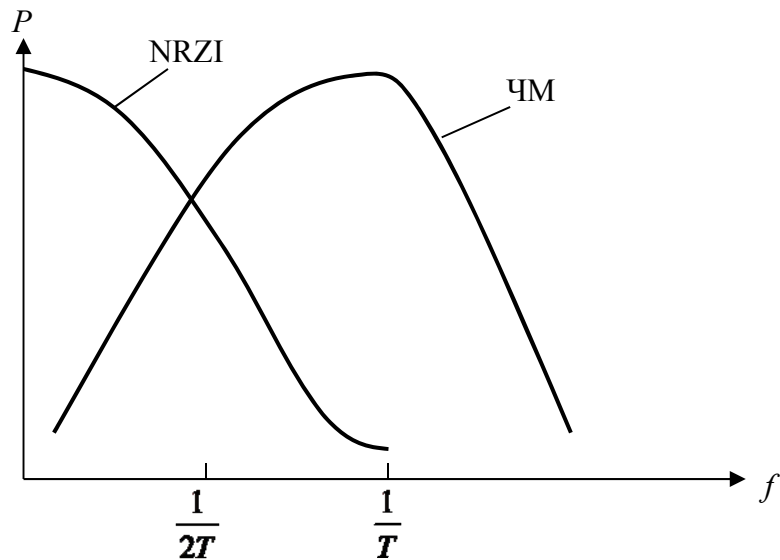


Рис. 1.9. Спектры мощности сигналов, модулированных по методу NRZI и по методу ЧМ.

T – длительность исходного бита данных

1.4. Итоговый перечень основных параметров канальной модуляции

Минимальная длина волны записи λ_{min} – суммарная длительность двух минимальных интервалов между соседними перепадами уровня - высокого уровня и низкого уровня (см. рис. 1.5). Соответствует наивысшей частоте в спектре модулированного сигнала ($f_{max} = 1/\lambda_{min}$) и, с учетом полосы пропускания канала записи/воспроизведения, характеризует предельные возможности системы с точки зрения повышения плотности записи при использовании данного способа модуляции.

Максимальная длина волны записи λ_{max} – суммарная длительность двух максимальных интервалов между соседними перепадами уровня - высокого уровня и низкого уровня. Соответствует наименьшей частоте в спектре модулированного сигнала ($f_{min} = 1/\lambda_{max}$), может быть использована для оценки спектральной плотности канального сигнала в области частот, близких к постоянной составляющей.

Окно детектирования t_w - промежуток времени, в пределах которого допустимы временные сдвиги между потоком данных и тактовым сигналом (см. рис. 1.2). Окно детектирования всегда равно периоду канального тактового сигнала T_k (или $\pm T_k/2$).

Минимальное расстояние между двумя соседними «единицами» d используется в канальном кодировании. Измеряется в периодах канального тактового сигнала T_k (см. рис. 1.7). Характеризует потенциальные возможности данного канального кода с точки зрения увеличения плотности записи. Чем больше величина d , тем большую плотность записи можно получить при прочих равных условиях.

Минимальная длина пробега T_{min} - минимальное расстояние между двумя соседними перепадами уровня модулированного сигнала, измеряемое в периодах канального тактового сигнала T_k (см. рис. 1.7). Так же, как связанная с ней величина d ($T_{min} = d + 1$), характеризует потенциальные возможности данного канального кода с точки зрения увеличения плотности записи. Чем больше T_{min} , тем большую плотность записи можно получить. Однозначно связана с минимальной длиной волны записи λ_{min} ($\lambda_{min} = 2 T_{min}$).

Максимальное расстояние между двумя соседними «единицами» k - используется в канальном кодировании. Измеряется в периодах канального тактового сигнала T_k (см. рис. 1.7). Характеризует кодированную последовательность с точки зрения самосинхронизации. Чем меньше величина k , тем лучше самосинхронизируемость кода.

Максимальная длина пробега T_{max} - максимальное расстояние между двумя соседними перепадами уровня модулированного сигнала, измеряемое в периодах канального тактового сигнала T_k (см. рис. 1.7). Так же, как связанная с ней величина k ($T_{max} = k + 1$), характеризует кодированную последовательность с точки зрения самосинхронизации. Чем меньше величина T_{max} , тем лучше самосинхронизируемость кода. Однозначно связана с максимальной длиной волны записи λ_{max} ($\lambda_{max} = 2 T_{max}$).

Скорость канального кода R равна отношению числа n исходных информационных бит к числу m канальных бит: $R = n/m$. Косвенно характеризует информационную емкость кода, т.е. какую часть из общего числа бит составляли бы информационные биты, если бы кодирование осуществлялось путем простого добавления к исходным n битам $m-n$ избыточных.

Коэффициент повышения плотности записи $K = (d + 1) \cdot \frac{n}{m}$ указывает, во сколько раз увеличится (или уменьшится) количество исходной информации на единице длины дорожки записи при использовании данного кода.

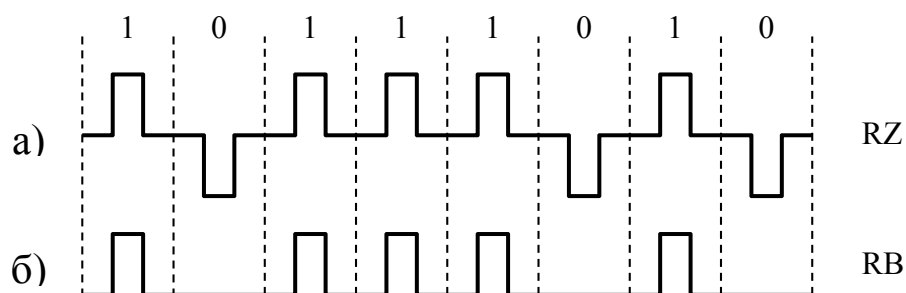
2. МЕТОДЫ КАНАЛЬНОЙ МОДУЛЯЦИИ

2.1. Метод записи «с возвращением к нулю»

Метод записи «с возвращением к нулю» (ВН) является одним из первых методов записи цифровой информации на магнитный носитель. В зарубежной литературе он известен под аббревиатурой **RZ (Return to Zero)**. Метод этот был разработан еще в 1955 году специально для записи информации на магнитные барабаны и использовался вплоть до середины 60-х годов. В нашей стране метод записи ВН применялся в накопителе на магнитном барабане НБ-9, а затем в его модернизированных вариантах – НБ-10 и НБ-11 [2].

Запись двоичных «единиц» и «нулей» в соответствии с данным методом производилась импульсами тока в обмотке магнитной головки. «Единицам» соответствовало одно направление тока записи, «нулям» - другое (рис. 2.1,а). Импульсы «единиц» намагничивали участки носителя до насыщения в одном направлении, импульсы «нулей» - в противоположном направлении. Фактически запись по методу RZ является трехуровневой. Существовала и двухуровневая разновидность метода RZ, которая предполагала отсутствие намагниченности на позициях «нулей» (отсутствие тока записи противоположного направления). Такая разновидность обозначалась аббревиатурой **RB (Return Back – «возврат назад»)** (рис. 2.1,б).

Из-за общего несовершенства аппаратуры записи/воспроизведения того времени между отдельными двоичными знаками необходимо было создавать защитный интервал, что и достигалось посредством того, что ток записи в промежутках между «нулями» и «единицами» информации принимал нулевое значение. Кроме того, одновременно с записью основной информации приходилось записывать отдельную синхронизирующую дорожку, для того чтобы обеспечить внешнюю синхронизацию при воспроизведении такой записи, так как системы тактовой синхронизации с фазовой автоподстройкой частоты были изобретены значительно позже – во второй половине 70-х годов.



2.1. Методы записи данных с возвращением к нулю

К недостаткам метода можно отнести также высокую чувствительность к шумовым помехам и временному сдвигу между информационным и синхронизирующим сигналами. Однако главным недостатком методов RZ и RB является их низкая информационная емкость: для передачи одного бита информации здесь требуется полтора периода сигнала максимальной частоты в спектре сигнала. Поэтому сегодня рассматривать эти методы можно только как первые исторические вехи на пути создания современных высокоэффективных групповых канальных кодов.

2.2. Методы записи «без возвращения к нулю»

Методы записи «без возвращения к нулю» (**БВН**) появились позже методов ВН, когда аппаратура записи стала более совершенной и необходимость в защитных интервалах между импульсами записи «единиц» и «нулей» информации отпала. В зарубежной терминологии и среди специалистов эти методы известны под аббревиатурой **NRZ (Non Return to Zero)**. Существует несколько разновидностей NRZ-представления.

Наиболее известным и распространенным является прямое NRZ-представление (рис. 2.2,а), где логической «единице» соответствует высокий уровень сигнала, а логическому «нулю» - низкий. NRZ-представление является уровневым, поэтому иногда его называют **NRZ-L**-представлением (от английского слова Level - уровень) [3]. Однако в чистом виде NRZ-представление для записи на носитель практически не используется из-за отсутствия ограничений на длину последовательности «нулей» или «единиц». Такие методы представления информации называются «неограниченными по длине пробега», т.е. расстояние между двумя соседними изменениями уровня может быть сколь угодно велико. По этой причине, во-первых, невозможно контролировать уровень низкочастотных составляющих в воспроизведенном сигнале; во-вторых, невозможно обеспечить надежную синхронизацию считанной с носителя информации по тактам. Вторым недостатком можно пренебречь только при многодорожечной записи на магнитную ленту с невысокой плотностью, поскольку здесь имеется возможность выделить для записи сигналов синхронизации специальную дорожку. Тем не менее NRZ является основной формой представления цифровых данных во всех системах обработки информации, в том числе в системах канального кодирования на этапе преобразования n исходных бит в m кодовые, после чего, как правило, m -разрядные кодовые слова модулируются по методу NRZI.

Метод **NRZ-I (Non Return to Zero Inverted)** – без возвращения к нулю инвертированный) не является уровневым. Здесь логическая «единица» обозначается изменением уровня сигнала в начале тактового интервала, а логический «нуль» - отсутствием такого изменения (сохранением прежнего уровня) (рис. 2.2,б). В этом заключается его преимущество перед методом NRZ, поскольку содержание информации не зависит от полярности сигнала, а зависит только от ее изменения [3]. Однако метод NRZ-I (в современной

литературе обозначается как NRZI – без черточки в середине) страдает тем же недостатком, что и NRZ, - отсутствием ограничений на длину последовательностей одного уровня, которые могут появиться при наличии в потоке данных длинных последовательностей «нулей». Поэтому в чистом виде он для записи на носитель не используется, зато широко используется для модуляции кодированной информации, когда количество «нулей» между соседними «единицами» ограничено.

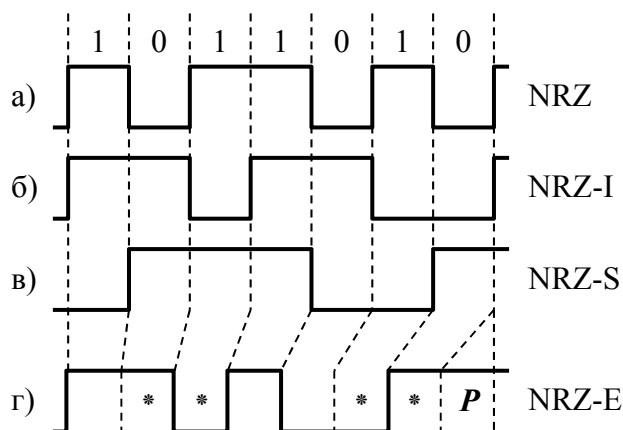


Рис. 2.2. Методы записи данных без возвращения к нулю. Звездочкой (*) отмечены инвертированные биты, символом *P* - бит четности

Метод **NRZ-S (NRZ Spaced – NRZ пространственный)** является аналогом NRZI, но здесь изменением уровня обозначаются логические «нули», а «единицы» - отсутствием изменения (рис. 2.2,в). На практике используется редко.

Технические характеристики методов представления NRZ, NRZI и NRZ-S:

- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$, где $T_{кан}$ – период канального тактового сигнала;
- максимальная длина волны записи $\lambda_{max} = \infty$ (высокий уровень спектральной мощности в области низких частот, включая постоянную составляющую);
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = \infty$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = \infty$ (самосинхронизируемость отсутствует).

Метод **NRZ-E (E от слова Enhanced – расширенный)** сильно отличается от других методов группы NRZ, поскольку предполагает введение некоторой избыточности в исходные данные. По этой причине его можно отнести скорее к методам канального кодирования, чем к методам канальной

модуляции. В соответствии с этим методом непрерывный поток исходных данных, прежде всего, делится на семиразрядные слова. Далее разряды 2, 3, 6 и 7 в каждом слове инвертируются. После этого к семиразрядному слову добавляется восьмой бит с таким расчетом, чтобы количество «единиц» в каждом 8-разрядном слове было нечетным (рис. 2.2,г) [4].

Кодирование здесь предназначено для получения сразу нескольких преимуществ над другими методами модуляции группы NRZ. Во-первых, обеспечивается достаточное количество переходов (хотя бы один на каждые 14 последовательных бит – см. рис.2.3) для того чтобы из сформированной последовательности при воспроизведении можно было выделить синхронизирующую информацию. Во-вторых, снижается содержание низкочастотных компонент в спектре записываемого и считываемого сигнала. И, наконец, в-третьих, с помощью восьмого контрольного бита возможна проверка принятых данных на наличие ошибок, хотя такая возможность приводит к уменьшению плотности записи на 12,5%.



Рис.2.3. Две наихудшие ситуации, которые могут возникнуть при соединении двух слов в коде NRZ-E

Технические характеристики NRZ-E-представления:

- количество битов в информационном слове $n = 7$;
- количество битов в кодовом слове $m = 8$;
- скорость кода $R = 7/8$;
- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 28T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 14$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = 14$ (самосинхронизируемость не очень хорошая, но она есть);
- коэффициент повышения плотности записи $K = 7/8$ (плотность записи меньше, чем до кодирования);
- окно детектирования $t_w = 7/8T = \pm 7/16 T$.

2.3. Двухфазные методы модуляции

Разновидности двухфазной модуляции известны под разными наименованиями: бифазное кодирование, модуляция с двойной частотой,

фазовая модуляция, модуляция частотным сдвигом, манчестерское кодирование и пр. Общим для этих методов является то, что в модулированном потоке данных всегда имеется по крайней мере один перепад уровня на бит информации (рис. 2.4). Это практически полностью освобождает записываемый сигнал как от компоненты постоянного тока, так и от низкочастотных составляющих спектра и обеспечивает идеальную самосинхронизацию потока данных.

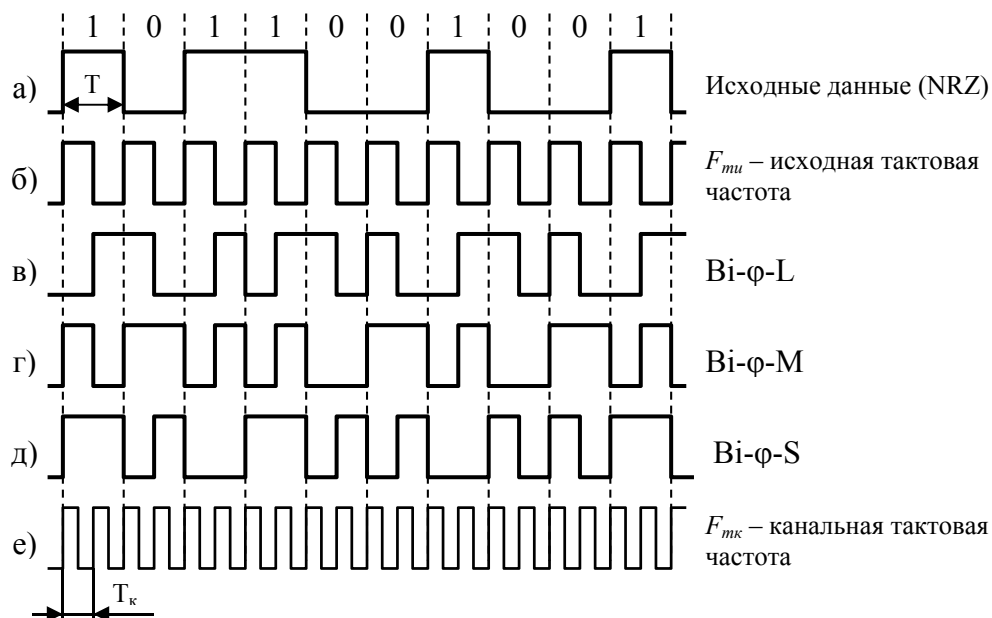


Рис. 2.4. Разновидности двухфазных методов модуляции

С другой стороны, наличие двух перепадов уровня на бит данных (либо на «ноль», либо на «единицу») приводит к необходимости удвоения канальной тактовой частоты F_{mk} (рис.2.4.е) в сравнении с исходной тактовой частотой F_{mi} (рис. 2.4,б), а также к уменьшению плотности записи в два раза. Такое неэффективное использование ширины полосы пропускания канала записи/воспроизведения сводит к минимуму все вышеуказанные преимущества двухфазной модуляции с точки зрения получения высокой плотности записи, что, разумеется, ограничивает ее применение на практике. Известны три разновидности двухфазной модуляции.

Двухфазная уровневая модуляция Vi-φ-L, известна также как **манчестерское кодирование**, **кодирование PE (Phase Encoding - фазовое кодирование)**, **фазовая модуляция (ФМ)** или **модуляция с расщеплением фазы** (рис. 2.4,в). Изменение уровня в центре тактового интервала от «высокого» к «низкому», означает «ноль», а от «низкого» к «высокому» - также в центре тактового интервала – «единицу». Если несколько «нулей» или «единиц» следуют подряд один за другим, то в начале каждого последующего тактового интервала происходит возврат уровня к требуемому исходному [5].

Двухфазная маркированная модуляция Vi-φ-M, или **частотная модуляция (ЧМ)**, была предложена А. Габором в 1959 году [6]. Характеризуется обязательной сменой уровня в начале каждого тактового

интервала (рис. 2.4,г). При этом «единица» кодируется дополнительным изменением уровня в центре тактового интервала. Отсутствие такого изменения обозначает «нуль». Используется для модуляции данных, передаваемых через интерфейсы AES3 (AES/EBU) и S/PDIF.

Двухфазная пространственная модуляция Vi-φ-S формируется так же, как и Vi-φ-M, но наличие изменения уровня в центре тактового интервала указывает на «нуль», а его отсутствие – на «единицу» (рис. 2.4,д).

Технические характеристики двухфазных методов модуляции:

- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 4T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 2$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = 2$ (самосинхронизация очень хорошая);
- коэффициент повышения плотности записи $K = 1/2$ (плотность записи вдвое меньше, чем до кодирования);
- окно детектирования $t_w = 1/2 T = \pm 1/4 T$.

