

Romm Yakov Evseevich – Taganrog State Pedagogical Institute; e-mail: romm@list.ru; 48, Initsiativnaya street, Taganrog, 347926, Russia; phones: +78634601753, +78634601812, +78634601807; the department of information science; head of the department; dr. of eng. sc.; professor.

Stakhovskaya Irina Illarionovna – e-mail: irihkast@gmail.ru; phone: +79034604006; the student grade.

УДК 681.3.06(075)

М.И. Ледовской

АЛГОРИТМ ВЫЧИСЛЕНИЯ ФУНКЦИЙ SIN(X) И COS(X) ДЛЯ 16-РАЗРЯДНЫХ МИКРОКОНТРОЛЛЕРОВ

При использовании микроконтроллеров во встраиваемых системах различного назначения возникает необходимость вычисления элементарных функций $\sin(x)$ и $\cos(x)$. Решение этой задачи с помощью стандартной библиотеки функций языка C приводит к существенному росту программного кода и времени вычислений для микроконтроллеров с ограниченными вычислительными возможностями. В этом случае альтернативой могут служить собственные подпрограммы на основе таблично-алгоритмического метода, реализуемого с помощью целочисленной арифметики микроконтроллера. Однако преимущества таблично-алгоритмического метода проявляются в том случае, если поправки к табличным значениям функций вычисляются с помощью простейших методов аппроксимации. Негативным следствием применения указанных методов является значительный объем табличных значений функций, которые требуется хранить в памяти микроконтроллера. По этой причине реализация таблично-алгоритмического метода затрудняется. В настоящей работе предлагается алгоритм таблично-алгоритмического метода, который по сравнению с аналогом обеспечивает требуемую точность и позволяет сократить в 8 раз объем табличных значений функций. Алгоритм может найти применение, например, при создании различных устройств со сверхнизким энергопотреблением на базе 16-разрядных микроконтроллеров MSP430 без аппаратного умножителя.

Функция $\sin(x)$; $\cos(x)$; таблично-алгоритмический метод; целочисленный алгоритм, микроконтроллер; погрешность; MATLAB; модель.

M.I. Ledovskoy

THE ALGORITHM OF CALCULATION OF THE FUNCTIONS SIN(X), COS(X) FOR 16-BIT MICROCONTROLLERS

When using microcontrollers and embedded systems for various purposes there is a necessity of calculation of elementary functions $\sin(x)$ and $\cos(x)$. The solution to this problem using standard C-library functions with leads to substantial growth of code and time of calculations for microcontrollers with limited computing power. In this case, the alternative can serve as your own routines based table-algorithmic method, which is implemented using integer arithmetic microcontroller. However, the advantages of table-algorithmic method appear in the case, if the amendments to the tabular values of the functions are computed using the simplest methods of approximation. A negative consequence of use of the mentioned methods is a significant amount of tabular values of the functions that you want to store in memory of the microcontroller. For this reason, the implementation of a table-algorithmic method difficult. In the present paper an algorithm is proposed table-algorithmic method, which in comparison with the similar provides the required accuracy and reduces to 8 times the amount of the tabular values of functions. The algorithm can find application, for example, the creation of different devices with ultra low power consumption-based 16-bit MSP430 microcontrollers without hardware multiplier.

The function $\sin(x)$; $\cos(x)$; table-algorithmic method; integer algorithm; microcontroller; error; MATLAB; model.

