



(19) RU⁽¹¹⁾ 2 120 141⁽¹³⁾ C1
(51) МПК⁶ G 10 L 9/18

РОССИЙСКОЕ АГЕНТСТВО
ПО ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ РОССИЙСКОЙ ФЕДЕРАЦИИ

(21), (22) Заявка: 95108221/09, 02.06.1994

(30) Приоритет: 11.06.1993 SE 9302025-3
06.12.1993 SE 9304058-2

(46) Дата публикации: 10.10.1998

(56) Ссылки: SU 1297098, A1, 15.03.87. US
4357491, A, 02.11.82. EP 0322213 A2,
13.1.93. GB 2239971 A, 17.07.91. US 5319736,
A, 07.06.94.

(71) Заявитель:

Телефонакtieболагет ЛМ Эрикссон (SE)

(72) Изобретатель: Вигрен Карл Торбьерн (SE),
Бергстрем Рольф Андерс (SE)

(73) Патентообладатель:

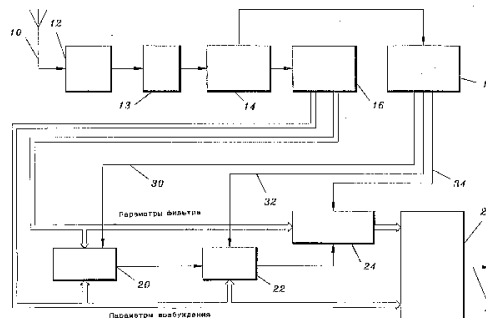
Телефонакtieболагет ЛМ Эрикссон (SE)

(54) УСТРОЙСТВО И СПОСОБ МАСКИРОВАНИЯ ПОСЛЕДСТВИЙ ПОТЕРИ КАДРОВ

(57) Реферат:

Приемник системы радиосвязи на основе передачи кадров содержит декодер речи типа источник-фильтр, который управляется средствами, содержащими переменные внутреннего состояния, обновляемые кадр за кадром, и предназначенными для модификации переданных по каналу связи параметров, определяющих фильтр и представляющих фоновые шумы. Приемник содержит средства для обнаружения потери кадров и средства маскирования последствий потери кадров путем ограничения обновления по меньшей мере одной из переменных внутреннего состояния. Технический результат настоящего изобретения состоит в разработке устройства и способа, в которых

применяется маскирование потерянных кадров принятого сигнала, чтобы сделать декодирование речи более устойчивым или нечувствительным к потере кадров. 2 с. и 14 з.п.ф-лы, 1 ил.





(19) **RU** ⁽¹¹⁾ **2 120 141** ⁽¹³⁾ **C1**
(51) Int. Cl.⁶ **G 10 L 9/18**

RUSSIAN AGENCY
FOR PATENTS AND TRADEMARKS

(12) **ABSTRACT OF INVENTION**

(21), (22) Application: 95108221/09, 02.06.1994

(30) Priority: 11.06.1993 SE 9302025-3
06.12.1993 SE 9304058-2

(46) Date of publication: 10.10.1998

(71) Applicant:
Telefonaktiebolaget LM Eriksson (SE)

(72) Inventor: Vigren Karl Torb'ern (SE),
Bergstrom Rol'f Anders (SE)

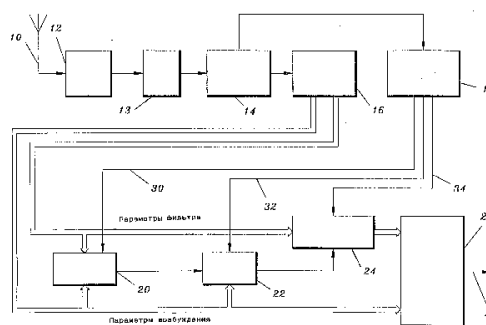
(73) Proprietor:
Telefonaktiebolaget LM Eriksson (SE)

(54) **METHOD AND DEVICE FOR FRAME LOSS RECOVERY**

(57) Abstract:

FIELD: radio communication. SUBSTANCE: device has voice decoder of "source-filter" type. It is controlled by units with variable internal registers which are update on frame basis and are used for modulation of characteristics which are sent through communication channel, determine filter and represent background noise. Receiver has units for detection of frame loss and units for recovery of lost frames by means of restriction on update of at least one variable in internal registers. EFFECT: increased reliability of decoding, decreased

sensitivity to frame losses. 16 cl, 1 dwg



Изобретение относится к маскированию потери кадров в системе связи, например в системе радиосвязи, и касается, в частности, устройства и способа улучшения декодирования фоновых шумов в таких системах.

Известные средства для улучшения кодирования/декодирования фоновых шумов в цифровых сотовых системах радиотелефонной связи предназначены главным образом для такой ситуации, когда связь между кодером и декодером речи близка к идеальной в том смысле, что кадры не теряются. Однако, например, согласно американскому стандарту IS-54 для цифровой сотовой связи быстродействующий совмещенный канал управления (FACCH) образуется при помощи изъятия кадров речи из канала полезного трафика (аналогичный канал предусмотрен в европейской сотовой системе подвижной связи GSM). Похожая ситуация возникает и в сетях с коммутацией пакетов, когда пакеты (кадры) теряются или приходят слишком поздно для того, чтобы использовать их для воспроизведения речи в реальном времени (пакеты могут иметь различные маршруты между передатчиком и приемником).