2.4. Модуляция задержкой

Так уж сложилось исторически, что методы модуляции задержкой чаще всего называют DM-кодами (DM – Delay Modulation), хотя, как говорилось выше, кодами в прямом смысле их назвать никак нельзя. Дело, видимо, в том, что в то время, когда они появились, канальных кодов в сегодняшнем понимании еще не существовало. Позволим и мы себе, отдавая дань сложившейся традиции, употреблять короткое, но емкое слово «код» вместо длинного слова «модуляция».

К DM-кодам относятся коды Миллера и M^2 . Характеризуются они тем, что имеют как минимум один перепад уровня на каждую пару битов. Такая плотность перепадов вполне достаточна для сведения низкочастотных компонент спектра мощности к минимуму, а также ограничивает длину последовательности битов одного уровня, что обеспечивает хорошую самосинхронизацию формируемой последовательности. Рассмотрим эти коды.

Код Миллера был предложен Армином Миллером еще в середине 60-х годов [7] и получил широкое распространение в технике записи цифровой информации [3]. В частности, он использовался в аппаратуре цифровой звукозаписи таких фирм, как 3M [8], Mitsubishi [9] и Matsushita [10]. Популярность его можно объяснить тем, что он имеет информационную емкость методов NRZ и NRZI (в 70-е годы считавшуюся наибольшей), но в отличие от них обладает самосинхронизацией. В отечественной литературе код Миллера иногда называют модифицированной фазовой модуляцией

(МФМ), в зарубежной - модифицированной частотной модуляцией (MFМ – modified frequency modulation).

По правилам кодирования МФМ логической «единице» соответствует изменение уровня в центре тактового интервала, а логическому «нулю» - изменение уровня в начале тактового интервала. Однако при этом не должно быть промежутка между двумя соседними изменениями уровня короче одного тактового интервала. Если не накладывать никаких ограничений на данный способ кодирования, то такие интервалы (длительностью в половину бита) всё же получатся, например, если за «единицей» следует «нуль». Для того чтобы таких интервалов не возникало, дополнительно оговаривается, что переход в начале тактового интервала, кодирующий «нуль», запрещается, если этот «нуль» следует за «единицей» (рис. 2.5,в). Таким образом, минимальный интервал T_{min} между соседними изменениями уровня равен одному тактовому интервалу T исходных данных (рис. 2.5,а). Кроме того, могут быть еще интервалы в $1,5T = 3T_k$ (сочетания бит 100 и 01) и в $2T = 4T_k$ (сочетание бит 101). По этой причине код Миллера иногда называют трехчастотным кодом.

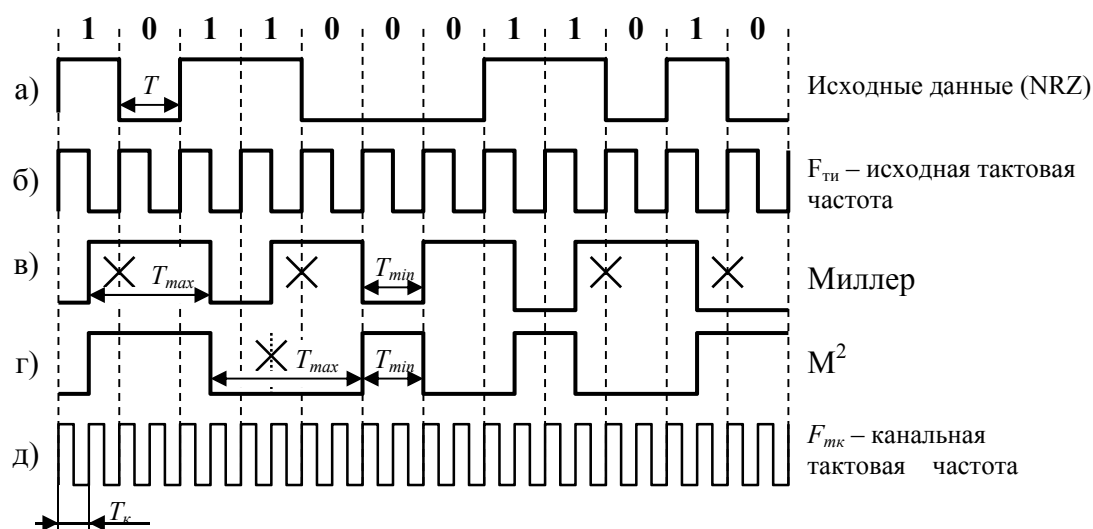


Рис.2.5. Формирование кодов Миллера и M^2

Недостатком модуляции задержкой можно считать увеличение канальной тактовой частоты $F_{ТК}$ (рис. 2.5,д) в сравнении с исходной тактовой частотой $F_{ТИ}$ (рис. 2.5,б) вдвое, обусловленное наличием интервалов между перепадами уровня, кратных половине длительности исходного бита данных. Однако здесь это не приводит к снижению плотности записи, поскольку минимальное расстояние между соседними перепадами, определяющее значение предельно достижимой плотности записи, в отличие от случая двухфазной модуляции, по-прежнему остается равным длительности одного исходного бита данных.

Технические характеристики кода Миллера:

- минимальная длина волны записи $\lambda_{min} = 4T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 8T_{кан}$;

- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 4$;
- минимальная длина пробега $T_{min} = 2$;
- максимальная длина пробега $T_{max} = 4$ (самосинхронизация хорошая);
- коэффициент повышения плотности записи $K = 1$ (плотность записи не изменилась);
- окно детектирования $t_w = 1/2 T = \pm 1/4 T$.

Код M^2 является усовершенствованным вариантом кода Миллера [11, 12]. Существует много разных его наименований: Миллер², модифицированный Миллер, ММФМ, M^2 ФМ и т.д. Такое разнообразие объясняется тем, что фамилии разработчика кода МФМ и того, кто усовершенствовал этот код, были Миллер, но инициалы у них разные, поскольку это разные люди. Первого из них зовут Армин, второго – Джерри. Использовался в магнитофоне для цифровой звукозаписи фирмы Аптех [13]. Формируется код M^2 почти так же, как и код Миллера, за исключением случая, когда за изолированным «нулем» следует четное число «единиц». Тогда переход, кодирующий последнюю «единицу», запрещается. За счет введения этого условия в спектре кода M^2 полностью уничтожается постоянная составляющая, хотя максимальная длина пробега увеличивается до $2,5T$ ($5T_k$).

Технические характеристики кода M^2 :

- минимальная длина волны записи $\lambda_{min} = 4T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 10T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 5$;
- минимальная длина пробега $T_{min} = 2$;
- максимальная длина пробега $T_{max} = 5$ (самосинхронизация хорошая);
- коэффициент повышения плотности записи $K = 1$ (плотность записи не изменилась);
- окно детектирования $t_w = 1/2 T = \pm 1/4 T$.

3. КАНАЛЬНЫЕ КОДЫ

3.1. Код ЗРМ

Данный код был предложен Джорджем Якоби в 1977 году [14]. В соответствии с правилами кодирования ЗРМ (**3 position modulation - трехпозиционная модуляция**) исходная последовательность символов делится на группы из трех символов ($n = 3$), после чего каждая 3-разрядная группа преобразуется в 6-разрядную ($m = 6$) в соответствии с табл. 3.1 (модуляция 3/6). Наконец, полученная последовательность модулируется по методу NRZI.

Таблица 3.1

Таблица соответствия кода ЗРМ

№	Исходное слово	Кодовое слово
1	000	000010
2	001	000100
3	010	010000
4	011	010010
5	100	001000
6	101	100000
7	110	100010
8	111	100100

Таблица соответствия кода ЗРМ составлена так, что количество «нулей» между двумя соседними «единицами» не может быть меньше двух ($d = 2$). Следовательно, после преобразования последовательности битов по способу NRZI минимальное расстояние между соседними перепадами уровня окажется не менее трех периодов канальной тактовой частоты ($T_{min} = 3$). Минимальная длина волны записи при этом составит 6 канальных интервалов ($\lambda_{min} = 6$).

Анализ таблицы соответствия показывает, что последним символом любого кодового слова всегда является «нуль». Но первым символом может быть как «нуль», так и «единица», поэтому при каскадном соединении двух кодовых слов ограничение на минимальное количество «нулей» между двумя соседними «единицами» может быть нарушено, т.е. может получиться так, что между двумя соседними «единицами» окажется всего лишь один «нуль». Например, слово 8 из табл. 3.1 следует за словом 1: 0000**1**0100100 - для этого случая правила кодирования ЗРМ предписывают замену сочетания 101 на сочетание 010: 0000**0**1000100.

Максимальное количество «нулей» между двумя соседними «единицами» в коде ЗРМ равно 11 ($k = 11$). Такой случай возникает, например, когда кодируется информационная последовательность 000 101 000 101. После кодирования получается последовательность 000010 100000 000010 100000. Сочетания 101 между двумя первыми и двумя последними парами кодовых слов по правилам кодирования (см. выше) должны быть заменены на сочетания 010. Следовательно, после замены получим последовательность 000001 000000 000001 000000. Между последней «единицей» первого кодового слова и последней «единицей» третьего кодового слова оказывается ровно 11 «нулей». После преобразования последовательности по методу NRZI расстояние между двумя соседними перепадами будет равно $12 T_{кан}$, т.е. максимальная длина пробега $T_{max} = 12$, а максимальная длина волны записи $\lambda_{max} = 24 T_{кан}$.

Повышение плотности записи при использовании кода ЗРМ составит $K = (d+1) \cdot n/m = 3/2 = 1,5$, что является очень хорошим показателем для любого канального кода. Не очень хорошо то, что при этом вдвое уменьшится окно детектирования $t_w = 3/6 T = T/2 = \pm T/4$, но с этим недостатком ради 50-процентного повышения плотности записи можно и примириться.

Что касается спектральной плотности мощности кода ЗРМ, то основная энергия кодированной последовательности, очевидно, будет находиться в диапазоне частот от $1/\lambda_{max} = 1/24 T_{кан} = f_K/24$ до $1/\lambda_{min} = 1/6 T_{кан} = f_K/6$ (рис. 3.1).

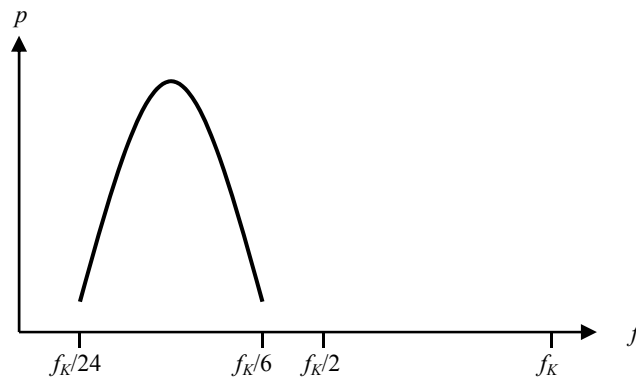


Рис. 3.1. Спектральная плотность мощности кода ЗРМ

Технические характеристики кода ЗРМ:

- минимальная длина волны записи $\lambda_{min} = 6T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 24T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 2$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 11$;
- минимальная длина пробега $T_{min} = 3$;
- максимальная длина пробега $T_{max} = 12$;
- скорость кода $R = 1/2$;

- коэффициент повышения плотности записи $K = 1,5$ (плотность записи увеличилась);
- окно детектирования $t_w = 1/2 T = \pm 1/4 T$.

3.2. Код 4/5

Правила кодирования кода 4/5 сильно отличаются от правил кодирования кода 3PM. Прежде всего, здесь не предполагается модуляция полученной в результате преобразования последовательности по методу NRZI, поскольку не предполагается ограничения количества «нулей» между двумя соседними «единицами». Кроме того, снижение уровня низкочастотных составляющих спектра достигается за счет непрерывного контроля за уровнем постоянной составляющей формируемой последовательности с помощью так называемой **величины числовой суммы DSV (Digital Sum Value)**. Метод использования DSV оказался очень эффективным способом уменьшения низкочастотных компонент в спектре сигнала и по этой причине до сих пор широко используется в технике канального кодирования. Работает он следующим образом.

Пусть на вход устройства, фиксирующего значение DSV, поступает последовательность «нулей» и «единиц» (рис. 3.2). Предположим, что в начальный момент времени t_0 значение DSV равно нулю. Первым битом последовательности является логическая «единица», которая добавит к значению DSV величину $+1$ (момент времени t_1). Вторым битом последовательности – логический «нуль». Этот «нуль» добавит к значению DSV величину -1 , после чего общее значение станет равным 0 (t_2). Следующий бит снова является логическим «нулем», который также внесет в значение DSV -1 , после чего оно станет равным -1 (t_3). Далее следует логическая «единица», которая изменит значение DSV на $+1$ и оно станет равным 0 (t_4). Таким образом, каждая последующая логическая «единица» этой последовательности будет увеличивать значение DSV на одну единицу, а каждый логический «нуль» - на единицу уменьшать его. Если в процессе кодирования некоторой произвольной последовательности тем или иным способом поддерживать значение DSV вблизи нуля, то можно считать, что за определенный, достаточно большой промежуток времени постоянная составляющая последовательности также будет равна нулю.

Как следует из названия кода 4/5, исходная последовательность делится здесь на группы (информационные символы) из четырех бит, и затем производится замена каждой такой группы на группу из пяти бит (кодированное слово). Т.е. для данного кода $n = 4$, а $m = 5$ (рис. 3.3).

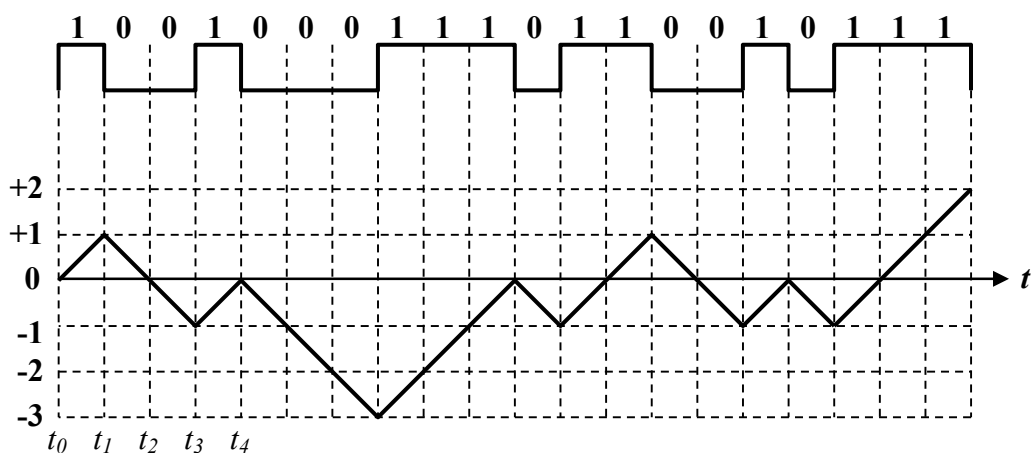


Рис.3.2. Принцип формирования DSV

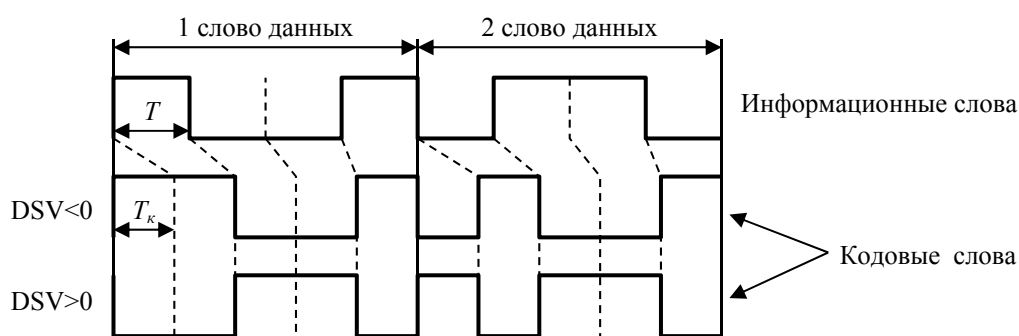


Рис. 3.3. Формирование кодовых слов кода 4/5 при разных исходных значениях DSV

В реальной последовательности информационных символов количество подряд следующих «нулей» или «единиц» может быть сколь угодно большим, и величина числовой суммы может уйти сколь угодно далеко от нулевого уровня в ту или иную сторону. Способ кодирования 4/5 предполагает, что в каждый момент времени по окончании передачи очередной группы известна величина мгновенного значения числовой суммы DSV. Известна также величина числовой суммы той группы из четырех бит, которую предстоит записать. Эта величина называется **DSP (Digital Sum Pattern)**. Кодирование состоит в том, что если эти две величины имеют разные знаки, то к исходной группе спереди добавляется «единица» и полученное 5-разрядное кодовое слово передается на запись вместо исходного 4-разрядного, уменьшая тем самым общую величину DSV. Если текущее значение DSV и DSP очередного 4-разрядного информационного слова имеют одинаковые знаки, то информационное слово инвертируется и спереди к нему добавляется «ноль». Полученное 5-разрядное кодовое слово будет иметь DSP противоположное по знаку исходному, за счет чего общее значение DSV формируемой последовательности уменьшится.

Таким образом, кодирование по методу 4/5 предполагает выбор кодового слова, соответствующего заданному информационному слову, из

двух альтернативных взаимно инверсных вариантов (табл.3.2). Один из них – это само информационное слово с добавленной «единицей» в качестве первого разряда, другой – его инверсия с «нулем» впереди. Первый разряд служит для декодера индикатором того, нужно ли инвертировать принятое слово или этого делать не требуется.

Таблица 3.2

Преобразование 4/5

Информационное слово	Кодовое слово +	DSP	Кодовое слово -	DSP
0000	10000	-3	01111	+3
0001	10001	-1	01110	+1
0010	10010	-1	01101	+1
0011	10011	+1	01100	-1
0100	10100	-1	01011	+1
0101	10101	+1	01010	-1
0110	10110	+1	01001	-1
0111	10111	+3	01000	-3
1000	11000	-1	00111	+1
1001	11001	+1	00110	-1
1010	11010	+1	00101	-1
1011	11011	+3	00100	-3
1100	11100	+1	00011	-1
1101	11101	+3	00010	-3
1110	11110	+3	00001	-3
1111	11111	+5	00000	-5

Поскольку модуляция по методу NRZI здесь не предусмотрена, то длина последовательностей одного уровня, характеризующих длину пробега, определяется количеством подряд следующих «единиц» или «нулей». Минимальная длина пробега T_{min} , как следует из табл. 3.2, равна одному каналному тактовому интервалу (изолированная «единица» либо изолированный «нуль»). Минимальная длина волны записи $\lambda_{min} = 2$.