При использовании микроконтроллеров во встраиваемых системах различного назначения зачастую возникает необходимость вычисления элементарных функций, например, $\sin(x)$ и $\cos(x)$. Решение этой задачи с помощью стандартной библиотеки функций языка C [1], основанной на арифметике вещественных чисел, приводит к существенному росту программного кода и времени вычислений для микроконтроллеров с ограниченной разрядностью, где аппаратная поддержка режима плавающей точки отсутствует. В этом случае альтернативой функциям языка C могут служить собственные подпрограммы на основе таблично-алгоритмического метода [2, 3], реализуемого с помощью целочисленной арифметики микроконтроллера. Такой прием положительно сказывается на объеме программного кода и скорости вычислений. Однако преимущества таблично-алгоритмического метода проявляются в том случае, если поправки к табличным значениям функций вычисляются с помощью простейших методов аппроксимации, например, одночленного ряда Тейлора или линейной интерполяции. Это наиболее заметно для тех микроконтроллеров, которые не поддерживают аппаратно целочисленную операцию умножения. Негативным следствием применения указанных методов аппроксимации является значительный объем табличных значений функций, которые требуется хранить в памяти микроконтроллера. По этой причине реализация таблично-алгоритмического метода в условиях ограниченных вычислительных возможностей микроконтроллера затрудняется.

В этой связи возникает необходимость в разработке алгоритмов таблично-алгоритмического метода, обеспечивающих сокращение количества табличных значений функций и не требующих использования целочисленной операции умножения. В настоящей работе рассматривается одно из возможных решений данной задачи на примере функций $\sin(x)$ и $\cos(x)$.

Как известно, таблично-алгоритмический метод основан на разбиении интервала изменения аргумента x и образовании узловых точек x_k , для которых заранее вычисляются значения функции, помещаемые в таблицу [2]. Причем $x_k = x_{k-1} + h$, где h – шаг таблицы по аргументу x . Тогда значения функции для промежуточных значений аргумента определяются путем выборки из таблицы ближайшего узлового значения и вычисления поправки. В рамках таблично-алгоритмического метода могут применяться различные методы аппроксимации функций для вычисления поправок. Предлагаемый алгоритм основан на вычислении поправок путем интегрирования порождающей системы дифференциальных уравнений Шеннона [4] для функций $\sin(x)$ и $\cos(x)$:

$$\begin{aligned} y_1(x) &= y_1(x_k) + \int_{x_k}^x y_2(x) dx, \\ y_2(x) &= y_2(x_k) - \int_{x_k}^x y_1(x) dx, \end{aligned} \tag{1}$$

где $y_1(x)=\sin(x)$ и $y_2(x)=\cos(x)$ – искомые значения функций в точке x ; $y_1(x_k)$ и $y_2(x_k)$ – ближайшие табличные значения этих функций в узловой точке x_k . Интегралы, стоящие в правой части уравнений (1), представляют собой поправки к табличным значениям функций. Для получения численного алгоритма необходимо подобрать разностную схему, которая аппроксимирует уравнения (1), обеспечивая баланс между методической и инструментальной погрешностью функций при известной разрядности микроконтроллера. Далее для определенности разрядность микроконтроллера принимается равной $n=16$.

Методическую погрешность можно получить, используя известную методику оценки погрешности воспроизведения элементарных функций путем численного интегрирования порождающих уравнений Шеннона [4]. В частности, если интегралы в уравнениях (1) вычисляются по формуле трапеций, то выражение для методической погрешности функции $y_1(x)=\sin(x)$ на интервале $[x_k, x]$ имеет вид

$$\varepsilon_{M1}(x) = \frac{1}{12}(\nabla x)^2(x - x_k)\sin(x), \quad (2)$$

где ∇x – шаг численного интегрирования по аргументу x . Поскольку $\sin(x)_{\max}=1$ и $(x-x_k)_{\max}=h$, где h – шаг таблицы по аргументу x для функции $\sin(x)$, то

$$\varepsilon_{M1} \leq \frac{1}{12}(\nabla x)^2 h. \quad (3)$$

Помимо влияния на величину методической погрешности ε_{M1} , параметр h определяет количество табличных значений функции $\sin(x)$ на заданном интервале изменения аргумента x , например $[0, 1-2^{-8}]$, а параметр ∇x – количество шагов интегрирования на интервале $[x_k, x]$, т.е. время вычисления текущего значения этой функции.