Целью настоящего изобретения является разработка устройства и способа, в которых применяется так называемое маскирование потерянных кадров принятого сигнала, чтобы сделать декодирование речи более устойчивым или нечувствительным к потере кадров.

В соответствии с изобретением в приемнике системы связи, основанной на передаче кадров, данная цель достигается при помощи устройства маскирования последствий потери кадров в декодере речи, который является декодером типа "источник-фильтр" и управляется средствами, в которых переменные внутреннего состояния обновляются кадр за кадром для модификации определяющих фильтр принятых параметров, представляющих фоновые шумы, указанное устройство отличается тем, что оно содержит:

а) средства обнаружения потери кадров;

б) средства маскирования последствий потери кадров путем ограничения обновления по меньшей мере одной из упомянутых переменных внутреннего состояния.

Кроме того, в соответствии с изобретением в приемнике системы радиосвязи, основанной на передаче кадров, данная цель достигается при помощи способа маскирования последствий потери кадров в декодере речи. Декодер речи является декодером типа "источник-фильтр" и включает средства, в которых переменные внутреннего состояния обновляются кадр за кадром, для модификации принятых определяющих фильтр параметров, представляющих фоновые шумы. Указанный способ отличается тем, что:

а) обнаруживают потерю кадров;

б) маскируют последствия потери кадров путем ограничения обновления по меньшей мере одной из упомянутых переменных внутреннего состояния.

Изобретение поясняется чертежом, который представляет собой блок-схему соответствующих частей приемника системы радиосвязи, содержащей устройство в

соответствии с настоящим изобретением.

Для пояснения изобретения кратко рассмотрим типичное соединение в цифровой сотовой системе радиосвязи и типичные методы и алгоритмы маскирования потерянных кадров.

В линии связи цифровой сотовой радиотелефонной системы звуковой сигнал сначала преобразуется в цифровую форму, а затем к нему применяется алгоритм кодирования речи (см., например, [1]). Этот алгоритм сжимает речевой сигнал и преобразует его в несколько квантованных параметров (обычно на основе кадров). Затем полученные в результате биты защищаются путем создания избыточности при кодировании с использованием методов канального кодирования (см., например, [2]). Полученная последовательность битов модулирует несущую (см., например, [3]) и передается, например, с использованием метода многостанционного доступа с временным разделением каналов. В приемнике сигнал демодулируется. Возможная временная дисперсия или дисперсия, обусловленная многоречевым распространением, может быть скомпенсирована различными способами выравнивания, например выравниванием по методу Витерби или коррекцией с решающей обратной связью (см., например, [3]). Затем используется канальное декодирование (см., например, [2]) для того, чтобы декодировать квантованные параметры, которые необходимы декодеру речи для восстановления переданного речевого сигнала. Из вышеизложенного очевидно, что потерянные кадры могут влиять на восстановленный речевой сигнал, ухудшая его качество.

Например, в американской цифровой сотовой системе связи в соответствии со стандартом IS-54 кадр, изъятый для быстродействующего совмещенного канала управления, приводит к потере кадра речи в декодере речи приемника. Декодер речи решает эту проблему путем "замещения" подходящей информацией. Обычно вместо потерянного кадра используется соответствующая информация из предыдущего кадра. Однако если декодер предусматривает так называемые "антивихревые" операции для фоновых шумов (эти операции будут описаны ниже), то в случае применения этого способа качество полученного в результате сигнала звуковой частоты будет неприемлемым.

С учетом этой информации о современном уровне техники обратимся к чертежу. На нем показаны необходимые для описания настоящего изобретения блоки приемника системы радиосвязи с подвижными объектами. Антенна 10 принимает переданный сигнал и подает его на демодулятор 12. Демодулятор 12 демодулирует принятый сигнал и подает его на выравниватель 13, например на выравниватель Витерби, который преобразует принятый и демодулированный сигнал в одну или несколько последовательностей битов, подаваемых на средства обнаружения потери кадров, выполненные в виде решающих средств 14. Решающие средства 14 определяют, содержит ли принятый кадр биты из канала

полезной нагрузки или же из быстродействующего совмещенного канала управления. Если принятый кадр содержит биты из канала полезной нагрузки, то последовательность битов направляется на канальный декодер 16. Канальный декодер 16 преобразует последовательность битов в последовательность параметров фильтра и последовательность параметров возбуждения для декодирования речи. Если же принятый кадр содержит биты из быстродействующего совмещенного канала управления, то последовательность битов на канальный декодер 16 не направляется. Вместо этого средства 18 маскирования информируются о том, что данный кадр не содержит данных речи.

Приемник содержит также детектор 20 речи (называемый также детектором голосовой активности). Подходящий детектор речи описан в заявке WO 89/08910 компании "British Telecom PLC". Детектор 20 речи с помощью упомянутых параметров фильтра и возбуждения определяет, содержит принятый кадр главным образом речь или же фоновые шумы. Решение детектора 20 речи может быть подано на детектор стационарности, выполненный в виде дискриминатора 22 сигналов (наличие которого необязательно), который использует некоторые из параметров возбуждения для того, чтобы определить, являются ли принятые сигналы, представляющие фоновые шумы, стационарными или нет. Если кадр объявлен содержащим стационарные фоновые шумы, выходной сигнал дискриминатора 22 сигналов управляет модификатор 24 параметров так, чтобы модифицировать принятые параметры фильтра. Указанные детектор 20 речи, дискриминатор 22 сигналов и модификатор 24 параметров представляют собой средства, переменные внутреннего состояния которых обновляются кадр за кадром, для модификации принятых определяющих фильтр параметров, представляющих фоновые шумы. Параметры фильтра, возможно модифицируемые (если принятый сигнал представляет стационарные фоновые шумы), и параметры возбуждения подаются на декодер 26 речи, который формирует на выходе сигнал звуковой частоты, подаваемый на выходную линию 28.

