

# Генератор гармонических колебаний звукового диапазона на базе синтезатора частоты AD9837B, микроконтроллера EFM8SB10 и LCD Nokia-5110 с батарейным питанием

Алексей Кузьминов

В статье описан генератор синусоидальных сигналов на основе DDS AD9837B, МК EFM8SB10, ИУ AD8231, ОУ ОРА333, стабилизаторов TPS78230, STLQ15M15R и дисплея Nokia-5110 с питанием от аккумулятора 18650-HG2 ёмкостью 3 А·ч, обеспечивающего непрерывную работу прибора до 200 часов без подзарядки. Диапазон частот генератора от 1 Гц до 100 кГц с амплитудой до 0,3 В и до 50 кГц с амплитудой до 1,4 В с дискретностью выбора частоты 1 Гц. Приведены принципиальные схемы, разводка платы и конструкция прибора, его программные средства, порядок работы с ним и результаты его работы.

## Введение

При разработке и тестировании звуковой радиоаппаратуры необходимым атрибутом является генератор синусоидальных колебаний звуковых частот, имеющий достаточно низкие искажения выходного сигнала. Подобные генераторы, выпускаемые промышленностью, достаточно дороги и, кроме того, они, как правило, имеют большие габариты и вес, так как оснащены встроенным источником питания, подключаемым к сетевому напряжению 220 В. В связи с этим многие разработчики конструируют свои собственные генераторы, которые, судя по публикациям в журналах или в Интернете, также имеют много недостатков, основным из которых является низкое качество выходного синусоидального сигнала, или, другими словами, этот сигнал имеет достаточно большие искажения.

Конструкции подобных генераторов условно можно разделить на два типа.

Первый основан на применении ОУ совместно с электронными компонентами, включаемыми в его обратную связь для получения синусоидального сигнала (диоды, миниатюрные лампы накаливания и т.п.). Как правило, подобные конструкции имеют достаточно большие искажения синусоидального сигнала на низких (единицы и десятки Гц) и повышенных (десятки кГц) частотах.

Второй тип подобных устройств основан на использовании микросхем прямого цифрового синтеза (Direct Digital Synthes – DDS), одной из которых является наиболее популярная микросхема DDS AD9833 и его малопотребляющий аналог AD9837. Высокое качество выходного синусоидального сигнала и его достаточно малые искажения обеспечиваются следующими причинами. У этих микросхем есть встроенный блок памяти (lookup table), где хранится 4096 значений, которые представляют собой 1/4 периода синусоидальной волны (т.е. от 0 до  $\pi/2$ ). Благодаря симметрии выходных сигналов можно считать объём этого блока памяти равным  $4 \times 4096$  (16 384) значений. То есть когда AD9833 генерирует синусоидальный сигнал определённой частоты, то сначала проходит данную таблицу в прямом направлении, потом в обратном и т.д. Здесь, однако, следует заметить, что встроенный 10-разрядный ЦАП существенно усекает эти значения до 10 бит (т.е. до 1024 отсчетов). Таким образом, положительная и отрицательная полуволны синусоиды имеют по 1024 значений, в связи с чем по вертикали вся синусоида представляется уже 2048 значениями (т.е. 11 бит). Разрешение же по горизонтали составляет от 10 до 14 бит и существенно зависит от частоты выходного сигнала. В звуковом диапазоне и даже несколько выше (до 100 кГц) разре-

ние по горизонтали составляет от 11–12 до 14 бит (на частотах в единицы Гц). Таким образом, вся синусоида представляет собой поле приблизительно  $11 \times 12$  бит, что и определяет высокую разрешающую способность и низкие искажения выходного сигнала подобных DDS. Однако использование AD9833/AD9837 наталкивается на ряд проблем (правда, к счастью, решаемых), которые заключаются в следующем.

Во-первых, выходной сигнал DDS изменяется в диапазоне от 0,37 мВ до 0,645 В (типовое значение), т.е. «сдвигнут» вверх относительно «земли» на постоянную составляющую, приблизительно равную чуть более 0,3 В, которая несколько больше самой амплитуды синусоидального сигнала. А для настройки звуковой аппаратуры требуется синусоидальный сигнал, изменяющийся относительно «земли» как в положительную, так и в отрицательную сторону. В связи с этим в подавляющем большинстве случаев для получения такого сигнала из сигнала DDS применяют разделительные (проходные) конденсаторы достаточно большой ёмкости, иногда даже электролитические, способные пропускать сигналы очень низких частот (единицы Гц), чтобы не препятствовать прохождению самых низкочастотных сигналов звукового диапазона (как правило, это 20 Гц). Однако, как известно, разделительные конденсаторы (особенно электролитические) сами по себе являются источником достаточно больших искажений сигнала, особенно на относительно высоких частотах (десятки кГц). Давняя разработка автора [1] показала, что вместо разделительных конденсаторов с успехом можно использовать инструментальный усилитель (ИУ), сдвигающий уровень сигнала DDS в «отрицательную» область относительно «земли», что позволяет отказаться от разделительных конденсаторов, так как

выходной сигнал, снятый с ИУ, может изменяться не только с самой низкой частотой (например, 1 Гц), но и вообще без искажений воспроизводить даже постоянную составляющую (0 Гц). В то же время любой ИУ, даже самый низкочастотный, легко и без искажений пропускает относительно высокие частоты звукового диапазона (20 кГц и выше).

Во-вторых, практически все без исключения разработки, связанные с получением сигналов звукового частотного диапазона от DDS, используют кварцевые резонаторы, как правило, частотой 25 МГц (обычно для DDS AD9833). Выпускаются даже готовые модули, в составе которых используется AD9833 и кварцевый резонатор частотой 25 МГц. Однако, как следует из описания (datasheet) DDS (AD9833/AD9837), их выходная частота определяется следующим выражением:

$$F_{\text{OUT}} = F_{\text{MCLK}} / 2^{28} \times \text{FREQREG}, \quad (1)$$

где  $F_{\text{OUT}}$  – частота [Гц] на выходе DDS,  $F_{\text{MCLK}}$  – частота [Гц] тактирования DDS, FREQREG – число, загружаемое в специальный регистр частоты DDS.

Из формулы (1) можно найти значение FREQREG в зависимости от частоты тактирования  $F_{\text{MCLK}}$  и требуемой частоты  $F_{\text{OUT}}$ :

$$\text{FREQREG} = (F_{\text{OUT}} \times 2^{28}) / F_{\text{MCLK}} \quad (2)$$

Из формулы (2) следует, что если  $F_{\text{MCLK}} = 25$  МГц, или  $25 \times 10^6$  Гц, и требуемая частота  $F_{\text{OUT}}$  должна быть выражена точным числом Гц, например, 500 Гц, а  $500 \times 2^{28}$  на  $25 \times 10^6$  нацело не делится, то для получения относительно приемлемой точности следует использовать деление с плавающей запятой (ПЗ), которое, в свою очередь, требует применения в программе соответствующей библиотеки с ПЗ, а такая библиотека отнимает до 3–4 кбайт оперативной памяти МК (RAM). Для плат Arduino, Raspberry Pi и им подобных (на базе которых в подавляющем большинстве случаев и конструируются генераторы на основе DDS) с достаточно большим по меркам МК объёмом RAM это не представляет никакой проблемы. Но если использовать обычный 8-разрядный МК с объёмом программной памяти 8 Кбайт и RAM всего в сотни байт, то использование библиотеки с ПЗ становится неприемлемым.

Однако эта проблема решается относительно просто, если использовать кварцевый резонатор с частотой, равной двойке в какой-либо степени, например,  $2^{24} = 16\,777\,216$ ,  $2^{23} = 8\,388\,608$  или  $2^{22} = 4\,194\,304$ . Поскольку максимальная частота так-

тирования DDS AD9837B – 16 МГц, а AD9837A – 5 МГц, то имеет смысл для AD9837B использовать кварцевый резонатор на частоту 16 777 216 Гц (16,777216 МГц), а для AD9837A – 4 194 304 Гц (4,194304 МГц). Кварцевые резонаторы на такие частоты выпускаются, довольно распространены, а потому легкодоступны и недороги. В настоящей статье применен AD9837B, поэтому для него и используется тактовая частота  $F_{\text{MCLK}} = 16\,777\,216$  Гц, или  $2^{24}$  Гц. В этом случае из формулы (2) следует, что  $\text{FREQREG} = (F_{\text{OUT}} \times 2^{28}) / 2^{24} = F_{\text{OUT}} \times 16$ , а сама формула (2) существенно упрощается и принимает вид:

$$\text{FREQREG} = F_{\text{OUT}} \times 16. \quad (3)$$

Таким образом, для загрузки FREQREG нужно просто умножить необходимую частоту в Гц на 16, и никаких библиотек с ПЗ не потребуется. Как будет видно из дальнейшего изложения, программная загрузка FREQREG в AD9837B и его запуск занимает всего несколько строчек кода на C.

В-третьих, для ручного ввода необходимой частоты в DDS, как правило, используется какой-либо дисплей и несколько кнопок или энкодер, для управления которыми в подавляющем большинстве случаев также используются платы Arduino и им подобные. Однако, как будет видно из дальнейшего изложения, для ввода частоты достаточно дисплея и всего двух кнопок, как, например, в электронных часах. Для этого вполне подойдёт простейший 8-разрядный МК.

Здесь следует заметить, что использование плат Arduino и им подобных, имеющих довольно большие габариты, потребляющих значительную энергию, а потому требующих внешнего источника питания (ИП), наталкивается на ещё одну проблему. Если ИП построен на базе сетевого трансформатора, выпрямителя и стабилизатора, то такой ИП является источником сетевых наводок, и в выходном сигнале DDS неизбежно появляется 50-герцовый шум, лежащий в звуковом частотном диапазоне, и избавиться от этого шума достаточно проблематично. Если же ИП построен на основе ВЧ-преобразователя, как, например, ИП для зарядки телефонов, то такой ИП также является источником шума на частотах в несколько кГц, также лежащих в звуковом диапазоне частот, и полностью избавиться от этого шума также не представляется возможным.

Если используется малопотребляющий DDS, как, например, AD9837B,

малопотребляющий дисплей, например LCD, малопотребляющие стабилизаторы, малопотребляющие ОУ и ИУ и, наконец, малопотребляющий МК, например EFM8SB10 (SB – сокращение от Sleepy Bee), то для питания генератора вполне достаточно применения небольшого аккумулятора, например, 18650 размером (d/h) 18×65 мм, напряжением 3,6 В и ёмкостью более 3 А·ч. В этом случае никаких внешних источников питания не потребуется (и, естественно, все вышеперечисленные наводки и шумы будут полностью исключены). Как будет видно из дальнейшего изложения, такой аккумулятор способен обеспечивать непрерывную работу генератора в течение, как минимум, 200 часов. А если на дисплей вывести состояние заряженности аккумулятора, то за ним можно легко следить и вовремя подзаряжать аккумулятор внешним ИП от того же телефона.