Максимальную длину пробега T_{max} можно определить при наихудшем сочетании кодовых слов с разными знаками DSP. Самая длинная последовательность «нулей» получается при кодировании исходных информационных слов 1100 1111 1000. Последовательность кодовых слов

при условии чередования DSP с разными знаками получается следующей: 11100 00000 00111 (DSP>0, DSP<0, DSP>0). Таким образом, $T_{max0} = 9$ (9 последовательных «нулей»). Самая длинная последовательность «единиц» получается при кодировании тех же самых исходных информационных слов, но при противоположном сочетании значений DSP: 00011 11111 11000 (DSP>0, DSP<0, DSP>0). Следовательно, $T_{max1} = 9$ (9 последовательных «единиц»). Самосинхронизация кодированной последовательности, таким образом, обеспечивается. Максимальная длина волны записи $\lambda_{max} = 18$.

Что касается повышения плотности записи, то его при использовании кода 4/5 не получается: $K = n/m = 4/5$, т. е. плотность записи уменьшается на 20%. Окно детектирования уменьшается на ту же самую величину: $t_w = 4/5 T = \pm 2/5 T$.

Основная энергия последовательности, кодированной по методу 4/5, будет располагаться в диапазоне частот от $1/\lambda_{max} = 1/18 T_{кан} = f_K/18$ до $1/\lambda_{min} = 1/2 T_{кан} = f_K/2$ (рис. 3.4).

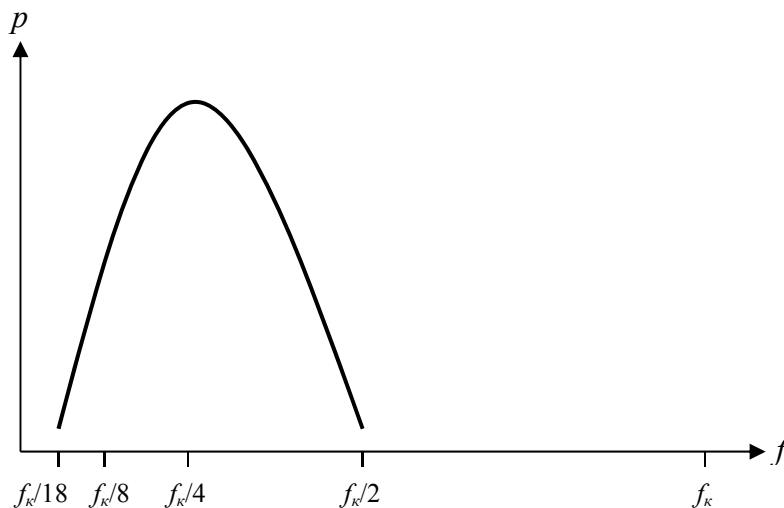


Рис. 3.4. Спектр мощности кода 4/5

Технические характеристики кода 4/5:

- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 18T_{кан}$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = 9$;
- скорость кода $R = 4/5$;
- коэффициент повышения плотности записи $K = 0,8$ (плотность записи уменьшилась);
- окно детектирования $t_w = 4/5 T = \pm 2/5 T$.

3.3. Код 4/5 NRZI

Несмотря на то, что данный код, так же, как и предыдущий, основан на преобразовании 4-разрядных информационных слов в 5-разрядные кодовые ($n = 4, m = 5$), правила кодирования обоих кодов существенно отличаются.

Преобразование информационных слов в кодовые здесь также осуществляется с помощью таблицы (табл. 3.3), но соответствие между словами вполне однозначное, а не альтернативное, как в предыдущем случае. Кроме того, после преобразования осуществляется модуляция кодовых слов по методу NRZI (что и отражено в названии кода) (рис. 3.5).

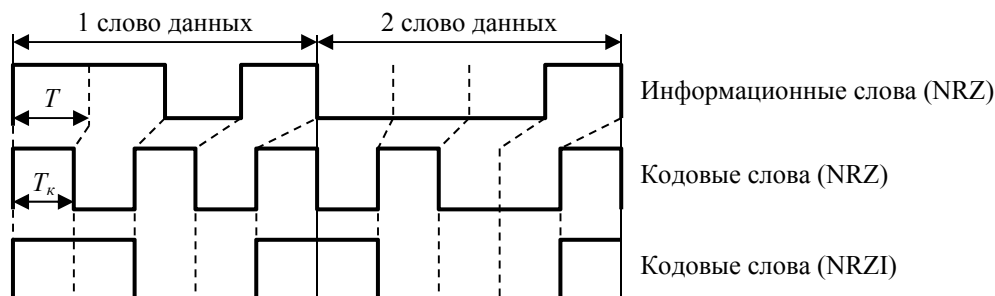


Рис. 3.5. Формирование кодовых слов кода 4/5 NRZI

Как видно из табл. 3.3, ограничение минимального количества «нулей» между двумя соседними «единицами» не предусмотрено ($d = 0$), максимальное же число «нулей» равно двум ($k = 2$). После модуляции по методу NRZI минимальная длина пробега получается равной одному периоду канальной тактовой частоты ($T_{min} = 1$), а максимальная – трем ($T_{max} = 3$). Следовательно, минимальная длина волны записи $\lambda_{min} = 2$, максимальная длина волны записи $\lambda_{max} = 6$, что указывает на то, что самосинхронизация данного кода значительно лучше, чем предыдущего. Окно детектирования и коэффициент повышения плотности записи у обоих кодов одинаковы: $t_w = 4/5 T = \pm 2/5 T$, $K = 0,8$.

Таблица 3.3

Преобразование 4/5 NRZI

Информационное слово	Кодовое слово	Информационное слово	Кодовое слово
0000	11001	1000	11010
0001	01001	1001	11011
0010	01010	1010	10010
0011	01011	1011	10011
0100	11101	1100	11110
0101	01101	1101	10101
0110	01110	1110	10110
0111	01111	1111	10111

Поскольку интервалы между соседними перепадами уровня в кодированном сигнале могут принимать только три значения - $1T_k$, $2T_k$ и $3T_k$, код 4/5 NRZI является трехчастотным, подобно коду МФМ. Основная энергия сигнала записи сосредоточена в полосе частот от $1/\lambda_{max} = 1/6 T_{кан} = f_k/6$ до $1/\lambda_{min} = 1/2 T_{кан} = f_k/2$ (рис. 3.6).

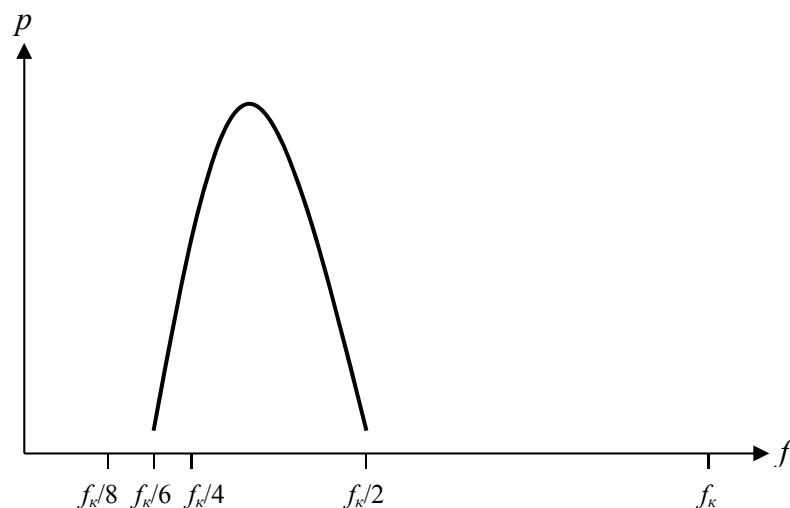


Рис. 3.6. Спектр мощности кода 4/5 NRZI

Код 4/5 NRZI использовался в свое время фирмой IBM в качестве кода записи на магнитную ленту запоминающих устройств для вычислительных машин [2,15].

Технические характеристики кода 4/5 NRZI:

- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 6T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 2$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = 3$;
- скорость кода $R = 4/5$;
- коэффициент повышения плотности записи $K = 0,8$ (плотность записи уменьшилась);
- окно детектирования $t_w = 4/5 T = \pm 2/5 T$.

3.4. Коды 2/3

Известно две разновидности кодов 2/3, в основе которых лежит преобразование 2-разрядных информационных символов 3-разрядными кодовыми. Оба кода имеют одинаковые или сходные технические характеристики и отличаются только способами альтернативной замены кодовых символов при невозможности основной замены из-за нарушения требований на минимальное количество «нулей» между двумя соседними «единицами», которое обеспечивает повышение плотности записи. Рассмотрим оба варианта кодирования.

3.4.1 Код 2/3 (I)

Данный метод кодирования, как уже говорилось выше, предполагает деление последовательности информационных бит на группы по два бита в каждой ($n = 2$) и преобразование каждой пары в кодовое слово из трех бит ($m = 3$) в соответствии с табл. 3.4.

Таблица 3.4

Преобразование 2/3

№	Информационное слово		Кодовое слово		
	a_1	a_2	b_1	b_2	b_3
1	0	0	0	1	0
2	0	1	0	0	1
3	1	0	1	0	0
4	1	1	1	0	1

В аналитическом виде преобразование выглядит следующим образом:

$$b_1 = a_1; b_3 = a_2; b_2 = \overline{b_1 + b_3},$$

где a_1 – старший, а a_2 – младший разряды исходного информационного слова, b_1 – b_3 – старший, средний и младший разряды кодового слова, а знак «+» обозначает операцию логического сложения (дизъюнкцию).

Если при каскадном соединении полученных трехразрядных кодовых слов в месте их соединения оказываются рядом две единицы (сочетания 2-3, 2-4, 4-3 и 4-4), то преобразование производится по альтернативному варианту в соответствии с табл. 3.5.

Таблица 3.5

Альтернативный вариант преобразования пары информационных слов

Пары	Информационные слова	Кодовые слова
	$a_{11} a_{12} \bullet a_{21} a_{22}$	$b_{11} b_{12} b_{13} \bullet b_{21} b_{22} b_{23}$
2•3	01•10	010•000
2•4	01•11	001•000
4•3	11•10	100•000
4•4	11•11	101•000

В аналитическом виде:

$$b_{11} = a_{11}, b_{13} = a_{22}, b_{12} = \overline{b_{11} + b_{13}} \text{ и } b_{21} = b_{22} = b_{23} = \overline{a_{12}} = \overline{a_{21}} = 0.$$

Если все соседние разряды, относящиеся к трем подряд следующим группам, равны «единицам», то первые две группы преобразуются по правилу, соответствующему табл. 3.5, а последняя – по правилу, соответствующему табл. 3.4.

После выполнения замен, полученная последовательность модулируется по методу NRZI [16].

Как видно из табл. 3.4 и 3.5, между двумя соседними «единицами» располагается по меньшей мере один «нуль» ($d = 1$), максимальное же количество «нулей» равно семи ($k = 7$). Таким образом, после выполнения преобразования по методу NRZI минимальное расстояние между перепадами уровня (минимальная длина пробега) равно двум периодам тактовой частоты ($T_{min} = 2$), максимальное (максимальная длина пробега) – восьми ($T_{max} = 8$), что позволяет повысить плотность записи и обеспечить самосинхронизацию формируемой последовательности бит. Минимальная длина волны записи составит $\lambda_{min} = 4T_{кан}$, максимальная $\lambda_{max} = 16T_{кан}$. Коэффициент повышения плотности записи $K = (d+1) \cdot n/m = 4/3 \approx 1,33$. Окно детектирования $t_w = 2/3T = \pm 1/3 T$.

Для того чтобы обеспечить снижение уровня постоянной и низкочастотных составляющих в спектре формируемого сигнала, предусмотрен расширенный алгоритм преобразования, в соответствии с которым последовательность трехразрядных кодовых слов делится на блоки по L кодовых слов в каждом и, либо первое, либо последнее кодовое слово заменяется балансирующей четырехразрядной группой p в соответствии с табл. 3.6. Для первых трех кодовых слов 010, 001 и 100 существует два альтернативных варианта балансирующей группы: p_1 и p_2 , одна из которых содержит четное число «единиц», другая – нечетное. Выбор варианта замены осуществляется следующим образом. Если балансирующая группа расположена в начале блока из L кодовых слов, то рассматриваются знаки DSV в конце предыдущего блока и DSP текущего. Если они совпадают, то выбирается балансирующая группа с нечетным числом «единиц», которая позволит изменить знак DSP текущего блока и, таким образом, минимизировать суммарное значение DSV в момент его окончания. Если знаки DSV в конце предыдущего блока и DSP текущего различны, то выбирается балансирующая группа с четным числом «единиц», которая сохраняет сложившуюся ситуацию, благоприятную для минимизации значения DSV в конце текущего блока.

Таблица 3.6

Выбор балансирующей группы

Кодовое слово $c_1c_2c_3$	Вариант	Балансирующая группа $b_1b_2b_3b_4$
010	p_1	0010
	p_2	0000
001	p_1	0101
	p_2	0001
100	p_1	1010
	p_2	1000
000	p_1	0100
101	p_1	1001

Расширенный вариант преобразования предполагает некоторое уменьшение окна детектирования, величина которого зависит от длины блока L :

$$t_w = \frac{3L}{3L+1} \cdot \frac{2}{3} T.$$

Очевидно, что чем больше длина блока L , тем меньше разница между величинами окна детектирования при обычном алгоритме кодирования и при расширенном, т.е. тем величина окна детектирования больше. При $L = 6$ $t_w = 0,63T$.

Основная энергия кодированного сигнала будет располагаться в полосе частот от $1/\lambda_{max} = 1/16 T_{кан} = f_k/16$ до $1/\lambda_{min} = 1/4 T_{кан} = f_k/4$ (рис. 3.7).

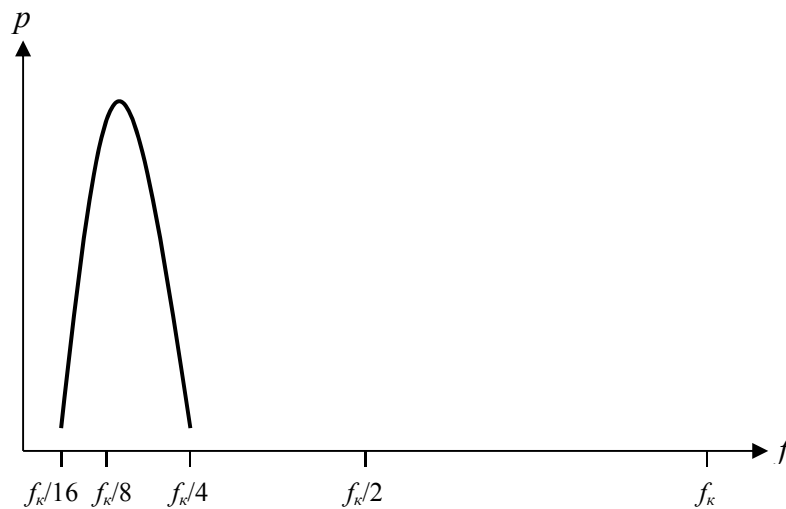


Рис. 3.7. Спектр мощности кода 2/3

3.4.2 Код 2/3 (II)

Данный код был предложен Дж.В. Якоби в 1982 году в качестве замены своему предшественнику – коду ЗРМ [17, 18]. В сравнении с кодом ЗРМ, имеет на 33% большее окно детектирования, что делает его более устойчивым к шумам. Использовался в качестве канального кода для записи информации на жесткие диски стандарта ISS-8470 емкостью 683 Мбайт.

Рассматриваемый код, также как и предыдущий, на первом этапе кодирования предусматривает деление последовательности информационных бит на пары ($n = 2$) и замену каждой такой пары на кодовое слово из трех бит ($m = 3$) в соответствии с табл. 3.7. После чего полученная последовательность кодовых слов модулируется по методу NRZI.

Минимальное количество «нулей» между двумя соседними «единицами» в соответствии с данной технологией кодирования должно быть не менее одного ($d = 1$). Поскольку при непосредственном соединении кодовых слов это правило, очевидно, будет нарушаться, то далее

рассматриваются пары кодовых слов и информационных слов, т.е. 16 их возможных сочетаний (табл. 3.8-3.11).

Таблица 3.7

Преобразование 2/3

№	Информационное слово	Кодовое слово
0	00	101
1	01	100
2	10	001
3	11	010

Таблица 3.8

Серия сочетаний 0-X

Сочетание	Информационные слова	Кодовые слова
0•0	00•00	101•101
0•1	00•01	101•100
0•2	00•10	101•001
0•3	00•11	101•010

Таблица 3.9

Серия сочетаний 1-X

Сочетание	Информационные слова	Кодовые слова
1•0	01•00	100•101
1•1	01•01	100•100
1•2	01•10	100•001
1•3	01•11	100•010

Таблица 3.10

Серия сочетаний 2-X

Сочетание	Информационные слова	Кодовые слова
2•0	10•00	001•101
2•1	10•01	001•100
2•2	10•10	001•001
2•3	10•11	001•010

Серия сочетаний 3-Х

Сочетание	Информационные слова	Кодовые слова
3•0	11•00	010•101
3•1	11•01	010•100
3•2	11•10	010•001
3•3	11•11	010•010

Анализ табл. 3.8 – 3.11 показывает, что в четырех случаях из 16 (сочетания 0-0, 0-1, 2-0 и 2-1) условие $d = 1$ нарушается. Для этих случаев предусмотрены альтернативные замены 3-разрядных кодовых слов 101 и 100, вызвавших нарушение, на 000 (табл. 3.12).

Таблица 3.12

Альтернативные замены кодовых слов

Сочетание	Информационные слова	Кодовые слова
0•0	00•00	101•000
0•1	00•01	100•000
2•0	10•00	001•000
2•1	10•01	010•000

Для того чтобы можно было правильно декодировать кодовые слова с учетом альтернативной замены, используется алгоритм, отображенный в виде табл. 3.13.

Таблица 3.13

Таблица декодирования кода 2/3

Последний бит предыдущего кодового слова	Декодируемое кодовое слово	Кодовое слово, следующее за декодируемым	Декодированное слово данных
N	101	XXX	00
N	100	XXX	01
N	001	XXX	10
N	010	XXX	11
N	101	000	00
N	100	000	00
N	001	000	10
N	010	000	10
0	000	NNN	01
1	000	NNN	00

XXX означает, что в кодовом слове не все биты «нули», N означает безразличное значение данного бита, т.е. он может быть и «нулем» и «единицей».

Как можно заметить из таблиц преобразования, максимальное количество «нулей» между двумя соседними «единицами» (наихудший случай: 100•000•001) равно 7 ($k = 7$). Следовательно, максимальная длина пробега $T_{max} = k + 1 = 8$, а минимальная длина пробега $T_{min} = d + 1 = 2$. Минимальная длина волны записи $\lambda_{min} = 4T_{кан}$, максимальная $\lambda_{max} = 16 T_{кан}$. Спектр мощности такой же, как и у предыдущего кода (см. рис. 3.4). Совпадают у них также и коэффициент повышения плотности записи $K = (d+1) \cdot n/m = 4/3 \approx 1,33$ и величина окна детектирования $t_w = 2/3 T = \pm 1/3 T$.