Вместо нахождения инструментальной погрешности, требующей сложных выкладок, можно ограничиться получением погрешности округления, возникающей при представлении вещественных значений переменных в целочисленном формате [5]. Например, для функции $y_1(x)=\sin(x)$ достаточно найти квант, соответствующий единице младшего разряда целочисленного формата со знаком: $\Delta y_1 = |y_1(x)|_{\max}/(2^{n-1}-1) \approx 2^{-(n-1)}$, что является предельным значением погрешности округления с недостатком. Если табличные значения функции $y_1(x)=\sin(x)$, являясь вещественными величинами, преобразуются в целочисленный формат по формуле $Y_{jk} = \text{round}(y_1(x_k)/\Delta y_1)$, где $\text{round}()$ – функция округления до ближайшего целого, то погрешность округления подчиняется оценке $\varepsilon_{o1} \leq 2^{-n}$ (округление по $1/2$).

Налагая более жесткие условия на методическую погрешность, определим баланс погрешностей в виде $\varepsilon_{M1} \leq \varepsilon_{o1}$ или

$$\frac{1}{12}(\nabla x)^2 h \leq 2^{-n}. \quad (4)$$

Выражение (4) содержит два неизвестных параметра ∇x и h , одним из которых необходимо задаться. В отличие от h , параметр ∇x будет явным образом присутствовать в алгоритме и влиять на его сложность. Поэтому его следует выбрать в виде степени числа 2, например, 2^{-8} . Такой выбор упрощает алгоритм и фактически исключает необходимость использования операции целочисленного умножения. Полагая $\nabla x = 2^{-8}$, из (4) при $n=16$ получаем $h \leq 1/12$.

С целью дальнейшего упрощения алгоритма шаг таблицы h по аргументу x также представим в виде степени числа 2. В этом случае старшие разряды целочисленного значения аргумента $X = \text{round}(x/\Delta x)$ представляют узловое (табличное) значение $X_k = \text{round}(x_k/\Delta x)$ и одновременно – адрес ближайшего табличного значения функции. Здесь $\Delta x = |x|_{\max}/(2^{n-1}-1)$ – квант аргумента x .

Учитывая ограничение $h \leq 1/12$, примем $h = 2^{-4}$. Тогда количество табличных значений функции $\sin(x)$ на интервале изменения аргумента $[0, 1-2^{-8}]$ составит 16 и максимальное количество шагов интегрирования $\nabla x = 2^{-8}$ на отрезке $(x-x_k)_{\max} = h = 2^{-4}$ тоже получается равным 16.

Заметим, что в случае вычисления поправки к табличным значениям функции $\sin(x)$ с помощью одночленного ряда Тейлора, шаг таблицы, найденный при тех же условиях из баланса методической погрешности и погрешности округления, будет равен $h = 2^{-7}$ [6]. При этом объем таблицы возрастет до 128. Следовательно, предлагаемый подход позволяет сократить объем таблицы для функции $\sin(x)$ в 8 раз.

Таким образом, формула трапеций подходит для построения разностной схемы, аппроксимирующей уравнения (1). В случае ее использования разностную схему можно представить следующим образом:

$$\begin{aligned}
 y_{2i}^* &= 2y_{2(i-1)} - y_{2(i-2)}, \\
 y_{1i} &= y_{1(i-1)} + 2^{-1}(y_{2i}^* + y_{2(i-1)})\nabla x, \\
 y_{2i} &= y_{2(i-1)} - 2^{-1}(y_{1i} + y_{1(i-1)})\nabla x, \\
 x_i &= x_{i-1} + \nabla x, \\
 x_0 &= x_k, \quad y_{10} = y_{1k}, \quad y_{20} = y_{2(-1)} = y_{2k}, \\
 i &= 1, 2, 3, \dots,
 \end{aligned}
 \tag{5}$$

где знак * обозначает экстраполированное значение.

При построении целочисленного алгоритма на основе разностной схемы (5), помимо основного 16-разрядного формата данных, используется внутренний 32-разрядный формат, что позволяет исключить накопление инструментальной погрешности. Обозначения этих форматов приведены в табл. 1.