Дальнейшее изложение построено следующим образом. Вначале приведены принципиальные схемы печатной платы и дополнительных устройств прибора, затем – программные средства для него, далее рассмотрены разводка и внешний вид его платы. После этого рассказано о настройке и конструкции прибора, порядке работе с ним и, наконец, показаны результаты его работы.

## Принципиальные схемы

Основой схемы платы генератора (рис. 1) является МК EFM8SB10-QFN20 (DD1) в корпусе для поверхностного монтажа QFN20 размером 3×3 мм. Конденсатор C8 блокировочный и положен по штату работы МК. Сопряжение МК с DDS и дисплеем Nokia-5110 организовано по одностороннему интерфейсу SPI, или, другими словами, в эти устройства только передаётся (но не принимается) информация с помощью сигналов SCK и MOSI, а также сигналов выбора кристалла (CS). Для DDS AD9837B (DD3) используется сигнал CSAD, а для дисплея CSD. Для тактирования DD3 используется системный тактовый сигнал МК SYS\_CLK частотой 16 777 216 Гц, который также подаётся на разъём X5, предназначенный для подключения к частотомеру и служащий для точной настройки частоты. Конденсаторы C4–C7 положены по штату работы DD3. Помимо сигналов интерфейса SPI, в дисплей передаются сигналы D/C и RES. Все дисплейные сигналы вместе с питанием и «землёй»

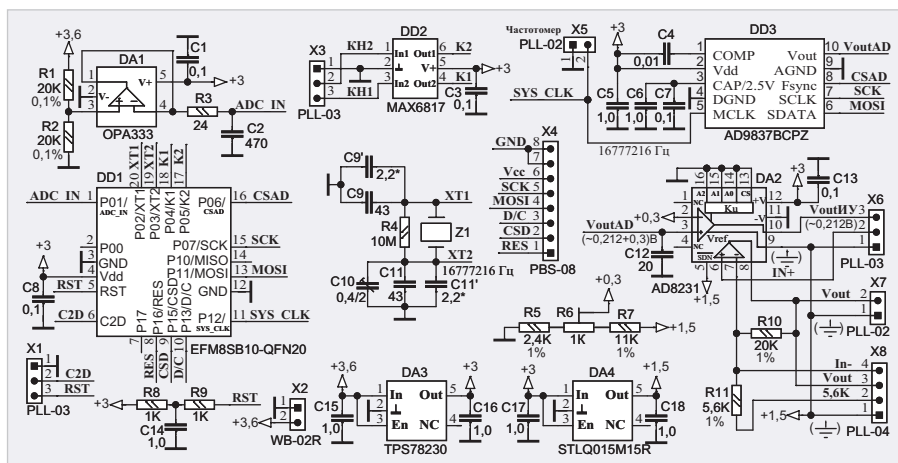


Рис. 1. Принципиальная схема платы генератора

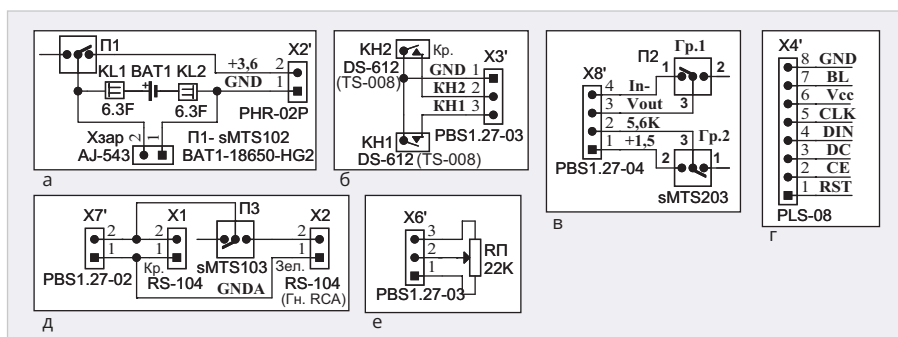


Рис. 2. Принципиальные схемы дополнительных устройств генератора: а – подключение аккумулятора, б – подключение кнопок, в – подключение тумблера для изменения коэффициента усиления, г – подключение LCD Nokia-5110, д – подключение тумблера для дополнительного разъёма RCA, е – подключение потенциометра для изменения амплитуды выходного сигнала генератора

выведены на разъём X4, к которому подключается плата дисплея ответным разъёмом X4' (рис. 2г). Кварцевый резонатор Z1 частотой 16 777 216 Гц вместе с резистором R4 и конденсаторами C9–C11 и C9', C11', подбираемыми при настройке (см. далее), подключается к выводам XT1 и XT2 МК.

Для устранения дребезга контактов кнопок используется специализированная микросхема MAX6817 (DD2). Сигналы (с дребезгом) непосредственно от кнопок КН1 и КН2 подключаются к его входам, а очищенные от дребезга выходные сигналы К1 и К2 подключаются к соответствующим выводам МК. Блокировочный конденсатор C3 положен по штату работы DD2. Сами сигналы от кнопок (и «земля») поступают на плату с разъёма X3, к которому ответным разъёмом X3' (рис. 2б) подключается кабель, второй конец которого соединён непосредственно с кнопками DS-612 (TS-008) – КН1 и КН2. При нажатии и отпуске кнопок слышен характерный щелчок, а также присутствует тактильная обратная связь.

RC-цепочка R8, R9, C14 предназначена как для штатной работы МК (она затягивает низкое состояние сигнала RST на время заряда конденсатора C14, требующееся при включении питания и формирующее сброс (RESET) МК – Power On Reset – POR), так и для программирования МК по интерфейсу C2 (резистор R9 позволяет легко управлять сигналом RST от этого интерфейса). Сигналы интерфейса C2 (C2D, RST) и «земля» выведены на разъём X1, к которому ответным разъёмом подключается кабель, второй конец которого соединён с соответствующим разъёмом USB DEBUG адаптера, который, в свою очередь, подключается к одному из разъёмов USB компьютера. Схема подключения подробно описана в [2].

Напряжение питания поступает на плату с разъёма X2. Это напряжение подаётся на вход микромощного стабилизатора TPS78230 (DA3), выходное стабилизированное напряжение которого (3 В) используется для питания всех микросхем платы. Микромощный стабилизатор STLQ15M15R (DA4) из входного напряжения 3 В формирует

выходное напряжение 1,5 В, которое используется как опорное ( $V_{REF}$ ) для ИУ AD8231 (DA2). Блокировочные конденсаторы C15–C18 положены по штату работы стабилизаторов DA3 и DA4. К разъёму X2 ответным разъёмом (X2', рис. 2а) подключается кабель, второй конец которого через тумблер sMTS102 (П1, рис. 2а) и ножевые клеммы 6.3F (KL1 и KL2) подключается к аккумулятору 18650-HG2 (BAT1), оборудованному двумя приваренными пластинами шириной 6 мм, на которые надеваются клеммы. К этим же клеммам подключается разъём А-543 – аудиогнездо 2,5 моно (Хзар), к которому ответным разъёмом (штекер 2,5 моно) подключается кабель, второй конец которого подключается к плате зарядного устройства на базе специализированной микросхемы TC4056A (см. далее).

Для определения степени разряженности аккумулятора входное напряжение питания, условно обозначенное как «+3,6», подаётся на делитель напряжения (в 2 раза), построенный на двух прецизионных (0,1%) и равных резисторах R1 и R2. С точки соединения этих резисторов уменьшенное в 2 раза входное напряжение подаётся на вход повторителя напряжения, построенного на микропотребляющем ОУ OPA333 (DA1), а с его выхода через RC-цепочку R3, C2 – на один из входов АЦП МК (сигнал ADC\_IN). МК с помощью встроенного высокоскоростного АЦП измеряет это напряжение, сравнивает его с тремя порогами (3,2 В, 3,5 В и 3,8 В) и, в зависимости от этого напряжения, выводит на дисплей рисунок аккумулятора с тремя сегментами, о степени заряженности которого можно судить по количеству включённых сегментов (см. далее). Блокировочный конденсатор C1 положен по штату работы ОУ DA1.

Выходное напряжение DDS D3 (сигнал VoutAD) с действующим значением ~0,212 В и постоянной составляющей чуть более +0,3 В подаётся на неинвертирующий вход ИУ AD8231 (3-й вывод DA2), к которому подключён конденсатор C12, сглаживающий дискретность представления выходного напряжения ЦАП DDS (как указывалось выше, эта дискретность представляется как 2048 точек по вертикали и 2048–4096 точек по горизонтали), в результате чего выходной синусоидальный сигнал с DDS становится практически гладкой функцией.

На инвертирующий вход ИУ (2-й вывод DA2) подаётся напряжение чуть

более +0,3 В, полученное с движка подстроечного резистора R6, входящего в состав делителя напряжения R5, R6, R7. Как известно, выходное напряжение ИУ определяется следующей формулой:

$$V_{out\ ИУ} = (V_{in+} - V_{in-}) \times K_u + V_{ref} \quad (4)$$

где  $V_{out\ ИУ}$  – выходное напряжение ИУ,  $V_{in+}$  – напряжение на неинвертирующем входе,  $V_{in-}$  – напряжение на инвертирующем входе,  $V_{ref}$  – опорное напряжение,  $K_u$  – коэффициент усиления ИУ. Если  $K_u = 1$ , формула (4) упрощается и принимает вид:

$$V_{out\ ИУ} = V_{in+} - V_{in-} + V_{ref} \quad (5)$$

Пусть постоянная составляющая выходного напряжения DDS равна, например, 0,35 В, тогда напряжение  $V_{in+} = \sim 0,212\text{ В} + 0,35\text{ В}$ . Если выходное напряжение с движка R6 делителя R5, R6, R7 настроить также на 0,35 В, то  $V_{in-} = 0,35\text{ В}$ . И, наконец, если  $V_{ref} = 1,5\text{ В}$ , то формула (5) ещё более упрощается:  $V_{out\ ИУ} = \sim 0,212\text{ В} + 0,35\text{ В} - 0,35\text{ В} + 1,5\text{ В}$  и приобретает вид:

$$V_{out\ ИУ} = \sim 0,212\text{ В} + 1,5\text{ В} \quad (6)$$

Смысл формулы (6) состоит в том, что теперь выходное напряжение ОУ  $V_{out\ ИУ}$  изменяется относительно +1,5 В как в положительную, так и в отрицательную сторону с амплитудой 0,3 В. Если принять напряжение  $V_{ref} = +1,5\text{ В}$  за, например, аналоговую «землю» (перевернутая ёлочка на рис. 1 рядом с ИУ DA2), то  $V_{out\ ИУ}$  также будет изменяться как в положительную, так и в отрицательную сторону относительно этой аналоговой «земли» с амплитудой 0,3 В. Другими словами, мы получили чистый синусоидальный сигнал без постоянной составляющей с амплитудным значением 0,3 В относительно аналоговой «земли».

