



(19) **RU** ⁽¹¹⁾ **2 233 472** ⁽¹³⁾ **C2**
 (51) МПК⁷ **G 06 F 17/30, H 04 Q 7/32**

РОССИЙСКОЕ АГЕНТСТВО
 ПО ПАТЕНТАМ И ТОВАРНЫМ ЗНАКАМ

(12) ОПИСАНИЕ ИЗОБРЕТЕНИЯ К ПАТЕНТУ РОССИЙСКОЙ ФЕДЕРАЦИИ

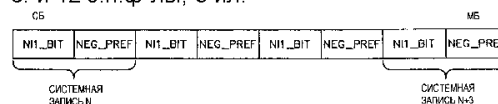
(21), (22) Заявка: 2001116091/09, 12.11.1999
 (24) Дата начала действия патента: 12.11.1999
 (30) Приоритет: 13.11.1998 US 09/191,205
 (43) Дата публикации заявки: 27.06.2003
 (46) Дата публикации: 27.07.2004
 (56) Ссылки: WO 9734240 A, 18.09.1997. WO 9730561 A, 21.08.1997. RU 2039376 C1, 09.07.1995. RU 2101762 C1, 10.01.1998.
 (85) Дата перевода заявки РСТ на национальную фазу: 13.06.2001
 (86) Заявка РСТ: US 99/26852 (12.11.1999)
 (87) Публикация РСТ: WO 00/33593 (08.06.2000)
 (98) Адрес для переписки: 129010, Москва, ул. Б. Спасская, 25, стр.3, ООО "Юридическая фирма Городисский и Партнеры", пат.пов. Ю.Д.Кузнецову, рег.№ 595

(72) Изобретатель: ТУНГ Хенти (US), СОРЕНСОН Джон Р. (US)
 (73) Патентообладатель: КВЭЛКОММ ИНКОРПОРЕЙТЕД (US)
 (74) Патентный поверенный: Кузнецов Юрий Дмитриевич

(54) СПОСОБ И УСТРОЙСТВО ДЛЯ БЫСТРОГО И ПРОИЗВОЛЬНОГО ДОСТУПА К ЗАПИСЯМ ПЕРЕМЕННОГО РАЗМЕРА, СОХРАНЕННЫМ В ПАМЯТИ В СЕКЦИОНИРОВАННОМ ФОРМАТЕ

(57) Изобретение относится к системам мобильной телефонной связи. Заявлены способ и устройство для сохранения списка предпочтительного роуминга и доступа к нему, причем список предпочтительного роуминга образован системными записями переменного размера, каждая из которых содержит поле системной идентификации, поле наличия сетевой идентификации, поле предпочтения отрицания, поле указателя географической области и поле информации параметров обнаружения. Каждая из

системных записей может также содержать поле сетевой идентификации, поле приоритета и поле указателя роуминга. Техническим результатом является создание нового секционированного формата памяти для загрузки списка предпочтительного роуминга в ОЗУ, который облегчает быструю загрузку системных записей и доступ к ним. 4 с. и 12 з.п.ф-лы, 5 ил.



Фиг. 1

RU 2 2 3 3 4 7 2 C 2

RU 2 2 3 3 4 7 2 C 2



(19) **RU** ⁽¹¹⁾ **2 233 472** ⁽¹³⁾ **C2**
 (51) Int. Cl.⁷ **G 06 F 17/30, H 04 Q 7/32**

RUSSIAN AGENCY
FOR PATENTS AND TRADEMARKS

(12) **ABSTRACT OF INVENTION**

(21), (22) Application: 2001116091/09, 12.11.1999
 (24) Effective date for property rights: 12.11.1999
 (30) Priority: 13.11.1998 US 09/191,205
 (43) Application published: 27.06.2003
 (46) Date of publication: 27.07.2004
 (85) Commencement of national phase: 13.06.2001
 (86) PCT application:
US 99/26852 (12.11.1999)
 (87) PCT publication:
WO 00/33593 (08.06.2000)
 (98) Mail address:
129010, Moskva, ul. B. Spasskaja, 25, str.3,
OOO "Juridicheskaja firma Gorodisskij i
Partnery", pat.pov. Ju.D.Kuznetsovu, reg.№ 595

(72) Inventor: TUNG Khenti (US),
SORENSEN Dzhon R. (US)
 (73) Proprietor:
KVEHLKOMM INKORPOREJTED (US)
 (74) Representative:
Kuznetsov Jurij Dmitrievich

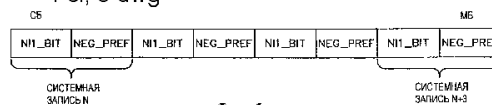
(54) **METHOD AND DEVICE FOR QUICK AND ARBITRARY ACCESS TO RECORDS OF VARIABLE SIZE STORED IN MEMORY IN SECTIONED FORMAT**

(57) Abstract:
 FIELD: mobile telephone service technologies.
 SUBSTANCE: method and device are used for storing a preferable roaming list and accessing it, while said preferable roaming list is formed by system records of variable size, each of which has a field of system identification, field of net identification presence, field of negation preference, field of geographical location pointer and field of detection parameters information.

Each system record may also have field of net identification, field of priority and field of roaming pointer.

EFFECT: new sectioned memory format for loading list of preferable roaming into RAM, which facilitates quick loading of system records and access to them.

4 cl, 8 dwg



RU 2 233 472 C2

RU ? 2 3 3 4 7 2 C2

Область техники

Настоящее изобретение относится к системам мобильной телефонной связи. Более конкретно, настоящее изобретение относится к системам и способам, обеспечивающим пользователю мобильного телефона возможность осуществлять роуминг вне исходной зоны обслуживания пользователя. Еще более конкретно, настоящее изобретение относится к системам и способам сохранения в памяти мобильного телефона записей Списка Предпочтительного Роуминга (СПР) и быстрого доступа к ним.

Уровень техники

В типичных системах мобильной телефонной связи, когда пользователь мобильного телефона перемещается вне исходной зоны обслуживания (осуществляет роуминг), телефон пользователя нередко отключается от поставщика услуг исходной системы и подключается к поставщику услуг альтернативной системы, в результате чего обслуживание пользователя может продолжаться. Чтобы пользователь мог переключиться на поставщика услуг альтернативной системы, мобильный телефон обычно обращается к системным записям СПР, который загружен в память мобильного телефона. На основании системных записей, хранящихся в СПР, мобильный телефон обычно выбирает поставщика услуг альтернативной системы, используемого для продолжения обслуживания пользователя, и, на основании этого выбора и соответствующей информации, хранящейся в системных записях СПР, мобильный телефон автоматически обращается к выбранному поставщику услуг от имени пользователя.

Ассоциация производителей средств связи/Ассоциация электронной промышленности (TIA/EIA) недавно опубликовала новый стандарт, в котором описан формат данных, подлежащий использованию при формировании записей системной таблицы в СПР (см. временный стандарт "Программирование услуг радиосвязи для мобильных станций в системах с расширением по спектру", далее именуемый стандартом IS-683-A), содержание которого полностью включено в настоящее описание изобретения посредством ссылки. Стандарт IS-683-A предусматривает, что каждая системная запись СПР содержит следующие восемь полей:

1. поле SID: 15-битовое поле, которые содержат числовое значение, представляющее собой Системный Идентификационный номер системы, связанной с системной записью;

2. поле NID_INCL: 2-битовое поле, младший бит которого указывает, существует ли Сетевой Идентификационный номер (NID), связанный с системной записью и хранящийся в поле NID, описанном ниже; старший бит этой записи используется для задания значения, присваиваемого NID, если нижеописанное поле NID не содержит никакого значения; значение 11 зарезервировано;

3. поле NID: 16-битовое поле, которое содержит числовое значение, представляющее собой Сетевой Идентификационный номер системы, связанной с системной записью;

4. поле PREF_NEG: 1-битовое поле, указывающее, является ли система, связанная с системной записью, предпочтительной системой или негативной системой;

5. поле GEO: 1-битовое поле, которое указывает, что система, связанная с системной записью, находится в географической области предыдущей системной записи СПР (если GEO=1) или же система находится в новой географической области (если GEO=0);

6. поле PRI: 1-битовое поле, которое указывает, имеет ли текущая системная запись более высокий приоритет по сравнению со следующей системной записью в СПР. Это поле не входит в системную запись, если поле PREF_NEG указывает, что система, связанная с системной записью, является негативной системой;

7. поле ACQ_INDEX: 9-битовое поле, которое содержит указатель на параметры обнаружения, необходимые мобильному телефону для доступа к системе, связанной с системной записью;

8. поле ROAM_IND: 8-битовое поле, которое указывает роуминговый статус соответствующей системной записи; это поле не входит в системную запись, если поле PREF_NEG указывает, что система, связанная с системной записью, является негативной системой.

