

# Низкочастотный генератор на основе синтезатора частоты AD9833

## Часть 2. Программные средства и результаты испытаний

Алексей Кузьминов (г. Москва)

**Во второй части статьи описываются программные средства и результаты испытаний низкочастотного генератора. Приведена тестовая программа для микроконтроллера C8051F330, который, в свою очередь, программирует DDS для вывода сигнала нужной частоты.**

В первой части статьи (СоЭл № 8, 2013, С. 58–65) приведены аппаратные средства генератора – принципиальные схемы и разводка печатной платы генератора и платы подключённого к нему графического дисплея марки MI6448, сконструированного с применением светодиодов на органической основе (OLED).

Во второй части подробно описываются программные средства, применённые в генераторе, и результаты его испытаний.

Результаты работы генератора иллюстрируются осциллограммами и спектрами, снятыми аналоговым и цифровым осциллографами, а также спектрами, рассчитанными программой спектрального анализа сигнала генератора, подключённого к линейному входу звуковой карты компьютера.

### ПРОГРАММНЫЕ СРЕДСТВА

Программные средства для микроконтроллера (МК), входящего в состав генератора, условно можно разделить на два типа. Первый – это программирование дисплея, ввод нужной частоты с помощью кнопок и расчёт необходимой кодовой последовательности, предназначенной для программирования синтезатора частоты. Второй – непосредственное программирование синтезатора.

Первый тип программного обеспечения (ПО), по мнению автора, имеет довольно широкий спектр приложений, необязательно использующий именно DDS. Помимо этого к генератору можно подключить и другие дисплеи, например, ЖКИ или светодиодные «восьмерки». Для ввода частоты также необязательно использовать кнопки. Кроме того, данный тип ПО

не совсем соответствует затронутой в статье теме, которая больше тяготеет к практической электронике.

В настоящее время многие производители программируемых микросхем и микроконтроллеров предоставляют разработчикам ПО, которое максимально автоматизировано и позволяет, особенно не вдаваясь в детали, быстро запрограммировать синтезатор или МК и, тем самым, оперативно оценить его работоспособность.

Компания Analog Devices предлагает бесплатное ПО Device Configuration Assistant, позволяющее оперативно получить коды загрузки в DDS, чтобы моментально заставить его работать. Эта программа предназначена, в том числе, и для программирования микросхемы AD9833.

Компания Silicon Laboratories также стала предоставлять бесплатное ПО, которое позволяет максимально автоматизировать процесс программирования МК (в основном это касается функций его инициализации). Это программа Config2, которая, в том числе, предназначена и для микроконтроллера C8051F330. Последней версией программы Config2 является 4.11 (2013 г.).

С использованием упомянутого выше ПО будет получена работоспособная программа, позволяющая проверить функционирование генератора в тестовом режиме.

Итак, заходим на интернет-страницу <http://www.analog.com> и ищем AD9833. В появившемся дополнительном меню выбираем «Visit the AD9833 DDS Device Configuration Assistant». На экран выводится следующая форма (см. рис. 19).

Прежде всего нажимаем нижнюю из двух радиокнопок, расположенных в правом верхнем углу экранной формы (на рис. 19 отмечена красным овалом), чтобы отключить расчёт фильтров (нам

потребуется только коды, загружаемые в DDS). Далее заполняем поле «MCLK» нашей тактовой частотой «20.971520» (отмечено овалом фиолетового цвета), затем заполняем поле «Desired FREQ0» требуемой частотой выходного сигнала DDS, т.е. 1 кГц, или «0.001» МГц. Это «нулевой» регистр частоты, который нам и нужен. Второй регистр частоты, который нам не требуется (FREQ1), заполняем нулями. На рисунке 19 оба поля отмечены фиолетовым овалом. Поля для обоих фазовых регистров «Desired PHASE0» и «Desired PHASE1» оставляем нулевыми (для нас начальная фаза сигнала DDS не важна). Далее в середине экранной формы в большом прямоугольнике, отмеченном слева как «Control register:», находим поле делителя на 2 «DIV2» и выбираем «Divide by 1», т.е. непосредственное тактирование DDS частотой 20,971520 МГц. После этого в поле «Codewords:» выбираем «FREQ0 LSB & MSB». Это означает, что нам требуются значения младшего и старшего байта нулевого регистра частоты «FREQ0».