Здесь следует заметить, что если DDS находится в состоянии сброса (такая команда на DDS относительно часто подаётся в программе на МК – см. далее), то его выходное напряжение принимает среднее значение, в связи с чем выходное напряжение  $V_{out\ ИУ}$  относительно аналоговой «земли» будет равно нулю ( $V_{out\ ИУ} = 0$ ). Это свойство DDS полезно использовать для точной настройки напряжения с движка R6 делителя R5, R6, R7 (см. далее).

Как видно из рис. 1, сигнал  $V_{out\ ИУ}$  подаётся на 3-й контакт разъёма X6, сигнал с неинвертирующего входа дополнительного ОУ, встроенного в DA2 (6-й вывод DA2), подаётся на его 2-й контакт, а аналоговая «земля» – на его 1-й контакт. К этому разъёму ответным разъёмом X6' (рис. 2е) подключён трёхпроводный кабель, второй конец

которого подключён к переменному резистору РП (СП4-1). Выход этого дополнительного ОУ (8-й вывод DA2, сигнал  $V_{out}$ ) подключён ко 2-му контакту разъёма X7, а его 1-й контакт соединён с аналоговой «землей». Кроме того, сигнал  $V_{out}$  подключён к 3-му контакту разъёма X8, сигнал с инвертирующего входа дополнительного ОУ (7-й вывод DA2, сигнал  $In-$ ) подключён к 4-му контакту X8, нижний конец резистора R11 подключён ко 2-му контакту X8, а аналоговая «земля» – к 1-му контакту. К разъёму X8 ответным разъёмом (X8', рис. 2в) подключается кабель, второй конец которого подключается к тумблеру sMTS203 (П2, рис. 2в) с двумя группами контактов.

Назначение разъёмов X6 и X8 – регулировка амплитуды выходного сигнала  $V_{out}$  и изменение коэффициента усиления дополнительного ОУ DA2.

Более подробно этот процесс схематично показан на рис. 3. С движка переменного резистора РП сигнал  $In+$  попадает на неинвертирующий вход ОУ. Резисторы R10 и R11 совместно с группами контактов Гр.1 и Гр.2 тумблера включены так, что в одном положении групп контактов (как на рис. 3) выводы ОУ 6 и 8 замыкаются накоротко, а левый по схеме конец резистора R11 оказывается свободным. В этом случае ОУ представляет собой обычный повторитель напряжения, поданного на его неинвертирующий вход с движка резистора РП. В ином положении групп контактов, когда тумблер переключён в противоположную сторону, выводы 6 и 8 ОУ размыкаются, и резистор R10 включается в обратную связь ОУ (между его выходом и инвертирующим входом), а резистор R11 подключается к аналоговой «земле». В этом состоянии ОУ становится неинвертирующим усилителем с коэффициентом усиления, равным  $K_u = 1 + R10/R11 = 1 + 20\text{K}/5,6\text{K} = 4,57$ .

Теперь, резюмируя вышесказанное, имеем следующее. В первом положении тумблера П2 (рис. 2в), когда ОУ является повторителем с  $K_u = 1$ , получаем, что выходное напряжение ОУ может регулироваться переменным резистором РП от  $V_{out} = 0\text{ В}$  до  $V_{out} = \sim 0,212\text{ В}$  (действующее значение) с амплитудным значением  $V_{outAMP} = 0,212\text{ В} \times \sqrt{2} \approx 0,2998 \approx 0,3\text{ В}$  ( $\sqrt{2} \approx 1,414$ ). Этот режим работы генератора целесообразно использовать для настройки различного рода предварительных усилителей, когда источником

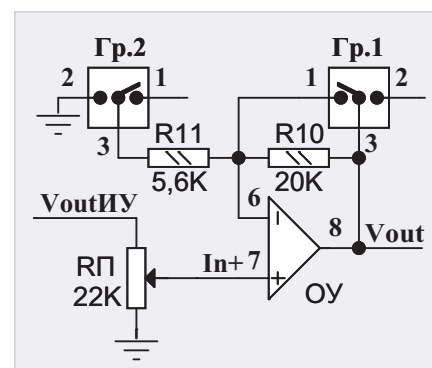


Рис. 3. Условная схема переключения коэффициента усиления ОУ

сигнала являются типовые аудиоустройства (проигрыватели CD, виниловых пластинок, аудиовыход телефона и т.п.) со стандартным для аудиосигнала выходным напряжением с амплитудным значением 0,3 В.

Во втором положении тумблера, когда  $K_u = 4,57$ , получаем, что выходное напряжение ОУ может регулироваться резистором РП от  $V_{out} = 0\text{ В}$  до  $V_{out} = \sim 0,212\text{ В} \times 4,57 = \sim 0,97\text{ В}$  (действующее значение) с амплитудным значением  $V_{outAMP} = 0,97\text{ В} \times \sqrt{2} \approx 1,37\text{ В}$ . Этот режим работы генератора обычно используют для настройки различных усилителей мощности звуковой частоты (УМЗЧ), источником сигнала для которых, как правило, является синусоидальный сигнал с максимальной амплитудой около 1 В.

Здесь может возникнуть вопрос: а зачем использован дополнительный ОУ, входящий в состав AD8231, коэффициент усиления которого переключается аппаратным способом – тумблером (рис. 3)? Ведь ИУ AD8231 оборудован встроенным средством переключения коэффициента усиления с помощью логических входов A0, A1 и A2 (рис. 1), позволяющих установить коэффициент усиления от 1 до 128 (в том числе и, например, 4), состояние которых можно менять, например, программным способом, если их подключить к МК. И почему эти входы заземлены, в связи с чем коэффициент усиления ИУ постоянно установлен и равен 1? Здесь дело в том, что в формуле (4), в которой присутствует этот коэффициент усиления ( $K_u$ ), скрыто некоторое лукавство. Оно заключается в том, что если  $K_u$  больше единицы (например, 2 и больше), то эта формула работает неадекватно. Эта неадекватность проявляется в том, что если  $(V_{in+} - V_{in-}) \times K_u$  (см. формулу (4)) больше 0,3 В, или, другими словами, если

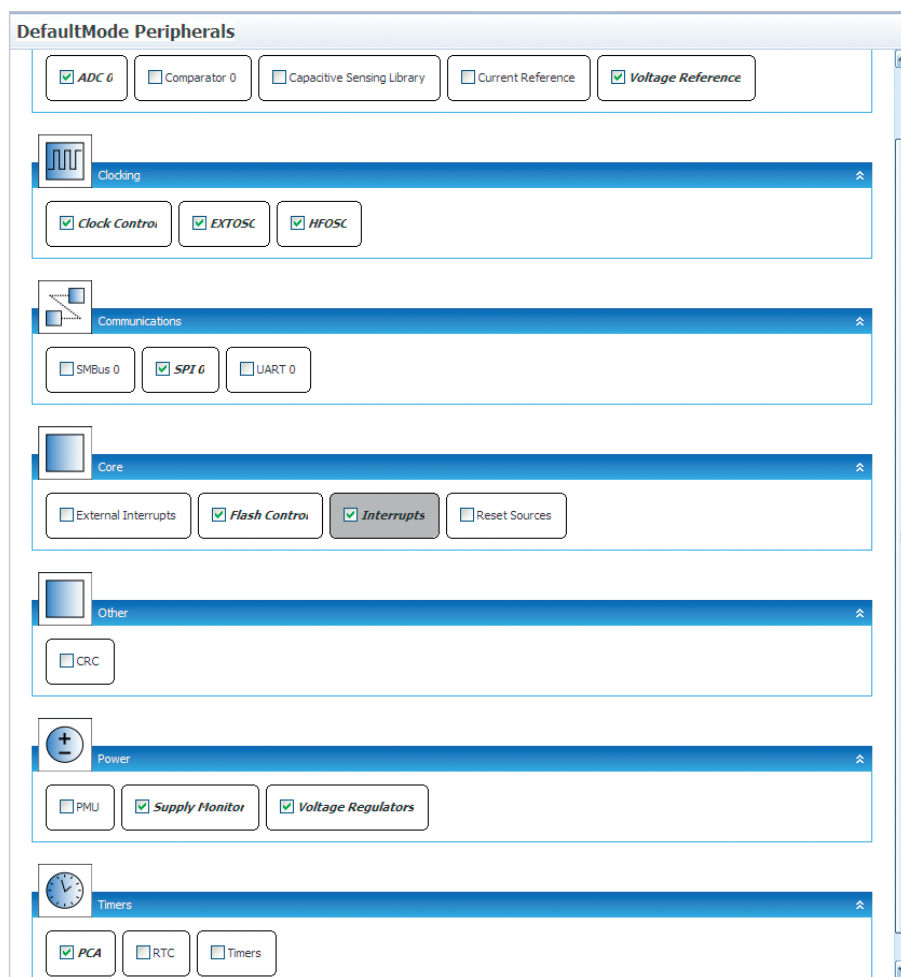


Рис. 4. Основное меню настройки устройств МК

мгновенное значение синусоидального сигнала опускается ниже  $-0,3$  В (относительно «земли»), а такое возможно, если  $K_u > 1$ , то нижняя часть синусоиды обрезается на уровне примерно  $-0,3$  В, и сигнал становится уже несинусоидальным. Но ведь в формуле (4) есть ещё член « $+V_{ref}$ », который равен  $+1,5$  В, и по формуле (4) он должен поднимать выходное напряжение на эти  $1,5$  В, что намного больше, чем  $0,3$  В. Хотя чисто математически это и верно, однако на практике, к сожалению, это не работает. Вот в этом и состоит лукавство формулы (4), которая (в данном случае) работает только тогда, когда  $K_u = 1$ .

Сигнал  $V_{out}$  и аналоговая «земля» также подаются на разъём X7 (рис. 1), к которому ответным разъёмом X7' подключён двухпроводный кабель, второй конец которого подключается к разъёму RCA красного цвета – RS-104 (X1, рис. 2д) и через тумблер sMTS103 (ПЗ, рис. 2д) – к разъёму RCA зелёного цвета – RS-104 (X2, рис. 2д). Разъём X1 служит для подключения выходного сигнала генератора к осциллографу для настройки, т.е. для получения сигнала необходимой

частоты и амплитуды, после чего тумблером ПЗ уже подключается к разъёму X2, к которому подключён кабель для тестируемого аудиоустройства.

Как видно из рис. 1, плата генератора не отличается особой сложностью, поэтому легко разводится и имеет размер всего  $22 \times 40$  мм (см. далее).