Таблица 1

Вид целочисленного формата	Обозначение формата	Обозначение старшего слова	Обозначение младшего слова
32-разрядный формат	16.16	16.0	0.16
16-разрядный формат	16	–	–

Целочисленный алгоритм вычисления функций $y_1(x)=\sin(x)$, и $y_2(x)=\cos(x)$ представлен на рис. 1. Переменные, входящие в данный алгоритм, связаны с вещественными переменными разностной схемы (5) следующими зависимостями: $X=\text{round}(x \cdot M_x)$, $X_k=\text{round}(x_k \cdot M_x)$, $\nabla X=\text{round}(\nabla x \cdot M_x)$, $Y_1=\text{round}(y_1 \cdot M_y)$, $Y_{1k}=\text{round}(y_{1k} \cdot M_y)$, $Y_2=\text{round}(y_2 \cdot M_y)$, $Y_{2k}=\text{round}(y_{2k} \cdot M_y)$, где M_x , M_y – масштабы, которые определяются как обратные значения соответствующих квантов и округляются до ближайшей степени числа 2. Значения масштабов и целочисленные значения переменных приведены в табл. 2.

Целочисленный шаг H , приведенный в табл. 2, не присутствует явно в алгоритме, но определяет фактический диапазон изменения узловых значений аргумента X_k . Если не считать нулевого значения, то $2^{11} \leq X_k \leq 2^{15} - 2^{11}$, так как $X_k = X_{k-1} + H$ и $H = 2^{11}$. Поэтому старшие разряды текущего значения целочисленного аргумента X , начиная с 11-го по 14-й, несут информацию о величине X_k и могут использоваться в качестве адреса ближайших табличных значений Y_{1k} и Y_{2k} .

Заметим также, что учет коэффициента 2^7 не требует выполнения операции умножения. Если представить $2^7 = 2^8 \cdot 2^{-1}$, то 2^{-1} реализуется операцией сдвига на 1 двоичный разряд, а 2^8 – перестановкой байтов в регистре микроконтроллера.

Для проверки точностных характеристик проведено моделирование целочисленного алгоритма в системе MATLAB. Цель эксперимента состояла в анализе полной, методической и инструментальной погрешностей вычисления функций $\sin(x)$ и $\cos(x)$ на интервале изменения аргумента $[0, 1-2^{-8}]$ при $h=2^{-4}$ и $\nabla x=2^{-8}$. Для этого использовалась методика [7], в соответствии с которой разработана структура модели (рис. 2) и создан m -файл модели на языке MATLAB.