В современных мобильных телефонах СПР хранится в энергонезависимой памяти мобильного телефона, и, в ходе инициализации системы, СПР целиком перемещается и загружается в ОЗУ мобильного телефона в виде единого побитово-упакованного массива, к которому, в дальнейшем, осуществляется доступ при выполнении операций Выбор Системы или Предпочтительный Роуминг. Одной из целей использования формы побитовой упаковки является экономия объема памяти. К сожалению, поскольку каждая запись СПР имеет переменный размер (поскольку, как описано выше, некоторые или все из полей NID, PRI и ROAM_IND в данной системной записи могут быть опущены), начальная позиция в ОЗУ каждой системной записи СПР является неопределенной. Эта неопределенность затрудняет доступ к данной системной записи и обуславливает дополнительные затраты времени на эту операцию в современных системах. В частности, в каком бы месте Системной Таблицы, загруженной в ОЗУ, ни располагалась системная запись, для осуществления доступа к ней, ввиду неопределенности начальной позиции в ОЗУ каждой системной записи СПР, современным программным средствам доступа к записи приходится последовательно перебирать системные записи в таблице, начиная с первой записи. Другим недостатком современных систем доступа к записям является то, что они не предусматривают "обратный поиск" записей СПР, загруженных в ОЗУ. Например, если система сначала обратилась к 164-й записи СПР, загруженной в ОЗУ, а потом потребовалось обратиться к 161-й записи, то программные средства доступа вместо того, чтобы просто перейти назад в таблице на три записи, вынуждены вновь осуществлять последовательный

перебор всех системных записей в Системной Таблице, загруженной в ОЗУ, начиная с первой записи в таблице, пока не дойдет до 161-й записи.

Вышеуказанные проблемы преодолены в настоящем изобретении, как описано ниже.

Сущность изобретения

Настоящее изобретение основано на использовании нового секционированного формата памяти для загрузки СПР в ОЗУ, который облегчает быструю загрузку системных записей СПР и доступ к ним. Секционированный формат памяти предусматривает использование пяти отдельных массивов для хранения различных полей системных записей СПР. Применяя индексный поиск в этих массивах, программные средства доступа могут быстро обращаться к одному или нескольким полям той или иной системной записи СПР, не прибегая к последовательному перебору всех полей в каждой системной записи, начиная с первой записи, пока не будет достигнута нужная запись. Этот аспект изобретения не только позволяет мобильному телефону быстро обращаться ко всем полям данной системной записи, но, в тех случаях, когда мобильному телефону нужно обратиться только к одному полю системной записи (например, полю SID), настоящее изобретение обеспечивает быстрый доступ к этой информации, исключая необходимость последовательного обращения к другой информации в СПР. Кроме того, система доступа, соответствующая настоящему изобретению, допускает обратный поиск записей в СПР, что невозможно в современных системах, согласно описанному выше.

Изобретение основано также на использовании новой системы считывания СПР из энергонезависимой памяти и загрузки СПР в ОЗУ в ходе инициализации системы. Эта новая система, по существу, не требует никакого дополнительного доступа к энергонезависимой памяти, помимо того, который необходим для перемещения СПР из энергонезависимой памяти в ОЗУ, благодаря тому, что весь процесс переформатирования системных записей происходит непосредственно в ОЗУ после их первоначального перемещения из энергонезависимой памяти в ОЗУ. Поскольку доступ к энергонезависимой памяти осуществляется медленно, этот аспект изобретения позволяет минимизировать время, затрачиваемое в процессе инициализации системы на переформатирование СПР в ОЗУ и загрузку данных в секционированном формате.

Согласно одному частному аспекту, задачей настоящего изобретения является создание способа и устройства для загрузки списка предпочтительного роуминга и доступа к нему, причем список образован системными записями переменного размера, каждая из которых содержит поле системной идентификации (SID), поле наличия сетевой идентификации (NID_INCL), поле предпочтения отрицания (PREF_NEG), поле указателя географической области (GEO) и поле информации параметра обнаружения (ACQ_INDEX). Каждая системная запись может также содержать поле сетевой идентификации (NID), поле приоритета (PRI) и

поле указателя роуминга (ROAM_IND). Поля сетевой идентификации, приоритета и указателя роуминга, входящие в состав записей, хранятся в первом массиве; поля системной идентификации и указателя географической области хранятся во втором массиве; первый бит поля наличия сетевой идентификации и поле предпочтения отрицания хранятся в третьем массиве; второй бит поля наличия сетевой идентификации каждой записи хранится в четвертом массиве и поля информации параметра обнаружения хранятся в пятом массиве. Принимают значение первого индекса, соответствующее записи, выбранной для доступа, и, в соответствии со значением первого индекса, оценивают элементы третьего массива, чтобы осуществить доступ к первому биту поля наличия сетевой идентификации и полю предпочтения отрицания, которые связаны с выбранной записью, и чтобы определить значение второго индекса для нахождения местоположения в первом массиве любого из полей сетевой идентификации, приоритета и указателя роуминга, связанных с выбранной записью. Затем, с помощью значения второго индекса осуществляют доступ в первом массиве к любому из полей сетевой идентификации, приоритета и указателя роуминга, связанных с выбранной записью, и с помощью значения первого индекса осуществляют доступ во втором массиве к полю системной идентификации и полю указателя географической области, связанным с выбранной записью. Доступ в остальных массивах к другим полям, связанным с выбранной записью, также осуществляют с помощью значения первого индекса.

Согласно еще одному аспекту, задачей настоящего изобретения является способ и устройство для быстрого поиска информации поля системной идентификации в списке предпочтительного роуминга. Согласно данному варианту осуществления, системные записи переменного размера загружают в память в виде пяти массивов, описанных в предыдущем абзаце. Однако, согласно еще одному варианту осуществления, принимают значение первого индекса, соответствующее одной из совокупности системных записей, выбранной для доступа, и, в соответствии со значением первого индекса, осуществляют быстрый доступ к полю системной идентификации, связанному с выбранной системной записью и хранящемуся во втором массиве.

Краткое описание чертежей

Признаки, задачи и преимущества настоящего изобретения поясняются в подробном описании, иллюстрируемом чертежами, на которых представлено следующее:

фиг.1 - блок-схема формата данных, используемого для хранения первого бита поля наличия сетевой идентификации и поля предпочтения отрицания, входящих в состав системных записей СПР, в виде массива, согласно предпочтительному варианту осуществления настоящего изобретения;

фиг.2 - блок-схема формата данных, используемого для хранения второго бита поля наличия сетевой идентификации, входящего в состав системных записей СПР,

в виде массива, согласно предпочтительному варианту осуществления настоящего изобретения;

фиг.3 - логическая блок-схема системы доступа к системным записям СПР в соответствии со значением индекса, согласно предпочтительному варианту осуществления настоящего изобретения;

фиг.4, 4А, 4В и 4С - логическая блок-схема системы загрузки в ОЗУ в секционированном формате системных записей СПР, считанных из энергонезависимой памяти, согласно предпочтительному варианту осуществления настоящего изобретения;

фиг.5 - блок-схема, иллюстрирующая компоненты системы загрузки системных записей СПР и доступа к ним в соответствии со значением индекса, согласно предпочтительному варианту осуществления настоящего изобретения.

Подробное описание предпочтительных вариантов осуществления

Настоящее изобретение основано на использовании секционированного формата памяти для загрузки СПР в ОЗУ. В частности, согласно настоящему изобретению, для загрузки различных полей системных записей СПР в ОЗУ мобильного телефона используют пять разных массивов. Первый массив (именуемый массивом SID_GEO) используют для хранения полей SID и GEO системных записей СПР. Каждый элемент массива SID_GEO образован двумя восьмибитовыми словами (всего 16 битов). 15 младших битов каждого элемента этого массива используют для хранения поля SID, связанного с данной системной записью, а старший бит этого элемента используют для хранения поля GEO, связанного с системной записью. 16-битовый элемент, содержащий поля SID и GEO, соответствующие первой системной записи СПР, сохраняют в массиве SID_GEO в качестве первой записи, 16-битовый элемент, содержащий поля SID и GEO, соответствующие второй системной записи СПР, сохраняют в массиве SID_GEO в качестве второй записи и т.д. Таким образом, весь массив SID_GEO состоит из N 16-битовых элементов, каждый из которых содержит поля SID и GEO, соответствующие той или иной системной записи СПР.

Второй массив (именуемый массивом NI_PR) используют для хранения первого бита поля NID_INCL и поля PREF_NEG системных записей СПР. На фиг.1 показана часть массива NI_PR. Каждый элемент массива NI_PR представляет собой двухбитовую запись (эти записи можно группировать по четыре с образованием восьмибитовых слов). Первый бит каждого элемента этого массива используют для хранения первого бита поля NID_INCL, связанного с данной системной записью (этот бит также обозначается как NI1_bit), а второй бит элемента используют для хранения поля PREF_REG, связанного с той же системной записью. Относительный порядок расположения двух битов не имеет существенного значения и может быть выбран произвольно. 2-битовый элемент, содержащий NI1_bit и поле PREF_REG, соответствующие первой системной записи в СПР, сохраняют в массиве NI_PR в качестве первой записи, 2-битовый элемент,

содержащий NI1_bit и поле PREF_REG, соответствующие второй системной записи в СПР, сохраняют в массиве NI_PR в качестве второй записи и т.д. Таким образом, весь массив NI_PR состоит из N 2-битовых элементов, каждый из которых содержит NI1_bit и поле PREF_REG, соответствующие той или иной системной записи в СПР.

Третий массив (именуемый массивом NI2_BIT) используют для хранения второго бита поля NID_INCL системных записей СПР. На фиг.2 показана часть массива NI2_BIT. Каждый элемент массива NI2_BIT представляет собой однобитовую запись (которые можно группировать по восемь с образованием восьмибитовых слов). Единственный бит каждого элемента этого массива используют для хранения второго бита поля NID_INCL, связанного с данной системной записью (этот бит обозначают также как NI2_bit). 1-битовый элемент, содержащий NI2_bit, соответствующий первой системной записи СПР, сохраняют в массиве NI2_BIT в качестве первой записи, 1-битовый элемент, содержащий NI2_bit, соответствующий второй системной записи СПР, сохраняют в массиве NI2_BIT в качестве второй записи и т.д. Таким образом, весь массив NI2_BIT состоит из N 1-битовых элементов, каждый из которых содержит NI2_bit, соответствующий той или иной системной записи в СПР.