Таким образом, всё, что нам требуется, введено, и осталось только нажать голубую кнопку «Update», расположенную в правом нижнем углу. После нажатия этой кнопки в окне «Init. Sequence» (последовательность инициализации) получаем 8 шестнадцатиразрядных (двухбайтовых) слов: 2108 7200 4000 8000 8000 C000 E000 2008.

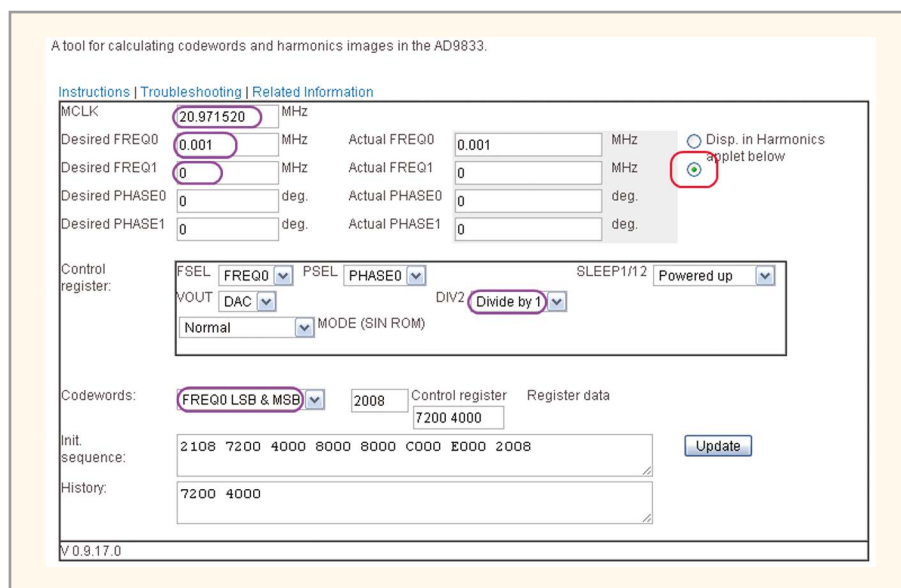
Первое слово 2108 – это начальная инициализация DDS, следующие 2 слова 7200 4000 – значение нулевого частотного регистра (FREQ0), при котором DDS генерирует синусоиду частотой 1 кГц; следующие слова 8000 8000 C000 E000 определяют значения второго частотного регистра и значения обоих фазовых регистров, и последнее слово 2008 определяет запуск DDS. Таким образом, в данной кодовой последовательности информационными и изменяемыми (если требуется изменить частоту) являются только два слова 7200 4000, загружаемые в нулевой частотный регистр. Так, например, если вместо частоты 1 кГц

выбрать частоту 50 кГц (0,05 МГц), то получим последовательность 2108 4400 4027 8000 8000 C000 E000 2008, которая отличается только кодами 4400 4027 вместо кодов 7200 4000.

Отсюда следует, что при программировании синтезатора на определённую частоту, следует изменить только два слова (второе и третье), а остальные оставить без изменений. Значения этих двух слов при различных частотах (от 10 Гц до 100 кГц) были найдены автором с помощью вышеуказанной программы Analog Devices, вставлены в программу для МК и закомментированы. Поэтому для вывода определённой частоты в программе требуется только убрать комментарии перед определённым словом, соответствующим этой частоте.

Следует отметить, что программа Device Configuration Assistant достаточно проста, очень удобна и позволяет получить требуемые кодовые слова, не вдаваясь в детали описания ИС AD9833.