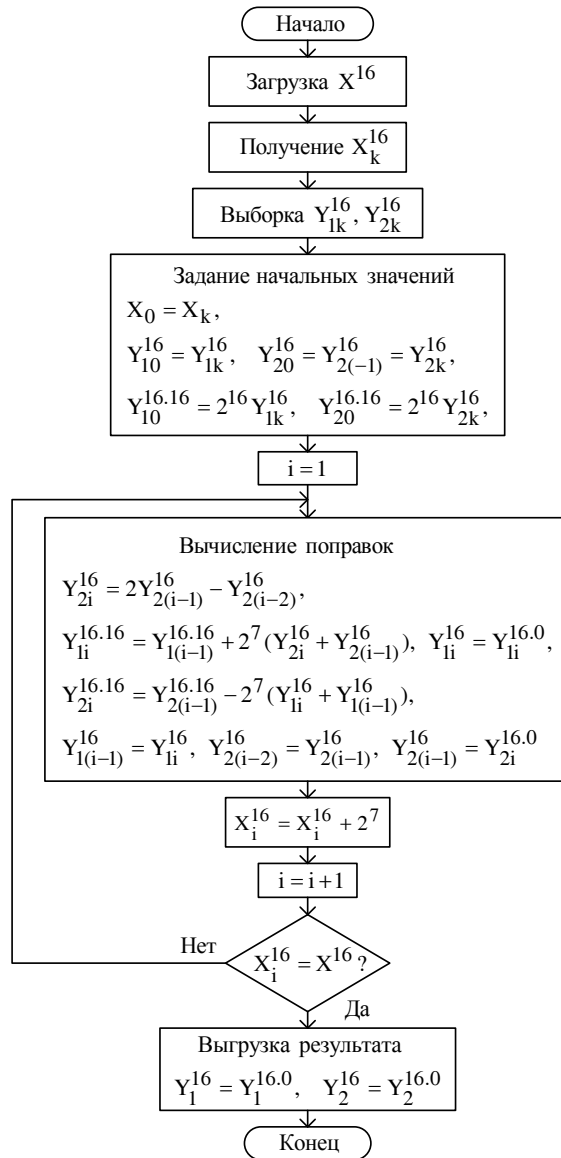


Рис. 1. Целочисленный алгоритм вычисления функций $\sin(x)$, $\cos(x)$ для 16-разрядных микроконтроллеров

Таблица 2

Вещественная переменная	Масштаб	Целочисленное значение
$y1$	$M_y=2^{15}$	$0 \leq Y1 < 2^{15} - 1$
$y2$	M_y	$0 < Y2 < 2^{15} - 1$
x	$M_x=2^{15}$	$0 \leq X < 2^{15} - 2^7$
x_k	M_x	$0 \leq X_k \leq 2^{15} - 2^{11}$
\sqrt{x}	M_x	$\sqrt{X} = 2^7$
h	M_x	$H = 2^{11}$

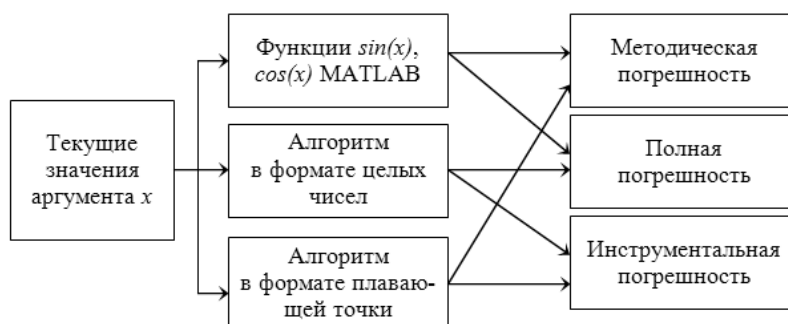


Рис. 2. Структура MATLAB-модели для экспериментального анализа погрешностей вычисления функций $\sin(x)$ и $\cos(x)$

Главный блок модели реализует целочисленный алгоритм, характеризующийся методической ε_m , инструментальной ε_u , а также полной погрешностью $\varepsilon = \varepsilon_m + \varepsilon_u$. Эталонный блок модели обеспечивает вычисление функций $\sin(x)$ и $\cos(x)$ в формате «двойное вещественное» (формат плавающей точки) с $\varepsilon_m \approx 0$ и $\varepsilon_u \approx 0$ путем вызова библиотечных функций MATLAB. Вспомогательный блок модели реализует разностную схему (5) также в формате «двойное вещественное» и характеризуется методической погрешностью ε_m , равной погрешности главного блока, и инструментальной погрешностью $\varepsilon_u \approx 0$. Поэтому сопоставление результатов указанных блоков, как это показано на рис. 2, обеспечивает достижение цели моделирования.