### Программные средства

Программа в уже готовом загрузочном формате EFM8SB10F8G\_A\_QFN20\_8.hex приведена в дополнительных материалах к статье на сайте журнала. Её можно запрограммировать в МК с помощью USB DEBUG адаптера (см. выше). Однако для тех, кто хочет написать свою программу, автор хотел бы поделиться некоторыми наиболее критическими её моментами.

Прежде всего, о настройках периферийных устройств МК.

В среде программирования Simplisity Studio v.4 в основном меню настройки периферийных устройств (рис. 4) галочками отмечены те устройства, которые необходимо настроить.

Вначале разрешаем встроенный внутренний ВЧ генератор (HFOSC)

частотой  $24,5$  МГц (рис. 5а), так как при включении питания он включается по умолчанию, и системная тактовая частота определяется именно им, а МК должен начать работать и выполнять команды программы, в том числе и те, которые переключают МК на внешний кварцевый резонатор частотой  $16\,777\,216$  Гц (рис. 5б) и устанавливают системную тактовую частоту SYSCLK, источником которой является внешний генератор (Clock Control, рис. 5в). Поэтому, если не разрешить внутренний генератор, то до этих команд МК просто «не дойдет», так как ему нечем будет выполнять команды программы, поскольку он не будет тактироваться. Также следует установить опцию Flash Control для работы с flash-памятью, т.е. определить, что для её чтения системная тактовая частота SYSCLK  $> 14$  МГц (рис. 5г).

В настройках интерфейса SPI (рис. 5д) необходимо выбрать частоту импульсов SCK ( $2,796$  МГц) почти в 2 раза ниже максимальной частоты ( $4$  МГц), на которой работает дисплей LCD5110 (для надёжности), поскольку DDS может работать на более высокой частоте, т.е. выбрать коэффициент деления SYSCLK Divider Coefficient, равный 2. Здесь необходимо отметить, что, хотя обмен информацией МК с DDS и дисплеем идёт по одному и тому же интерфейсу SPI, у этих устройств настройка этого интерфейса разная. В частности, если фаза (Clock Phase) у них одинаковая, т.е. данные тактируются первым же импульсом SCK (Data sample on first edge), то полярность разная. У дисплея состояние импульса SCK в отсутствие обмена (Clock Polarity) находится в низком уровне (Low in idle state), а у DDS – в высоком. Но поскольку обмен информацией МК с дисплеем происходит на порядок чаще, выбираем правильную полярность для дисплея (Low in idle state), а перед началом обмена с DDS уже программным способом придётся переключить полярность на обратную, а после окончания обмена – восстановить прежнюю (см. далее).

Далее необходимо разрешить в матрице соединений (CROSSBAR0) подключение сигналов SPI и системной тактовой частоты SYSCLK\_OUT (для тактирования DDS) к портам МК, поставив галочки в квадратах Clock/Data и Clock (рис. 5е).

Для работы АЦП (при измерении напряжения аккумулятора) выбираем внутренний источник опорного напря-

**Properties of HFOSC**

Property	Value
<b>High Frequency Oscillator Control</b>	
High Frequency Oscillator Frequency	24.5 MHz
Enable High Frequency Oscillator	Enabled
Enable Spread Spectrum	Enabled

**Properties of Flash Control**

Property	Value
<b>Flash Scale</b>	
Flash Read Timing One-Shot Bypass	SYSCLK determines flash read time (SYSCLK < 14 MHz)
One shot determines flash read time (SYSCLK < 14 MHz)	
SYSCLK determines flash read time (SYSCLK > 14 MHz)	

**Properties of Voltage Reference**

Property	Value
<b>Voltage Reference Control</b>	
Select Voltage Reference	1.65 V High Speed Ref...
Temperature Sensor	Disabled
Analog Ground Reference	GND pin

**Properties of ADC 0**

Property	Value
<b>View</b>	Advanced
<b>Control</b>	
Enable ADC	Enabled
Enable Burst Mode	Disabled
Start of Conversion	Write to 1 of ADBUSY
<b>Multiplexer Selection</b>	
Positive Input Selection	ADC0.1 (P0.1)
<b>Configuration</b>	
Resolution	10-bit
SARCLK (Actual)	4.194 MHz
SAR Clock Source	16.777 MHz
SAR Clock Divider	4 (0x4)
Enable 8-Bit Mode	Normal mode (10- or 12-...)
Gain Control	0.5x gain
Result Shift and Justify	Right justified
Track Mode	Normal tracking
<b>Throughput</b>	
Conversion Time	3.100 us
Minimum Tracking Time	1.500 us
Maximum Throughput	217.407 ksp/s
<b>Burst Mode Control</b>	
Burst Mode Repeat Count	1 conversion
<b>Power Control</b>	
Enable Low Power Mode	Disabled
<b>Accumulator Configuration</b>	
Enable 12-Bit Mode	Disabled
Enable Accumulate	Disabled
<b>Window Compare</b>	
Greater-Than Value	0 (0x0)
Less-Than Value	0 (0x0)

**Properties of EXTOSC**

Property	Value
<b>External Oscillator Control</b>	
External Oscillator Frequency	16.777 MHz
External Oscillator Frequency Coefficient	8.2 MHz < Crystal freq...
External Oscillator Mode	Crystal
External Crystal Frequency	16777216 (0x100000)

**Properties of SPI 0**

Property	Value
<b>View</b>	Advanced
<b>Control</b>	
SPI Enable	Enabled
Bus Mode (Master)	Master 3-wire mode
<b>Configuration</b>	
SPI Mode	Master
Clock Phase	Data sample on first edge
Clock Polarity	Low in idle state
<b>Clock Rate</b>	
SYSCLK	16.777 MHz
SPI Clock Frequency (Actual)	2.796 MHz
SYSCLK Divider Coefficient (SPI0)	2 (0x2)

**Properties of Clock Control**

Property	Value
<b>Clock Control</b>	
<b>Clock Select</b>	
SYSCLK	16777216
Clock Source Divider	SYSCLK / 1
Select Clock Source	External Oscillator

**Properties of P0.1**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Analog I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
<b>Capacitive Sense Settings</b>	
Capacitive Sense Input	Disabled
Average Touch Delta	1000 (0x3E8)
Active Threshold	70 (0x46)
Inactive Threshold	30 (0x1E)

**Properties of P0.4**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Digital OpenDrain I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
Label	

**Properties of P1.2**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of P1**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of PBCFG\_0**

Property	Value
<b>Port Config</b>	
<b>Settings</b>	
Disable Port I/O Weak Pullup	Pull-ups enabled
Enable Crossbar	Enabled

**Port I/O Mapping**

Property	Value
<b>DefaultMode: CROSSBAR0</b>	
SPI 0	<input checked="" type="checkbox"/> Clock / Data
SPI0_SCK	P0.7
SPI0_MISO	P1.0
SPI0_MOSI	P1.1
SPI0_NSS	
SMBus 0	<input type="checkbox"/> Clock / Data
SMBUS0_SDA	
SMBUS0_SCL	
Comparator 0	<input type="checkbox"/> Sync Output
CMP0_S	
<input type="checkbox"/> Async Output	
CMP0_A	
Clock Control	<input checked="" type="checkbox"/> Clock
SYSCLK_OUT	P1.2

**Properties of P0.1**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Analog I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
<b>Capacitive Sense Settings</b>	
Capacitive Sense Input	Disabled
Average Touch Delta	1000 (0x3E8)
Active Threshold	70 (0x46)
Inactive Threshold	30 (0x1E)

**Properties of P1.2**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of P1**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of PBCFG\_0**

Property	Value
<b>Port Config</b>	
<b>Settings</b>	
Disable Port I/O Weak Pullup	Pull-ups enabled
Enable Crossbar	Enabled

**Port I/O Mapping**

Property	Value
<b>DefaultMode: CROSSBAR0</b>	
SPI 0	<input checked="" type="checkbox"/> Clock / Data
SPI0_SCK	P0.7
SPI0_MISO	P1.0
SPI0_MOSI	P1.1
SPI0_NSS	
SMBus 0	<input type="checkbox"/> Clock / Data
SMBUS0_SDA	
SMBUS0_SCL	
Comparator 0	<input type="checkbox"/> Sync Output
CMP0_S	
<input type="checkbox"/> Async Output	
CMP0_A	
Clock Control	<input checked="" type="checkbox"/> Clock
SYSCLK_OUT	P1.2

**Properties of P0.1**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Analog I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
<b>Capacitive Sense Settings</b>	
Capacitive Sense Input	Disabled
Average Touch Delta	1000 (0x3E8)
Active Threshold	70 (0x46)
Inactive Threshold	30 (0x1E)

**Properties of P0.4**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Digital OpenDrain I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
Label	

**Properties of P1.2**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of P1**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of PBCFG\_0**

Property	Value
<b>Port Config</b>	
<b>Settings</b>	
Disable Port I/O Weak Pullup	Pull-ups enabled
Enable Crossbar	Enabled

**Port I/O Mapping**

Property	Value
<b>DefaultMode: CROSSBAR0</b>	
SPI 0	<input checked="" type="checkbox"/> Clock / Data
SPI0_SCK	P0.7
SPI0_MISO	P1.0
SPI0_MOSI	P1.1
SPI0_NSS	
SMBus 0	<input type="checkbox"/> Clock / Data
SMBUS0_SDA	
SMBUS0_SCL	
Comparator 0	<input type="checkbox"/> Sync Output
CMP0_S	
<input type="checkbox"/> Async Output	
CMP0_A	
Clock Control	<input checked="" type="checkbox"/> Clock
SYSCLK_OUT	P1.2

**Properties of P0.1**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Analog I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
<b>Capacitive Sense Settings</b>	
Capacitive Sense Input	Disabled
Average Touch Delta	1000 (0x3E8)
Active Threshold	70 (0x46)
Inactive Threshold	30 (0x1E)

**Properties of P0.4**

Property	Value
<b>Settings</b>	
Drive Strength	Low drive
IOMode	Digital OpenDrain I/O
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Skipped
Label	

**Properties of P1.2**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of P1**

Property	Value
<b>Settings</b>	
Drive Strength	High drive
IOMode	Digital Push-Pull Output
Latch	High
Mask Value	Ignored
Match Value	High
Skip	Not skipped
Label	

**Properties of PBCFG\_0**

Property	Value
<b>Port Config</b>	
<b>Settings</b>	
Disable Port I/O Weak Pullup	Pull-ups enabled
Enable Crossbar	Enabled

**Port I/O Mapping**

Property	Value
<b>DefaultMode: CROSSBAR0</b>	
SPI 0	<input checked="" type="checkbox"/> Clock / Data
SPI0_SCK	P0.7
SPI0_MISO	P1.0
SPI0_MOSI	P1.1
SPI0_NSS	
SMBus 0	<input type="checkbox"/> Clock / Data
SMBUS0_SDA	
SMBUS0_SCL	
Comparator 0	<input type="checkbox"/> Sync Output
CMP0_S	
<input type="checkbox"/> Async Output	
CMP0_A	
Clock Control	<input checked="" type="checkbox"/> Clock
SYSCLK_OUT	P1.2

Рис. 5. Настройка устройств МК: (а-г) – генераторов, (д-е) – SPI, (ж-з) – АЦП, (и-н) – портов, о – матрицы соединений

жения (Voltage Reference, рис. 5ж) 1,65 В и настраиваем работу АЦП (рис. 5з). Здесь основными опциями являются следующие. Разрешаем работу АЦП (Enable ADC

– enable), начало преобразования определяем с помощью записи 1 в регистр ADBUSY (Start of Conversion – Write to 1 of ADBUSY), входной сигнал для АЦП

подключаем к порту P1.0, устанавливаем коэффициент усиления входного усилителя (см. далее) на 0,5 (0.5x gain). Остальные опции АЦП – согласно рис. 5з.