Четвертый массив (именуемый массивом ACQ_INDEX) используют для хранения полей ACQ_INDEX системных записей СПР. Каждый элемент массива ACQ_INDEX представляет собой 9-битовую запись, загруженную в ОЗУ в побитово-упакованном виде. Каждый 9-битовый элемент этого массива используют для хранения поля ACQ_INDEX, связанного с данной системной записью. 9-битовый элемент, содержащий поле ACQ_INDEX, соответствующее первой системной записи СПР, сохраняют в массиве ACQ_INDEX в качестве первой записи, 9-битовый элемент, содержащий поле ACQ_INDEX, соответствующее второй системной записи СПР, сохраняют в массиве ACQ_INDEX в качестве второй записи в побитово-упакованном формате сразу после первого 9-битового элемента и т.д. Таким образом, весь массив ACQ_INDEX состоит из N 9-битовых элементов, каждый из которых содержит поле ACQ_INDEX, соответствующее той или иной системной записи в СПР.

Пятый массив (именуемый массивом MISC) используют для хранения полей NID, PRI и ROAM_IND системных записей СПР. Как было отмечено выше, не все записи обязаны содержать поля NID, PRI и ROAM_IND. Таким образом, некоторые записи СПР могут не содержать информацию, соответствующую полям NID, PRI и ROAM_IND массива MISC. Каждый элемент массива MISC представляет собой запись переменного размера, в зависимости от того, какие из полей NID, PRI и ROAM_IND содержатся в данной системной записи. Соответственно, если в конкретной системной записи присутствуют поля PRI и ROAM_IND, а поле NID отсутствует, то для данной системной записи в массиве MISC выделяется 9 битов (что соответствует суммарному размеру полей PRI и ROAM_IND). Если в конкретной системной записи присутствует поле NID, а поля PRI и

ROAM_IND отсутствуют, то для данной системной записи в массиве MISC выделяется 16 битов (что соответствует размеру поля NID). Если в конкретной системной записи присутствуют все три поля NID, PRI и ROAM_IND, то для данной системной записи в массиве MISC выделяется 25 битов (что соответствует суммарному размеру полей NID, PRI и ROAM_IND). Если в данной системной записи отсутствуют все три поля NID, PRI и ROAM_IND, то для этой системной записи в массиве MISC не выделяется никакого места. Поля NID, PRI и ROAM_IND (если присутствуют), связанные с каждой записью СПР, сохраняют в массиве MISC последовательно, в побитово-упакованном виде. Таким образом, элемент переменного размера (0 битов, 9 битов, 16 битов или 25 битов), содержащий некоторые или все из полей NID, PRI и ROAM_IND, соответствующие первой системной записи СПР, сохраняют в массиве MISC в качестве первой записи, элемент переменного размера (0 битов, 9 битов, 16 битов или 25 битов), содержащий некоторые или все из полей NID, PRI и ROAM_IND, соответствующие второй системной записи СПР, сохраняют в массиве MISC в качестве второй записи в побитово-упакованном формате сразу после первого элемента переменного размера и т.д.

На фиг.3 представлена логическая блок-схема системы быстрого доступа к системным записям СПР в соответствии со значением (M) индекса, с использованием вышеописанных массивов SID_GEO, NI_PR, NI2_BIT, ACQ_INDEX и MISC. Значение индекса выражает местоположение в СПР системной записи, к которой требуется осуществить доступ. Так, например, первой системной записи в СПР соответствует значение M=1 индекса, второй системной записи в СПР соответствует значение M=2 индекса и т.д. Процесс начинается с этапа 302, на котором СПР из энергонезависимой памяти мобильного телефона перемещают в ОЗУ мобильного телефона, а затем преобразуют в секционированный формат, распределяя между вышеописанными массивами SID_GEO, NI_PR, NI2_BIT, ACQ_INDEX и MISC. Как было отмечено выше, этот этап предпочтительно осуществлять в ходе инициализации системы. Подробное описание, демонстрирующее предпочтительный вариант осуществления этапа 302, приведено ниже со ссылкой на фиг.4, 4А, 4В и 4С.

Согласно фиг.3, на этапе 304 значение (M) индекса, соответствующее местоположению в СПР системной записи, к которой нужно осуществить доступ, получают с помощью программных средств доступа. На этапе 306 значение M индекса применяют к массиву NI_PR, чтобы определить биты NI1_bit и PREF_NEG, связанные с системной записью, к которой осуществляется доступ. На этапе 306 сначала значение (M-1) умножают на размер каждой записи массива NI_PR (т.е. 2 бита), а затем полученную величину складывают с начальным адресом в ОЗУ массива NI_PR (ni_pr_startp), чтобы найти начальный адрес элемента массива NI_PR, соответствующего M-й записи. На основании этого начального адреса можно легко осуществить доступ к битам NI1_bit и

PREF_NEG, соответствующим нужной системной записи.

Если биты NI1_bit и PREF_NEG указывают наличие поля NID и/или полей PRI и ROAM_IND, связанных с нужной системной записью (этап 308), то на этапе 309 система последовательно оценивает каждый элемент массива NI_PR, начиная с первого элемента и вплоть до (M-1)-го элемента массива. В ходе этого процесса сначала обнуляют счетчик, а затем каждый элемент массива NI_PR (который принимает одно из следующих значений: 00, 01, 10 или 11) вплоть до (M-1)-го элемента последовательно применяют к таблице преобразования, которая управляет счетчиком, обеспечивающим в массиве MISC внутренний отсчет количества битов, связанных с первыми M-1 элементами массива NI_PR. В частности, каждый элемент массива NI_PR, имеющий значение 00 (указывающее, что поля NID, PRI и ROAM_IND входят в запись), увеличивает значение внутреннего счетчика на 0 битов; каждый элемент массива NI_PR, имеющий значение 01 (указывающее, что поля PRI и ROAM_IND входят в запись, а поле NID не входит в запись), увеличивает значение внутреннего счетчика на 9 битов; каждый элемент массива NI_PR, имеющий значение 10 (указывающее, что поля PRI и ROAM_IND не входят в запись, а поле NID входит в запись), увеличивает значение внутреннего счетчика на 16 битов и каждый элемент массива NI_PR, имеющий значение 11 (указывающее, что все три поля, NID, PRI и ROAM_IND, входят в запись), увеличивает значение внутреннего счетчика на 25 битов. После того, как все M-1 записей в NI_PR оценены подобным образом, конечное значение внутреннего счетчика соответствует вышеупомянутому значению второго индекса. На этапе 310 значение второго индекса складывают с начальным адресом в ОЗУ массива MISC (misc_startp), чтобы найти начальный адрес элемента массива MISC, соответствующего M-й записи. На основании этого начального адреса можно легко осуществить доступ к информации в любом из полей NID, PRI или ROAM_IND, соответствующих нужной системной записи.

Хотя, согласно вышеописанному варианту осуществления этапа 309, чтобы определить конечное значение внутреннего счетчика, каждую запись в массиве NI_PR по отдельности применяют к таблице преобразования в последовательном порядке, специалистам в данной области очевидно, что для ускорения процесса вычисления значения второго индекса множество элементов массива NI_PR можно группировать и совокупно применять к таблице преобразования. Например, чтобы ускорить процесс вычисления значения второго индекса, соседние пары элементов массива NI_PR можно применять в виде 4-битовых групп к таблице преобразования, содержащей 16 элементов (а не к вышеописанной 4-элементной таблице).

На этапе 312 значение M индекса применяют к массиву SID_GEO, чтобы определить значения полей SID и GEO, связанных с системной записью, к которой осуществляется доступ. На этапе 312 сначала умножают значение (M-1) на размер каждой записи массива SID_GEO (т.е. 16 битов), а

затем полученную величину складывают с начальным адресом в ОЗУ массива SID_GEO (sid_geo_startp), чтобы найти начальный адрес элемента массива SID_GEO, соответствующего M-й записи. На основании этого начального адреса легко осуществить доступ к полям SID и GEO, соответствующим нужной системной записи.

На этапе 314 значение M индекса применяют к массиву ACQ_INDEX, чтобы определить значение поля ACQ_INDEX, связанного с системной записью, к которой осуществляется доступ. На этапе 314 сначала умножают значение (M-1) на размер каждой записи в массиве ACQ_INDEX (т.е. 9 битов), а затем полученную величину складывают с начальным адресом в ОЗУ массива ACQ_INDEX (acq_index_startp), чтобы найти начальный адрес элемента массива ACQ_INDEX, соответствующего M-й записи. На основании этого начального адреса можно легко осуществить доступ к полю ACQ_INDEX, соответствующему нужной системной записи.

Наконец, на этапе 316 значение M индекса применяют к массиву NI2_BIT, чтобы определить значение NI2_bit, связанное с системной записью, к которой осуществляется доступ. На этапе 316 сначала умножают значение (M-1) на размер каждой записи в массиве NI2_BIT (т.е. 1 бит), а затем полученную величину складывают с начальным адресом в ОЗУ массива NI2_BIT (ni2_bit_startp), чтобы найти начальный адрес элемента массива NI2_BIT, соответствующего M-й записи. На основании этого начального адреса можно легко осуществить доступ к NI2_bit, соответствующему нужной системной записи.