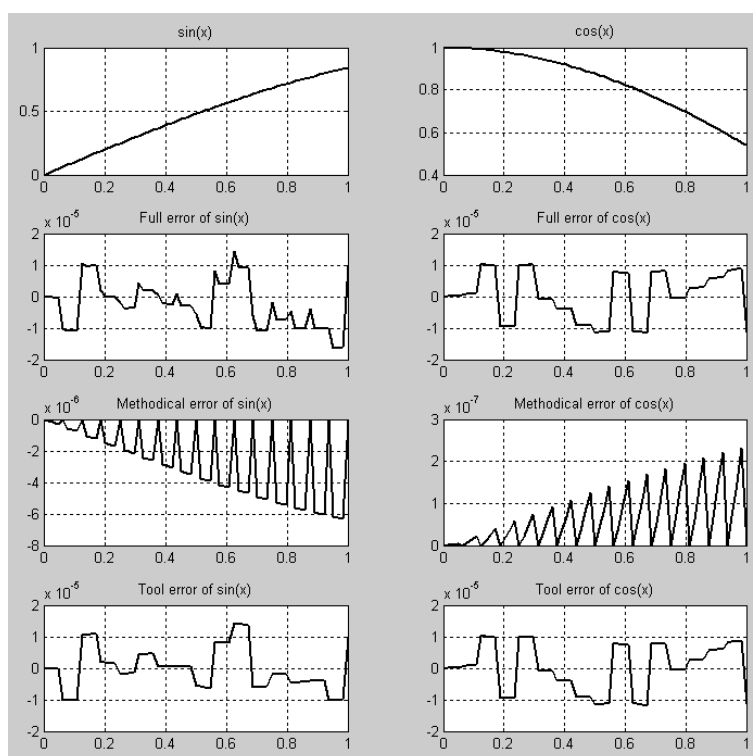


Рис. 3. Результаты моделирования целочисленного алгоритма в системе MATLAB

Из результатов моделирования, представленных на рис. 3, видно, что полная погрешность вычисления функций $\sin(x)$ и $\cos(x)$ не превышает величины $1,5 \cdot 10^{-5}$, что соответствует погрешности округления для 16-разрядного формата данных. Заданный баланс методической и инструментальной погрешностей $\varepsilon_m \leq \varepsilon_u$ соблюдается.

Наряду с приемлемыми точностными характеристиками предлагаемый алгоритм, как отмечалось выше, обеспечивает сокращение в 8 раз объема табличных значений функций по сравнению с аналогом.

Алгоритм может найти применение при создании различных устройств со сверхнизким энергопотреблением на базе 16-разрядных микроконтроллеров MSP430 без аппаратного умножителя, например, MSP430F135.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Керниган Б., Ритчи Д. Язык программирования С. – 2-е изд.: Пер. с англ. – М.: Вильямс, 2010. – 304 с.
2. Попов Б.А., Теслер Г.С. Вычисление функций на ЭВМ: Справочник. – Киев: Наукова думка, 1984. – 600 с.
3. Ромм Я.Е., Аксайская Л.Н. Кусочно-полиномиальная аппроксимации функций, производных и определенных интегралов на основе интерполяции по Ньютону // Известия ЮФУ. Технические науки. – 2008. – № 2 (79). – С. 14-21.
4. Каляев А.В. Теория цифровых интегрирующих машин и структур. – М.: Советское радио, 1970. – 472 с.
5. Ледовской М.И. Обработка вещественных данных в микроконтроллерах с арифметикой фиксированной точки // Известия ТРТУ. – 2004. – № 2 (37). – С. 52-58.
6. Байков В.Д., Смолков В.Б. Специализированные процессоры: итерационные алгоритмы и структуры. – М.: Радио и связь, 1985. – 288 с.
7. Ледовской М.И. Моделирование алгоритма инерциальной навигации в MATLAB-SIMULINK // Ползуновский вестник. – 2011. – № 3/1. – С. 9-11.

Статью рекомендовал к опубликованию д.т.н., профессор А.А. Зори.

Ледовской Михаил Иванович – Южный федеральный университет; e-mail: kafmps@tppark.ru; 347900, г. Таганрог, ул. Петровская, 81; тел.: 88634328025; кафедра микропроцессорных систем; к.т.н.; доцент.

Ledovskoy Mikhail Ivanovich – Southern Federal University; e-mail: kafmps@tppark.ru; 81, Petrovsky street, Taganrog, 347930, Russia; phone: +78634328025; the department of microprocessor systems; cand. of eng. sc.; associate professor.