Чтобы описать методы маскирования ошибок в соответствии с настоящим изобретением, необходимо кратко рассмотреть воздействия потери кадров (которые появляются, когда в кадре быстродействующего совмещенного канала управления заменяют кадры речи) на так называемые "антивихревые" алгоритмы. Эти воздействия можно грубо разделить на следующие.

1. Детектор 20 голосовой активности или речи, который используется для управления антивихревым алгоритмом, обычно является адаптивным (см. [4] и заявку WO 89/089910 компании "British Telecom PLC"). Это означает, что имеются пороги и соответствующие состояния, которые автоматически обновляются внутри в детекторе голосовой активности с использованием или измеренного сигнала, или, если этот детектор входит в состав приемника, как в данном описании, декодированных параметров от канального

декодера. Когда кадры теряются, необходимые параметры должны формироваться в приемнике. Одним из способов является использование тех же самых параметров, что и в предыдущем кадре. Однако при этом пороги или переменные внутреннего состояния не обновляются правильным образом, что может привести к ошибочным решениям. Следствием этого будет ухудшенное качество восстановления звукового сигнала.

2. Детектор 20 речи или голосовой активности формирует решение о наличии речи или фоновых шумов, используя входящие параметры фильтра и возбуждения, а также обновленные внутренние состояния, то есть старые входящие параметры и дополнительную априорную информацию. Поэтому потерянные кадры могут сразу же привести к неправильным решениям в приемнике, что приводит к ухудшению качества восстановленного звукового сигнала. Кроме того, поскольку текущее решение зависит также от старых входящих параметров, потерянные кадры могут влиять и на будущие решения.

3. Дискриминатор 22 сигналов, который может быть частью системы в предпочтительном варианте осуществления настоящего изобретения, исследует входящие статистические моменты, предпочтительно значения энергии, которые характеризуют среднюю энергию сигнала для каждого кадра. Необходимо хранить большое число таких значений энергии как для текущего кадра, так и для прежних кадров в одном или нескольких буферах. Если в этих входящих значениях энергии будут иметься ошибки, то эти ошибки также будут храниться в буферах, вызывая ошибочные решения в течение значительного времени. Результатом этого будет ухудшение качества восстановленных сигналов фоновых шумов.

4. Обусловленное потерей кадров неправильное решение отрицательно влияет на операции, используемые для предотвращения "завихрения", которые применяются в случае стационарного сигнала фонового шума. Одним из последствий является разрушение речи, когда ошибочно опознается стационарный фоновый шум и навиваются антивихревые операции. Может появиться также противоположное ошибочное решение (обнаружение речи, когда в действительности присутствует стационарный фоновый шум), вследствие чего характер фонового шума мгновенно изменится, что может вызывать неприятные ощущения при восприятии. Многократные переходы от решения, указывающего на речь, к решению, указывающему на стационарные фоновые шумы, и наоборот, также являются нежелательными, так как такие переходы имеют постоянные времени. Если будут иметь место частые переходы от одного решения к другому, то это будет создавать значительные помехи.

5. Потеря кадров также отрицательно влияет на реальные антивихревые операции в модификаторе 24 параметров (по существу спектральную фильтрацию нижних частот в сочетании с расширением полосы частот). При этом одно из последствий возникает из-за неправильных решений, поступающих от

детектора 20 речи или голосовой активности, или от дискриминатора 22 сигналов. В этих случаях обновление фильтра нижних частот может начинаться или прерываться в отличие от случая, когда не было потеряно никаких кадров. Другое последствие потери кадров проявляется в поражении спектральной информации, которая используется в фильтрах нижних частот и для увеличения полосы пропускания. Оба эти эффекта могут вызывать ухудшение качества.

6. В пост-фильтре декодера 26 речи (если он имеется) возникают проблемы, аналогичные описанным выше в п. 5. Кроме того, чувствительность слуха к восприятию так называемого спектрального наклона велика и, так как на этот наклон может влиять антивихревой алгоритм, потеря кадров может значительно ухудшить качество восстановленного речевого сигнала.

7. Описанные выше эффекты могут сочетаться и влиять друг на друга. Например, неправильное решение в детекторе 20 речи может привести к прекращению обновления содержимого буферов дискриминатора 22 сигналов. В свою очередь, это будет оказывать влияние на дискриминатор 22 сигналов в течение значительного периода времени, вызывая ухудшение качества антивихревых операций.

Из вышеизложенного ясно, что потерянные кадры могут приводить к неправильному обновлению внутренних переменных детектора 20 речи, дискриминатора 22 сигналов, модификатора 24 параметров или к комбинации этих эффектов. В соответствии с настоящим изобретением эти недостатки полностью или частично устраняются при помощи модификации процесса обновления, если обнаружена потеря кадров. Это модифицированное обновление более подробно описано ниже.