Таким образом, технические характеристики обоих кодов 2/3 следующие:

- минимальная длина волны записи $\lambda_{min} = 4T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 16T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 1$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 7$;
- минимальная длина пробега $T_{min} = 2$;
- максимальная длина пробега $T_{max} = 8$;
- скорость кода $R = 2/3$;
- коэффициент повышения плотности записи $K = 1,33$ (плотность записи увеличилась);
- окно детектирования $t_w = 2/3 T = \pm 1/3 T$.

3.5. Код 2/4

Последовательность битов данных в данном методе кодирования делится на группы из двух бит ($n = 2$), и каждая такая группа заменяется 4-разрядной кодовой ($m = 4$) в соответствии с одним из вариантов, приведенным в табл. 3.14 и 3.15. Полученная последовательность затем модулируется по способу NRZI [19].

Таблица 3.14

Преобразование 2/4 (вариант «а»)

Информационное слово	Кодовое слово	Условия
11	Y000	
01	0010	
10	0100	$E_2E_1 \neq 10$ и $L_1L_2 = 00$
10	0001	$E_2E_1 = 10$ и $L_1L_2 = 00$
10	0000	$E_2E_1 = 10$, $L_1L_2 = 10$ и $L_3L_4 = 00$
10	0100	$E_2E_1 \neq 10$, $L_1L_2 = 10$ и $L_3L_4 = 00$
10	Y001	Во всех других случаях
00	0000	$E_2E_1 = 10$
00	0100	Во всех других случаях

E_n – разряд исходных данных, стоящий на n -й позиции перед первым разрядом рассматриваемой группы (рис. 3.8);

L_n – разряд исходных данных, располагающийся на n -й позиции после последнего разряда рассматриваемой группы (рис. 3.8);

Y – инверсия логической суммы двух разрядов последовательности кодовых бит, предшествующих разряду Y (рис. 3.9).

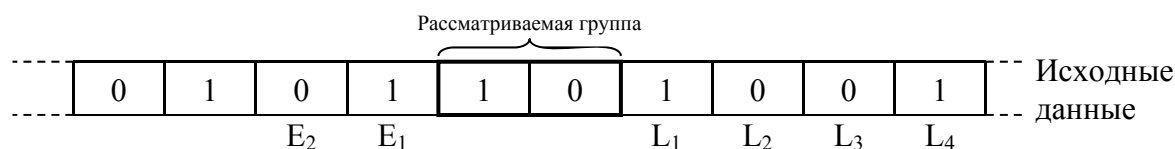


Рис. 3.8. Расположение символов E и L в потоке исходных данных

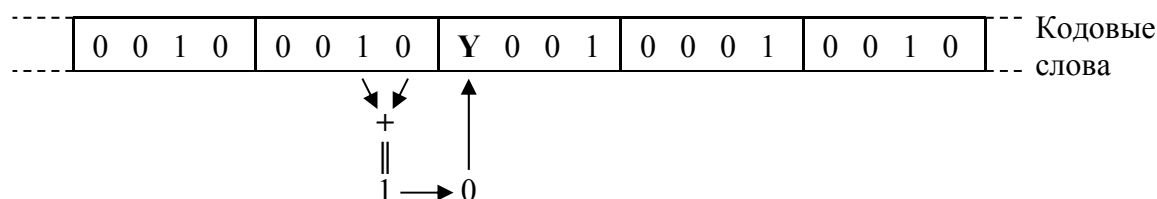


Рис. 3.9. Образование символа Y

Таблица 3.14

Преобразование 2/4 (вариант б)

Информационное слово	Кодовое слово	Условия
11	Y000	
01	0010	
10	0100	$E_2E_1 \neq 10$ и $L_1L_2 = 00$
10	0001	$E_2E_1 = 10$ и $L_1L_2 = 00$
10	0000	$E_2E_1 = 10$, $L_1L_2 = 10$ и $L_3L_4 = 00$
10	0100	$E_2E_1 \neq 10$, $L_1L_2 = 10$ и $L_3L_4 = 00$
10	Y001	Во всех других случаях
00	0000	$E_4E_3 \neq 10$, $E_2E_1 = 10$ и $L_1L_2 \neq 01$
00	0001	$E_4E_3 \neq 10$, $E_2E_1 = 10$ и $L_1L_2 = 01$
00	0000	$E_2E_1 = 10$
00	0100	Во всех других случаях

В результате преобразования минимальное число «нулей» между двумя соседними «единицами» оказывается равным двум ($d = 2$), а максимальное - при варианте «а» - восьми ($k = 8$), при варианте «б» - семи ($k = 7$). Минимальная длина пробега $T_{min} = 3$, максимальная - при варианте «а» $T_{max} = 9$, при варианте «б» $T_{max} = 8$. Минимальная длина волны записи $\lambda_{min} = 6T_{кан}$, максимальная при варианте «а» - $\lambda_{max} = 18T_{кан}$, при варианте «б» - $\lambda_{max} = 16T_{кан}$. Спектр мощности кодированного сигнала сосредоточен в достаточно узкой полосе частот от $f_k/6$ до $f_k/18$ при варианте «а» или до $f_k/16$ при

варианте «б» (рис. 3.10). Коэффициент повышения плотности записи $K = (d+1) \cdot n/m = 6/4 = 1,5$. Окно детектирования $t_w = 2/4 T = 1/2 T = \pm 1/4 T$.

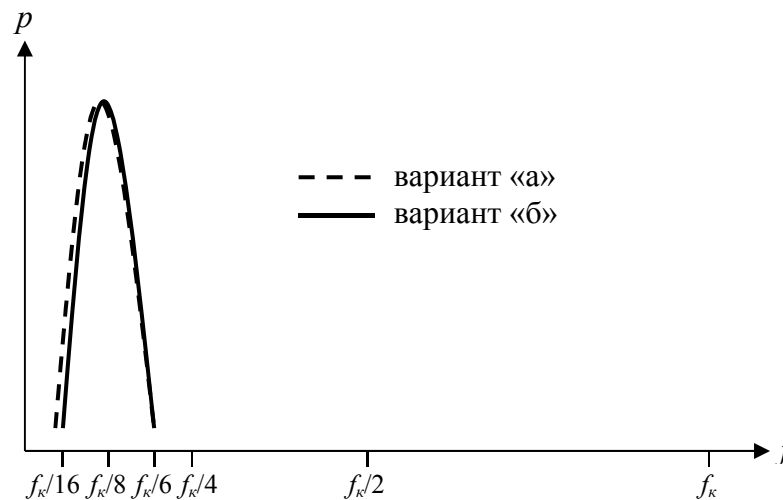


Рис. 3.10. Спектр мощности кода 2/4

Код обладает следующими техническими характеристиками:

- минимальная длина волны записи $\lambda_{min} = 6T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 18 T_{кан}$ («а»); $\lambda_{max} = 16T_{кан}$ («б»);
- минимальное количество «нулей» между двумя соседними «единицами» $d = 2$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 8$ («а»); $k = 7$ («б»);
- минимальная длина пробега $T_{min} = 3$;
- максимальная длина пробега $T_{max} = 9$ («а»); $T_{max} = 8$ («б»);
- скорость кода $R = 1/2$;
- коэффициент повышения плотности записи $K = 1,5$ (плотность записи увеличилась);
- окно детектирования $t_w = 1/2 T = \pm 1/4 T$.

3.6. Код EFM

Канальный код EFM (Eight to Fourteen Modulation – модуляция 8/14) был предложен в конце 70-х годов специально для разрабатываемого в то время формата лазерной звукозаписи, который впоследствии стал известен как компакт-диск (CD) [20-25]. В соответствии с правилами кодирования EFM, каждый 8-разрядный символ заменяется однозначно ему соответствующим 14-разрядным символом из таблицы преобразования (табл.3.16).

Таблица 3.16-1

Таблица преобразования 8-14 кода EFM

№п/п	Информационное слово	Кодовое слово	№п/п	Информационное слово	Кодовое слово
0	00000000	01001000100000	32	00100000	00000000100000
1	00000001	10000100000000	33	00100001	10000100001000
2	00000010	10010000100000	34	00100010	00001000100000
3	00000011	10001000100000	35	00100011	00100100100000
4	00000100	01000100000000	36	00100100	01000100001000
5	00000101	00000100010000	37	00100101	00000100001000
6	00000110	00010000100000	38	00100110	01000000100000
7	00000111	00100100000000	39	00100111	00100100001000
8	00001000	01001001000000	40	00101000	01001001001000
9	00001001	10000001000000	41	00101001	10000001001000
10	00001010	10010001000000	42	00101010	10010001001000
11	00001011	10001001000000	43	00101011	10001001001000
12	00001100	01000001000000	44	00101100	01000001001000
13	00001101	00000001000000	45	00101101	00000001001000
14	00001110	00010001000000	46	00101110	00010001001000
15	00001111	00100001000000	47	00101111	00100001001000
16	00010000	10000000100000	48	00110000	00000100000000
17	00010001	10000010000000	49	00110001	10000010001000
18	00010010	10010010000000	50	00110010	10010010001000
19	00010011	00100000100000	51	00110011	10000100010000
20	00010100	01000010000000	52	00110100	01000010001000
21	00010101	00000010000000	53	00110101	00000010001000
22	00010110	00010010000000	54	00110110	00010010001000
23	00010111	00100010000000	55	00110111	00100010001000
24	00011000	01001000010000	56	00111000	01001000001000
25	00011001	10000000010000	57	00111001	10000000001000
26	00011010	10010000010000	58	00111010	10010000001000
27	00011011	10001000010000	59	00111011	10001000001000
28	00011100	01000000010000	60	00111100	01000000001000
29	00011101	00001000010000	61	00111101	00001000001000
30	00011110	00010000010000	62	00111110	00010000001000
31	00011111	00100000010000	63	00111111	00100000001000

Таблица 3.16-2

Таблица преобразования 8-14 кода EFM (продолжение)

№п/п	Информационное слово	Кодовое слово	№п/п	Информационное слово	Кодовое слово
64	01000000	01001000100100	96	01100000	01001000100010
65	01000001	10000100100100	97	01100001	10000100100010
66	01000010	10010000100100	98	01100010	10010000100010
67	01000011	10001000100100	99	01100011	10001000100010
68	01000100	01000100100100	100	01100100	01000100100010
69	01000101	00000000100100	101	01100101	00000000100010
70	01000110	00010000100100	102	01100110	01000000100100
71	01000111	00100100100100	103	01100111	00100100100010
72	01001000	01001001000100	104	01101000	01001001000010
73	01001001	100000001000100	105	01101001	100000001000010
74	01001010	100100001000100	106	01101010	100100001000010
75	01001011	10001001000100	107	01101011	10001001000010
76	01001100	010000001000100	108	01101100	010000001000010
77	01001101	000000001000100	109	01101101	000000001000010
78	01001110	000100001000100	110	01101110	000100001000010
79	01001111	001000001000100	111	01101111	001000001000010
80	01010000	10000000100100	112	01110000	10000000100010
81	01010001	10000010000100	113	01110001	10000010000010
82	01010010	10010010000100	114	01110010	10010010000010
83	01010011	00100000100100	115	01110011	00100000100010
84	01010100	0100000010000100	116	01110100	010000001000010
85	01010101	0000000010000100	117	01110101	000000001000010
86	01010110	00010010000100	118	01110110	00010010000010
87	01010111	0010000010000100	119	01110111	001000001000010
88	01011000	01001000000100	120	01111000	01001000000010
89	01011001	100000000000100	121	01111001	00001001001000
90	01011010	100100000000100	122	01111010	10010000000010
91	01011011	100010000000100	123	01111011	10001000000010
92	01011100	010000000000100	124	01111100	01000000000010
93	01011101	000010000000100	125	01111101	00001000000010
94	01011110	000100000000100	126	01111110	00010000000010
95	01011111	001000000000100	127	01111111	00100000000010

Таблица 3.16-3

Таблица преобразования 8-14 кода EFM (продолжение)

№п/п	Информационное слово	Кодовое слово	№п/п	Информационное слово	Кодовое слово
128	10000000	01001000100001	160	10100000	00001000100001
129	10000001	10000100100001	161	10100001	10000100001001
130	10000010	10010000100001	162	10100010	01000100010000
131	10000011	10001000100001	163	10100011	00000100100001
132	10000100	01000100100001	164	10100100	01000100001001
133	10000101	00000000100001	165	10100101	00000100001001
134	10000110	00010000100001	166	10100110	01000000100001
135	10000111	00100100100001	167	10100111	00100100001001
136	10001000	01001001000001	168	10101000	01001001001001
137	10001001	10000001000001	169	10101001	10000001001001
138	10001010	10010001000001	170	10101010	10010001001001
139	10001011	10001001000001	171	10101011	10001001001001
140	10001100	01000001000001	172	10101100	01000001001001
141	10001101	00000001000001	173	10101101	00000001001001
142	10001110	00010001000001	174	10101110	00010001001001
143	10001111	00100001000001	175	10101111	00100001001001
144	10010000	10000000100001	176	10110000	00000100100000
145	10010001	10000010000001	177	10110001	100000100001001
146	10010010	10010010000001	178	10110010	100100100001001
147	10010011	00100000100001	179	10110011	00100100010000
148	10010100	01000010000001	180	10110100	010000100001001
149	10010101	00000010000001	181	10110101	000000100001001
150	10010110	00010010000001	182	10110110	000100100001001
151	10010111	00100010000001	183	10110111	001000100001001
152	10011000	01001000000001	184	10111000	010010000001001
153	10011001	10000010010000	185	10111001	10000000001001
154	10011010	10010000000001	186	10111010	100100000001001
155	10011011	10001000000001	187	10111011	100010000001001
156	10011100	01000010010000	188	10111100	010000000001001
157	10011101	00001000000001	189	10111101	000010000001001
158	10011110	00010000000001	190	10111110	000100000001001
159	10011111	00100010010000	191	10111111	001000000001001

Таблица 3.16-4

Таблица преобразования 8-14 кода EFM (продолжение)

№п/п	Информационное слово	Кодовое слово	№п/п	Информационное слово	Кодовое слово
192	11000000	01000100100000	224	11100000	01000100000010
193	11000001	10000100010001	225	11100001	00000100000010
194	11000010	10010010010000	226	11100010	10000100010010
195	11000011	00001000100100	227	11100011	00100100000010
196	11000100	01000100010001	228	11100100	01000100010010
197	11000101	00000100010001	229	11100101	00000100010010
198	11000110	00010010010000	230	11100110	01000000100010
199	11000111	00100100010001	231	11100111	00100100010010
200	11001000	00001001000001	232	11101000	10000100000010
201	11001001	10000100000001	233	11101001	10000100000100
202	11001010	00001001000100	234	11101010	00001001001001
203	11001011	00001001000000	235	11101011	00001001000010
204	11001100	01000100000001	236	11101100	01000100000100
205	11001101	00000100000001	237	11101101	00000100000100
206	11001110	00000010010000	238	11101110	00010000100010
207	11001111	00100100000001	239	11101111	00100100000100
208	11010000	00000100100100	240	11110000	00000100100010
209	11010001	10000010010001	241	11110001	10000010010010
210	11010010	10010010010001	242	11110010	10010010010010
211	11010011	10000100100000	243	11110011	00001000100010
212	11010100	01000010010001	244	11110100	01000010010010
213	11010101	00000010010001	245	11110101	00000010010010
214	11010110	00010010010001	246	11110110	00010010010010
215	11010111	00100010010001	247	11110111	00100010010010
216	11011000	01001000010001	248	11111000	01001000010010
217	11011001	10000000010001	249	11111001	10000000010010
218	11011010	10010000010001	250	11111010	10010000010010
219	11011011	10001000010001	251	11111011	10001000010010
220	11011100	01000000010001	252	11111100	01000000010010
221	11011101	00001000010001	253	11111101	00001000010010
222	11011110	00010000010001	254	11111110	00010000010010
223	11011111	00100000010001	255	11111111	00100000010010

Количество исходных 8-разрядных символов, очевидно, равно $2^8=256$, а 14-разрядных символов может быть $2^{14}=16384$. Но из этого множества 14-разрядных комбинаций подобрано 256 таких, у которых между двумя соседними единицами не меньше двух ($d = 2$), но не больше 10 «нулей» ($k = 10$). На самом деле символов, удовлетворяющим таким требованиям на ограничение последовательности «нулей», чуть больше – 267, но 11 оказались уже лишними.

Кроме того, к каждому полученному в результате преобразования 14-разрядному символу добавляются еще 3 **соединительных разряда**. Эти соединительные разряды никакой информации не несут и нужны только для того, чтобы при сопряжении двух символов, один из которых заканчивается «единицей», а другой с нее начинается, создать между ними интервал в три «нулевых» бита, а в случае, если при сопряжении возникнет последовательность «нулей» длиной больше десяти, изменить значение одного из расположенных между ними «нулевых» соединительных разрядов на «единицу». После этого полученная последовательность модулируется по способу NRZI (рис.3.11).

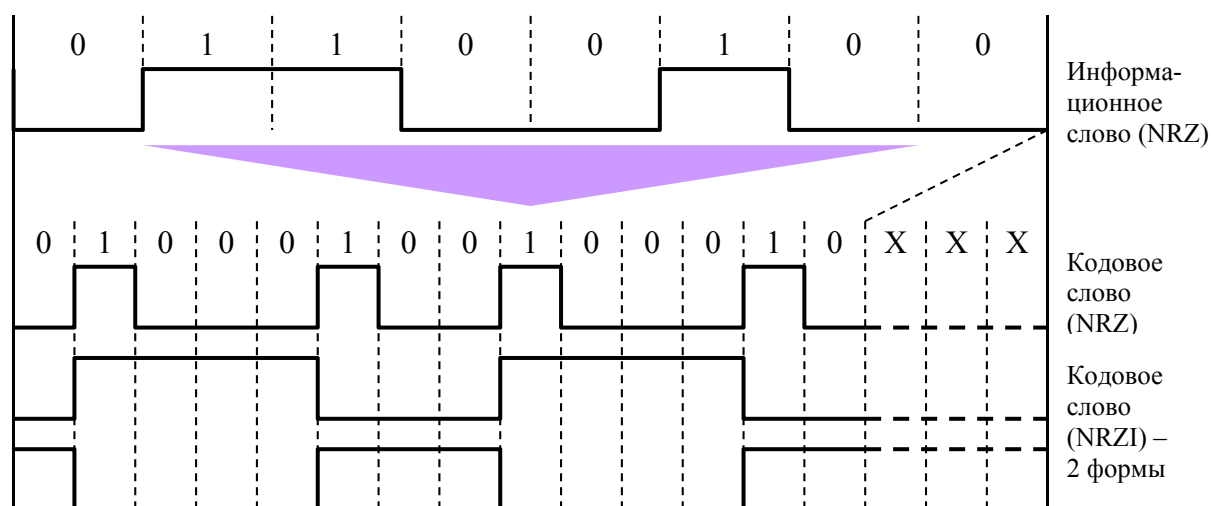


Рис.3.11.Преобразование символа по способу EFM

Таким образом, длина исходного информационного слова составляет 8 бит ($n = 8$), а длина кодового слова $14+3 = 17$ бит ($m = 17$). Минимальная длина последовательности одного уровня для кода EFM составляет три канальных разряда ($T_{min} = 3$), а максимальная – 11 канальных разрядов ($T_{max} = 11$). Следовательно, минимальная длина волны записи $\lambda_{min} = 6T_k$, максимальная длина волны записи $\lambda_{max} = 22T_k$. Поскольку канальная тактовая частота в системе CD известна и составляет 4,3218 МГц, то можно рассчитать верхнюю и нижнюю частоту спектра сигнала записи (рис. 3.12):

$$F_{max} = 4,3218 \text{ МГц}/6 = 720 \text{ кГц.}$$

$$F_{min} = 4,3218 \text{ МГц}/22 = 196 \text{ кГц.}$$

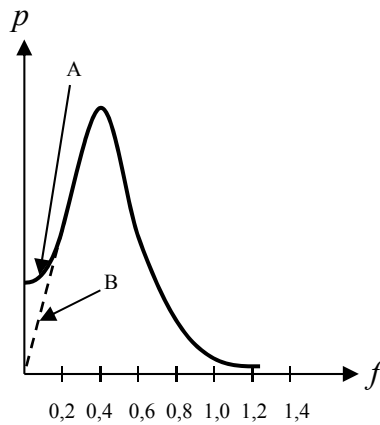


Рис. 3.12. Спектр кода EFM

Если посмотреть на спектр мощности информационной последовательности, модулированной по правилам EFM, показанный на рис.3.12, то видно, что основная энергия находится в промежутке между двумя этими частотами.