Компания Silicon Laboratories, как уже говорилось выше, разработала программу для компьютера Config2, которая позволяет получить не только кодо-



**Рис. 19.** Снимок экрана компьютера при работе интерактивной программы **DDS Configuration Assistant (Analog Devices)** для получения кодов, загружаемых в DDS, при тактовой частоте **20,971520 МГц** и выходной частоте DDS **1 кГц (0,001 МГц)**

вую последовательность, а и готовый текст программы для МК на языке C51. Этот текст на 90% подходит для выполнения задачи.

Для получения программы для МК C8051F330, которая вводит в синтеза-

тор частоты кодовую последовательность из 8 двухбайтных слов, прежде всего детализируем задачу. Чтобы записать кодовую последовательность в ИС синтезатора из МК и запустить DDS, необходимо:

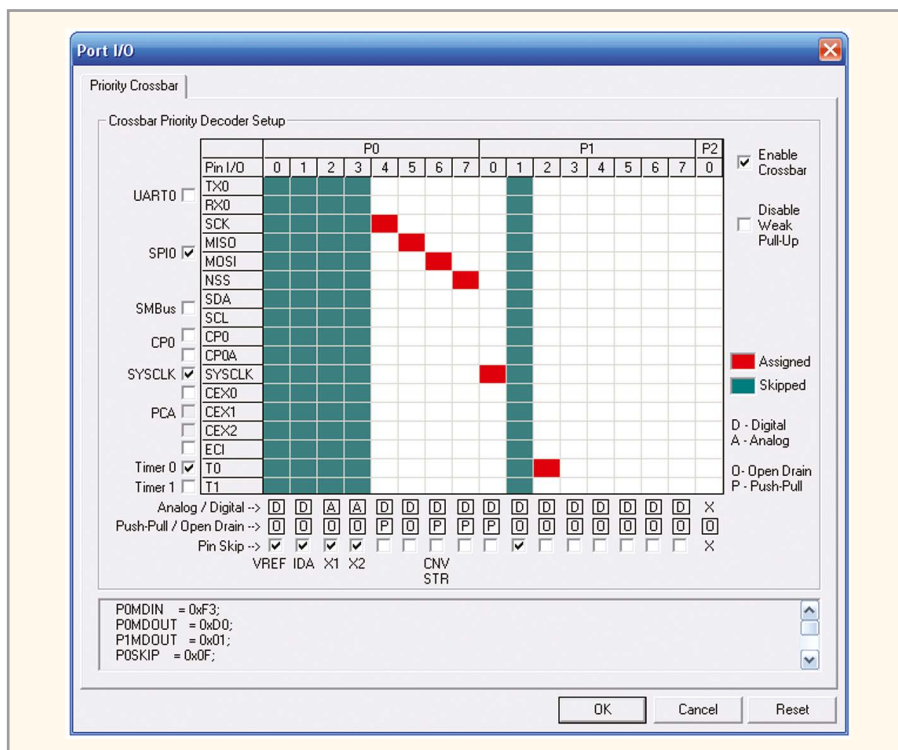


Рис. 20. Снимок экрана компьютера при работе программы Config2 (Silabs) для настройки матрицы соединений МК C8051F330

1. Инициализировать МК для работы с тактовым генератором, используя внешний кварцевый резонатор 20,97152 МГц;
2. Инициализировать интерфейс SPI МК, работающего в режиме, аналогичном SPI в ИС синтезатора;
3. Настроить матрицу соединений МК в соответствии со схемой, приведённой на рисунке 1 (СоЭл № 8, 2013, С. 60). При использовании программы Config2, это не вызывает затруднений;
4. Вывести в синтезатор частоты кодовую последовательность. Это – единственный шаг, требующий написание подпрограммы вывода двухбайтного кода по интерфейсу SPI, однако, в связи с тем, что этот интерфейс аппаратно интегрирован в МК, такая подпрограмма не является сложной.