Согласно предпочтительному варианту осуществления настоящего изобретения, предусматривающему обращение только к полю SID, соответствующему той или иной системной записи, этапы 306, 308-310, 314 и 316, представленные на фиг.3, можно опустить. Согласно этому варианту осуществления, после загрузки массивов на этапе 302 и приема значения (M) индекса на этапе 304 процесс сразу переходит к этапу 312, на котором значение M индекса используют, согласно описанному выше, для осуществления доступа к полю SID, связанному с нужной системной записью.

Согласно другому варианту осуществления, который можно использовать после того, как система, показанная на фиг.3, использована для обращения к M-й записи в СПР, настоящее изобретение можно использовать для обратного поиска в СПР, чтобы быстро осуществить доступ к записи, которая предшествует M-й записи СПР. Например, если необходимо обратиться к (M-2)-й записи, система просто использует начальные адреса в массивах, вычисленные при обращении к M-й записи, после чего сообщает этим адресам соответствующие отрицательные приращения, определяя местоположения полей, соответствующих (M-2)-й записи, в пяти рассмотренных выше массивах.

На фиг.4, 4А, 4В и 4С изображена логическая блок-схема системы для осуществления в ОЗУ преобразования системных записей СПР, считанных из энергонезависимой памяти, в

секционированный формат. Первоначально СПР полностью считывают из энергонезависимой памяти в ОЗУ и определяют начальную позицию первой записи в ОЗУ. Затем, на этапе 402, считывают из ОЗУ первую группу системных записей (например, 128 записей или все записи, если их полное количество меньше 128). Размер группы записей, считанной на этапе 402, может быть больше или меньше 128, но предпочтительно кратно 8. На этапе 404 поля SID и GEO, связанные с первой группой системных записей, загружают в первый локальный буфер; биты NI1_bit и поля PREF_NEG, связанные с первой группой системных записей, загружают во второй локальный буфер; биты NI2_bit, связанные с первой группой системных записей, загружают в третий локальный буфер и поля ACQ_INDEX, связанные с первой группой системных записей, загружают в четвертый локальный буфер.

На этапе 406 поля NID, PRI и ROAM_IND (т.е. информационные поля массива MISC), связанные с первой группой системных записей, загружают в первую область ОЗУ в побитово-упакованном виде. Начальную позицию в ОЗУ этой первой области, используемой для загрузки побитово-упакованных информационных полей массива MISC, запоминают как адрес misc_startp. Затем, на этапе 408, конечную позицию в ОЗУ полей NID, PRI и ROAM_IND, загруженных на этапе 406, помечают как позицию MISC_END (т.е. как текущий конец массива MISC в ОЗУ). Затем, на этапе 410, все остальные системные записи, которые не были считаны из энергонезависимой памяти (т.е. все системные записи, кроме считанных в ОЗУ в составе первой группы), перемещают в пределах ОЗУ из их текущей позиции и сохраняют в ОЗУ, начиная сразу после позиции MISC_END.

На этапе 412 помечают конечную позицию необработанных системных записей, сохраненных в ОЗУ на этапе 410, и адрес первого 16-битового слова, следующего после этой позиции, запоминают как значение sid_geo_startp (это значение будет использовано для сохранения в ОЗУ начальной позиции массива SID_GEO в процессе секционирования данных). На этапе 414 часть массива SID_GEO сохраняют в ОЗУ, копируя в ОЗУ содержимое первого локального буфера, начиная с позиции sid_geo_startp.

На этапе 416 бит, следующий за конечной позицией информации SID_GEO, сохраненной в ОЗУ на этапе 414, помечают и запоминают как значение ni_pr_startp (это значение будет использовано для сохранения в ОЗУ начальной позиции массива NI_PR в процессе секционирования данных). На этапе 418 часть массива NI_PR сохраняют в ОЗУ, копируя в ОЗУ содержимое второго локального буфера, начиная с позиции ni_pr_startp.

На этапе 420 бит, следующий за конечной позицией информации NI_PR, сохраненной в ОЗУ на этапе 418, помечают и запоминают как значение ni2_bit_startp (это значение будет использовано для сохранения в ОЗУ начальной позиции массива NI2_BIT в процессе секционирования данных). На этапе 422 часть массива NI2_BIT сохраняют в ОЗУ, копируя в ОЗУ содержимое третьего

локального буфера, начиная с позиции ni2_bit_startp.

На этапе 424 бит, следующий за конечной позицией информации NI2_BIT, сохраненной в ОЗУ на этапе 422, помечают и запоминают как значение acq_index_startp (это значение будет использовано для сохранения в ОЗУ начальной позиции массива ACQ_INDEX в процессе секционирования данных). На этапе 426 часть массива ACQ_INDEX сохраняют в ОЗУ, копируя в ОЗУ содержимое четвертого локального буфера, начиная с позиции acq_index_startp.

Если полное количество записей превышает количество записей в первой группе системных записей, то после вышеописанной обработки первой группы системных записей переходят к этапу 428, на котором следующую группу необработанных системных записей (например, 128 записей или оставшееся количество записей, если их осталось меньше 128) считывают из ОЗУ, начиная непосредственно с позиции MISC_END (т.е. текущего конца массива MISC в ОЗУ). На этапе 430 поля SID и GEO, связанные со следующей группой системных записей, сохраняют в первом локальном буфере, биты NI1_bit и поля PREF_NEG, связанные со следующей группой системных записей, сохраняют во втором локальном буфере; биты NI2_bit, связанные со следующей группой системных записей, сохраняют в третьем локальном буфере и поля ACQ_INDEX, связанные со следующей группой системных записей, сохраняют в четвертом локальном буфере.

На этапе 432 поля NID, PRI и ROAM_IND (т.е. информационные поля массива MISC), связанные со следующей группой системных записей, сохраняют в ОЗУ в побитово-упакованном виде, начиная с позиции MISC_END. Затем, на этапе 434, конечную позицию полей NID, PRI и ROAM_IND, сохраненных в ОЗУ на этапе 432, помечают и запоминают как новое значение позиции MISC_END (т.е. как текущий конец массива MISC в ОЗУ). Затем, на этапе 436, все остальные необработанные системные записи в ОЗУ (т.е. все системные записи, кроме тех, которые были считаны в первой группе или в ходе любых повторений этапа 428) считывают из ОЗУ и перемещают в позицию ОЗУ, следующую сразу после нового значения позиции MISC_END.

На этапе 438 адрес первого 16-битового слова, следующего за конечной позицией в ОЗУ последнего бита необработанных системных записей, перемещенных на этапе 436, помечают и запоминают как новое значение sid_geo_startp (т.е. текущей начальной позиции массива SID_GEO в ОЗУ). Затем, на этапе 440, записи SID_GEO, ранее сохраненные в ОЗУ, начиная с позиции, адресуемой предыдущим значением sid_geo_startp (т.е. значением sid_geo_startp, которое существовало до обновления этого адреса на этапе 438), перемещают в пределах ОЗУ, начиная с позиции, соответствующей новому значению sid_geo_startp. На этапе 442 следующую часть массива SID_GEO сохраняют в ОЗУ, копируя содержимое первого локального буфера в ОЗУ, начиная сразу после конца записей, перемещенных на этапе 440.

На этапе 444 бит, следующий за конечной

позицией информации SID_GEO, сохраненной в ОЗУ на этапе 442, помечают и запоминают его позицию как новое значение ni_pr_startp (текущей начальной позиции массива NI_PR в ОЗУ). Затем, на этапе 446, записи NI_PR, ранее сохраненные в ОЗУ, начиная с позиции, адресуемой предыдущим значением ni_pr_startp (т.е. значением ni_pr_startp, которое существовало до обновления этого адреса на этапе 444), перемещают в пределах ОЗУ, начиная с позиции, соответствующей новому значению ni_pr_startp. На этапе 448 часть массива NI_PR сохраняют в ОЗУ, копируя содержимое второго локального буфера в ОЗУ, начиная сразу после конца записей, перемещенных на этапе 446.

На этапе 450 бит, следующий за конечной позицией информации NI_PR, сохраненной в ОЗУ на этапе 448, помечают и запоминают его позицию как новое значение ni2_bit_startp (текущей начальной позиции массива NI2_BIT в ОЗУ). Затем, на этапе 452, записи NI2_BIT, ранее сохраненные в ОЗУ, начиная с позиции, адресуемой предыдущим значением ni2_bit_startp (т.е. значением ni2_bit_startp, которое существовало до обновления этого адреса на этапе 450), перемещают в пределах ОЗУ, начиная с позиции, соответствующей новому значению ni2_bit_startp. На этапе 454 часть массива NI2_BIT сохраняют в ОЗУ, копируя содержимое третьего локального буфера в ОЗУ, начиная сразу после конца записей, перемещенных на этапе 452.

На этапе 456 бит, следующий за конечной позицией информации NI2_BIT, сохраненной в ОЗУ на этапе 454, помечают и запоминают его позицию как новое значение acq_index_startp (текущей начальной позиции массива ACQ_INDEX в ОЗУ). Затем, на этапе 458, записи ACQ_INDEX, ранее сохраненные в ОЗУ в позиции, адресуемой предыдущим значением acq_index_startp (т.е. значением acq_index_startp, которое существовало до обновления этого адреса на этапе 456), перемещают в пределах ОЗУ, начиная с позиции, соответствующей новому значению acq_index_startp. На этапе 460 часть массива ACQ_INDEX сохраняют в ОЗУ, копируя содержимое четвертого локального буфера в ОЗУ, начиная сразу после конца записей, перемещенных на этапе 458.