Однако в области низких частот наблюдается довольно высокий уровень составляющих спектра, которые могут затруднить работу сервосистем проигрывателя – автофокусировки и автотрекинга [23, 26]. Текущие изменения в спектре информационного сигнала могут быть восприняты как ошибки позиционирования оптической головки.

Для борьбы с низкочастотными составляющими спектра в правилах кодирования EFM предусмотрена следующая процедура, предусматривающая контроль DSV формируемой последовательности (рис.3.13).

При формировании соединительных разрядов между двумя последовательными канальными символами рассматриваются четыре варианта с разными значениями DSV. Один из них – когда «единица» в соединительные разряды не вставляется, и три варианта – с формированием «единицы» на одной из трех возможных позиций. Из них выбирается тот, при котором величина DSV минимальна. Из рассмотрения исключаются только те варианты, когда нарушаются требования на ограничение длины пробега снизу или сверху.

В примере на рис. 3.13.а текущая величина DSV в конце предыдущего канального символа, закончившегося в момент времени t_0 , равна +3. Последний его разряд – низкого уровня. Следующий символ, расположенный во временном промежутке между t_1 и t_2 , имеет DSP = +4. Если в соединительных разрядах, располагающихся между ними, нет «единицы», то их DSP = -3. Общая числовая сумма такого варианта равна +3-3+4 = +4. Наглядно текущее изменение DSV от разряда к разряду показано на рис. 3.13,б.

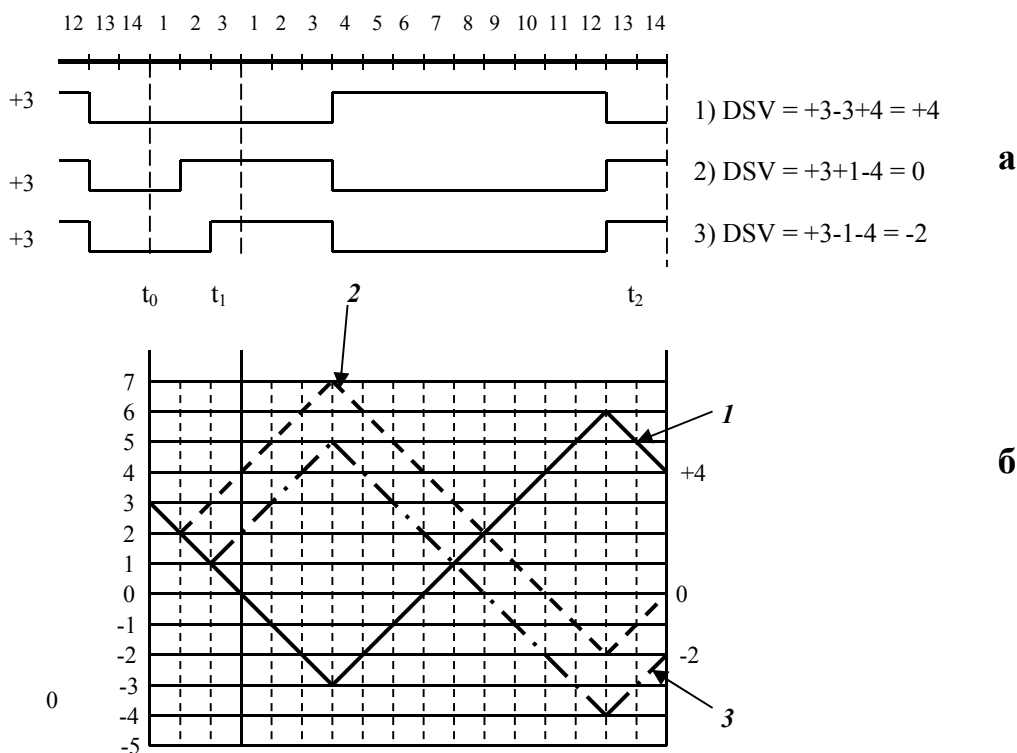


Рис.3.13. Формирование соединительных разрядов

Следующий вариант, когда «единица» формируется в первом соединительном разряде, исключается, так как при этом длина последовательности «нулей» получается равной двум, что меньше минимально допустимой в три разряда.

А во второй соединительный разряд «единицу» вставлять можно. При этом их DSP будет равна +1, а величина DSP следующего за ними символа инвертируется и превратится из +4 в -4. Общая числовая сумма при этом станет равной $+3+1-4 = 0$.

Последний вариант – с «единицей» в третьем соединительном разряде – даст $DSV = -2$.

Очевидно, что минимальная абсолютная величина DSV, равная 0, получается при формировании «единицы» во втором соединительном разряде. Значит, именно этот вариант и будет использован.

Следует заметить, однако, что стратегия формирования соединительных разрядов по минимуму абсолютной величины DSV в конце рассматриваемого канального символа не является единственно возможной. Кроме нее могут использоваться и другие стратегии, например, по критерию максимального числа пересечений нулевой линии DSV. В этом случае вариант 1 был бы предпочтительнее, так как тут два пересечения против одного у остальных двух. Также может использоваться критерий минимального удаления изломов линии DSV от оси. Здесь предпочтительнее вариант 3, где наиболее удаленный от нулевой линии излом расположен на уровне +5, против +6 и +7 у первого и второго вариантов соответственно.

Эти и другие стратегии могут использоваться в различных сочетаниях друг с другом и с разными приоритетами. Как правило, так и делается, потому что одна стратегия в некоторых случаях может дать два одинаковых результата. Чтобы выяснить, какой из них предпочтительнее, прибегают к другой стратегии с более низким приоритетом.

Контроль DSV помогает устранить из спектра кода EFM постоянную составляющую и существенно снизить уровень примыкающих к ней низкочастотных компонент [26-30]. На рис. 3.12 кривая А соответствует спектру мощности кода EFM без использования технологии снижения уровня постоянной составляющей с помощью контроля DSV, а кривая В – с использованием такой технологии.

Для реализации преобразования 8-разрядных информационных слов в 14-разрядные кодовые можно использовать постоянное запоминающее устройство (ПЗУ) емкостью 256×14 бит, которое программируется исходя из заданной стандартом таблицы преобразования (табл. 3.15). При этом каждые 8 разрядов входного информационного слова, которые можно обозначить как X_1, X_2, \dots, X_8 , будут служить адресом соответствующих 14 ячеек, в которых содержатся 14 бит (Y_1, Y_2, \dots, Y_{14}) выходного кодового слова (рис. 3.14,а).

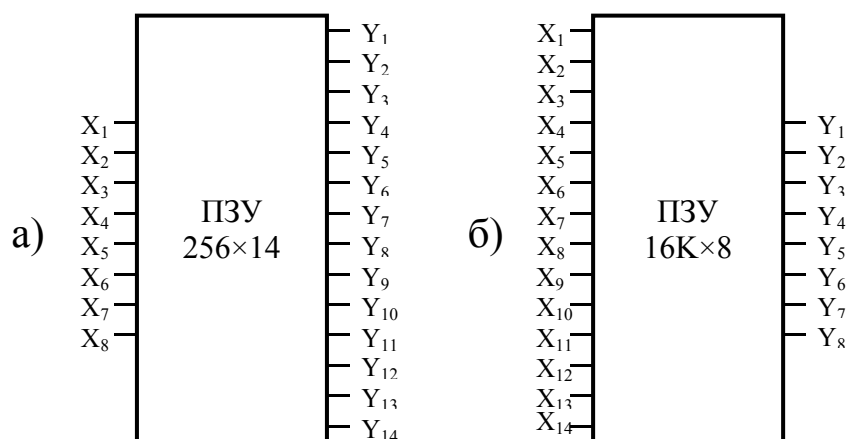


Рис. 3.14. Преобразователи 8/14 (а) и 14/8 на основе ПЗУ

Такую же процедуру можно использовать и для обратного преобразования 14/8 на стороне декодера (рис. 3.14,б). Однако для реализации преобразователя 14/8 использование ПЗУ было бы невыгодным, так как потребовался бы объем его памяти более 16000 байт, а фактически использовалось бы всего 256 ячеек. В этом случае значительно экономнее применить устройство, которое реализовывало бы заданную таблицу соответствия с помощью набора **логических функций** $Y_k = F(X_1, X_2, \dots, X_{14})$, где $X_1 \dots X_{14}$ – значения разрядов воспроизведенного 14-разрядного кодового слова, а Y_k – значение k -го разряда декодированного 8-разрядного информационного слова. При этом в качестве базовых элементов использовались бы не ячейки памяти, а логические схемы И, ИЛИ, НЕ и

ИСКЛЮЧАЮЩЕЕ ИЛИ [31]. Такие функции были выведены в [32]. Схема реализации первого разряда показана в качестве примера на рис. 3.15.

$$\begin{aligned}
 Y_1 &= X_{14} + X_{10}(X_7 + X_{13} + X_6 Z_{17}) + \overline{X_3} Z_5 (\overline{Z_9} + \overline{Z_4}) + Z_1 \overline{X_{11}} (X_8 + Z_4) Z_{16} + Z_4 Z_8 + Z_{11} (\overline{X_6} Z_6 + Z_{10}); \\
 Y_2 &= Z_4 + Z_1 X_8 + X_6 (Z_9 Z_{14} + \overline{X_9} \overline{X_{11}} X_{14}) + X_{10} [X_{14} + X_7 (X_4 + \overline{Z_3})]; \\
 Y_3 &= X_{11} + X_{13} + Z_8 X_{12} + Z_{24} + Z_{18} (Z_3 X_{10} + X_3 Z_{23}) + \overline{Z_3} \overline{X_{12}} [(X_5 + \overline{X_4} X_{14}) X_9 + (X_9 + \overline{X_{10}} X_{14}) X_6]; \\
 Y_4 &= \overline{Z_{17}} Z_{18} [X_{11} \overline{Z_4} \overline{Z_{21}} + X_9 (\overline{X_1} + \overline{Z_4})] + Z_1 (Z_{11} + X_8 X_{11} \overline{X_{14}}) + X_6 X_{10} \overline{Z_{12}} (X_1 + X_3) + X_7 (\overline{X_{10}} + Z_{12} + \\
 &+ X_1 \overline{X_4} + Z_{17}) + \overline{X_6} \overline{X_7} \overline{Z_{16}} + Z_7 (X_3 + X_1 \overline{X_4} X_5); \\
 Y_5 &= X_8 + Z_{10} Z_{11} + \overline{X_4} \overline{Z_{12}} Z_{19} + \overline{X_7} \overline{X_9} [X_{12} + \overline{X_{10}} (\overline{X_{11}} X_{14} + Z_{13} X_{13}) + X_6 (X_{13} + Z_{21})]; \\
 Y_6 &= (X_3 \overline{Z_7} + Z_6) (\overline{Z_4} Z_{15} Z_{18} + X_6 X_{13} \overline{Z_{15}}) + Z_3 \{X_4 + X_5 \overline{Z_{16}} + \overline{X_5} [X_6 (X_9 + \overline{Z_{21}} Z_{22}) + Z_4 Z_{14}]\}; \\
 Y_7 &= X_3 + X_4 + Z_2 + X_{10} \overline{X_{14}} [Z_{13} + \overline{X_{13}} (X_7 \overline{Z_9} + X_2 X_6)] + Z_5 (X_1 \overline{X_{14}} \overline{Z_4} + Z_9 X_{14}) + \\
 &+ Z_{20} [Z_{14} + X_8 (\overline{X_{11}} X_{14} + X_{11} X_{14})] + Z_{24}; \\
 Y_8 &= Z_2 + X_3 + Z_{20} (\overline{Z_{16}} + X_8 Z_{22} + X_9 Z_4) + Z_{13} (X_9 + \overline{X_{13}}) + \\
 &+ \overline{X_2} \overline{X_4} X_5 \{X_1 \overline{X_6} X_9 + \overline{X_1} [Z_{12} (Z_{23} + Z_{19} + Z_8 \overline{Z_{21}} X_{12}) + Z_5 \overline{X_{14}}]\};
 \end{aligned}$$

где $Z_1 \dots Z_{24}$ – вспомогательные функции:

$$\begin{aligned}
 Z_1 &= \overline{X_1} \overline{X_2} X_5; & Z_{13} &= X_1 X_6; \\
 Z_2 &= X_1 X_5; & Z_{14} &= X_9 \overline{X_{14}}; \\
 Z_3 &= X_1 + X_2 + X_3 = \overline{\overline{\overline{X_1 X_2 X_3}}}; & Z_{15} &= X_9 + X_{10} = \overline{\overline{\overline{X_9 X_{10}}}}; \\
 Z_4 &= X_{12} + X_{13} = \overline{\overline{\overline{X_{12} X_{13}}}}; & Z_{16} &= X_8 + X_9 = \overline{\overline{\overline{X_8 X_9}}}; \\
 Z_5 &= X_6 X_9; & Z_{17} &= X_2 + X_3 = \overline{\overline{\overline{X_2 X_3}}}; \\
 Z_6 &= X_2 \overline{X_5}; & Z_{18} &= X_6 \overline{X_{14}}; \\
 Z_7 &= \overline{X_6} X_9; & Z_{19} &= X_7 X_{10}; \\
 Z_8 &= X_6 \overline{X_9}; & Z_{20} &= \overline{X_2} X_5; \\
 Z_9 &= X_1 + X_2 = \overline{\overline{\overline{X_1 X_2}}}; & Z_{21} &= X_{10} + X_{11} = \overline{\overline{\overline{X_{10} X_{11}}}}; \\
 Z_{10} &= \overline{X_1} X_4; & Z_{22} &= \overline{X_{12}} \overline{X_{14}}; \\
 Z_{11} &= X_9 X_{13}; & Z_{23} &= X_9 \overline{X_{12}}; \\
 Z_{12} &= X_{13} + X_{14} = \overline{\overline{\overline{X_{13} X_{14}}}}; & Z_{24} &= Z_6 Z_7.
 \end{aligned}$$

В процессе модуляции в начало каждой структурной единицы формата CD – кадра - вставляется кадровая **синхрогруппа** стандартной конфигурации, содержащая два максимальных интервала между перепадами уровня T_{max} (рис.3.16). Сформированная таким образом последовательность подается на устройство оптической записи.

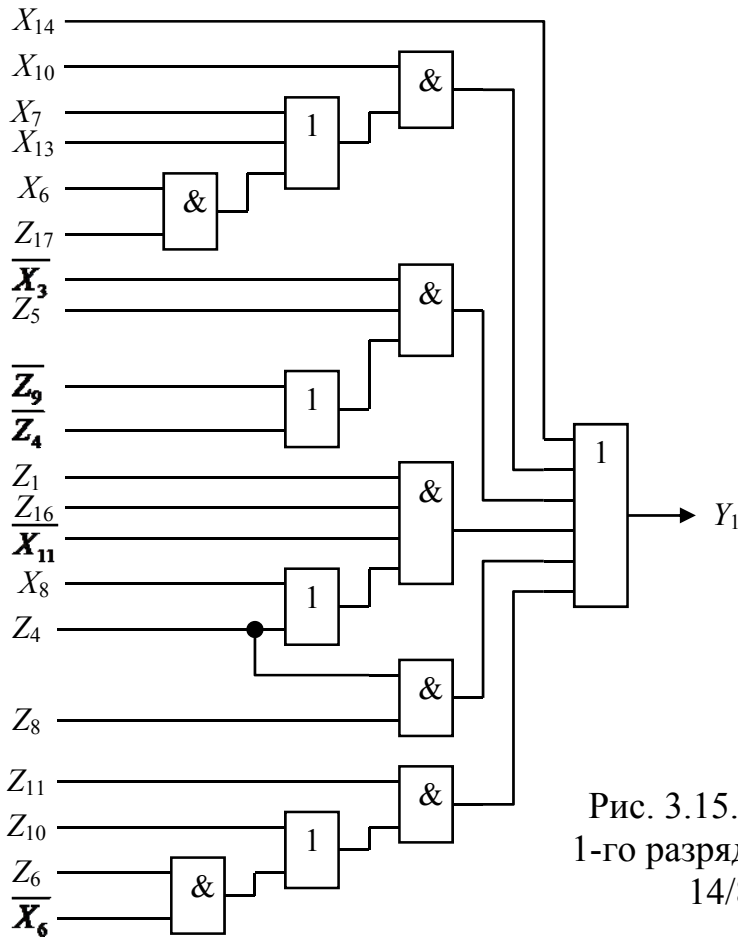


Рис. 3.15. Схема формирования 1-го разряда Y_1 в преобразователе 14/8 декодера EFM

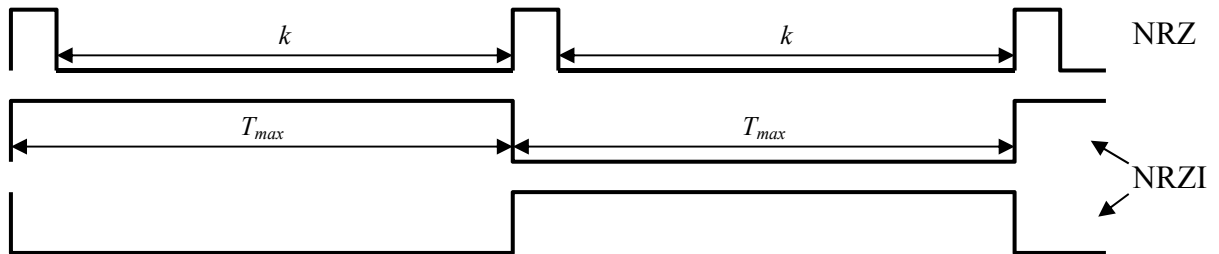


Рис. 3.16. Конфигурация кадровой синхрогруппы в формате CD

Технические характеристики кода EFM:

- минимальная длина волны записи $\lambda_{min} = 6T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 22T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 2$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 10$;
- минимальная длина пробега $T_{min} = 3$;
- максимальная длина пробега $T_{max} = 11$;
- скорость кода $R = 8/17$;
- коэффициент повышения плотности записи $K = 1,41$ (плотность записи увеличилась);
- окно детектирования $t_w = 8/17 T = \pm 4/17 T$.