Итак, запускаем программу Config2 и выбираем МК C8051F330. Вначале выбираем «Peripherals», в появившемся меню – «Oscillators» (генераторы), далее – «External Oscillator» (внешний генератор), в появившемся окне выбираем «Cristal Oscillator» (кварцевый резонатор), «Use External Oscillator as SYSCLK» (использовать внешний генератор как системный); нажимаем радиокнопку «10 MHz <math>f <= 30 \text{ MHz}</math>» и в поле «External Oscillator Frequency» вводим нашу частоту «20971520». После этого нажимаем кнопку ОК и получа-

ем готовый текст программы инициализации генератора:

```
void Oscillator_Init()
{
    int I = 0;
    OSCXCN = 0x67;
    for (I = 0; I < 30000; i++); //
    Wait 1ms for initialization
    while ((OSCXCN & 0x80) == 0);
    CLKSEL = 0x01;
    OSCICN = 0x00;
}
```

Снова заходим в меню «Peripherals» и, чтобы отключить Watchdog Timer (WDT), который по умолчанию включён, выбираем «PCA» (программируемый массив счётчиков, в котором расположен WDT), далее – «Module2/WDT» и снимаем галочку в окне «Enable Watchdog Timer». После этого нажимаем ОК и получаем готовый текст программы отключения WDT:

```
void PCA_Init()
{
    PCA0MD &= ~0x40;
    PCA0MD = 0x00;
}
```

Ещё раз заходим в меню «Peripherals» и, чтобы инициализировать интерфейс SPI, выбираем «SPI». В этом окне выби-

раем «Operate in master mode» (работать в ведущем режиме), «Data centered on First Edge of SCK» (данные центрировать по первому фронту импульса SCK), «SCK Line High in Idle State» (линия SCK находится в высоком состоянии при отсутствии обмена), «4 Wire Single Master Mode» (4-проводный режим единственного ведущего); ставим галочку в поле «Set NSS signal to 1», т.е. устанавливаем сигнал NSS (CS DDS) в «1», далее нажимаем кнопку «Change SCK» и вводим в поле частоты обмена «Target SCK Frequency» 1048576 (1/20 от 20971520 Гц). После этого нажимаем кнопку ОК и получаем готовый текст подпрограммы инициализации SPI:

```
void SPI_Init()
{
    SPI0CFG = 0x50;
    SPI0CN = 0x0D;
    SPI0CKR = 0x09;
}
```

Далее опять входим в меню «Peripherals» и, чтобы инициализировать матрицу соединений, выбираем «Port I/O» (порты ввода/вывода). На экран выводится пустая матрица соединений, которую необходимо заполнить.

Двигаемся по левой части экранной формы сверху вниз и ставим галочки в полях «SPI0», «SYSCLK» и «T0», выбирая, тем самым, эти устройства и определяя для них выходы МК. Далее двигаемся по нижней части матрицы слева направо и ставим галочки в полях P01, P00 (распределяя эти выходы на входы прерываний INT0 и INT1, т.е. пропуская их), P02, P03 (также указывая, что эти выходы должны быть пропущены и освобождены для входов X1 и X2, к которым подключён кварцевый резонатор). Кроме того, устанавливаем эти два порта, как аналоговые (для подключения резонатора), т.е. в полях «Analog/Digital» ставим «A». Последнюю галочку ставим у порта P11, пропуская этот вывод, чтобы его впоследствии можно было заземлить и, тем самым, подать «землю» на вывод 2 DD3.

Далее в полях «Push-Pull/Open Drain» портов P04(SCK), P06(MOSI), P07(NSS) и P10(SYSCLK) устанавливаем «P» (двухтактный выход). И последнее, что необходимо сделать, – установить галочку в поле «Enable Crossbar», т.е. активировать матрицу соединений. После всех манипуляций матрица соединений примет вид, показанный на рисунке 20.