На этапе 462 система проверяет, все ли системные записи СПР обработаны вышеописанным способом; если нет, то система переходит к этапу 428, и все последующие действия повторяются до тех пор, пока все системные записи не будут обработаны. Таким образом, этот процесс позволяет сохранять СПР в ОЗУ в секционированном формате с использованием пяти вышеописанных массивов.

Преимущество способа, описанного выше со ссылкой на фиг.4, 4А, 4В и 4С, состоит, в частности, в том, что весь процесс переформатирования данных осуществляется в ОЗУ, где размещается СПР. Для осуществления операций переформатирования помимо памяти, необходимой для хранения значений geo_sid_startp, ni_pr_startp, ni2_bit_startp и acq_index_startp, требуется небольшая дополнительная память.

Вышеописанные способы доступа к данным и сохранения данных предпочтительно осуществлять программными средствами с помощью контроллера 510 на микропроцессоре (показанного на фиг.5), который обычно применяют для обращения к ОЗУ 520 и к энергонезависимой памяти 530 мобильного телефона. ОЗУ 520, изображенное на фиг.5, представляет формат ОЗУ по окончании процесса переформатирования, описанного со ссылкой на фиг.4, 4А, 4В и 4С.

Вышеприведенное описание предпочтительных вариантов осуществления дает возможность специалистам в данной области реализовать и использовать изобретение. Специалисты в данной области могут предлагать различные модификации этих вариантов осуществления и применять общие принципы настоящего изобретения к другим вариантам осуществления, без дополнительного изобретательства. Таким образом, настоящее изобретение не ограничивается представленными способами и устройствами, но охватывает широкий объем, соответствующий нижеприведенной формуле изобретения.

Формула изобретения:

1. Способ сохранения списка предпочтительного роуминга (СПР), хранящегося в мобильном телефоне, и доступа к упомянутому списку, причем СПР образован множеством системных записей переменного размера, каждая из которых содержит поле системной идентификации, поле наличия сетевой идентификации, поле предпочтения отрицания, поле указателя географической области и поле информации параметров обнаружения и дополнительно содержит поле сетевой идентификации, поле приоритета и поле указателя роуминга, включающий этапы, на которых

(А) сохраняют поля сетевой идентификации, поля приоритета и поля указателя роуминга в упомянутом множестве системных записей в первом массиве; сохраняют поля системной идентификации и поля указателя географической области в упомянутом множестве системных записей во втором массиве; сохраняют первый бит каждого из полей наличия сетевой идентификации и поля предпочтения отрицания в упомянутом множестве системных записей в третьем массиве; сохраняют второй бит каждого из полей наличия сетевой идентификации в упомянутом множестве системных записей в четвертом массиве; и сохраняют поля информации параметров обнаружения в упомянутом множестве системных записей в пятом массиве,

(В) принимают значение первого индекса, соответствующее одной из упомянутого множества системных записей, выбранной для доступа,

(С) оценивают, в соответствии со значением первого индекса, записи в третьем массиве, чтобы осуществить доступ к первому биту поля наличия сетевой идентификации и полю предпочтения отрицания, ассоциированным с выбранной системной записью, и чтобы определить значение второго индекса для нахождения местоположения в первом массиве поля сетевой идентификации, поля приоритета или

поля указателя роуминга, ассоциированных с выбранной системной записью в первом массиве,

(D) осуществляют доступ в первом массиве в соответствии со значением второго индекса к полю сетевой идентификации, полю приоритета или к полю указателя роуминга, ассоциированным с выбранной системной записью, и

(Е) осуществляют доступ в соответствии со значением первого индекса к полю системной идентификации и к полю указателя географической области, ассоциированным с выбранной системной записью и сохраненным во втором массиве.

2. Способ по п.1, отличающийся тем, что дополнительно содержит этап, на котором

(F) осуществляют доступ в соответствии со значением первого индекса к полю информации параметров обнаружения, ассоциированному с выбранной системной записью и сохраненному в пятом массиве.

3. Способ по п.2, отличающийся тем, что дополнительно содержит этап, на котором

(G) осуществляют доступ в соответствии со значением первого индекса ко второму биту поля наличия системной идентификации, ассоциированному с выбранной системной записью и сохраненному в четвертом массиве.

4. Способ по п.1, отличающийся тем, что этап (С) включает в себя этап, на котором последовательно применяют элементы третьего массива к таблице преобразования, чтобы вычислить значение второго индекса.

5. Способ по п.1, отличающийся тем, что значение первого индекса, принятое на этапе (В), соответствует смещению позиции выбранной системной записи относительно системной записи, к которой обращались ранее.

6. Способ по п.1, отличающийся тем, что на этапе (А) дополнительно считывают множество системных записей из энергонезависимой памяти мобильного телефона и сохраняют упомянутое множество системных записей в оперативной памяти (ОЗУ) мобильного телефона.

7. Способ по п.6, отличающийся тем, что множеством системных записей манипулируют в ОЗУ на этапе (А), чтобы секционировать упомянутое ОЗУ на первый, второй, третий, четвертый и пятый массивы.

8. Способ по п.7, отличающийся тем, что этап (А) включает этапы, на которых

(i) считывают первую группу системных записей из энергонезависимой памяти, сохраняют поля сетевой идентификации, поля приоритета и поля указателя роуминга, ассоциированные с первой группой системных записей, в первой области упомянутого ОЗУ; сохраняют поля системной идентификации и поля указателя географической области, ассоциированные с первой группой системных записей, в первом буфере, сохраняют первый бит каждого из полей наличия сетевой идентификации и поля предпочтения отрицания, ассоциированные с первой группой системных записей, во втором буфере; сохраняют второй бит каждого из полей наличия сетевой идентификации, ассоциированных с первой группой системных записей, в третьем буфере; сохраняют поля информации параметров обнаружения, ассоциированные с первой

группой системных записей, в четвертом буфере;

(ii) маркируют первую концевую позицию в ОЗУ полей сетевой идентификации, полей приоритета и полей указателя роуминга, сохраненных на этапе (i), считывают все оставшиеся не считанными системные записи и сохраняют оставшиеся не считанными системные записи в упомянутом ОЗУ после первой концевой позиции.

9. Способ по п.8, отличающийся тем, что этап (A) дополнительно включает этап, на котором

(iii) маркируют вторую концевую позицию в ОЗУ, соответствующую концевой позиции в ОЗУ оставшихся не считанными системных записей, сохраненных на этапе (ii), а затем сохраняют часть второго массива путем копирования первого буфера в упомянутое ОЗУ после второй концевой позиции.

10. Способ по п.9, отличающийся тем, что этап (A) дополнительно включает этап, на котором

(iv) маркируют третью концевую позицию в ОЗУ, соответствующую концевой позиции в ОЗУ первого буфера, сохраненного на этапе (iii), а затем сохраняют часть третьего массива путем копирования второго буфера в упомянутое ОЗУ после третьей концевой позиции.

11. Способ по п.10, отличающийся тем, что этап (A) дополнительно включает этап, на котором

(v) маркируют четвертую концевую позицию в ОЗУ, соответствующую концевой позиции в ОЗУ второго буфера, сохраненного на этапе (iv), а затем сохраняют часть четвертого массива путем копирования третьего буфера в упомянутое ОЗУ после четвертой концевой позиции.

12. Способ по п.11, отличающийся тем, что этап (A) дополнительно включает этап, на котором

(vi) маркируют пятую концевую позицию в ОЗУ, соответствующую концевой позиции в ОЗУ третьего буфера, сохраненного на этапе (v), а затем сохраняют часть пятого массива путем копирования четвертого буфера в упомянутое ОЗУ после пятой концевой позиции.

13. Способ по п.12, отличающийся тем, что этап (A) дополнительно включает этапы, на которых

(vii) считывают, начиная с первой концевой позиции в ОЗУ, следующую группу системных записей; сохраняют, начиная с первой концевой позиции в упомянутом ОЗУ, поля сетевой идентификации, поля приоритета и поля указателя роуминга, ассоциированные со следующей группой системных записей; сохраняют поля системной идентификации и поля указателя географической области, ассоциированные со следующей группой системных записей, в первом буфере; сохраняют первый бит каждого из полей наличия сетевой идентификации и поля предпочтения_отрицания, ассоциированных со следующей группой системных записей, во втором буфере; сохраняют второй бит каждого из полей наличия сетевой идентификации, ассоциированных со следующей группой системных записей, в третьем буфере; сохраняют поля информации параметров обнаружения,

ассоциированные со следующей группой системных записей, в пятом буфере,

(viii) обозначают концевую позицию в ОЗУ полей сетевой идентификации, полей приоритета и полей указателя роуминга, сохраненных на этапе (vii), как первую концевую позицию, а затем перемещают все оставшиеся системные записи, следующие за первой концевой позицией в ОЗУ, в первую концевую позицию;

(ix) обозначают концевую позицию в ОЗУ системных записей, перемещенных на этапе (viii), как вторую концевую позицию и перемещают второй массив во вторую концевую позицию,

(x) сохраняют в ОЗУ следующую часть второго массива, присоединяя первый буфер к концу второго массива, перемещенного на этапе (ix),

(xi) обозначают концевую позицию в ОЗУ второго массива, присоединенного на этапе (x), как третью концевую позицию и перемещают третий массив в третью концевую позицию,

(xii) сохраняют в ОЗУ следующую часть третьего массива, присоединяя второй буфер к концу третьего массива, перемещенного на этапе (xi),

(xiii) обозначают концевую позицию в ОЗУ третьего массива, присоединенного на этапе (xii), как четвертую концевую позицию и перемещают четвертый массив в четвертую концевую позицию,

(xiv) сохраняют в ОЗУ следующую часть четвертого массива, присоединяя третий буфер к концу четвертого массива, перемещенного на этапе (xiii),

(xv) обозначают концевую позицию в ОЗУ четвертого массива, присоединенного на этапе (xiv), как пятую концевую позицию и перемещают пятый массив в пятую концевую позицию,

(xvi) сохраняют в ОЗУ следующую часть пятого массива, присоединяя четвертый буфер к концу пятого массива, перемещенного на этапе (xv),

(xvii) повторяют этапы (vii-xvi), пока не обработают все системные записи.