Далее переключаемся в опцию настройки портов, где на экран выводится корпус МК со всеми портами ввода/вывода (рис. 5и). Порты с сигналами ADC\_IN (P0.1), XTAL1 (P0.2) и XTAL2(P0.3) определяем как аналоговые входы/выходы (Analog I/O). Пример для порта P0.1 – рис. 5к. Порты, к которым подключаются сигналы от кнопок (P0.4, P0.5), и порт для SPI0\_MISO (P1.0) определяем как цифровые входы (Digital OpenDraine I/O). Пример для порта P0.4 – рис. 5л. Остальные порты, кроме неиспользованных P0.0 и P1.7 (которые также настраиваем как цифровые входы), определяем как цифровые слаботочные (Low drive) выходы (Digital Push-Pull Output). Пример для порта P1.2 – рис. 5м. Здесь необходимо отметить, что при настройке системной тактовой частоты, сигнал которой (SYSClk\_OUT) выходит с порта P1.2, тактирует DDS и подключается к частотомеру относительно длинным кабелем, во время настройки частоты можно установить «сильноточным» (High drive – рис. 5н), так как в этом случае его фронты будут существенно круче, и поэтому частота будет более точно измеряться частотомером. После настройки этот порт необходимо переключить обратно на слаботочный (Low drive), так как DDS расположен близко к МК, и для его тактирования вполне достаточен слаботочный выход.

На порты с P0.0 по P0.6 необходимо обязательно установить опцию пропуска (Skip), отмеченную красными крестиками на рис. 5и.

В конце настройки устройств необходимо разрешить работу матрицы соединений (Enable Crossbar – enabled) – рис. 5о.

Остальные опции меню рис. 4, отмеченные галочками, в связи с простотой подробно не описаны, но их необходимо также установить следующим образом. В Interrupt – запретить все прерывания, Supply Monitor – разрешить, Voltage Regulators – разрешить, в PCA – запретить Watchdog Timer.

После проделанной описанной выше настройки устройств МК созданную конфигурацию необходимо записать на диск, нажав в левом верхнем углу экрана кнопку с двумя дискетами. В этом случае средой Simplisity Studio сгенерируется программа инициализации устройств на C51 с названием InitDevice.c, которая автоматически подключится к основной программе. Текст этой программы, если её распечатать, займет 7 страниц формата A4 шрифтом размера 10. Можно, конеч-

но, написать такую программу своими силами, но на это придётся потратить несколько дней достаточно кропотливой работы, так как самостоятельное написание такой программы достаточно сложно и не гарантировано от ошибок. Автоматическая генерация такой программы средой Simplisity Studio произойдёт всего за пару-тройку секунд, и, что главное, если вышеуказанные настройки сделаны верно, такая программа гарантирована от ошибок. Вот поэтому на рис. 4 и рис. 5 так подробно и описаны все настройки устройств МК.

Теперь по поводу критически важных особенностей некоторых подпрограмм.

Основой всех подпрограмм вывода информации по интерфейсу SPI служит простейшая подпрограмма вывода байта по этому интерфейсу:

```
// Вывод байта по SPI
void outspi(uint8_t byte)
{SPI0DAT = byte; //
Вывод байта по SPI
while (!SPI0CN0_SPIF); //
Ожидание окончания вывода байта
SPI0CN0_SPIF = 0; //
Сброс флага окончания передачи
}
```

На базе этой подпрограммы основаны подпрограммы вывода команд и данных в дисплей:

```
// Вывод команды по SPI
для дисплея
void outcmd(uint8_t byte) {
DC = 0;
CSD = 0;
outspi(byte);
CSD = 1;
}
```

```
// Вывод данных по SPI
для дисплея
void outdat(uint8_t byte) {
DC = 1;
CSD = 0;
outspi(byte);
CSD = 1;
}
```

Эти подпрограммы используются для инициализации и очистки дисплея с максимальной контрастностью, которая зависит не только от самой команды, определяющей контрастность, но и от команды, определяющей температурный коэффициент (и, как ни странно, в большей степени), о чём часто забывают, а для некоторых этот факт вообще неизвестен. Подпрограмма инициализации и очистки дисплея приведена ниже.

```
// Подпрограмма инициализации
и очистки ЖКИ
void CLS(void) {
uint16_t i;
RES = 1;
DEL3MS();
}
```

```
RES = 0;
DEL3MS();
RES = 1;
DEL3MS();
outcmd(0x21); //
Режим внешних команд.
outcmd(0x04); // temperature
coefficient 0 - высокая
контрастность
outcmd(0xC0); // Set LCD
Vop(Contrast = 80h + 40h).
outcmd(0x13); //
LCD bias mode 1:48
outcmd(0x20); //
Режим стандартных (внутр.)
команд, горизонт. адресация
outcmd(0x0C); //
Нормальный режим отображения
(черным по белому)
outcmd(0x80); // x=0;
outcmd(0x40); // y=0;
for (i = 0; i < 504; i++) {
outdat(0x00); //Очистить lcd
}
outcmd(0x22); // Стандартные
команды, вертикальная адресация
}
```

О выводе символов в дисплей Nokia-5110 подробно написано в статье автора [3], и, чтобы не повторяться, на взгляд автора, описывать этот вывод здесь не имеет смысла. Для цифр использован шрифт Clarendon жирный, размер символов 14×16 точек.

Теперь по поводу вывода информации в DDS.

Цифровое значение частоты выводится в DDS по SPI двумя двухбайтными (16-разрядными) словами. Но поскольку интерфейс SPI однобайтный, имеет смысл сделать следующее совмещение (объединение) одного двухбайтного числа и массива из двух однобайтных:

```
//-----
union {
uint16_t US; // U.US[0]- Ст.6.
uint8_t UB[1]; // U.UB[1]-
Мл.6.
} U; // U.US - 2-байтное
uint16_t число.
//-----
```

В этом совмещении 2-байтное число U.US и два элемента двухбайтного массива UB[1]: U.UB[0] (старший байт) и U.UB[1] (младший байт) перераспределяют одно и то же место в памяти МК. Это означает, что два однобайтных числа U.UB[0] и U.UB[1] полностью определяют двухбайтное число U.US, и наоборот, число U.US полностью определяет два числа U.UB[0] и U.UB[1]. Другими словами, если задано двухбайтное (uint16\_t) число U.US, то однобайтные (uint8\_t) числа U.UB[0] и U.UB[1] автоматически определяются как его старший и младший байты соответственно. Верно также и обратное.

Здесь необходимо напомнить (см. выше), что перед тем как выводить информацию по SPI в DDS, необходимо

в регистре SPI0CFG переключить бит полярности с «дисплейного» (заданного в начальной конфигурации – рис. 5д) на «DDS-й», а после окончания вывода восстановить первоначальное его (бита) значение, иначе дисплей работать не будет.

Подпрограмма вывода двухбайтного числа в DDS по SPI с учётом вышеуказанного совмещения, переключения бита полярности и вышеописанной подпрограммы вывода байта по SPI `outspi()` приведена ниже.

```
//Вывод 16-разрядного слова
в DDS
void outspi16(uint16_t wor) {
    U.US = wor;
    SPI0CFG = SPI0CFG | 0x10; //
    IDLE_HIGN для AD9837
    CSAD = 0; // CSAD=0.
    outspi(U.UB[0]); // Ст.б.
    outspi(U.UB[1]); // Мл.б.
    CSAD = 1; // CSAD=1.
    SPI0CFG = SPI0CFG & 0xef; //
    IDLE_LOW - Для дисплея
}
```

Теперь о том, как вывести в DDS само значение частоты. Как указано в справочном листке (datasheet) на AD9837, численное значение частоты представляется 28-разрядным кодом и вводится в DDS двумя двухбайтными (16-разрядными) словами. В первом слове в 14 его младших битах содержится 14 младших бит 28-разрядного кода, во втором, в его 14 младших битах, – 14 старших бит кода. А в двух старших битах обоих слов содержится управляющий код. Если, например, этот управляющий код равен 01b (01<sub>2</sub>), то DDS будет воспринимать его следующим образом. Во-первых, будет выводиться синусоида, во-вторых, численное значение частоты будет вводиться в нулевой регистр частоты (FREG0), а численное значение фазы (как правило, нулевое значение) – в нулевой регистр фазы (PHASE0). Есть ещё FREG1 и PHASE1, но их использования не требуется.

Пусть задано 32-разрядное (`uint32_t`) значение частоты F. Каким образом сформировать из этого числа два двухбайтных слова с указанным управляющим кодом для ввода в DDS? Сделать это достаточно просто.

Во-первых, определим это число F как 32-разрядное:

```
uint32_t F; // частота в Гц
```

Во-вторых, сделаем ещё одно совмещение:

```
union {
    uint32_t FRL; // FR.FRS[0]- Ст.
    uint16_t слово.
    uint16_t FRS[1]; // FR.FRS[1]-
    Мл. uint16_t слово.
} FR; // FR.FRL - 4-байтное
uint32_t число.
```

В этом совмещении 4-байтное (`uint32_t`) число FR.FRL и два двухбайтных (`uint16_t`) числа FR.FRS[0] и FR.FRS[1] двухэлементного массива FRS[1] также перераспределяют одно и то же место в памяти МК и являются соответственно старшим и младшим словами числа FR.FRL.