Нажав кнопку ОК, получаем готовый текст подпрограммы для матрицы соединений, т.е. инициализации портов ввода/вывода:

```
void Port_IO_Init()
{
    P0MDIN = 0xF3;
    P0MDOUT = 0xD0;
    P1MDOUT = 0x01;
    P0SKIP = 0x0F;
    P1SKIP = 0x02;
    XBR0 = 0x0A;
    XBR1 = 0x50;
}
```

Кроме этого, сгенерируется текст подпрограммы инициализации всех устройств МК, объединяющей подпрограммы, которые уже были получены ранее:

```
void Init_Device(void)
{
    PCA_Init();
    SPI_Init();
    Port_IO_Init();
    Oscillator_Init();
}
```

Далее выбираем *File*, и в открывшемся меню выбираем *Save Source As...* В открывшемся окне назначаем название файла, например *DDS.c*. Это будет «болванка» нашей программы, содержащая инициализацию всех устройств МК. К ней требуется добавить только вывод кодов в синтезатор частоты.

Приступаем к «ручному» программированию (в файле *DDS.c*). Во-первых, заводим переменную в виде следующей структуры, а точнее – совмещения:

```
union {
    unsigned short US; // U.UB[0]-
    Ст.б.
    unsigned char UB[1]; //
    U.UB[1]- Мл.б.
}U; // U.US - 2-байтное ushort
число.
```

Несмотря на кажущуюся простоту, вышеприведённый текст, на взгляд автора, нуждается в пояснении. Дело в том, что интерфейс SPI является 8-рядным (или однобайтовым, т.е. выводимые числа типа `unsigned char`), а числа, которые требуется ввести в DDS, – двухбайтовые (типа `unsigned short`). Для

вывода двухбайтового числа по однобайтовому интерфейсу его необходимо разбить на младший и старший байты и по очереди их вывести. Процедура отнимает некоторое время и дополнительную память МК, отводимую подпрограмме разбивки. Если в такой подпрограмме присутствуют операции умножения и деления, то может потребоваться обращение к библиотечным функциям, а это ещё более усложнит процедуру разбивки. Кроме того, в памяти МК должны храниться как само 2-байтовое число, так и два 1-байтовых числа, т.е. использоваться, как минимум, 4 байта.

Всего этого можно избежать, если применить совмещение, в котором использован массив из двух однобайтовых элементов `U.UB[0]` и `U.UB[1]`. Этот массив совмещён (т.е. перераспределяет одно и то же место в памяти) с двухбайтовым числом `U.US`. Если, например, задать какие-либо значения указанным элементам массива, то совмещённое с массивом число `U.US` автоматически получит такое значение, что `U.UB[0]` будет старшим байтом этого числа, а `U.UB[1]` – младшим. Справедливо

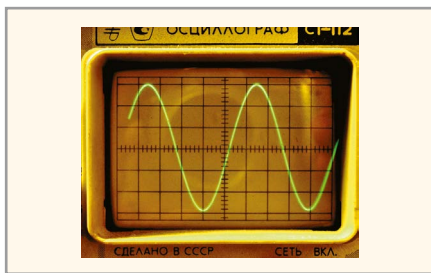


Рис. 21. Фотография осциллограммы сигнала генератора (частота  $F = 1$  кГц, амплитуда  $A \approx 0,3$  В), снятого с экрана аналогового осциллографа (С1-112)

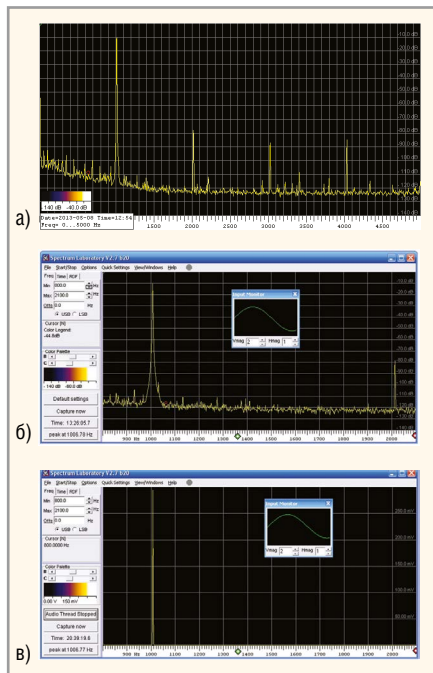


Рис. 22. Снимок экрана компьютера при работе программы SpecLab. Результаты вычисления спектральной плотности сигнала генератора частотой  $F = 1$  кГц, амплитудой  $A = 0,3$  В в логарифмическом масштабе ( $-140 \dots 0$  дБ) в диапазоне частот:

а –  $(0 \dots 5)$  кГц, б –  $(0,8 \dots 2,1)$  кГц; в – в линейном масштабе  $(0 \dots 0,3)$  В в диапазоне частот  $(0,8 \dots 2,1)$  кГц

и обратное: если записать в переменную  $U.US$  какое-либо 2-байтовое число, то переменная  $U.UB[0]$  автоматически получит значение старшего байта этого числа, а  $U.UB[1]$  – младшего байта. Такое совмещение, во-первых, требует всего двух байт памяти (а не 4) и, во-вторых, отпадает надобность в процедуре разбивки.

Подпрограмма вывода 2-байтового числа по интерфейсу SPI, благодаря совмещению, значительно упрощается и выглядит следующим образом (см. листинг 1).

Таким образом, с учётом всех предыдущих подпрограмм, основная программа (main) приобретает следующий вид:

```
void main (void)
{
    Init_Device();
    outspi16(0x2108);
    outspi16(0x7200); // - 1 kHz
    outspi16(0x4000);
    // outspi16(0x4400); // - 50 kHz
    // outspi16(0x4027);
    outspi16(0xC000);
    outspi16(0xE000);
    outspi16(0x2008);
    while (1); // loop forever
}
```

Эта программа выводит в синтезатор коды для генерации частоты 1 кГц. Если закомментировать нижеследующие строки:

```
outspi16(0x2108);
outspi16(0x7200); // - 1 kHz
и убрать комментарии (//) перед
строками:
// outspi16(0x4400); // - 50 kHz
// outspi16(0x4027);
```

то программа будет выводить коды для генерации частоты 50 кГц.

Полученную программу необходимо оттранслировать и запрограммировать в МК. Для трансляции и получения файла программы в \*.hex-формате понадобится файл \*.bat, содержимое которого приведено в листинге 2.

Естественно, потребуется и сам транслятор C51.exe (автор использовал транслятор Keil C-51 v.6.14), сборщик b151.exe и преобразователь кодов из объектного в \*.hex-формат oh51.exe (все из одного пакета).

Как уже упоминалось, программирование МК осуществляется с помощью USB-DEBUG-адаптера производства Silicon Labs, подключённого к плате генератора по схеме, приведённой на рисунке 3 (СоЭЛ № 8, 2013, С. 61). Хотя полный текст программы приведён на интернет-странице журнала, по мнению автора, проще (и надёжнее) воспользоваться программой Config2.exe, которую можно бесплатно скачать с интернет-страницы Silicon Labs (<http://www.silabs.com>).

## ИСПЫТАНИЯ ГЕНЕРАТОРА

Для проверки работы генератора использовались три различных метода.

1. Визуальное наблюдение сигнала генератора на обычном аналоговом осциллографе в диапазоне частот от 10 Гц до 100 кГц и диапазоне амплитуд от 0,3 до 10 В. Преимущество этого

метода – возможность визуально оценить чистоту сигнала и, в некоторой степени, – его форму во всём рабочем диапазоне частот и амплитуд. Недостаток – низкая информативность, т.к. визуальная оценка осциллограммы достаточно субъективна.

2. Звуковая карта компьютера и программа, рассчитывающая спектральную плотность входного сигнала, подключённого к линейному входу звуковой карты. Преимущества – возможность вычисления спектра по большому количеству реализаций, позволяющая получить достаточно высокую точность и подробность всех спектральных составляющих входного сигнала. Недостаток – узкий частотный и амплитудный диапазон, невысокая точность измерения частоты на пике спектральной плотности. Диапазон частот, в котором такой способ позволяет адекватно оценить спектр, приблизительно от 30–40 Гц до 18–19 кГц. Вне этого диапазона звуковая карта даёт неадекватные показания. Амплитудный диапазон, состоящий из одной амплитуды 0,3 В, определяется звуковой картой.