3.7. Код 8/10

Канальный код 8-10, основанный на замене 8-разрядных информационных символов ($n = 8$) 10-разрядными канальными ($m = 10$), используется в системе цифровой магнитной записи с вращающимися головками формата R-DAT (Rotary Digital Audio Tape) [33-36]. Здесь, так же как и в случае кодирования по методу EFM, полученная последовательность преобразуется затем по способу NRZI (рис. 3.17).

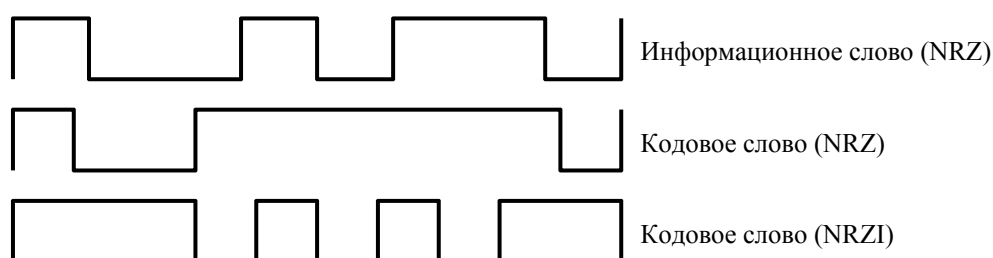


Рис. 3.17. Преобразование 8/10

Для преобразования всех 8-разрядных символов потребовалось бы $2^8 = 256$ 10-разрядных комбинаций.

Однако для обеспечения самосинхронизируемости кода нужно ограничить длину последовательностей одного уровня, иными словами – количество подряд следующих «нулей» между двумя соседними «единицами». В коде 8-10 такое ограничение равно трем ($k = 3$). Это максимальное количество «нулей», которые могут существовать между двумя соседними «единицами». Следовательно, после NRZI-преобразования длина последовательностей одного уровня будет ограничена четырьмя канальными интервалами ($T_{max} = 4$). Однако минимальное количество «нулей» между соседними «единицами» в коде 8/10 не ограничивается ($d = 0$), т.е. «единицы» могут следовать подряд ($T_{min} = 1$) – в отличие от кода EFM, где минимальное число «нулей» между «единицами» равно двум и минимальная длина пробега $T_{min} = 3$. В магнитной записи, чем выше частота записываемого сигнала (до определенных пределов), тем выше его уровень при воспроизведении. Но бесконечно повышать частоту записи тоже нельзя, так как при этом уменьшается окно детектирования t_w , т.е. допуск на смещение фронтов выделяемых импульсов (джиттер) [37].

Кроме того, поскольку соединительные разряды в коде 8-10 не используются, то необходимо так выбирать 10-разрядные комбинации, чтобы слишком длинные последовательности «нулей» не получались бы и при их соединении.

С учетом вышесказанного, из $2^{10} = 1024$ 10-разрядных комбинаций всем перечисленным требованиям удовлетворяют лишь 153. Оставшиеся 103 комбинации пришлось выбрать из тех, которые имеют ненулевое значение DSP. Однако вместо одной 10-разрядной комбинации каждому из этих 103 8-разрядных символов поставлены в соответствие две, отличающиеся друг от друга только знаком DSP. Одна из них имеет значение DSP = +2, другая –

DSP = -2. Причем, и конфигурации их отличаются только первым разрядом: у одной он – «единица», у другой – «нуль».

Таблица 3.17.

Фрагмент таблицы соответствия информационных символов и канальных кода 8-10 с учетом DSP и параметра Q

Информационный символ	Q' = -1			Q' = 1		
	Кодовый символ	DSP	Q	Кодовый символ	DSP	Q
00000000	0101010101	0	1	0101010101	0	-1
00000001	0101010111	0	-1	0101010111	0	1
00000010	0101011101	0	-1	0101011101	0	1
00000011	0101011111	0	1	0101011111	0	-1
00000100	0101001001	0	-1	0101001001	0	1
00000101	0101001011	0	1	0101001011	0	-1
00000110	0101001110	0	1	0101001110	0	-1
00000111	0101011010	0	1	0101011010	0	-1
00001000	0101110101	0	-1	0101110101	0	1
00001001	0101110111	0	1	0101110111	0	-1
00001010	0101111101	0	1	0101111101	0	-1
00001011	0101111111	0	-1	0101111111	0	1
00001100	0101101001	0	1	0101101001	0	-1
00001101	0101101011	0	-1	0101101011	0	1
00001110	0101101110	0	-1	0101101110	0	1
00001111	0101111010	0	-1	0101111010	0	1
00010000	1101010010	0	1	1101010010	0	-1
00010001	0100010010	2	-1	1100010010	-2	-1
00010010	0101010010	0	-1	0101010010	0	1
00010011	0101110010	0	1	0101110010	0	-1
00010100	1101110001	2	1	0101110001	-2	1
00010101	1101110011	2	-1	0101110011	-2	-1
00010110	1101110110	2	-1	0101110110	-2	-1
00010111	1101110010	0	-1	1101110010	0	1
00011000	0101100101	2	-1	1101100101	-2	-1
00011001	0101100111	2	1	1101100111	-2	1
00011010	0101101101	2	1	1101101101	-2	1
00011011	0101101111	2	-1	1101101111	-2	-1
00011100	0101111001	2	1	1101111001	-2	1
00011101	0101111011	2	-1	1101111011	-2	-1
00011110	0101111110	2	-1	1101111110	-2	-1
00011111	0101101010	2	-1	1101101010	-2	-1

В процессе кодирования значение используемой комбинации из такой пары выбирается исходя из значения DSP предыдущей ненулевой комбинации. Например, если для кодирования 8-разрядных символов используются 10-разрядные наборы, не имеющие постоянной составляющей, то и DSP всей последовательности будет равно 0. Если затем встретится байт, которому соответствует 10-разрядная комбинация с ненулевым DSP, то

DSV всей последовательности приобретет то значение, какое имеет выбранная 10-разрядная комбинация. Если использована комбинация с DSP = +2, то и DSV всей последовательности станет равной +2. Если в дальнейшем вновь встретится байт, которому соответствует 10-разрядная комбинация с ненулевым DSP, то в этом случае будет выбрана комбинация с DSP = -2. Таким образом, DSV всей последовательности снова станет равной нулю. Тем самым сохраняется отсутствие постоянной составляющей в потоке данных [38].

Соответствие 8-разрядных и 10-разрядных символов иллюстрируется табл. 3.17. Такое преобразование можно осуществить с помощью ПЗУ (постоянного запоминающего устройства) объемом $256 \times 10 = 2560$ бит. При этом 8-разрядный информационный символ будет служить адресом 10 запоминающих элементов с битами 10-разрядного канального символа. Однако это не единственный способ реализации. Можно использовать логическую матрицу, состоящую из элементов И, ИЛИ, НЕ и ИСКЛЮЧАЮЩЕЕ ИЛИ, соединенных по определенной схеме.

С помощью компьютерного моделирования можно вывести формулы, отражающие зависимость каждого разряда 10-разрядного канального символа ($Y_1 - Y_{10}$) от значений разрядов 8-разрядного информационного символа ($X_1 - X_8$). Такие формулы называются **логическими функциями**. В них значения $X_1 - X_8$ являются входными переменными, а значения разрядов канального символа $Y_1 - Y_{10}$ – функциями от них.

$$Y_i = F(X_1, X_2, \dots, X_8)$$

где $i = 1, 2, \dots, 10$

В результате моделирования получается следующий набор логических функций [39]:

$$\begin{aligned} Y_1 &= X_1 + X_3 Z_2 + Z_1 [\overline{X_3} \oplus \overline{X_6} (X_7 + X_8)] \\ Y_2 &= X_1 (X_2 + X_4 \overline{X_5}) + \overline{X_1} (\overline{X_2} + \overline{X_3}); \\ Y_3 &= \overline{X_1} X_3 + X_1 (\overline{X_4} + X_5) + X_2 X_4 X_5; \\ Y_4 &= X_1 (X_3 + X_2 X_4 \overline{X_5}) + X_3 X_4 X_5 + X_3 Z_2 + (\overline{X_1} \overline{X_2} \oplus \overline{X_6} \overline{X_7} X_8 Z_1); \\ Y_5 &= (X_1 X_2 + \overline{X_4}) \overline{X_5} + \overline{X_1} \overline{X_2} X_3 X_4 X_5 + Z_1 \overline{X_6} (\overline{X_7} + \overline{X_8}); \\ Y_6 &= \overline{X_1} \overline{X_5} [X_3 + (X_2 \oplus X_4)] + [(\overline{X_4} + X_3 \overline{X_5}) \oplus X_6 (\overline{X_7} + \overline{X_8})]; \\ Y_7 &= \overline{X_6} \overline{X_7} + Z_1 + (X_2 + X_3) Z_2; \\ Y_8 &= X_6 X_7 \overline{X_8} + \overline{X_6} Z_1; \\ Y_9 &= X_8 + X_6 X_7 + \overline{X_6} Z_1; \\ Y_{10} &= X_6 \overline{X_7} + \overline{X_6} Z_1. \\ Z_1 &= \overline{X_1} (\overline{X_2} + X_3) X_4 \overline{X_5}; \\ Z_2 &= \overline{X_1} X_4 \overline{X_5} X_6 (\overline{X_7} + \overline{X_8}). \end{aligned}$$

Функции Z_1 и Z_2 являются вспомогательными, упрощающими запись и реализацию основных функций $Y_1 - Y_{10}$. На рис. 3.18 показан пример реализации функции $Y_2 = F(X_1, X_2, \dots, X_8)$ с помощью элементов И, ИЛИ и

НЕ. Подобным же образом реализуются и другие функции $Y_i = F(X_1, X_2, \dots, X_8)$.

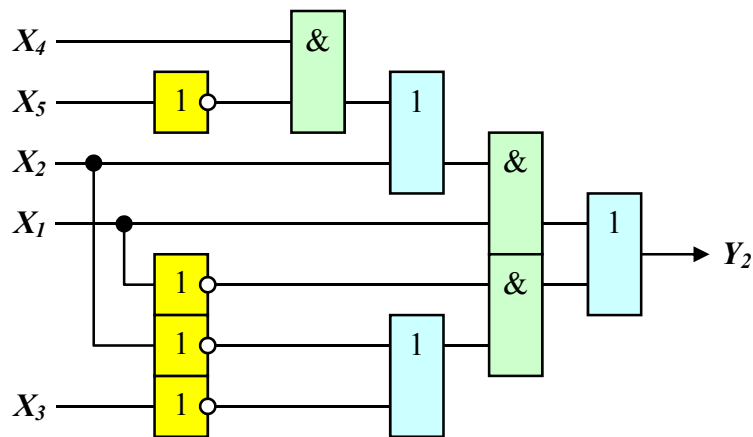


Рис. 3.18. Пример реализации логической функции $Y_2 = F(X_1, X_2, \dots, X_8)$

С помощью полученных логических функций формируются только комбинации с $DSP = 0$ и $DSP = +2$. Комбинации с $DSP = -2$ легко получаются из наборов с $DSP = +2$ простым инвертированием первого разряда Y_1 :

$$Y_i(DSP = -2) = \overline{Y_i}(DSP = +2)$$

Для того, чтобы принять решение о том, какой вариант набора из пары с разными DSP использовать, нужно знать значение DSP предыдущего набора. Определяется оно весьма интересным способом. Так как «единица» в NRZI-представлении изменяет значение уровня сигнала на противоположное, очевидно, что если в паре канальных битов второй бит – «единица» (независимо от значения первого бита), то такая пара будет иметь $DSP = 0$. Если проверить на четность пять четных разрядов 10-разрядного канального символа, и результат окажется равным «единице», то весь символ может иметь значение DSP, равное или 0, или ± 14 , или ± 8 . Если результат окажется равным нулю, то DSP может принимать значения ± 2 , ± 6 или ± 10 . Однако символы, формируемые логической матрицей, могут иметь значение DSP только 0 или ± 2 , и никакое другое. Поэтому результат вышеописанной проверки однозначно характеризует DSP рассматриваемого символа. Схема для осуществления такой проверки показана на рис. 3.19.

Работает она следующим образом. Элементы ИСКЛЮЧАЮЩЕЕ ИЛИ D1-D4 осуществляют проверку на четность четных разрядов (Y_2, Y_4, Y_6, Y_8 и Y_{10}) 10-разрядного канального символа, формируемого логической матрицей.

Элемент D7 служит управляемым инвертором, с помощью которого можно изменять полярность разряда Y_1 , когда на выходе логической матрицы появляются канальные символы с $DSP = +2$. При этом, если на выходе элемент D6 сформировать «единицу», то элемент D7 будет инвертировать значение разряда Y_1 , а если сформировать «нуль», то значение разряда Y_1 не изменится.

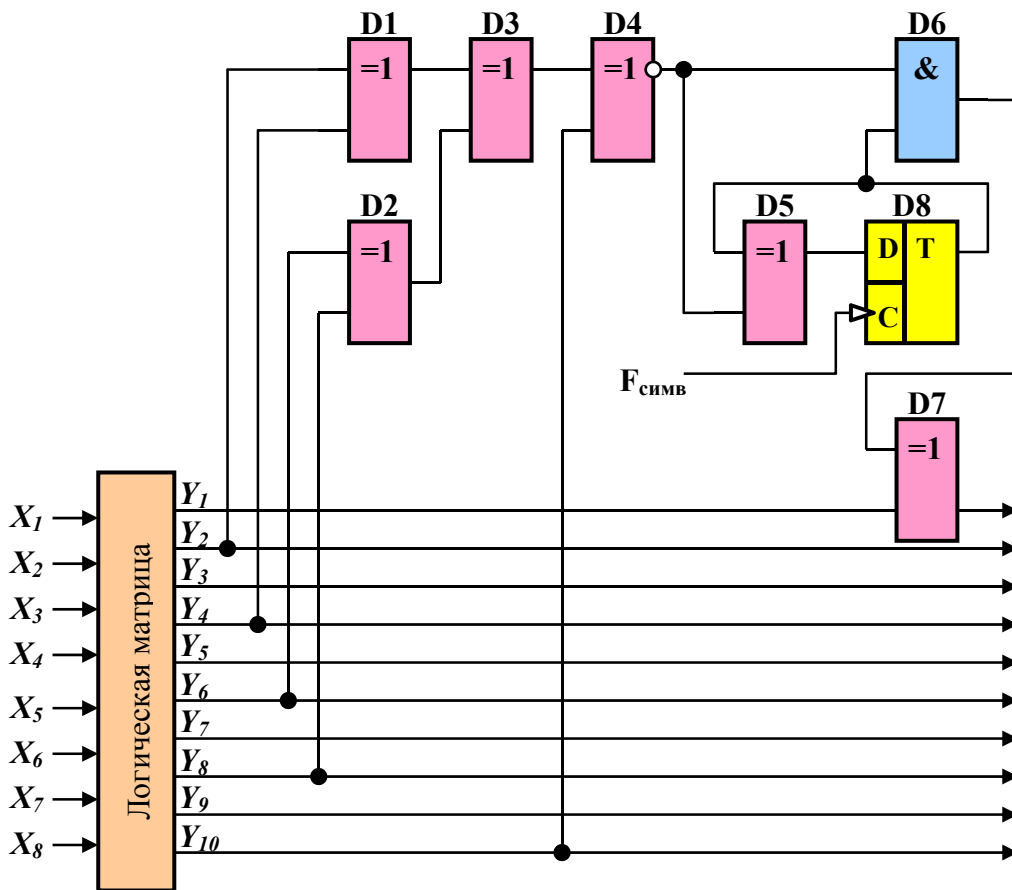


Рис. 3.19. Схема контроля DSP канального символа

Чтобы на выходе элемента D6 сформировалась «единица», нужно, чтобы «единицы» были на обоих его входах. На верхнем (по схеме) входе «единица» будет тогда, когда сумма четных разрядов канального символа будет равна «нулю» (что и укажет на то, что его DSP = +2). Поскольку элемент D4 инвертирующий, то он при этом вырабатывает необходимую «единицу». «Единица» на втором (нижнем по схеме) входе элемента D6 будет в том случае, если таково состояние триггера D8. Если это так, то на выходе элемента D6 будет «единица», и элемент D7 проинвертирует разряд Y_1 , изменив DSP канального символа с +2 на -2. У следующего же канального символа с DSV = +2, для поддержания общей суммы DSV последовательности равной нулю, разряд Y_1 инвертироваться не должен. Поэтому триггер D8 должен изменить свое состояние на противоположное. Поскольку его следующее состояние зависит от потенциала на входе D в момент появления на входе C переднего фронта тактового импульса частоты $F_{\text{симв}}$, то необходимый для этого «нуль» формируется элементом ИСКЛЮЧАЮЩЕЕ ИЛИ D5. При наличии на его входах двух «единиц» на выходе будет «нуль» и триггер изменит своё состояние.

Теперь, если вновь появится канальный символ с DSP = +2, то, хотя на выходе элемента D4 будет «единица», инвертирования разряда Y_1 не произойдет. «Нуль» с выхода триггера D8 закроет элемент D6. После этого сформированная элементом D5 «единица» на входе D триггера снова

заставит его изменить свое состояние, подготавливая к очередной операции инвертирования.

Когда сумма по модулю 2 разрядов Y_2, Y_4, Y_6, Y_8 и Y_{10} равна «единице», DSP канального символа равно «нулю». Это означает, что никаких действий с разрядом Y_1 производить не следует. В этом случае элемент D4 инвертирует полученную «единицу» и «нулем» на своем выходе закроет элемент D6. На выходе элемента D6, вне зависимости от состояния триггера D8, также будет «нуль», и элемент D7 пропустит разряд Y_1 без изменения. При этом, поскольку на выходе элемента D4 логический «нуль», то состояние D-входа триггера, определяемое элементом D5, будет таким же, как и состояние его выхода, и передним фронтом импульса частоты $F_{симв}$ это состояние подтвердится.