14. Способ сохранения списка предпочтительного роуминга (СПР), хранящегося в мобильном телефоне, и доступа к упомянутому списку, причем СПР образован множеством системных записей переменного размера, каждая из которых содержит поле системной идентификации, поле наличия сетевой идентификации, поле предпочтения_отрицания, поле указателя географической области и поле информации параметров обнаружения, а также дополнительно содержит поле сетевой идентификации, поле приоритета и поле указателя роуминга, включающий этапы, на которых

(A) сохраняют поля сетевой идентификации, поля приоритета и поля указателя роуминга в упомянутом множестве системных записей в первом массиве; сохраняют поля системной идентификации и поля указателя географической области в упомянутом множестве системных записей во втором массиве; сохраняют первый бит каждого из полей наличия сетевой идентификации и поля предпочтения_отрицания в упомянутом множестве системных записей в третьем

массиве; сохраняют второй бит каждого из полей наличия сетевой идентификации в упомянутом множестве системных записей в четвертом массиве; сохраняют поля информации параметров обнаружения в упомянутом множестве системных записей в пятом массиве,

(B) принимают значение первого индекса, соответствующее одной из упомянутого множества системных записей, выбранной для доступа,

(C) осуществляют доступ в соответствии со значением первого индекса к полю системной идентификации, ассоциированному с выбранной системной записью и сохраненному во втором массиве.

15. Устройство для сохранения списка предпочтительного роуминга (СПР), хранящегося в мобильном телефоне, и доступа к упомянутому списку, причем СПР образован множеством системных записей переменного размера, каждая из которых содержит поле системной идентификации, поле наличия сетевой идентификации, поле предпочтения отрицания, поле указателя географической области и поле информации параметров обнаружения, а также дополнительно содержит поле сетевой идентификации, поле приоритета и поле указателя роуминга, содержащее

(A) оперативную память, в которой хранятся поля сетевой идентификации, поля приоритета и поля указателя роуминга в упомянутом множестве системных записей в первом массиве, причем в упомянутой оперативной памяти хранятся поля системной идентификации и поля указателя географической области в упомянутом множестве системных записей во втором массиве, и в упомянутой оперативной памяти хранятся первые биты каждого из полей наличия сетевой идентификации и поля предпочтения отрицания в упомянутом множестве системных записей в третьем массиве, кроме того, в упомянутой оперативной памяти хранятся вторые биты каждого из полей наличия сетевой идентификации в упомянутом множестве системных записей в четвертом массиве, и в упомянутой оперативной памяти хранятся поля информации параметров обнаружения в упомянутом множестве системных записей в пятом массиве,

(B) микропроцессор, связанный с упомянутой оперативной памятью, который принимает значение первого индекса, соответствующее одной из упомянутого множества системных записей, выбранной для доступа, а затем оценивают в соответствии со значением первого индекса записи в третьем массиве, чтобы осуществить доступ к первому биту поля наличия сетевой идентификации и полю предпочтения отрицания, связанным с выбранной системной записью, и чтобы определить значение второго индекса для

нахождения местоположения в первом массиве поля сетевой идентификации, поля приоритета или поля указателя роуминга, ассоциированных с выбранной системной записью,

5 (C) причем упомянутый микропроцессор осуществляет доступ, в соответствии со значением второго индекса, к полю сетевой идентификации, полю приоритета или к полю указателя роуминга, ассоциированным с выбранной системной записью, в первом массиве, и

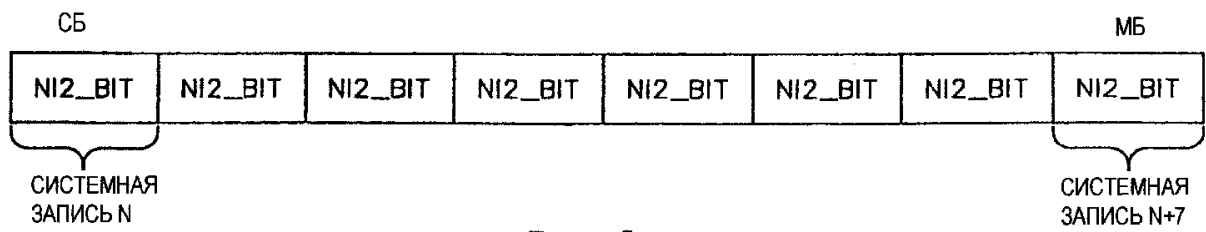
10 (D) упомянутый микропроцессор осуществляет доступ в соответствии со значением первого индекса к полю системной идентификации и к полю указателя географической области, ассоциированным с выбранной системной записью и сохраненным во втором массиве.

16. Устройство для сохранения списка предпочтительного роуминга (СПР), хранящегося в мобильном телефоне, и доступа к упомянутому списку, причем СПР образован множеством системных записей переменного размера, каждая из которых содержит поле системной идентификации, поле наличия сетевой идентификации, поле предпочтения отрицания, поле указателя географической области и поле информации параметров обнаружения, а также дополнительно содержит поле сетевой идентификации, поле приоритета и поле указателя роуминга, содержащее

20 (A) оперативную память, в которой хранятся поля сетевой идентификации, поля приоритета и поля указателя роуминга в упомянутом множестве системных записей в первом массиве, причем упомянутая оперативная память хранит поля системной идентификации и поля указателя географической области во множестве системных записей во втором массиве, и упомянутая оперативная память хранит первые биты каждого из полей наличия сетевой идентификации и поля предпочтения отрицания в упомянутом множестве системных записей в третьем массиве, кроме того, упомянутая оперативная память хранит вторые биты каждого из полей наличия сетевой идентификации в упомянутом множестве системных записей в четвертом массиве, и упомянутая оперативная память хранит поля информации параметров обнаружения в упомянутом множестве системных записей в пятом массиве, и

30 (B) микропроцессор, связанный с упомянутой оперативной памятью, который принимает значение первого индекса, соответствующее одной из упомянутого множества системных записей, выбранной для доступа, а затем осуществляют доступ в соответствии со значением первого индекса к полю системной идентификации, ассоциированному с выбранной системной записью и сохраненному во втором массиве.