Теперь, как указывалось выше (см. формулу (3)), заданное число F умножим на 16, а результат запишем в 4-байтное число FR.FRL вышеуказанного совмещения, т.е.  $FR.FRL = F \times 16$ . Если теперь сдвинуть 32-разрядное число FR.FRL влево на 2 бита, то в младшем слове FR.FRS[1] получим 14 младших бит 28-разрядного кода, только сдвинутых влево на 2 бита, а в старшем слове FR.FRS[0] – полностью 14 старших бит этого кода. Если сдвинуть младшее слово FR.FRS[1] вправо на 2 бита (т.е. восстановить сдвинутое влево значение), то в его 14 младших битах как раз и будет содержаться 14 младших бит 28-разрядного кода. Таким образом, оба слова FR.FRS[0] и FR.FRS[1] для вывода теперь подготовлены и остаётся только добавить к ним 2 старших управляющих бита 01b (или 0x4000). Эту «добавку» можно, например, просто прибавить, но проще каждое из слов логически сложить (|) с числом 0x4000.

Для вывода нулевого значения фазы в нулевой регистр фазы (PHASE0) требуется подать команду 0xC000 (см. справочный листок AD9837).

Но прежде чем выводить в регистр частоты (FREG0) эти два двухбайтных слова и нулевое значение фазы в нулевой регистр фазы (PHASE0), DDS нужно сбросить (т.е. установить бит RESET'a, подав команду 0x2108 – см. справочный листок на AD9837), а после вывода частоты и фазы – запустить, т.е. обнулить бит сброса, подав команду 0x2008.

Несмотря на такое пространное объяснение, подпрограмма вывода в DDS заданной частоты занимает всего 9 строчек кода на C51:

```
//Вывод частоты в DDS
FR.FRL = F * 16;
FR.FRL = FR.FRL << 2; //
Сдвиг всего числа uint32_t влево
на 2 бита
FR.FRS[0] = (FR.FRS[0] |
0x4000); //Добавка 2-х ст.бит 01b
к ст. слову для Freq 0
FR.FRS[1] = ((FR.FRS[1]) >> 2)
| 0x4000; //Сдвиг мл. слова
вправо на 2 бита (восст.)
//и добавка 2-х ст.бит 01b к
мл. слову
outspi16(0x2108); // Сброс DDS
(стоп)
outspi16(FR.FRS[1]); //
вывод мл. слова частоты в FREG_0
```

```
outspi16(FR.FRS[0]); //
вывод ст. слова частоты в FREG_0
outspi16(0xC000); //
WRITE_TO_PHASE0_REG 0xC000
outspi16(0x2008); //
Выход из сброса (запуск)
```

Таким образом, программирование DDS примитивно просто, а приведённая выше подпрограмма занимает ничтожное место в программной памяти МК.

Подпрограмма для измерения напряжения аккумулятора и вывода на дисплей состояния его разряженности также проста. Она десятикратно измеряет напряжение, поступающее на вход ADC0.1 (P0.1) МК, осредняет его, сравнивает с порогами (3,2 В; 3,5 В и 3,8 В) и в зависимости от этого напряжения выводит на дисплей рисунок аккумулятора с тем или иным количеством сегментов. С необходимыми комментариями эта подпрограмма приведена ниже. Поскольку подпрограмма взята из одного из примеров работы АЦП, приведённых в среде Simplicity Studio, и адаптирована под настоящую задачу, некоторые комментарии оставлены в оригинальном виде. Вывод на дисплей опущен, поскольку, как указывалось выше, более подробно о таком выводе написано в [3]. Учитывая заданную конфигурацию (см. рис. 5ж и рис. 5з), её достаточно просто понять:

```
void BAT() {
    uint32_t accumulator = 0; //
    accumulator for averaging
    uint8_t measurements; // = 10;
    // measurement counter
    uint32_t result = 0;
    uint32_t mV; // measured
    voltage in mV
    for (measurements =
0; measurements < 10;
measurements++) {
        // clear ADC0 conv. complete
        flag
        ADC0CN0_ADINT = 0;
        // Start a conversion by
        setting ADBUSY
        ADC0CN0_ADBUSY = 1;
        while (!ADC0CN0_ADINT); //wait
        conversionComplete
        accumulator += ADC0; // ADC0
    }
    result = accumulator / 10;

    // Vref (mV)
    // measurement (mV) = -----
    ----- * result (bits)
    // (2^10)-1 (bits)
    mV = result * 6600 / 1023;
    //2*1650=3300 и *2 = 6600, так
    как 2 резистора
    // и повторитель на ОРА333;0.5x
    gain => 2*1650
    if (mV < 3200) {
        //вывод пустого аккумулятора
    }
    if ((mV >= 3200) & (mV <=
3500)) {
        //вывод одного сегмента
    }
    if ((mV > 3500) & (mV <= 3800))
    {
        //вывод двух сегментов
    }
    if (mV > 3800) {
        //вывод трех сегментов
    }
}
```



Теперь по поводу того, как работает подпрограмма, определяющая состояние кнопок.

Во-первых, как можно увидеть из схем рис. 1 и рис. 2, сигналы от кнопок, очищенные от дребезга, поступают на порты P0.4 и P0.5 МК, для чего в основной программе сделаны следующие назначения:

```
sbit KN1 = P0 ^ 4; //
Правая кнопка
sbit KN2 = P0 ^ 5; //
Левая кнопка
```

Отсутствие дребезга позволяет в подпрограмме, связанной с кнопками, легко определять их состояние всего двумя простыми командами, которые выполняются в бесконечном цикле (**while** (1)), выход из которого (**break**) осуществляется при нажатии и отпуске соответствующей кнопки:

```
while (1) {
  if (!KN1) { // Ждем нажатия
кнопки KN1
    while (!KN1); // Ждем отпущения
кнопки KN1
    ... // Вывод на дисплей цифр и
запуск/стоп
    break;
  }
  if (!KN2) { // Ждем нажатия
кнопки KN2
    while (!KN2); // Ждем отпущения
кнопки KN2
    ... // Вывод на дисплей режима
    break;
  }
}
```

Вывод на дисплей, как указывалось выше, опущен.

После трансляции всей программы в среде Simplisity Studio внизу экрана появляется следующее сообщение:

```
Program Size: data=38.0 xdata=0
const=672 code=1727
LX51 RUN COMPLETE. 0
WARNING(S), 0 ERROR(S)
Finished building target:
EFM8S8B10F8G-A-QFN20_8.omf
```

Из этого сообщения следует, что программа использует менее трети (всего 38 байт) внутренней оперативной памяти с прямой адресацией (data=38.0), размер которой 128 байт, внешняя дополнительная оперативная память с косвенной адресацией размером 512 байт не используется вообще (xdata=0), размер кодовой части программы составляет 1727 байт (code=1727) плюс константы 672 байта (const=672) в сумме составляют  $1727 + 672 = 2399$  байт  $\approx 2,34$  КБ, т.е. менее 30% от максимального размера программной памяти для этого МК, равного 8 КБ. Программа использует так называемую small – model, где все данные располагаются

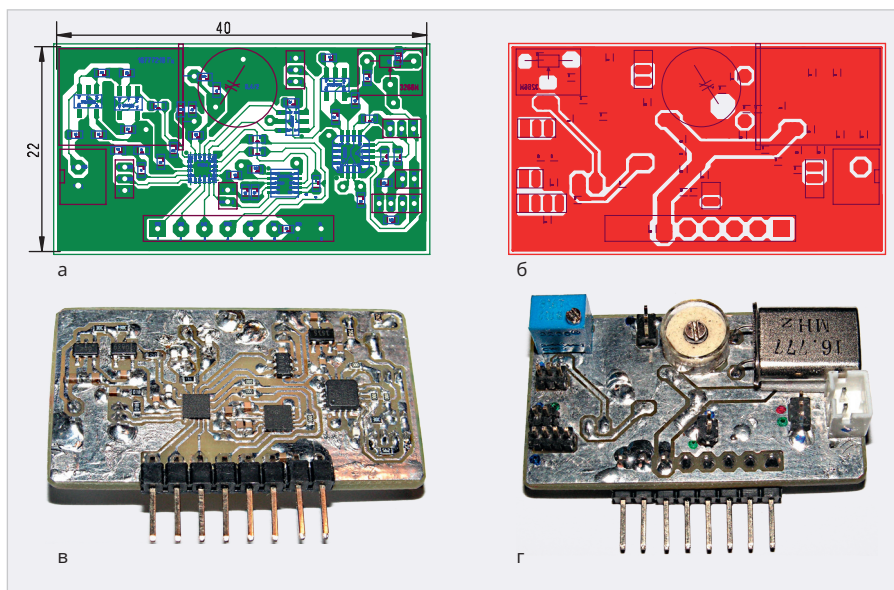


Рис. 6. Разводка и внешний вид платы прибора: а, в – вид со стороны расположения компонентов для поверхностного монтажа; б, г – вид с обратной стороны

в памяти с прямой адресацией (data), и которая работает наиболее быстро.

### Разводка и внешний вид платы генератора

Разводка платы устройства сделана автором с помощью программы SprintLayout v.6. Файл разводки в формате \*.lay6 приведён в дополнительных материалах к статье на сайте журнала. Из рисунков разведённой платы и её внешнего вида (рис. 6) можно заключить, что её разводка достаточно проста, а сама плата имеет размер всего 22×40 мм.

Все отверстия на разводке (кроме отверстий для подстроечного конденсатора и подстроечного резистора) – это переходные отверстия со слоя на слой. Если через них проходят штыри разъемов и выводы кварцевого резонатора, то их следует пропаять с двух сторон платы. Во все остальные переходные отверстия следует вставить тонкий лужёный медный провод и также пропаять его с двух сторон платы. Эта процедура позволяет не использовать металлизацию отверстий, технология которой в домашних условиях весьма проблематична, а потому неприемлема.

### Конструкция прибора

Генератор расположен в корпусе от распределительной коробки для электротрики размером 75×75×30 мм (рис. 7). В верхней части корпуса (рис. 7г) для дисплея прорезано окно, а сама плата дисплея приклеена к внутренней поверхности корпуса пористой лентой с двусторонним липким слоем. Плата прибора с одной стороны вставлена в

ответный разъем платы дисплея, а с другой – приклеена к ней пористой лентой с двусторонним липким слоем. Такой же лентой к внутренней поверхности корпуса приклеен аккумулятор. В верхней части корпуса просверлены отверстия, через которые тумблеры, переменный резистор и кнопки укреплены соответствующими гайками, а ручки управления выведены на лицевую поверхность корпуса (рис. 7в, рис. 7г). Два разъёма RCA красного и зелёного цвета и гнездо 2,5 моно через отверстия на боковых стенках корпуса прикручены к нему соответствующими гайками (рис. 7а, рис. 7в). К внутренней поверхности стенок корпуса дихлорметаном приклеены Г-образные пластиковые уголки, в которых просверлены отверстия и нарезана резьба М2. К этим уголкам винтами М2 с головками впотай прикручена крышка корпуса, в которой в соответствующих местах просверлены и раззенкованы отверстия диаметром 2 мм (рис. 7б). Чтобы корпус не скользил, к внешней поверхности крышки пористой лентой с двусторонним липким слоем приклеены 4 резиновые самоклеющиеся ножки. Под две передние ножки подложены пластиковые шайбы (они красного цвета на рис. 7б, рис. 7в). Это сделано для того, чтобы плоскость дисплея отклонить на небольшой угол в сторону пользователя, поскольку в этом случае контраст изображения на дисплее максимален, а потому само изображение наиболее читабельно.