Таким образом, при появлении канальных символов с $DSP = 0$, никаких изменений в их конфигурации производиться не будет. При появлении же символов с $DSP = +2$, будет происходить поочередное изменение знака DSP, с тем, чтобы результирующее значение оставалось равным «нулю».

В табл. 3.17 рядом со значением каждого 10-разрядного символа указано еще и значение уровня сигнала Q в его конце после осуществления NRZI-преобразования. Значение Q зависит от уровня сигнала Q' перед началом данного символа. Если $Q' = +1$, то уровень – высокий, если $Q' = -1$, то уровень – низкий. Левая колонка 10-разрядных символов соответствует низкому уровню сигнала в конце предыдущего символа ($Q' = -1$), правая – высокому уровню ($Q' = +1$).

Каждый очередной 10-разрядный канальный символ выбирается из таблицы исходя из значения 8-разрядного информационного символа и значения параметра Q' (рис. 3.20).

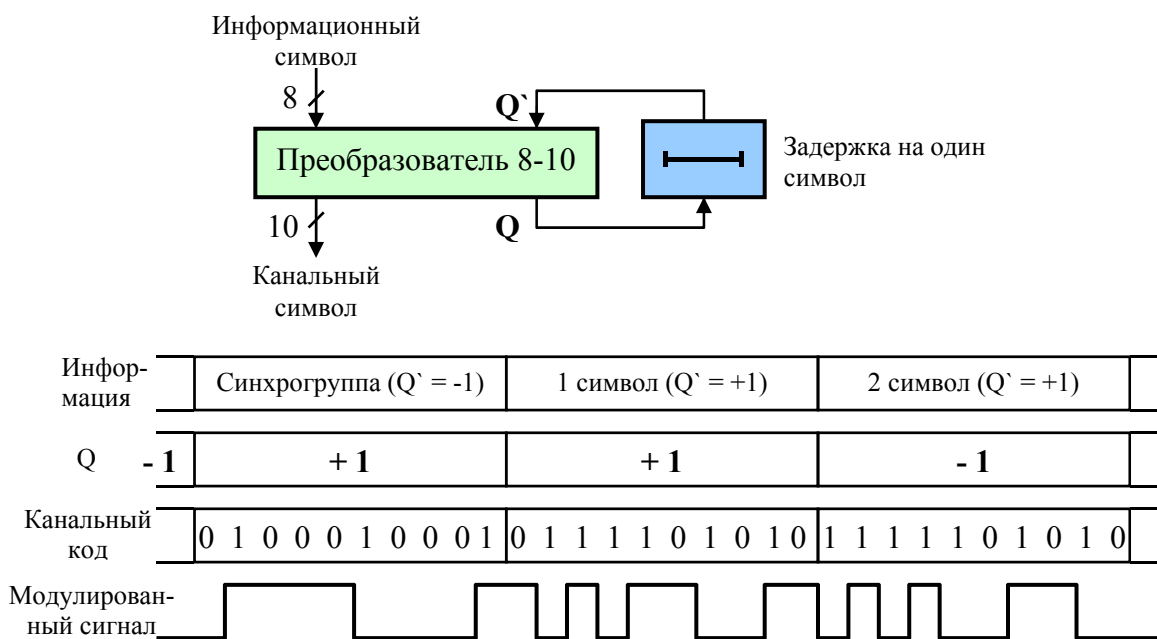


Рис. 3.20. Процедура кодирования информации кодом 8-10 с учетом параметра Q

Поскольку у кода 8/10 $T_{max} = 4$ а $T_{min} = 1$, то максимальная длина волны записи $\lambda_{max} = 8T_{к}$, а минимальная длина волны записи $\lambda_{min} = 2T_{к}$. Канальная тактовая частота в системе R-DAT составляет $f_{к} = 9,408$ МГц. Следовательно, основная энергия кодированного сигнала сосредоточена в полосе частот между $f_{к}/2 = 4,704$ МГц и $f_{к}/8 = 1,176$ МГц (рис. 3.21). Окно детектирования кода 8/10 составляет $t_w = 4/5 T = \pm 2/5 T$, а коэффициент повышения плотности записи $K = (d+1) \cdot n/m = 4/5$.

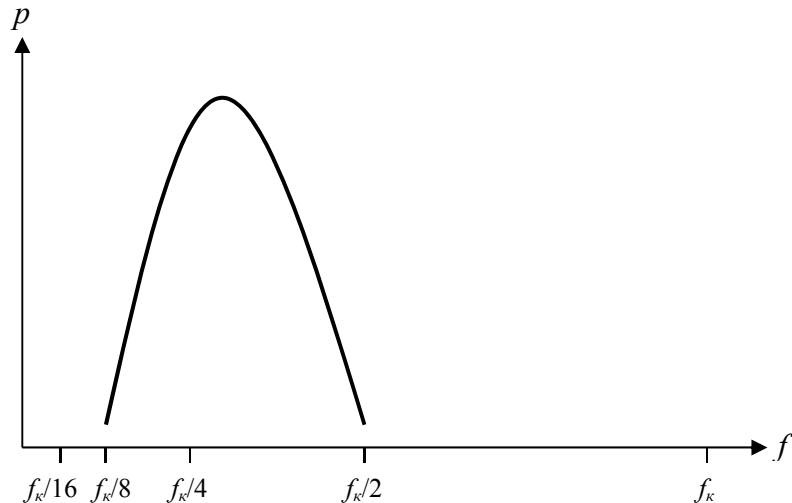


Рис. 3.21. Спектр мощности кода 8/10

Для цикловой синхронизации в системе R-DAT используется блочная синхрогруппа из 10 канальных бит, которая включает в себя два подряд следующих максимальных интервала между соседними перепадами уровня $4T_{к}$ (рис. 3.22).

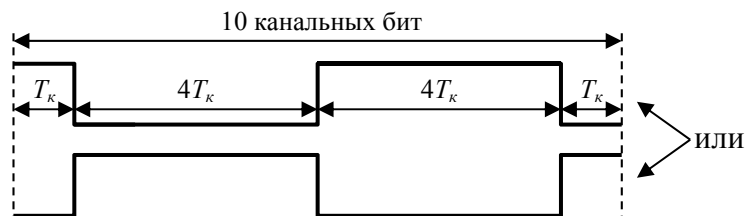


Рис. 3.22. Конфигурация блочной синхрогруппы в формате R-DAT

Таким образом, код 8/10 обладает следующими техническими характеристиками:

- минимальная длина волны записи $\lambda_{min} = 2T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 8T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 0$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 3$;
- минимальная длина пробега $T_{min} = 1$;
- максимальная длина пробега $T_{max} = 4$;
- скорость кода $R = 4/5$;

- коэффициент повышения плотности записи $K = 0,8$ (плотность записи уменьшилась);
- окно детектирования $t_w = 4/5 T = \pm 2/5 T$.

3.8. Код EFM Plus

Канальный код EFM Plus (8-16), как следует из его обозначения, является усовершенствованным вариантом кода EFM (8-14), используемого в формате CD, и был разработан в середине 90-х годов для записи информации на новый оптический носитель – мультимедийный компакт-диск – MMCD (MultiMedia Compact Disc) [40, 41]. Впоследствии формат MMCD стал одним из прототипов широко распространенного сегодня формата DVD.

Основные параметры кода EFM Plus те же, что и у его предшественника – кода EFM: минимальное расстояние между двумя соседними «единицами» $d = 2$, максимальное $k = 10$, а следовательно $T_{min} = 3$ и $T_{max} = 11$. Однако алгоритмы кодирования и декодирования довольно сильно отличаются. Если в коде EFM каждому входному 8-разрядному символу однозначно соответствовал один выходной (кодовый), то в коде EFM Plus такого соответствия нет. Одному и тому же входному символу могут соответствовать по меньшей мере четыре выходных, выбираемых из четырех столбцов таблицы преобразования по определенному закону. Кроме того, для первых 88 символов (от 0 до 87) существует еще альтернативная таблица, из которой могут быть выбраны выходные символы. Эта таблица используется для того, чтобы уменьшить уровень низкочастотных составляющих (значение DSV) в результирующей выходной последовательности. Выходные (кодовые) слова в коде EFM Plus – 16-разрядные, а не 14-разрядные, как в коде EFM. Соединительные разряды здесь не используются. Поэтому другое название кода EFM Plus – код 8-16. В некотором смысле аббревиатура EFM (Eight to Fourteen Modulation) здесь совершенно не соответствует сути преобразования (правильнее было бы назвать этот код ESM – Eight to Sixteen Modulation), и является лишь указанием на своего предшественника.

Стратегия кодирования кодом EFM Plus состоит в следующем. Общее число 16-разрядных слов, удовлетворяющих условиям на ограничение расстояния между двумя соседними «единицами» $d = 2$ и $k = 10$, и, следовательно, пригодных для использования в качестве кодовых слов кода EFM Plus, равно 351. Семь из них исключены из этого числа, поскольку они могут стать причиной образования ложных синхрогрупп (о синхрокомбинациях, используемых в потоке данных, модулированных кодом EFM Plus, будет рассказано несколько позже). Остается, таким образом, $351-7=344$ слов. Для кодирования 256 исходных 8-разрядных информационных символов, в принципе, достаточно такого же количества 16-разрядных кодовых символов. Поэтому оставшиеся $344-256=88$ символов можно использовать в качестве альтернативных основному множеству

символов для минимизации низкочастотных составляющих в спектре формируемой последовательности.

Для того чтобы в процессе кодирования при каскадном соединении двух соседних 16-разрядных символов не произошло нарушения заданных ограничений на минимальное и максимальное расстояние между двумя соседними «единицами» ($d = 2, k = 10$), каждое очередное кодовое слово выбирается из четырех возможных вариантов, т.е. однозначного соответствия между входным 8-разрядным символом и 16-разрядным кодовым, как уже говорилось выше, в методе EFM Plus нет (в отличие от его предшественника – кода EFM). Для того чтобы обеспечить такую свободу выбора, из 256 кодовых слов основного множества образованы четыре группы, каждая из которых сама содержит 256 16-разрядных кодовых символов. Очевидно, что для этого каждое 16-разрядное кодовое слово должно входить в такие группы неоднократно – в среднем четыре раза. И действительно, один и тот же кодовый символ может, как входить в разные группы, соответствующие одному и тому же входному 8-разрядному символу, так и в одну и ту же группу, но соответствовать при этом разным входным символам. Группы символов формируются следующим образом.

В 1-ю группу входят кодовые слова, которые могут следовать за словами, заканчивающимися «единицей» или одним «нулем». То есть в начале слов этой группы может быть не менее 2 и не более 9 «нулей». Состояние кодера, когда очередное кодовое слово выбирается из 1-й группы, называется состоянием 1.

Во 2-ю группу входят кодовые слова, которые могут следовать за словами, у которых в конце от 2 до 5 «нулей». Кроме того, у всех слов 2-й группы 1-й и 13-й биты равны «нулю». Состояние кодера, когда очередное кодовое слово выбирается из 2-й группы, называется состоянием 2.

Во 3-ю группу входят кодовые слова, которые, также как и слова 2-й группы, могут следовать за словами, у которых в конце от 2 до 5 «нулей». Однако, у всех слов 3-й группы, в отличие от слов 2-й группы, либо 1-й, либо 13-й биты, не равны «нулю». Состояние кодера в этом случае соответствует состоянию 3.

В 4-ю группу входят кодовые слова, которые могут следовать за словами, у которых в конце от 6 до 9 «нулей». То есть в начале слов этой группы может быть не более 1-го «нуля» (или слово должно начинаться с «единицы»). Состояние кодера в этом случае соответствует состоянию 4.

Заметим, что 1-я и 4-я группы не могут иметь общих слов, хотя слова как 1-й, так и 4-й группы могут также входить во 2-ю и 3-ю группы.

Описание работы кодера будет более понятным, если воспользоваться следующими обозначениями:

x_t – кодовое слово, сформированное кодером и переданное на выход в момент времени t ;

b_t – входной информационный символ (слово источника), поступивший на вход кодера в момент времени t , в результате чего кодер сформировал на своем выходе кодовое слово x_t ;

s_t – состояние кодера в момент времени t , когда на его вход поступил информационный символ b_t и когда он сформировал на своем выходе отклик x_t ;

$h(b_t, s_t)$ – функция, которая описывает состояние выхода кодера и которая указывает на то, что это состояние зависит от значения входного информационного символа b_t и от состояния кодера s_t в момент времени t ;

$g(b_t, s_t)$ – функция следующего состояния кодера, которая, так же как и функция $h(b_t, s_t)$, зависит от значения входного информационного символа b_t и от состояния кодера s_t в момент времени t .

Таким образом, можно записать

$$\begin{aligned}x_t &= h(b_t, s_t); \\s_{t+1} &= g(b_t, s_t),\end{aligned}$$

где s_{t+1} – состояние кодера в момент времени $t+1$, т.е. после сразу после завершения формирования и передачи на выход очередного кодового слова.

Для кодирования входного потока данных, представленного в виде последовательности 8-разрядных информационных символов (b_t), используется специальная таблица, фрагмент которой показан ниже (табл. 3.18). Эта таблица содержит четыре столбца, каждый из которых включает в себя 256 16-разрядных кодовых слов, обозначенных функцией состояния выхода h , и указанных рядом значений следующего состояния кодера g , в которое он должен перейти после формирования и передачу на выход данного кодового символа. Входные информационные символы обозначены их порядковыми номерами i ($i = 0 \dots 255$). Физически процедура кодирования реализуется с помощью памяти с произвольным доступом, объемом $256 \times 4 = 1024$ 18-разрядных слов. В каждую 18-разрядную ячейку записывается значение 16-разрядного кодового символа (b_t) из соответствующего столбца таблицы и 2-разрядное значения одного из четырех состояний кодера (s_t), в которое кодер должен перейти сразу же после того, как очередной символ передан на его выход (т.е. в момент времени $t + 1$).

В качестве примера, иллюстрирующего работу кодера, рассмотрим процесс кодирования информационных символов с номерами 8, 3 и 4. Пусть исходным состоянием кодера будет состояние 1. Откликом на входное воздействие «8» (информационный символ с номером $i = 8$) при наличии состояния 1 кодера будет значение функции выхода $h(8, 1) = 0010000010010000$. Следующим состоянием кодера должно стать состояние $g(8, 1) = 3$. Поэтому откликом на следующее входное воздействие «3» (информационный символ с номером $i = 3$) при наличии состояния кодера 3 будет значение функции выхода $h(3, 3) = 0010000001001000$. При этом следующим состоянием кодера должно стать состояние $g(3, 3) = 2$. В состоянии «2» при наличии входного воздействия «4» (информационный символ с номером $i = 4$) кодер сформирует отклик $h(4, 2) = 0010000010010000$, а также определит свое следующее состояние $g(4, 2) = 2$. Отметим, что отклики $h(8, 1)$ и $h(4, 2)$ совершенно одинаковы и равны 0010000010010000 . Также одинаковы и отклики $h(7, 1)$ и $h(3, 1)$ (0010000001001000), т.е. отклики одной и той же группы, но на разные

входные воздействия. Таким образом, налицо некоторая двусмысленность, которую предстоит разрешать при декодировании. В связи с этим здесь следует обратить на то, что после кодирования входного слова «7» кодер переходит в состояние «3», а после кодирования слова «3» - в состояние «2». Очевидно, этот факт и будет служить указателем декодеру на тот символ, который был закодирован.

Таблица 3.18

Фрагмент таблицы кодирования кодом EFM Plus

<i>i</i>	<i>h(i,1), g(i,1)</i>	<i>h(i,2), g(i,2)</i>	<i>h(i,3), g(i,3)</i>	<i>h(i,4), g(i,4)</i>
0	0010000000001001, 1	0100000100100000, 2	0010000000001001, 1	0100000100100000, 2
1	00100000000010010, 1	00100000000010010, 1	1000000100100000, 3	1000000100100000, 3
2	0010000100100000, 2	0010000100100000, 2	1000000000010010, 1	1000000000010010, 1
3	0010000001001000, 2	0100010010000000, 4	0010000001001000, 2	0100010010000000, 4
4	0010000010010000, 2	0010000010010000, 2	1000000100100000, 2	1000000100100000, 2
5	0010000000100100, 2	0010000000100100, 2	1001001000000000, 4	1001001000000000, 4
6	0010000000100100, 3	0010000000100100, 3	1000100100000000, 4	1000100100000000, 4
7	0010000001001000, 3	0100000000010010, 1	0010000001001000, 3	0100000000010010, 1
8	0010000010010000, 3	0010000010010000, 3	1000010010000000, 4	1000010010000000, 4

В общем виде операция декодирования может быть записана следующим образом:

$$b_t = h^{-1}(x_t, s_t),$$

из чего следует, что для правильного декодирования на приемном конце необходимо знать состояние кодера в момент формирования данного кодового слова. В упомянутом выше случае, когда слова «7» и «3» имеют одно и то же кодовое отображение, это делается путем анализа предыдущего кодового символа, который во время кодирования следовал за рассматриваемым. Если предыдущий символ принадлежал ко 2-й группе кодовых слов, то рассматриваемый символ «3», если к 3-й – то это «7». Как говорилось выше, у кодовых слов 2-й группы 1-й и 13-й биты всегда «нули», а у слов 3-й группы – один из этих бит должен быть равным «единице», что легко проверить и, таким образом, декодировать данный символ единственным образом. Другими словами, для декодирования требуется выполнить операцию:

$$b_t = h^{-1}(x_t, x_{t+1,1}, x_{t+1,13}).$$

Как уже говорилось, для кодирования с помощью представленной выше таблицы используются только 256 из 344 16-разрядных кодовых слов, удовлетворяющих условиям ограничения $d = 2, k = 10$. Оставшиеся 88 слов используются для минимизации энергии сигнала на низких частотах. Для этого из «лишних» 88 слов образована таблица замещения, слова из которой используются в качестве альтернативного канального представления слов источника 0...88. Таким образом, полное описание кодера включает в себя две таблицы – главную (которая рассматривалась выше) и похожую на нее вспомогательную (таблицу замещения) из 88 слов (рис. 3.23).

Подавление низкочастотных компонент, т.е. контроль постоянной составляющей в коде EFM Plus осуществляется теми же методами, что и в обычном коде EFM, а именно – путем контроля текущего значения числовой суммы DSV (Digital Sum Value). Таблицы (главная и таблица замещения) составлены таким образом, что вклад первых 88 символов главной таблицы в текущее значение DSV является преимущественно отрицательным, а вклад соответствующих символов из таблицы замещения – преимущественно положительный. Для слов источника 0...87 кодер выбирает такой вариант представления из главной таблицы и таблицы замещения, который позволяет минимизировать абсолютную величину DSV.