Когда обнаружена потеря кадра, средства 18 маскирования могут управлять ходом процесса обновления соответствующих переменных внутреннего состояния в детекторе 20 речи, дискриминаторе 22 сигналов и модификаторе 24 параметров, как показано линиями 30, 32 и 34 управления соответственно. Эти модификации включают в себя прекращение обновления переменных внутреннего состояния (например, порогов) в детекторе 20 речи, если обнаружена потеря кадра. Это подразумевает, что внутренние переменные детектора 20 речи сохраняются равными тем же значениям, что и в предыдущем кадре, или что обновление этих переменных состояния ограничивается (переменные состояния могут корректироваться на меньшие, чем обычно, величины).

Другим действием является сохранение в детекторе 20 речи решения из предыдущего кадра в случае, когда обнаружена потеря кадра.

Если обнаружена потеря кадра, то обновление буферов в дискриминаторе 22 сигналов может быть остановлено или ограничено.

Другой возможностью является сохранение в дискриминаторе 22 сигналов решения для предыдущего кадра.

Если обнаружена потеря кадра, то обновление внутренних коэффициентов

фильтра модификатора 24 параметров, управляющего фильтрацией нижних частот и/или расширением полосы пропускания, может быть остановлено или ограничено.

Спектральный наклон пост-фильтра (если он имеется), может быть сохранен равным наклону для предыдущего кадра.

Так как различные параметры влияют на различные блоки, показанные на чертеже (детектор 20 речи, дискриминатор 22 сигналов, модификатор 24 параметров и пост-фильтр, если он имеется), ясно, что в случае потери кадра может выполняться одно или несколько из этих действий.

Из вышесказанного также понятно, что если теряется несколько последовательных кадров, то переменные внутреннего состояния приемника будут сохраняться (или по существу сохраняться) равными значениям, соответствующим последнему принятому кадру.

Изобретение было описано применительно к системе радиосвязи, основанной на передаче кадров, в которой кадры иногда "изымаются" из канала основного трафика, чтобы быть использованными для других целей. Однако те же самые принципы могут быть применены, если кадры теряются по другим причинам, например, в сети с коммутацией пакетов, где пакеты (они могут рассматриваться или как целые кадры, или как части кадра) используют различные маршруты от передатчика до приемника и могут быть "потеряны" из-за слишком позднего прибытия или действительной потери пакетов во время передачи.

Предпочтительный вариант осуществления способа согласно изобретению поясняется с помощью двух программных модулей на языке Паскаль в Приложении.

Очевидно, что возможны различные усовершенствования и изменения настоящего изобретения, не выходящие за пределы его сущности и объема, определенного формулой изобретения.

Список литературы

1. Atal B.S., Cuperman V. and A. Gersho, eds. *Advances in Speech Coding*. - Kluwer Academic Publishers, 1991.
2. Clark G. C., Cain J.B. *Error Correction Coding for Digital Communication*. - Plenum Press, 1981.
3. Proakis J.G. *Digital Communication*. - 2nd edition, McGraw-Hill, 1989.
4. Voice Activity Detection. Recommendation GSM 06.32, ETSI/GSM, 1991.

Формула изобретения:

1. Устройство для маскирования последствий потери кадров в декодере речи приемника системы связи, основанной на передаче кадров, который представляет собой декодер типа источник-фильтр, управляемый средствами, переменные внутреннего состояния которых обновляются кадр за кадром, для модификации принятых определяющих фильтр параметров, представляющих фоновые шумы, отличающееся тем, что в него введены средства обнаружения потери кадров и средства маскирования последствий потери кадра путем ограничения обновления по меньшей мере одной из указанных переменных внутреннего состояния.

2. Устройство по п. 1, отличающееся тем, что указанные средства модификации параметров фильтра содержат детектор голосовой активности, который имеет по меньшей мере порог для принятия решений о наличии речи/фонового шума, а средства маскирования выполнены с возможностью ограничения обновления указанного порога в случае потери кадра.

3. Устройство по п. 1, отличающееся тем, что указанные средства модификации параметров фильтра содержат детектор голосовой активности для принятия решений о наличии речи/фонового шума кадр за кадром, а средства маскирования выполнены с возможностью запрещения, в случае потери кадра, обновления решения о наличии речи/фонового шума, которое получено для предыдущего кадра.

4. Устройство по п. 2 или 3, отличающееся тем, что указанные средства модификации параметров фильтра дополнительно включают детектор стационарности для распознавания стационарных и нестационарных фоновых шумов, подключенный к выходу детектора голосовой активности и включающий по меньшей мере один буфер, содержащий оценки статистических моментов последних кадров с преобладанием фоновых шумов для принятия решения о стационарности, причем средства маскирования выполнены с возможностью ограничения обновления содержимого буфера в случае потери кадра.

5. Устройство по п. 2 или 3, отличающееся тем, что указанные средства модификации параметров фильтра дополнительно содержат подключенный к выходу детектора голосовой активности детектор стационарности для распознавания стационарных и нестационарных фоновых шумов, причем средства маскирования выполнены с возможностью запрещения, в случае потери кадра, обновления решения о стационарности/нестационарности, полученного для предыдущего кадра.

6. Устройство по одному из пп. 2 - 5, отличающееся тем, что указанные средства модификации параметров фильтра содержат средства фильтрации нижних частот параметров, определяющих фильтр, а средства маскирования выполнены с возможностью ограничения, в случае потери кадра, обновления коэффициентов фильтра процесса фильтрации нижних частот.