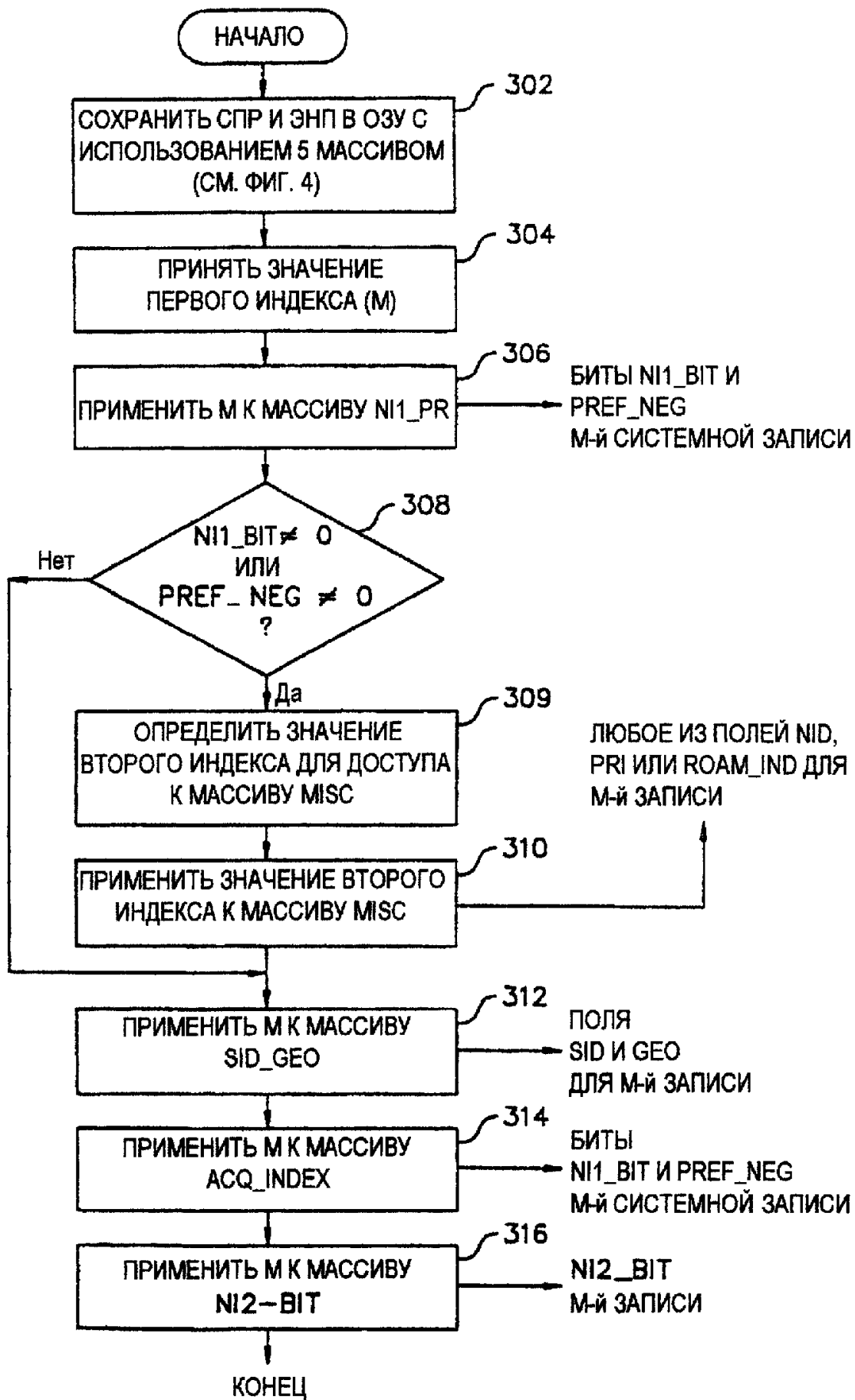
60



Фиг. 2

RU 2233472 C2

RU 2233472 C2



Фиг. 3

НАЧАЛО

ПОЛНОСТЬЮ СЧИТАТЬ СПР ИЗ ЭНП В ОЗУ

ОПРЕДЕЛИТЬ ПОЛОЖЕНИЕ НАЧАЛЬНОГО БИТА ПЕРВОЙ СИСТЕМНОЙ ЗАПИСИ

СЧИТАТЬ ПЕРВЫЕ 128 СИСТЕМНЫХ ЗАПИСЕЙ (ИЛИ ВСЕ ЗАПИСИ, ЕСЛИ ИХ ПОЛНОЕ КОЛИЧЕСТВО МЕНЬШЕ 128)

402

СОХРАНИТЬ ИНФОРМАЦИЮ SID_GEO, NI_PR, NI2_BIT И ACQ_INDEX ИЗ ПЕРВЫХ 128 ЗАПИСЕЙ В ЛОКАЛЬНЫХ БУФЕРАХ

404

УПАКОВАТЬ НА ПОБИТОВОЙ ОСНОВЕ ИНФОРМАЦИЮ МАССИВА MISC ИЗ ПЕРВЫХ 128 ЗАПИСЕЙ В ОЗУ, НАЧИНАЯ С ПЕРВОЙ ПОЗИЦИИ СИСТЕМНОЙ ТАБЛИЦЫ

406

ПОМЕТИТЬ КОНЕЦ МАССИВА MISC В ОЗУ (КАК ПОЗИЦИЮ MISC_END)

408

ПЕРЕМЕСТИТЬ В ОЗУ ОСТАВШИЕСЯ СИСТЕМНЫЕ ЗАПИСИ В ПОЗИЦИЮ СРАЗУ ПОСЛЕ КОНЦА МАССИВА MISC

410

ПОМЕТИТЬ ПОСЛЕДНИЙ БИТ СИСТЕМНЫХ ЗАПИСЕЙ, СКОПИРОВАННЫХ НА ЭТАПЕ 410, И ПОМЕТИТЬ НАЧАЛЬНЫЙ БИТ СЛЕДУЮЩЕГО 16-БИТОВОГО СЛОВА (SID_GEO_STARTP)

412

КОПИРОВАТЬ СОДЕРЖИМОЕ ЛОКАЛЬНОГО БУФЕРА SID_GEO (ПОЛУЧЕННОЕ НА ЭТАПЕ 404) В ОЗУ, НАЧИНАЯ С ПОЗИЦИИ SID_GEO_STARTP

414

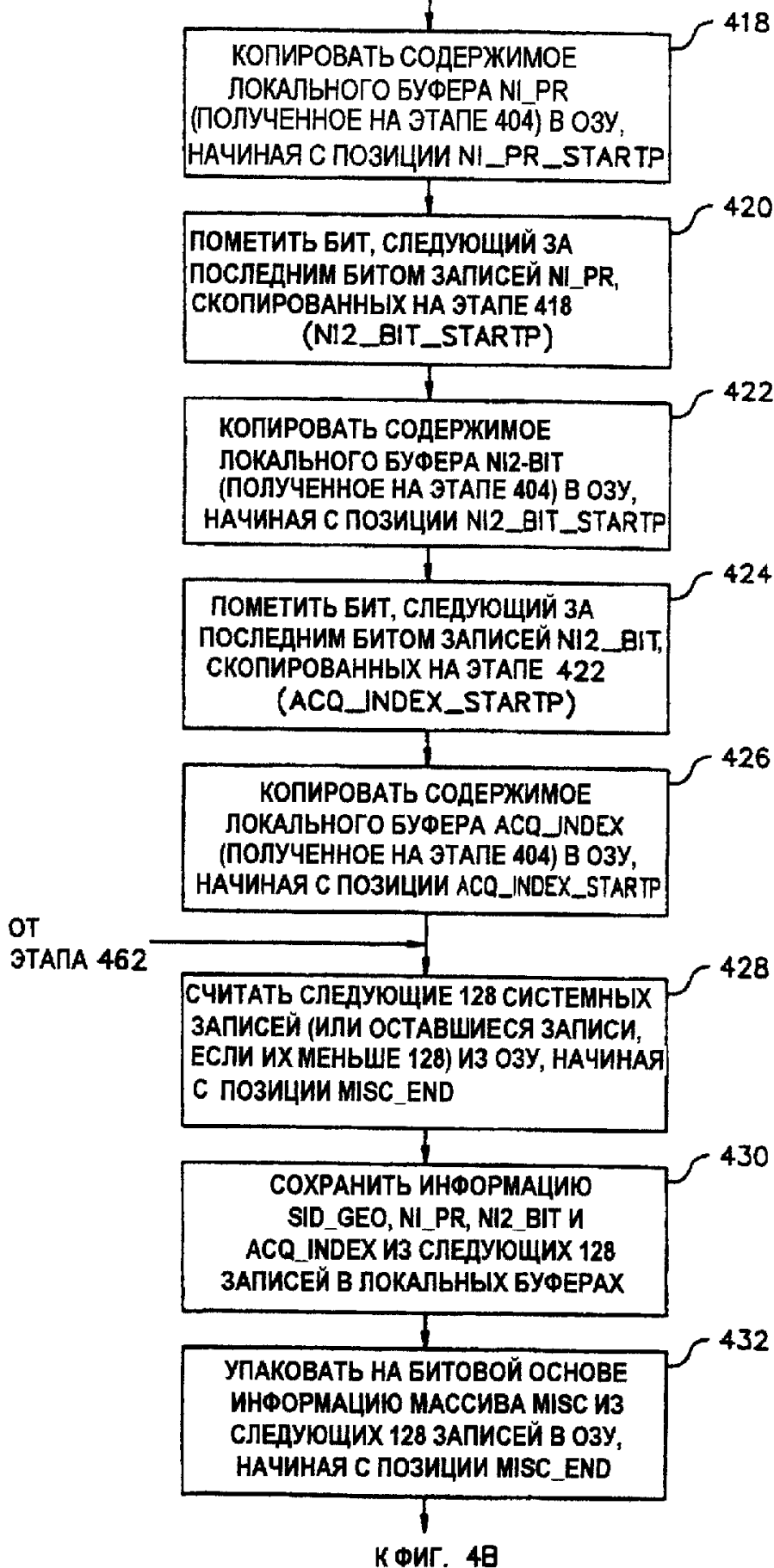
ПОМЕТИТЬ БИТ, СЛЕДУЮЩИЙ ЗА ПОСЛЕДНИМ БИТОМ ЗАПИСЕЙ SID_GEO, СКОПИРОВАННЫХ НА ЭТАПЕ 414 (NI_PR_STARTP)