Следует отметить, что для принятия решения здесь необходимо выполнить не более двух вычислений и сравнений для каждого кодового слова, в то время как в обычном коде EFM для этого должны быть выполнены не более четырех вычислений и сравнений (существует не более четырех возможных вариантов формирования соединительных разрядов). То обстоятельство, что только к 88 из 256 слов источника может быть применена технология контроля DSV, делает систему чувствительной к «наихудшему случаю» на выходе. Однако это не так опасно, как может показаться, поскольку аналогичные трудности могут возникать и при кодировании кодом EFM, который, тем не менее, работает вполне удовлетворительно в течение более четверти века. Компьютерное моделирование показало, что уровень низкочастотных компонент у кода EFM Plus примерно на 3 дБ выше, чем у кода EFM.

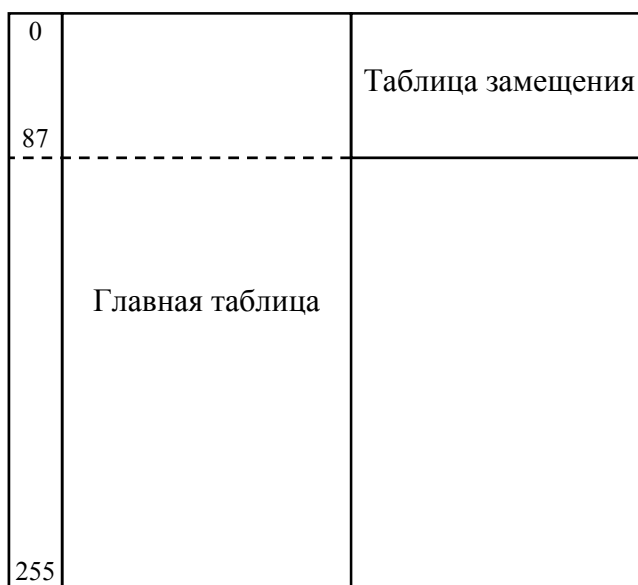


Рис. 3.23. Структура таблиц кодера EFM Plus

В формате DVD для обозначения начала очередного кадра используются 32-разрядные синхрогруппы трех типов:

- S_1 : x001000001000000000010000000001;
- S_2 : x001001001000000000010000000001;
- S_3 : x000010001000000000010000000001.

Значение символа, обозначенного символом «х», зависит от того, в каком из четырех состояний находится кодер непосредственно перед началом синхрогруппы. Если он находится в состоянии 1 или 2, то «х» принимает значение «0», если в состоянии 3 или 4, то $x = 1$. Это правило может быть отменено в случае угрозы формирования ложной синхрогруппы из сочетания бит двух предыдущих кодовых символов и первых бит синхрогруппы. Как и в формате CD, каждая из синхрогрупп характеризуется наличием двух подряд следующих интервалов максимальной длины $k = 10$.

После формирования синхрогруппы кодер всегда устанавливается в состояние 1. Поскольку кодовые слова первой группы начинаются не более чем с девяти «нулей», то тем самым устраняется угроза формирования в месте соединения ложной синхрогруппы.

Поскольку у кода EFM Plus $T_{min} = 3$ и $T_{max} = 11$, то $\lambda_{min} = 6T_{кан}$ и $\lambda_{max} = 8T_{кан}$. Коэффициент повышения плотности записи $K = (d+1) \cdot n/m = (2+1) \cdot 8/16 = 3/2 = 1,5$, т.е. плотность записи здесь несколько выше, чем у простого кода EFM. Окно детектирования $t_w = 8/16 T = T/2 = \pm T/4$ здесь также шире, чем у кода EFM ($t_w = 8/17 T = \pm 4/17 T$).

Таким образом, технические характеристики кода EFM Plus следующие:

- минимальная длина волны записи $\lambda_{min} = 6T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 8T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 2$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 10$;
- минимальная длина пробега $T_{min} = 3$;
- максимальная длина пробега $T_{max} = 11$;
- скорость кода $R = 1/2$;
- коэффициент повышения плотности записи $K = 1,5$ (плотность записи увеличилась);
- окно детектирования $t_w = T/2 = \pm T/4$.

3.9. Код 17 PP

Данный код используется в новейшей системе оптической записи с высокой плотностью BluRay [42]. Обозначение 17PP (Parity preserve Prohibit) должно указывать на то, что код имеет не менее одного «нуля» между двумя соседними «единицами» ($d = 1$) и не более семи ($k = 7$). Кодированная последовательность затем модулируется по способу NRZI, следовательно, минимальная длина пробега $T_{min} = 2$, максимальная – $T_{max} = 8$. Минимальная длина волны записи $\lambda_{min} = 4T_{к}$, максимальная $\lambda_{max} = 16T_{к}$. Коэффициент повышения плотности записи $K = (d+1) \cdot n/m = (1+1) \cdot 2/3 = 4/3 = 1,33$. Окно детектирования $t_w = 2/3 T = \pm 1/3 T$.

По сути, код 17 PP является кодом 2/3 ($n = 2$, $m = 3$), однако здесь одновременно могут кодироваться (подвергаться преобразованию) не только

пары исходных информационных битов, но и четверки, шестерки и даже восьмерки битов ($i = 1, 2, 3$ и 4). Для минимизации постоянной составляющей используется контроль текущего значения DSV, и все замены производятся исходя из того, чтобы значение DSV было возможно ближе к нулю. Основной алгоритм замены показан в табл. 3.19.

Таблица 3.19

Алгоритм кодирования кодом 17 PP

Информационные слова	Кодовые слова
$i = 1$	
11	*0*
10	001
01	010
$i = 2$	
00 11	010 100
00 10	010 000
00 01	000 100
$i = 3$	
00 00 11	000 100 100
00 00 10	000 100 000
00 00 01	010 100 100
00 00 00	010 100 000
$i = 4$	
00 00 10 00	000 100 100 100
00 00 00 00	010 100 100 100

Обозначение *0* говорит о том, что если предыдущее кодовое слово оканчивается на «1» (xx1), то *0* = 000, а если на «0» (xx0), то *0* = 101.

Синхрогруппа, используемая в формате BluRay, содержит два подряд следующих интервала между соседними перепадами уровня, равных $9T_k$, т.е. превышающих максимально возможный в потоке кодированных данных ($8T_k$) и имеет конфигурацию, показанную на рис. 3.24. Наличие таких нестандартных интервалов облегчает опознание синхрокомбинации блоком цикловой синхронизации при наличии искажений.

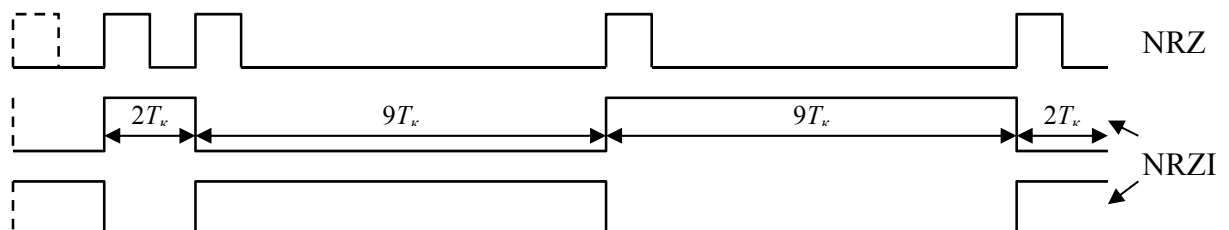


Рис. 3.24. Конфигурация синхрогруппы в формате BluRay

Символ, обозначенный как «х», имеет значение «1», если предыдущее кодовое слово заканчивается «нулем», и значение «0», если предыдущее кодовое слово заканчивается «единицей».

Код 17 PP имеет следующие технические характеристики:

- минимальная длина волны записи $\lambda_{min} = 4T_{кан}$;
- максимальная длина волны записи $\lambda_{max} = 16T_{кан}$;
- минимальное количество «нулей» между двумя соседними «единицами» $d = 1$;
- максимальное количество «нулей» между двумя соседними «единицами» $k = 7$;
- минимальная длина пробега $T_{min} = 2$;
- максимальная длина пробега $T_{max} = 8$;
- скорость кода $R = 2/3$;
- коэффициент повышения плотности записи $K = 1,33$ (плотность записи увеличилась);
- окно детектирования $t_w = 2/3 T = \pm 1/3 T$.

ЗАКЛЮЧЕНИЕ

Методов канальной модуляции и канального кодирования существует достаточно много. В данной публикации были рассмотрены только некоторые из них – прежде всего, конечно, те, которые сегодня используются в широко распространенных форматах записи цифровой информации на магнитный, оптический и магнитооптический носители: CD, DVD, BD, MD, DAT. Некоторые методы модуляции, которые уже давно не используются (RZ, NRZ-E, 4/5 NRZI и пр.), тем не менее, представляют интерес как своего рода исторические вехи, на примере которых можно проследить весь путь развития техники и технологии записи цифровой информации на носители различного типа. Некоторые канальные коды интересны используемыми в них оригинальными приемами формирования последовательностей с нужными свойствами (коды 2/3), которые могут быть взяты на вооружение и в будущем – при разработке каких-то новых кодов для записи информации на новые носители. Технологии записи информации постоянно развиваются и совершенствуются, появляются новые, ранее не существовавшие, или не существовавшие как массовые, носители. Процесс поиска оптимальных методов записи информации на такие носители для разработчика может оказаться проще, если он хорошо знаком с тем, что было сделано раньше. И если не само техническое решение, то хотя бы какой-то элемент его, могут подсказать ему тот подход, который позволит отыскать оптимальное решение задачи.

ЛИТЕРАТУРА

1. Franaszek P.A. Sequence-state Methods for Run-Length-Limited Coding. – IBM J. Res. & Dev., 14, No. 4, July 1970, pp. 376-383.
2. Рыжков В.А., Сергеев Н.П., Раков Б.М. Внешние ЗУ на магнитном носителе. М., «Энергия», 1978. 224 с.
3. Itoh Y., Nishimura K. Modulation-Demodulation Methods for Digital Magnetic Recording. -Nikkey Electron., 1978, vol. 12, No. 11, pp. 1-164.
4. Severt R.H. Encoding Schemes Support High Density Digital Data Recording. – Computer Design, vol. 19, May 1980, No. 5, pp. 181-190.
5. Knoll A.L. Spectrum Analysis of Digital Magnetic recording Waveforms. – IEEE Transaction on Electronic Computers, December 1967, vol. EC-16, No. 6, pp. 723-743.
6. Gabor A. Adaptive Coding for Self-clocking Recording. – IEEE Transaction on Magnetics, December 1967, vol. 4, pp.866-868.
7. Miller A., U.S. Patent 3 108 261, October 22nd 1963.
8. McCracken. A High Performance Digital Audio Recorder, presented at the 58th Convention of the Audio Engineering Society, J. Audio Eng. Soc. (Abstracts), vol. 25, Dec. 1977, p. 1072, preprint No. 1268.
9. Tanaka K., Yamaguchi T., Sugiyama Y. Improved Two-Channel PCM Tape Recorder for Professional Use, presented at the 64th Convention of the Audio Engineering Society, J. Audio Eng. Soc. (Abstracts), vol. 27, Dec. 1979, p. 1026, preprint No. 1533.
10. Matsushima H., Kanai K., Miura T., Kogure T. A New Digital Audio Recorder for Professional Applications, presented at the 62nd Convention of the Audio Engineering Society, J. Audio Eng. Soc. (Abstracts), vol. 27, May 1979, p. 428, preprint No. 1447.
11. Miller J.W., U.S. Patent 4 027 335, May 31st 1977.
12. Mallinson J.C., Miller J.W. Optimum Codes for Digital Magnetic Recording. – Radio and Electron. Eng., April 1977, vol. 47, pp. 172-176.
13. Engberg E.W. A Digital Audio Recorder Format for Professional Applications, presented at the 61nd Convention of the Audio Engineering Society, J. Audio Eng. Soc. (Abstracts), vol. 26, Dec. 1978, p. 990, preprint No. 1413.
14. Jacoby G.V. A New Look-Ahead Code for Increased Data Density. - IEEE Transaction on Magnetics, vol. MAG-13, Sept. 1977, pp. 1201-1204.
15. ANSI X3.56-1977, Recorded Magnetic Tape Cartridge for Information Interchange (4-track, 0.250 Inch. 1600 BPI, Phase Encoded).
16. Патент ФРГ №3240219, МКИ H03K 13/00, H04L 25/48, G11B 5/09, приор. 29.10.81 г.
17. Cohn M., Jacoby G.V., Bates A. Data Encoding Method and System Employing Two-Thirds Code Rate With Full Word Look-Ahead. – U.S. Patent 4 337 458, issued June 29, 1982.
18. Jacoby G.V., Kostil R. Binary Two-Thirds Rate Code With Full Word Look-Ahead. - IEEE Transaction on Magnetics, Sept. 1984, vol. MAG-20, No. 5, pp. 709-714.

19. Европейский патент №00592246, МКИ G11B 5/09 (H04L 25/49), приор. 05.09.80 Jr., публ. 08.09.82.
20. Standard on Compact Disc Digital Audio System, JEC Techn. Com. No.60: Recording, Sub-committee 60A: Sound Recording, EL 15-14-299/82, 1982.
21. Vries L.B., Immink K.A., Nijb J.G., Doi T.T., Odaka K. etc. The Digital Compact Disc: Modulation and Error Correction Schemes. 67th Convention of the Audio Engineering Society, J. Audio Eng. Soc. (Abstracts), vol. 22, Dec. 1980, p. 931.
22. Doi T.T. General Information on Compact Disc. - JAES, vol. 29, 1981, No. 1/2, pp. 60-66.
23. Doi T.T. Channel Coding for Digital Audio Recordings, JAES, vol. 31, 1983, No. 4, pp. 224-236.
24. Heemskerk J.R.J., Immink K.A. Compact Disc: System Aspects and Modulation, Philips Techn. Rev., vol. 40, 1982, No. 6, pp. 157-164.
25. Immink K.A., Ogava H. EFM – The Modulation Method for the Compact Disc Digital Audio System. AES Digital Audio (Collected Papers from AES Premiere Conference), New York, 1982, pp. 117-126, J. Audio Eng. Soc., 1982, vol. 30, No. 3-6, pp. 117-124.
26. Immink K.A., Gross U. Optimization of Low-Frequency Properties of Eight-to-Fourteen Modulation. – Radio & Electronic Engineer, vol. 53, Febr. 1983, No. 2, pp. 63-66.
27. Исида Ёсинобу. Цифровые звуковые диски. – Тэрэбидзен Гаккайси, т. 36, 1982, №3, с. 194-200.
28. Никамин В.А. Цифровая звукозапись. Технологии и стандарты. - СПб., Наука и Техника, 2002. - 256 с.
29. Накадзима Х., Огава Х. Цифровые грампластинки: Пер. с яп. - М., Радио и связь, 1988. - 168 с.
30. Цифровая звукозапись: Пер. с англ./Под ред. Дж. Маеса и М. Веркаммена.– М.: Мир, 2004. – 352 с.
31. Владимиров Д.А. Булевы алгебры. - М., Наука, 1969.
32. Никамин В.А. Преобразователь 14-8 кода EFM. – Рукопись, депонированная в ЦНИИ «Румб» 1.04.85 г., справка № ДР-2096, реферат опубл. в БАУ «Судостроение», сер. 4, вып. 3, 1985 г.
33. Digital Audio Recording and Reproducing System on Magnetic Tape for Domestic Use. DAT Cassette System. Draft JEC Publication. – Doc. 60A (CO) 115-I,II/ 1988.
34. Immink K.A. Coding Techniques for the Noisy Magnetic Recording Channel: A State of the Art Report. – IEEE Trans. Commun., 1989, vol. 37, No. 5, pp. 413-419.
35. Immink K.A. Graceful Degradation of Magnetic Recording Channel. - IEEE Transaction on Magnetics, 1989, vol. 25, No. 3, pp. 2780-2786.
36. Watkinson J. Inside R-DAT. Part III. – Hi-Fi News, July 1987, pp. 31-33.

37. Newman J., Fisher R. Detection Window and Linear Density Relationship in Digital Recording Systems. – Dig. INTERMAG-81: Int. Magn. Conv., Grenoble, May 1981-12-15, New York, N.Y. 1981, 37/11.
38. Furukawa T. et.al. On a DC Free Block Modulation Code. - IEEE Transaction on Magnetics, 1984, vol. 20, No. 5, Pt. 1: Int. Magn. Conv. (INTERMAG), Hamburg, Apr. 9-13, 1984, pp. 878-880.
39. Никамин В.А. Форматы цифровой звукозаписи. - СПб.: - ЭЛБИ, 1998. - 264 с.
40. Immink K.A. Constructions of Almost Block-Decodable Runlength-Limited Codes. – IEEE Trans. Inform. Theory, vol. 41, Jan. 1995, No. 1, pp. 284-287.
41. Immink K.A. EFMPlus: The Coding Format of the MultiMedia Compact Disc. – IEEE Trans. on Consum. Electron., vol. 41, Aug. 1995, No. 3, pp. 491-497.
42. Balanced disparity channel code for DC control. - U.S. Patent №7218254, May 15, 2007.

ПРЕДМЕТНЫЙ УКАЗАТЕЛЬ

А

автоподстройка частоты фазовая - 7

В

величина числовой суммы (DSV) – 26,45

Д

длина пробега максимальная (λ_{max}) – 9-12, 15

- - минимальная (λ_{min}) – 9-12, 15

- волны записи максимальная (T_{max}) – 9, 14

- - - минимальная (T_{min}) – 9, 14

К

код манчестерский - 20

- Миллера - 22

- EFM – 39-49

- EFM Plus – 57-62

- M² - 23

- 17 PP – 62-64

- 2/3 (I) – 32-34

- 2/3 (II) – 34-37

- 2/4 – 37-39

- 3PM – 24-26

- 4/5 – 26-29

- 4/5 NRZI – 29-31

- 8/10 – 50-56

кодирование манчестерское - 20

- фазовое - 21

- частотное - 22

- DM – 21-22

- PE - 20

коэффициент повышения плотности записи - 15

М

метод записи NRZ (БВН) - 17

- - NRZE - 18

- - NRZI (БВНМ) - 17

- - NRZ-L - 17

- - NRZ-S - 18

- - RB - 16

- - RZ (ВН) - 16

модуляция двухфазная маркированная (Bi-φ-M) - 21

- - пространственная (Bi-φ-S) - 21

- - уровневая (Vi-φ-L) - 21
- задержкой (DM) – 21, 22
- модифицированная фазовая (МФМ) - 22
- - частотная (МФМ) - 22
- с расщеплением фазы - 21
- трехпозиционная (ЗРМ) – 24-26
- фазовая (ФМ) - 21
- частотная (ЧМ) – 8, 20

О

окно детектирования – 6-7

П

последовательность самосинхронизирующаяся - 8
 - с ограниченной длиной пробега - 9

Р

разряд соединительный - 44

С

самосинхронизация – 5-8
 символ информационный - 10
 - канальный - 10
 синхрогруппа – 48, 56, 61, 63
 система тактовой синхронизации - 6
 скорость кода - 15

Ф

функция логическая – 47, 52

Ч

частота тактовая - 5