7. Устройство по одному из пп. 2 - 6, отличающееся тем, что средства модификации параметров фильтра содержат средства увеличения полосы пропускания фильтра, представленного определяющими его параметрами, а средства маскирования выполнены с возможностью ограничения, в случае потери кадра, обновления коэффициентов фильтра.

8. Устройство по п. 6 или 7, отличающееся тем, что в него введен пост-фильтр для модификации наклона спектра декодированного сигнала, а средства маскирования выполнены с возможностью ограничения, в случае потери кадра, обновления информации о наклоне спектра.

9. Способ маскирования последствий потери кадров в декодере речи приемника системы радиосвязи, основанной на передаче кадров, в соответствии с которым используют

декодер речи типа источник-фильтр, включающий средства, переменные внутреннего состояния которых обновляются кадр за кадром, для модификации принятых определяющих фильтр параметров, представляющих фоновые шумы, отличающийся тем, что обнаруживают потерю кадров и маскируют последствия потери кадров путем ограничения обновления по меньшей мере одной из указанных переменных внутреннего состояния.

10. Способ по п. 9, отличающийся тем, что используют средства модификации параметров фильтра, включающие детектор голосовой активности, имеющий по меньшей мере порог для принятия решения о наличии речи/фонового шума, а при выполнении операции маскирования ограничивают обновление указанного порога в случае потери кадра.

11. Способ по п. 9, отличающийся тем, что используют средства модификации параметров фильтра, включающие детектор голосовой активности для принятия решения о наличии речи/фонового шума кадр за кадром, а при выполнении операции маскирования в случае потери кадра запрещают обновление решения о наличии речи/фонового шума, которое получено для предыдущего кадра.

12. Способ по п. 10 или 11, отличающийся тем, что используют средства модификации параметров фильтра, включающие детектор стационарности для распознавания стационарных и нестационарных фоновых шумов, соединенный с выходом детектора голосовой активности и содержащий по меньшей мере один буфер для хранения оценок статистических моментов последних кадров с преобладанием фоновых шумов для принятия решения о стационарности, а при выполнении операции маскирования ограничивают обновление содержимого указанного буфера в случае потери кадра.

13. Способ по п. 10 или 11, отличающийся тем, что используют средства модификации параметров фильтра, включающие детектор стационарности для распознавания стационарных и нестационарных фоновых шумов, соединенный с выходом детектора голосовой активности, а при выполнении операции маскирования в случае потери кадра запрещают обновление решения о стационарности/нестационарности, полученного для предыдущего кадра.

14. Способ по одному из пп. 10 - 13, отличающийся тем, что используют средства модификации параметров фильтра, включающие средства фильтрации нижних частот параметров фильтра, и ограничивают обновление коэффициентов фильтра процесса фильтрации нижних частот в случае потери кадра.

15. Способ по одному из пп. 10 - 14, отличающийся тем, что используют средства модификации параметров фильтра, включающие средства расширения полосы пропускания фильтра, представленного этими параметрами, и ограничивают обновление коэффициентов фильтра в случае потери кадра.

16. Способ по п. 14 или 15, отличающийся тем, что наклон спектра декодированного сигнала модифицируют с помощью пост-фильтра и ограничивают обновление

информации о наклоне спектра в случае потери кадра.

5

10

15

20

25

30

35

40

45

50

55

60

-7-

RU 2120141 C1

RU 2120141 C1

```
[INHERIT ('spd$def')]
MODULE vad_dtx_rx (input,output);
```

CONST

```
nr_sub_blocks      =      4;
nr_acf_lags_in     =     10;
nr_acf_lags_used   =      8;
burstconst         =      3;
hangconst          =      5;
frames_av0         =      4;
```

VAR

```
adaptcount      : [STATIC] INTEGER;   { For threshold }
thvad           : [STATIC] DOUBLE;    { For threshold }
rvad            : [STATIC] ARRAY [0..nr_acf_lags_used]
                  OF REAL;
burstcount      : [STATIC] INTEGER;   { For overhang }
hangcount       : [STATIC] INTEGER;   { For overhang }
n               : [STATIC] ARRAY [-1..nr_sub_blocks-1]
                  OF INTEGER;
last_dm         : [STATIC] REAL;      { For VAD }
old_lag_count   : [STATIC] INTEGER;
very_old_lag_count : [STATIC] INTEGER;
thresh          : [STATIC] REAL;
lthresh         : [STATIC] INTEGER;
nthresh         : [STATIC] INTEGER;

acf_old,av0     : [STATIC] ARRAY [-frames_av0..0,
                                  0..nr_acf_lags_used]
                  OF REAL;
aav1,rav1,av1   : [STATIC] ARRAY [0..nr_acf_lags_used]
                  OF REAL;

elapsed_frames  : [STATIC] INTEGER;
```



```

sp_hangover      : [STATIC] INTEGER;
speech_dtx       : [STATIC] BOOLEAN;

sp_old           : [STATIC] BOOLEAN;

```

```

PROCEDURE schur_pas ( acf : ARRAY [A1..A2:INTEGER] OF REAL;
                     VAR rc : ARRAY [B1..B2: INTEGER] OF
                           REAL;
                     mdim  : INTEGER ); EXTERNAL;

```

```

PROCEDURE stepup_pas ( rc  : ARRAY [A1..A2:INTEGER] OF REAL;
                     VAR a : ARRAY [B1..B2:INTEGER] OF
                           REAL;
                     mdim  : INTEGER ); EXTERNAL;