Левый верхний тумблер (рис. 7г) предназначен для включения питания. В его верхнем положении пита-

ние включено, в нижнем выключено. Положение левого нижнего тумблера определяет коэффициент усиления усилителя и соответственно максимально возможное напряжение синусоидального сигнала генератора, зависящее от положения ручки управления переменным резистором. В нижнем положении этого тумблера коэффициент усиления равен 1, а максимальная амплитуда выходного напряжения генератора равна чуть более 0,3 В. В верхнем положении коэффициент усиления, как указывалось выше, равен 4,57, а максимальная амплитуда около 1,4 В. Правый тумблер подключает выходное напряжение генератора к зелёному разъёму RCA (рис. 7в). В верхнем положении тумблера напряжение подключено, в нижнем отключено. Ручка потенциометра (РП) регулирует выходное напряжение генератора от нуля до максимума. При повороте ручки против часовой стрелки до упора выходное напряжение нулевое, при повороте по часовой стрелке до упора – оно максимальное. Левая кнопка предназначена для ввода цифр числа частоты генератора и для его запуска и остановки. Правая определяет режим работы: набор числа или запуск/стоп генератора (см. далее в разделе о порядке работы с генератором).

## Настройка генератора

Настройка генератора состоит из двух пунктов: настройки нуля выходного напряжения генератора в режиме сброса с помощью подстроечного резистора R6 на рис. 1 и настройки тактовой частоты 16 777 216 Гц подстроечным конденсатором C10, рис. 1.

Вначале о настройке нуля выходного напряжения.

Квыходному разъёму RCA X1 на рис. 2д (разъём красного цвета на рис. 7в) необходимо подключить цифровой вольтметр или цифровой тестер. Переключить диапазон измерения постоянного напряжения (DC) в тестере на минимальное значение в милливольтках. Включить прибор, переключить левый нижний тумблер в нижнее положение (единичный коэффициент усиления) и, вращая шлиц подстроечного резистора отвёрткой, желательно с керамическим наконечником, в ту или иную сторону, добиваются нулевого показания тестера. Автор использовал тестер на милливольтном диапазоне с одним разрядом после запятой. Показания тестера при правильной настройке составили 000,0 мВ. Далее переключить тум-

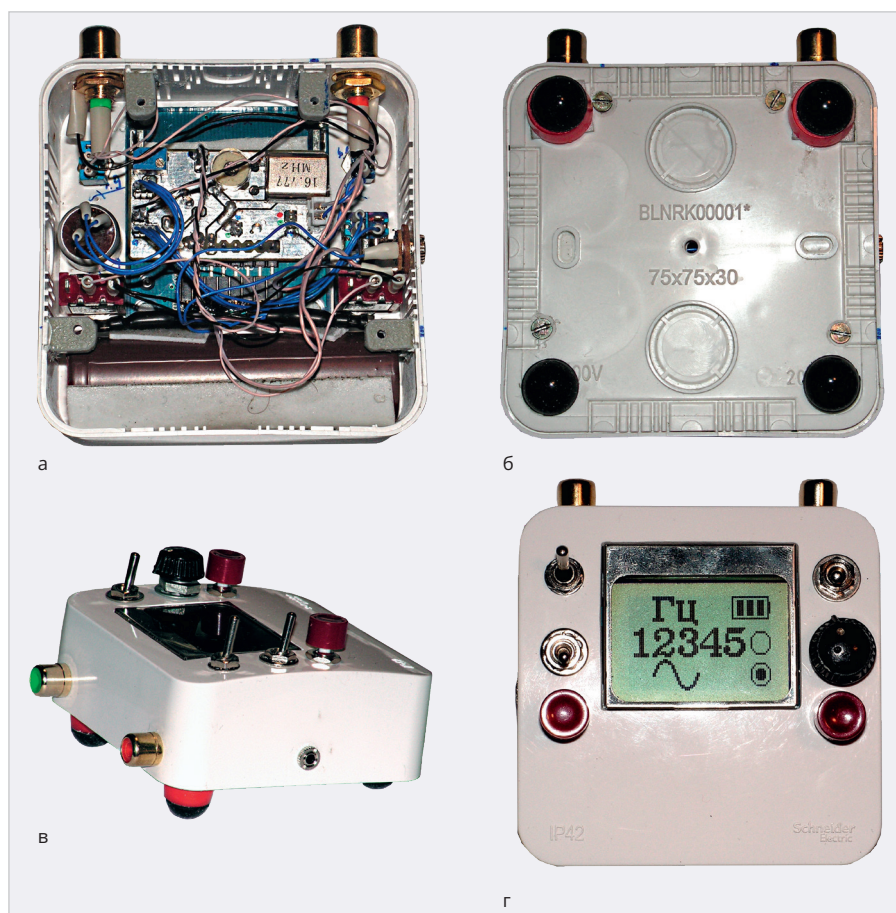


Рис. 7. Конструкция и внешний вид прибора: а – вид в открытом корпусе без крышки, б – вид снизу, в – вид сбоку, г – вид сверху

блер в верхнее положение (коэффициент усиления 4,57). Подстроить ноль, вращая шлиц подстроечного резистора до показания тестера, равного 000,0 мВ. Если после этого переключить тумблер в нижнее положение, и ноль в показаниях тестера останется, то настройка нуля считается законченной.

Теперь о настройке тактовой частоты МК, а точнее – о настройке частоты кварцевого резонатора.

Здесь следует сделать некоторое отступление относительно ёмкости конденсаторов C9 и C11 на рис. 1. Как правило, если обозначить эти ёмкости как  $C_g$ , нагрузочную ёмкость кварцевого резонатора, указанную в его справочном листке (datasheet), как  $C_L$ , а паразитную ёмкость как  $C_s$  (это обычно 3–5 пФ), то для расчёта  $C_g$  используется следующая эмпирическая формула:

$$C_g = 2 \times (C_L - C_s). \quad (7)$$

Если, например,  $C_L = 25$  пФ (как в кварцевом резонаторе, применённом автором), то по формуле (7)  $C_g = 2 \times (25 - 3) = 44$  пФ. В связи с этим на рис. 1 ёмкости C9 и C11 по 43 пФ, а добавочные ёмкости C9' и C11' – по 2 пФ. Кроме того, ёмкость подстро-

ечного конденсатора C10 варьируется в пределах 0,4–2 пФ.

Возвращаемся к прерванной последовательности изложения.

Подключить частотомер к прибору кабелем, один конец которого подключить ответным разъёмом к разъёму X5 на рис. 1, а второй, с разъёмом BNC, – к частотному входу частотомера. Автор использовал достаточно прецизионный частотомер ЧЗ-57. Включить прибор и, вращая шлиц подстроечного конденсатора C10 отвёрткой, желательно с керамическим наконечником, добиться показания частоты частотомера, равной 16 777 216 Гц (рис. 8). В частотомере ЧЗ-57 имеются несколько диапазонов времени T, при которых измеряется частота. В частности, есть 3 диапазона T: T = 0,1 с, T = 1 с и T = 10 с. На диапазоне T = 0,1 с показание частотомера должно равняться «16777,22» кГц (рис. 8а). На диапазоне T = 1 с, в связи с ограниченным количеством цифр числа частоты индикатора в данном частотомере, левая цифра (единица) исчезает (она неинтересна, поскольку её значение уже известно), зато появляется 3-я цифра после запятой, и показания частотомера становятся «6777,216» кГц



Рис. 8. Настройка кварцевого резонатора на частоту 16 777 216 Гц, а –  $T = 0,1$  с; б –  $T = 1$  с; в, г –  $T = 10$  с

(рис. 8б). В принципе, на этом можно и закончить настройку, однако, потратив еще немного времени и переключив диапазон на  $T = 10$  с, можно добиться показаний «777,2161» кГц (рис. 8в). Потратив ещё немного времени на этом диапазоне, можно вообще частоту настроить «в ноль», т.е. получить «777,2160» кГц (рис. 8г), хотя, конечно, такая точность совсем не обязательна. На этом настройка тактовой частоты заканчивается.

### Порядок работы

При включении прибора на дисплее появляется изображение, показанное на рис. 9а. Курсор устанавливается под первой цифрой, в данном случае это «0», а справа от цифр появляется так называемая радиокнопка, которая в активном состоянии индицируется чёрным кружком внутри. Предположим, требуется набрать частоту 12 345 Гц. Нажимаем и отпускаем левую кнопку, и левая цифра становится равной «1» (рис. 9б). Далее нажимаем правую кнопку, и курсор

передвигается на следующую цифру. Нажимаем и отпускаем левую кнопку 2 раза, и на второй позиции появляется цифра «2» с курсором под ней (рис. 9в). Таким же способом набираем остальные цифры «12345» (рис. 9г). После того как набрана последняя цифра «5», нажимаем правую кнопку. Радиокнопка у цифр становится неактивной (чёрный кружок пропадает), и справа внизу появляется новая радиокнопка в активном состоянии, сигнализирующая, что генератор переключён в режим запуска/стопа. В данном конкретном случае он в режиме стопа, о чём сигнализирует длинная черта слева от радиокнопки (рис. 9д). Нажимаем и отпускаем левую кнопку. В этом случае генератор запускается, о чём свидетельствует значок синусоиды, появляющийся вместо длинной черты, а в верхнем правом углу дисплея появляется рисунок аккумулятора с тремя сегментами, сигнализирующими о том, что аккумулятор полностью заряжен (рис. 9е). Этот режим работы генератора является

наиболее энергопотребляющим, поэтому состояние заряженности аккумулятора проверяется именно в этом режиме. Если ещё раз нажать и отпустить левую кнопку, то генератор остановится, и на экране опять появится длинная черта (как на рис. 9д). Таким образом, нажимая и отпуская левую кнопку, можно запускать и останавливать генератор с набранным числовым значением частоты сколько угодно раз. Но если в этом режиме нажать и отпустить правую кнопку, то генератор, во-первых, остановится, а во-вторых, переключится в режим набора новой частоты (как на рис. 9а). При этом рисунок аккумулятора в правом верхнем углу останется. Как видно из вышеописанного, управление генератором достаточно просто.