3. Цифровой осциллограф, позволяющий наблюдать как сам сигнал, так и его спектр. Преимущества – работа во всём рабочем диапазоне частот и амплитуд входного сигнала, возможность оценки спектра и относительно точные измерения частоты пика спектральной плотности. Недостатки – осциллограмма имеет «цифровой» вид и не позволяет визуально оценить чистоту входного сигнала (как с помощью аналогового осциллографа). Спектр не очень подробный, т.к. он получен по одной реализации и адекватно отображается только в линейном масштабе (хотя осциллограф и способен воспроизводить спектр в логарифмическом масштабе, такое отображение мало информативно).

На рисунке 21 приведена типичная осциллограмма выходного сигнала генератора. В диапазоне частот от 10 Гц до 100 кГц и диапазоне амплитуд от 0,3 до 10 В все наблюдаемые осциллограммы практически ничем не отличались. Как можно заметить, синусоида является гладкой и на ней отсутствуют какие-либо «цифровые» искажения.

На рисунке 22 приведены спектры сигнала генератора, подключённого к линейному входу звуковой карты компьютера и рассчитанные программой SpecLab по 65536 реализациям. Можно заметить, что вторая гармоника имеет частоту в 2 раза выше частоты

**Листинг 1**

```

void outspi16(unsigned short wor) {
    U.US=wor; // Загрузка 2-байтного числа U.US
    NSSMD0=0; // CSAD=0.
    SP10DAT=U.UB[0]; // Вывод ст. байта по SPI.
    while (!SPIF); // Ожидание окончания вывода байта.
    SPIF=0; // Сброс флага окончания передачи.
    SP10DAT=U.UB[1]; // Вывод мл. байта по SPI.
    while (!SPIF); // Ожидание окончания вывода байта
    SPIF=0; // Сброс флага окончания передачи.
    NSSMD0=1; // CSAD=1.
}

```

**Листинг 2**

```

c51 DDS.c print small symbols code debug object(DDS.obj) objectextend
pause
bl51 DDS.obj to DDS ixref print(bl_DDS.lst)
oh51 DDS
pause

```

основной гармонике, а по амплитуде – меньше амплитуды основной гармоники приблизительно на 70 дБ (пик основной гармоники расположен примерно на уровне –10 дБ, а второй – на уровне –80 дБ, т.е. их разность составляет около 70 дБ). Может показаться, что амплитуда второй гармоники очень велика (судя по графику, она всего в 2,5 раза меньше амплитуды основной гармоники). Однако не следует забывать, что график приведён в логарифмическом масштабе.

В левом нижнем углу на рисунках 22б и 22в показано небольшое окно, в котором приведена частота пика спектральной плотности (1006,78 Гц и 1006,77 Гц соответственно), хотя этот пик должен находиться на частоте ровно 1000 Гц.

Для решения этого вопроса сигнал с генератора был подан на частотный вход поверенного частотомера. Частота, измеренная частотомером, составляла 1,0000 кГц. Кроме того, был измерен период сигнала, который составлял 1,000000 мс и 1000.0001 мкс (это также показания частотомера, снятые при двух разных режимах измерения).

Это означает, что использованная программа (или звуковая карта компьютера) не совсем верно отражает частоту.

Ниже показаны снимки экрана цифрового осциллографа при различных частотах и одной и той же амплитуде (0,3 В) сигнала (см. рис. 23а-г) и снимок экрана при максимальной амплитуде сигнала (10 В). Можно заметить, что сигнал имеет абсолютно идентичную форму спектра, которая не зависит ни от частоты, ни от амплитуды. Кстати, на рисунке 23а осциллограф показал

частоту ровно 1,000 кГц. Что касается осциллограмм самого сигнала (жёлтого цвета), то их дискретность – свойство цифрового осциллографа.