```

```

PROCEDURE Flstat_det_rx ( pow  : REAL;
                        sp_l  : BOOLEAN;
                        VAR sp : BOOLEAN); EXTERNAL;

```

```

PROCEDURE Flstat_det_rx_init; EXTERNAL;

```

```

[GLOBAL]
PROCEDURE FLvad_init;          { MUST be called from start}

```

```

VAR
    i,j      : INTEGER;

```

```

BEGIN

```

```

    { threshold }

```

```

    adaptcount := 0;
    thvad := 1000000;
    rvad[0] := 6;
    rvad[1] := -4;
    rvad[2] := 1;

```

RU 2120141 C1

```

FOR i := 3 TO nr_acf_lags_used DO BEGIN
    rvad[i] := 0;
END;

    { end threshold }

    { vad }

old_lag_count := 0;
very_old_lag_count := 0;
n[3] := 19;
hangcount := -1;
burstcount := 0;
last_dm := 0;
thresh := 0.05;

lthresh := 2;
nthresh := 4;

FOR i := -frames_av0 TO -1 DO BEGIN
    FOR j := 0 TO nr_acf_lags_used DO BEGIN
        acf_old[i,j] := 0;
        av0[i,j] := 0;
    END;
END;

    { end vad }

elapsed_frames := 24;
sp_hangover := 0;
speech_dtx := TRUE;
sp_old := TRUE;
Flstat_det_rx_init;
END;    { init_vad_dtx }

PROCEDURE vad_thresh (
    acf0 : REAL;
    { Input }

```

RU 2120141 C1

```

        rav1 : ARRAY [A1..A2: INTEGER]
                OF REAL;                { Input }
        stat : BOOLEAN;                 { Input }
        ptch : BOOLEAN;                 { Input }
        pvad : DOUBLE                   { Output }
    );

```

{ Common variables used: (all output)

```

    adaptcount    initially set to      0
    thvad         initially set to 1000000
    rvad          initially set to
        rvad[0] :    6
        rvad[1] :   -4
        rvad[2] :    1
        rvad[3-8] :  0 ]

```

CONST

```

    pth    = 300000;
    plev   = 800000;
    fac    =      3;
    adp    =      8;
    inc    =     16;
    dec    =     32;
    margin = 80000000;

```

VAR.

```

    i      : INTEGER;

```

BEGIN

```

    IF acf0 < pth THEN BEGIN
        thvad := plev;
    END ELSE BEGIN

```

```

IF NOT (stat AND NOT ptch) THEN BEGIN
    adaptcount := 0;
END ELSE BEGIN
    adaptcount := adaptcount + 1 ;
    IF adaptcount > adp THEN BEGIN
        thvad := thvad - thvad / dec;
        IF thvad < pvad*fac THEN BEGIN
            thvad := MIN (
                thvad + thvad/inc,
                pvad*fac);
        END;

        IF thvad > pvad+margin THEN BEGIN
            thvad := pvad + margin;
        END;

        FOR i := 0 TO nr_acf_lags_used DO
            BEGIN
                rvad[i] := ravl[i];
            END;

            adaptcount := adp + 1;
        END;
    END;
END;

END; { Procedure }

PROCEDURE FLvad_rx_1 (
    acf_in : realACFType;           { Input }
    ltp_lags : integersubframeltptype; { Input }
    VAR vad : BOOLEAN);           { Output }

{ Common variables used:

    n[-1..3]          : ltp_lags   Input/Output
    oldlagcount        :           Input/Output
    veryoldlagcount    :           Input/Output

```

thvad : threshold }

VAR

ptch,
vvad,
stat : BOOLEAN;
lag_count,
smallag,
i,j,k : INTEGER;
acf0,
dm,difference : REAL;
pvad : DOUBLE;
rc : ARRAY [1..nr_acf_lags_used]
OF REAL;

BEGIN

n[-1] := n[3];
FOR i := 0 TO 3 DO BEGIN
n[i] := ltp_lags[i];
END;

FOR i := -frames_av0 TO -1 DO BEGIN
FOR k := 0 TO nr_acf_lags_used DO BEGIN
acf_old[i,k] := acf_old[i+1,k];
av0[i,k] := av0[i+1,k];
END;
END;

FOR k := 0 TO nr_acf_lags_used DO BEGIN
acf_old[0,k] := acf_in[k];
END;

```

{ Adaptive filtering and energy computation. }

pvad := rvad[0] * acf_old[0,0];
FOR k := 1 TO nr_acf_lags_used DO BEGIN
    pvad := pvad + 2.0 * rvad[k] * acf_old[0,k];
END;

{ ACF averaging }

FOR k := 0 TO nr_acf_lags_used DO BEGIN
    av0[0,k] := 0;
    FOR j := 0 TO frames_av0-1 DO BEGIN
        av0[0,k] := av0[0,k] + acf_old[-j,k];
    END;
    avl[k] := av0[-frames_av0,k];
END;

{ Solve the equations system }
schur_pas (avl,rc,nr_acf_lags_used);
stepup_pas (rc,aavl,nr_acf_lags_used);

FOR i := 0 TO nr_acf_lags_used DO BEGIN
    ravl[i] := 0;
    FOR k := 0 TO nr_acf_lags_used-i DO BEGIN
        ravl[i] := ravl[i] + aavl[k] * aavl[k+i];
    END;
END;

IF av0[0,0] <= 0 THEN BEGIN
    dm := 0;
END ELSE BEGIN
    dm := ravl[0] * av0[0,0];
    FOR i := 1 TO nr_acf_lags_used DO BEGIN
        dm := dm + 2*ravl[i]*av0[0,i];
    END;
    dm := dm/av0[0,0];
END;