416

К ФИГ. 4А

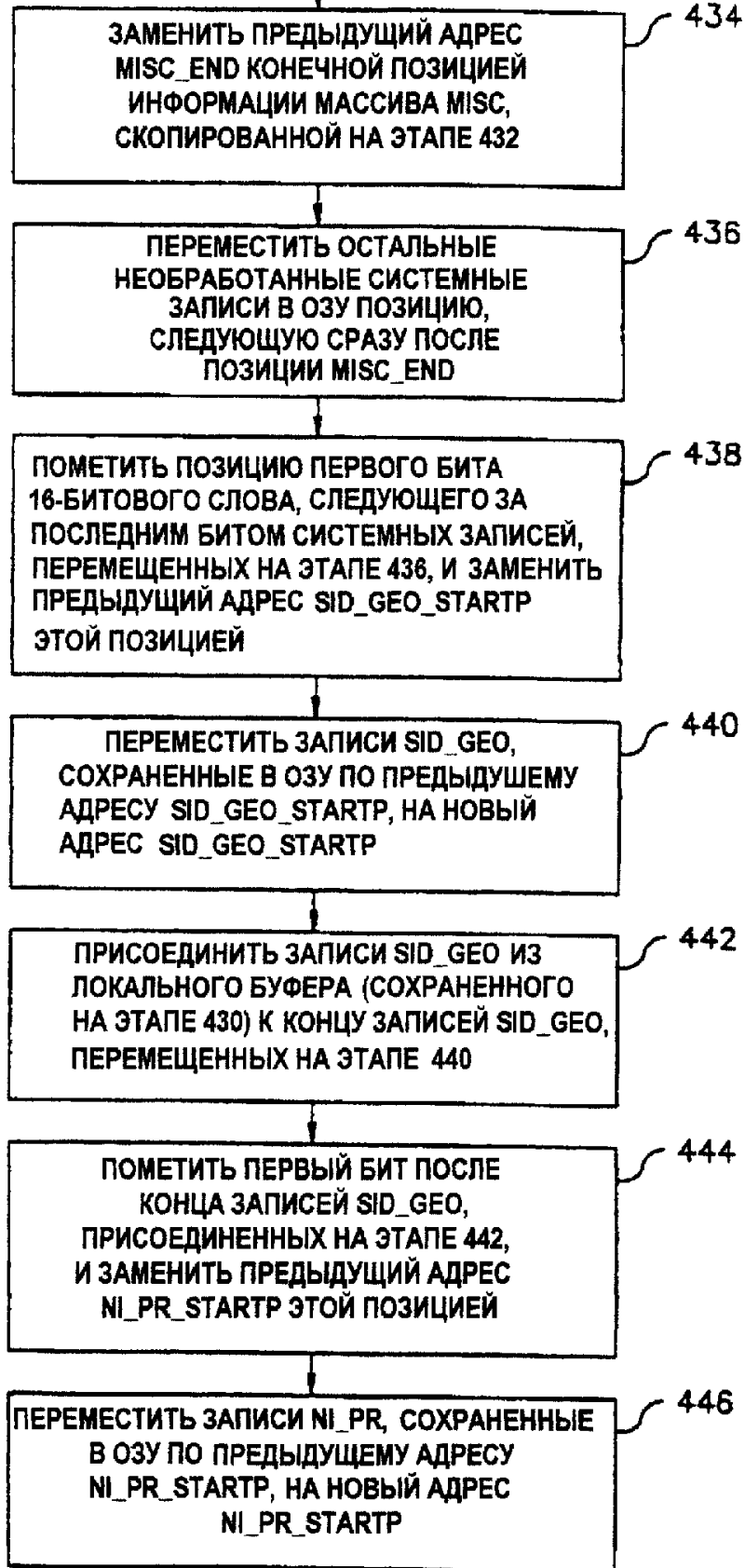
Фиг. 4

(ОТ ФИГ. 4)



Фиг. 4А

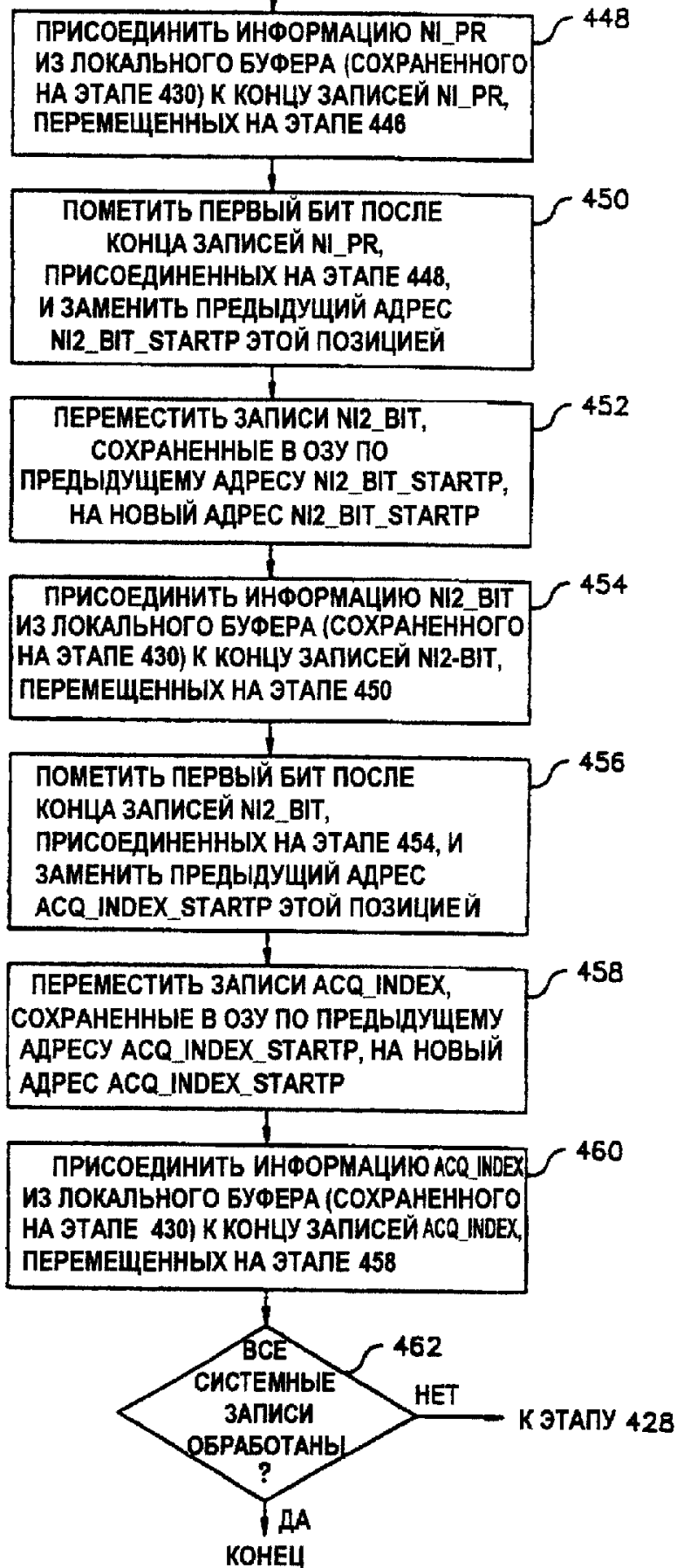
(ОТ ФИГ. 4А)



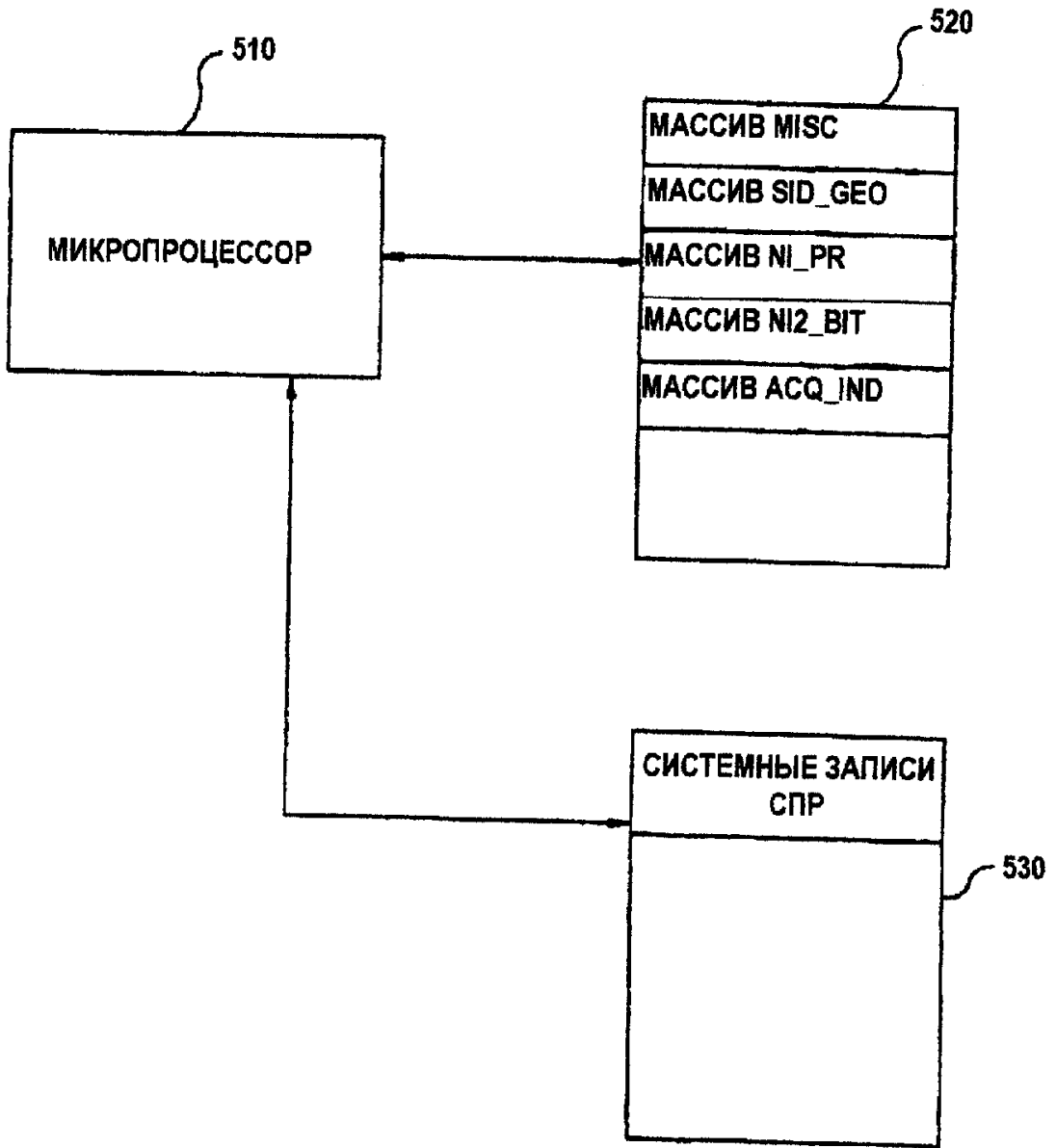
К ФИГ. 4С

Фиг. 4В

(ОТ ФИГ. 4B)



Фиг. 4C



Фиг. 5

RU 2233472 C2

RU 2233472 C2