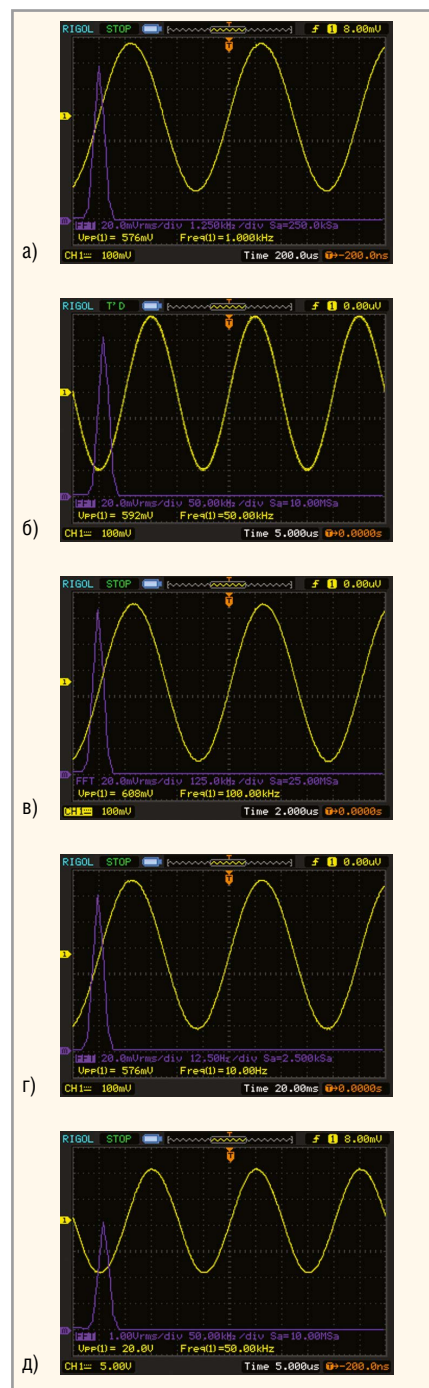
Приведённые на рисунках 21–23 иллюстрации, по мнению автора, достаточно красноречиво свидетельствуют о том, что во всём рабочем диапазоне частот и амплитуд генератор выдаёт синусоидальный сигнал, имеющий высокую чистоту (в том числе и спектральную) и адекватно отражающий частоту.

**ЗАКЛЮЧЕНИЕ**

Применение в конструкции генератора ИС синтезатора частоты AD9833, микроконтроллера C8051F330 и инструментального усилителя AD8295 обеспечило выходной сигнал, точно воспроизводящий установленную частоту и обладающий высокой спектральной чистотой. Тактирование ИС синтезатора системной частотой МК и использование ИУ для компенсации постоянной составляющей в выходном сигнале ИС синтезатора позволили реализовать высокую временную и эксплуатационную стабильность работы устройства.

**ЛИТЕРАТУРА**

1. Кузьминов А. Метод фоторепродуцирования для изготовления фотошаблона печатных плат в домашних условиях. Технологии в электронной промышленности, № 5–7, 2010.
2. Кузьминов А. Изготовление устройств на печатных платах с высоким разрешением в домашних условиях. Технологии в электронной промышленности, № 8–10, 2010.



**Рис. 23. Снимок с экрана цифрового осциллографа, показывающего сигнал генератора (жёлтый цвет – осциллограмма, фиолетовый цвет – спектр в линейном масштабе):**  
а –  $F = 1$  кГц,  $A \approx 0,3$  В; б –  $F = 50$  кГц,  $A \approx 0,3$  В;  
в –  $F = 100$  кГц,  $A \approx 0,3$  В; г –  $F = 10$  кГц,  $A \approx 0,3$  В;  
д –  $F = 50$  кГц,  $A \approx 10$  В

3. Кузьминов А. Регулируемый двухполярный блок питания. Радио, № 5, 2012.
4. Кузьминов А. Преобразователь интерфейсов USB\_SPI с гальванической развязкой. Современная электроника, № 1, 2012.
5. Бирюков С.А. Цифровые устройства на МОП-интегральных микросхемах. Радио и связь, 1996.
6. Рутковски Дж. Интегральные операционные усилители. Мир, 1978.