```

```

difference := dm - last_dm;
stat := ABS(difference) < thresh;
last_dm := dm;
ptch := ((old_lag_count+very_old_lag_count) >= nthresh );
acf0 := acf_in[0];
vad_thresh (acf0,ravl,stat,ptch,pvad);
vvad := (pvad>thvad);
IF vvad THEN BEGIN
    burstcount := burstcount + 1;
END ELSE BEGIN
    burstcount := 0;
END;

IF burstcount >= burstconst THEN BEGIN
    hangcount := hangconst;
    burstcount := burstconst;
END;

vad := vvad OR (hangcount >= 0);

IF hangcount >= 0 THEN BEGIN
    hangcount := hangcount -1;
END;

lag_count := 0;

FOR j := 0 TO 3 DO BEGIN
    IF n[j] > 19 THEN BEGIN
        smallag := MAX(n[j],n[j-1]) MOD
                    MIN(n[j],n[j-1]);
        IF MIN(smallag,MIN(n[j],n[j-1]))-smallag <
            lthresh THEN BEGIN
            lag_count := lag_count + 1;
        END;
    END;
END;
END;

```

```
very_old_lag_count := old_lag_count;  
old_lag_count := lag_count;
```

```
END;
```

```
PROCEDURE FLdtx_hand_rx_1 ( vad : BOOLEAN;      { Input }  
                           VAR sp : BOOLEAN);    { Output }
```

```
BEGIN
```

```
IF elapsed_frames < 24 THEN BEGIN  
    elapsed_frames := elapsed_frames + 1;  
END;
```

```
IF speech_dtx THEN BEGIN
```

```
    IF vad THEN BEGIN  
        sp := TRUE;  
    END ELSE BEGIN  
        sp_hangover := 1;  
        IF elapsed_frames = 23 THEN BEGIN  
            elapsed_frames := 22;  
        END;  
        sp := (elapsed_frames > 23);  
        speech_dtx := FALSE;
```

```
    END;
```

```
END ELSE BEGIN
```

```
    IF vad THEN BEGIN  
        sp := TRUE;  
        speech_dtx := TRUE;
```

```
    END ELSE BEGIN
```

```
        IF sp_hangover < 5 THEN BEGIN  
            sp_hangover := sp_hangover + 1;  
            IF elapsed_frames = 23 THEN BEGIN
```


RU 2120141 C1

RU ? 1 2 0 1 4 1 C 1

```

        END ELSE BEGIN
            sp := sp_old;
        END;
    END;
END.

```

```

[INHERIT ('spd$def')]
MODULE as_actions_rx (input,output);

```

```

    { "Global" (for the module) variables }

```

```

VAR

```

```

    FLfilter_filter_state      : [STATIC] realArray10Type;
    FLfilter_filter_coeff     : [STATIC] REAL;
    FLfilter_filter_gain      : [STATIC] REAL;
    FLfilter_post_state       : [STATIC] realArray10Type;
    FLfilter_post_coeff       : [STATIC] REAL;
    FLfilter_post_gain        : [STATIC] REAL;
    FLfilter_my_state         : [STATIC] REAL;
    FLfilter_my_coeff         : [STATIC] REAL;
    FLfilter_my_gain          : [STATIC] REAL;

```

```

    FExpand_factor            : [STATIC] REAL;
    first_sp                  : [STATIC] BOOLEAN;

```

```

    { External routines }

```

```

PROCEDURE stepdn_unstable_special_pas (
    a          : realArray10Type;      { Input }
    VAR rc      : RealArray10Type;     { Output }
    VAR unstable : BOOLEAN);           { Output }

```

```

EXTERNAL;

```

```

PROCEDURE stepdn_special_pas (
    a          : realArray10Type;      { Input }
    VAR rc      : RealArray10Type);    { Output }

```

EXTERNAL;

```
PROCEDURE FLpostCoeffCalculation(
    ZFLacfW          : realACFType;
    VAR ZFLetaCurr   : realArray10Type );
EXTERNAL;
```

```
PROCEDURE FLcalculateACF(
    FLalphaCurr      : realArray10Type;
    VAR FLacfW       : realACFType );
EXTERNAL;
```

```
PROCEDURE FLcalculateautocorrfunction(
    FLalphaCurr      : realArray10Type;
    VAR FLacfW       : realACFType );
EXTERNAL;
```

[GLOBAL]

```
PROCEDURE FLas_actions_rx_init;
    { MUST be called first to initilize }
    { some things. }

    VAR
        m      : INTEGER;

        { NOTE FLbw_exp is transferred as COMMON }
```

BEGIN

```
    FOR m := 1 TO nrCoeff DO BEGIN
        FLfilter_filter_state[m] := 0;
        FLfilter_post_state[m] := 0;
    END;
```

```
    FLfilter_my_state := 0;
    first_sp := TRUE;
```

{ The following could be placed in ROM }

FLfilter_filter_coeff := EXP(-1.0/(4.0*50.0));

FLfilter_my_coeff := EXP(-1.0/(0.25*50.0));

FLfilter_post_coeff := FLfilter_my_coeff;

FLfilter_filter_gain := 1 - FLfilter_filter_coeff;

FLfilter_post_gain := 1 - FLfilter_post_coeff;