### Зарядка аккумулятора

Как уже упоминалось выше, для зарядки аккумулятора используется специальная плата на базе микросхемы TP4056, оборудованная разъёмом MicroUSB для подключения к ней устройства для зарядки телефона с выходным напряжением 5 В. К выходным контактам платы припаян двухпроводный кабель, а на другой его конец – аудиоразъём – штекер моно 2,5 мм (рис. 10). Провода в месте пайки кабеля к плате укреплены каплей термоклея. Иначе от частого изгибания кабеля провода в месте пайки могут отломиться. Этот штекер вставляется в ответный разъём – розетку (Хзар, рис. 2а), расположенную на боковой поверхности прибора (рис. 7в), а разъём microUSB от устройства зарядки телефона – к плате. При зарядке аккумулятора питание прибора должно быть выключено, а зарядка для телефона подключена к сети. При правильном подключении на плате включится красный светодиод, и начнется зарядка аккумулятора. По завершении (когда аккумулятор зарядится до напряжения 4,2 В) красный светодиод погаснет, и включится синий, сигнализируя о конце зарядки (рис. 10). Здесь следует заметить, что на плате с TP4056 по умолчанию установлено максимальное значение зарядного тока в 1 А. Такой ток обеспечивают далеко не все зарядные устройства для телефона. Кроме того, при токе 1 А плата с TP4056 достаточно сильно нагревается и может выйти из строя от перегрева. Поэтому, если использовать зарядку для телефона (или, например, для зарядки аккумуляторов IQOS), обеспечивающую ток 1 А (или более), то плату необходимо установить на радиатор.

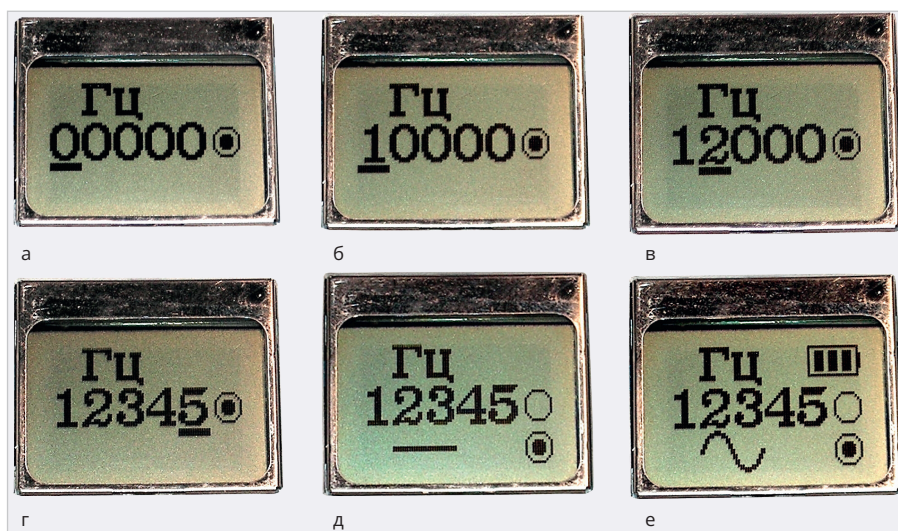


Рис. 9. Ввод частоты и запуск прибора

Автор использовал игольчатый радиатор с площадью поверхности около  $70 \text{ см}^2$ . Плата приклеена к радиатору гибкой термопрокладкой с двусторонним липким слоем (она белого цвета на рис. 10). Если же зарядка для телефона обеспечивает ток не более  $0,5 \text{ А}$ , то радиатор не нужен, так как в этом случае нагрев платы небольшой; правда, время зарядки аккумулятора будет существенно больше.

## Результаты работы генератора

О результатах работы генератора свидетельствует рис. 11, где приведены показания дисплея с набранной частотой и показания частотомера – с измеренной. На рис. 11б дополнительно показана осциллограмма сигнала аналогового осциллографа, к которому подключён сигнал от прибора параллельно частотомеру. О степени синусоидальности сигнала генератора также можно судить по спектрам, снятым вместе с осциллограммами цифровым осциллографом и опубликованным в [1] (поэтому, чтобы не повторяться, они не приводятся). Как следует из анализа спектров, все они представляют собой единственный пик на соответствующей частоте. Никаких дополнительных лепестков рядом с основным пиком, которые свидетельствуют о несинусоидальности сигнала, а также никаких дополнительных пиков на других частотах, кроме основной, на спектрах не обнаружено. Это свидетельствует о том, что все сигналы генератора, независимо от частоты, строго синусоидальны.

## Заключение

Применение малопотребляющих (с током потребления до единиц мА) DDS AD9837B, МК EFM8SB10, ИУ AD8231, дисплея Nokia-5110 и микропотребляющих (с током до  $20 \text{ мкА}$ ) ОУ ОРА333, стабилизаторов TPS78230 ( $3 \text{ В}$ ), STLQ15M15R ( $1,5 \text{ В}$ ) и подавителя дребезга контактов кнопок MAX6817 позволило сконструировать недорогой генератор синусоидальных сигналов с питанием от аккумулятора 18650-HG2 с напряжением  $3,6 \text{ В}$ , ёмкостью  $3 \text{ А}\cdot\text{ч}$ , обеспечивающим непрерывную работу прибора до  $200$  часов без подзарядки. Отсутствие сетевого источника питания гарантирует чистоту выходных сигналов генератора, свободных от сетевых или ВЧ-помех. Простота набора частоты с помощью всего двух кнопок, высокий контраст изображения на дисплее Nokia-5110 способствовали удобству работы с прибором, а его небольшой корпус ( $75 \times 75 \times 30 \text{ мм}$ ) и вес дали возмож-

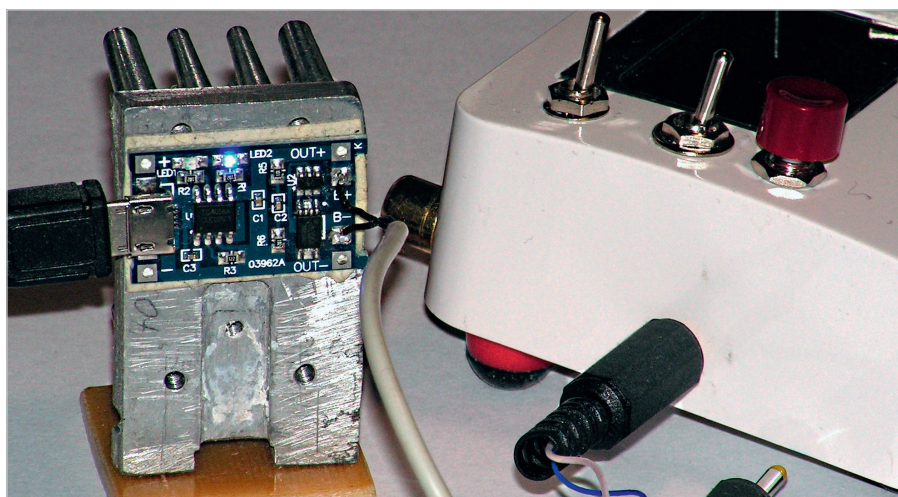


Рис. 10. Процесс зарядки аккумулятора генератора

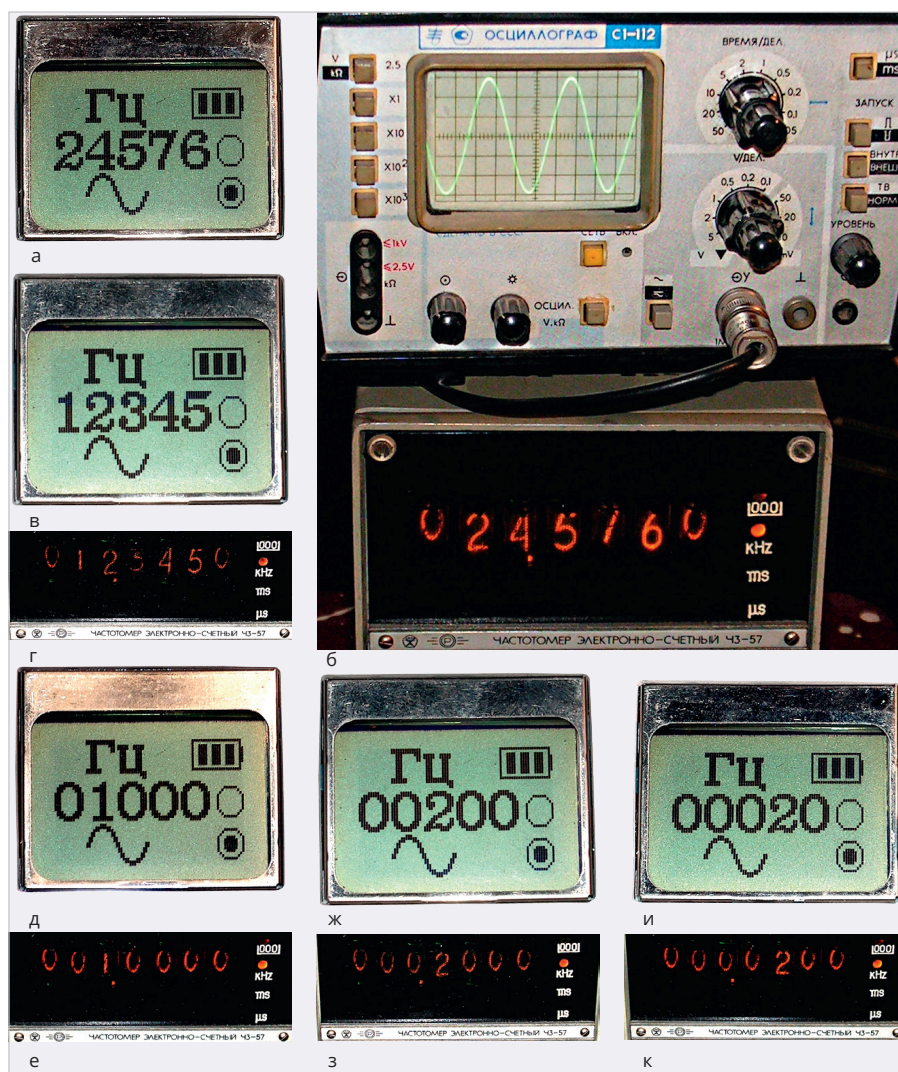


Рис. 11. Результаты работы генератора при выходных частотах: а, б –  $24,576 \text{ кГц}$ ; в, г –  $12,345 \text{ кГц}$ ; д, е –  $1000 \text{ Гц}$ ; ж, з –  $200 \text{ Гц}$ ; и, к –  $20 \text{ Гц}$

ность брать его с собой для работы, как говорят, «в полевых условиях».

## Литература

1. Кузьминов А. Низкочастотный генератор на основе синтезатора частоты AD9833 // Современная электроника. 2013. № 8, № 9.
2. Кузьминов А.Ю. Связь между компьютером и микроконтроллером. Современные аппаратные и программные средства. М.: Перо, 2018.
3. Кузьминов А. Миллиомметр с графическим LCD Nokia-5110 // Современная электроника. 2021. № 4.