FLfilter_my_gain := 1 - FLfilter_my_coeff;

IF FLbw_exp >= 0 THEN BEGIN

 FLeexpand_factor := EXP(-FLpi*FLbw_exp/8000.0);

END ELSE BEGIN

 FLeexpand_factor := 1;

END;

{FLeexpand_factor := EXP(-FLpi*100.0/8000.0);}

{FLeexpand_factor := EXP(-FLpi*400.0/8000.0);} { ### }

{ ### }

WRITELN('FLfilter_filter_coeff:',FLfilter_filter_coeff);

WRITELN('FLfilter_filter_gain: ',FLfilter_filter_gain);

WRITELN('FLfilter_my_coeff: ',FLfilter_my_coeff);

WRITELN('FLfilter_my_gain: ',FLfilter_my_gain);

WRITELN('FLfilter_post_coeff: ',FLfilter_post_coeff);

WRITELN('FLfilter_post_gain: ',FLfilter_post_gain);

WRITELN('FLbw_exp: ',FLbw_exp);

WRITELN('FLeexpand_factor: ',FLeexpand_factor);

WRITELN('FLv_post: ',FLv_post);

{ ### }

END;

[GLOBAL]

PROCEDURE FLas_actions_rx (

 sp

 : BOOLEAN;

 { In }

```

        FLa_in          : realArray10Type;          { In  } ,
        FLrc_in         : realArray10Type;          { In  }
        FllostFrame     : BOOLEAN;                  { In  }
VAR   FLa_pres         : realArray10Type;          { Out }
VAR   FLrc_pres        : realArray10Type;          { Out }
VAR   FLa_post_pres    : realArray10Type;          { Out }
VAR   FLetacurr        : realArray10Type;          { In/Out }
VAR   FLmy_use         : REAL);                    { Out }

```

VAR

```

        m              : INTEGER;
        FLdum          : REAL;
        FLRC_temp      : realArray10Type;
        unstable       : BOOLEAN;
        FLacfw         : realACFType;
        i_ab           : INTEGER;   { ### }
        eta_temp       : realArray10Type; { ### }

```

BEGIN

```

FOR m := 1 TO nrCoeff DO BEGIN
    FLa_post_pres[m] := FLa_in[m] * FLnyweight[m];
END;

```

IF sp THEN BEGIN

```

    {FLfilter_my_state := FLfilter_my_coeff *
      FLfilter_my_state +
      FLfilter_my_gain * FLmy;} { ### }

```

FLmy_use := FLmy;

```

FOR m := 1 TO nrCoeff DO BEGIN
    FLa_pres[m] := FLa_in[m];

```

```

        FLrc_pres[m] := FLrc_in[m];
    END;

    IF first_sp THEN BEGIN
        FLdum := FExpand_factor;
        FOR m := 1 TO nrCoeff DO BEGIN
            Flfilter_filter_state[m] :=
                FLa_in[m] * FLdum;

            FLfilter_post_state[m] :=
                FLa_post_pres[m] * FLdum;

            FLdum := FLdum * FExpand_factor;
        END;
    END;
END ELSE BEGIN

    IF NOT FllostFrame THEN BEGIN
        FLfilter_my_state := FLfilter_my_coeff *
            FLfilter_my_state +
            FLfilter_my_gain * FLmy_off;
    END;

    FLmy_use := FLfilter_my_state;

    FLdum := FExpand_factor;
    FOR m := 1 TO nrCoeff DO BEGIN
        IF NOT FllostFrame THEN BEGIN
            FLfilter_filter_state[m] :=
                FLfilter_filter_state[m] *
                FLfilter_filter_coeff +
                FLfilter_filter_gain *
                FLa_in[m] * FLdum;

            FLfilter_post_state[m] :=
                FLfilter_post_state[m] *
                FLfilter_post_coeff +

```

```

                                FLfilter_post_gain *
                                FLa_post_pres[m] * FLdum;
END;

FLa_pres[m] :=
    FLfilter_filter_state[m];

FLa_post_pres[m] :=
    FLfilter_post_state[m];

FLdum := FLdum * FLeexpand_factor;
END;

{ Check for stability }

stepdn_unstable_special_pas (
                                FLa_pres,      { In }
                                FLRC_pres,     { Out }
                                unstable);     { Out }

IF unstable THEN BEGIN
WRITELN('Unstable a-parameter (as_actions_rx)');
    FOR m := 1 to nrCoeff DO BEGIN
        FLa_pres[m] := FLa_in[m];
        FLrc_pres[m] := FLrc_in[m];
    END;
END;

. stepdn_unstable_special_pas (
                                FLa_post_pres, { In }
                                FLRC_temp,     { Out }
                                unstable);     { Out }

IF unstable THEN BEGIN
WRITELN('Unstable post_filter (as_actions_rx)');
    FLdum := FLeexpand_factor;
    FOR m := 1 TO nrCoeff DO BEGIN

```

```

        FLa_post_pres[m] := FLa_in[m] *
                                FLnyweight[m]* Fldum;
        FLdum := FLdum * FExpand_factor;
    END;
END;

FLcalculateACF (
    FLa_post_pres,
    FLACFw);

{Flcalculateautocorrfunction (
    FLa_pres,
    FLACFw);}

FLpostCoeffCalculation (
    FLACFw,
    FLetaCurr);

END;

first_sp := (sp AND first_sp);
END;
END.

```

RU 2120141 C1

RU 2120141 C1