

Анализ эффективности модифицированного алгоритма измерения физических величин для сенсорных систем

А. А. Койбаш

Донецкий национальный технический университет
mr.koibash@gmail.com

Аннотация

Рассмотрены детали реализации модифицированного алгоритма измерения физических величин для сенсорных систем, в основе которого лежит постбинарное кодирование информации. С целью проверки работоспособности алгоритма выполнен анализ его эффективности с использованием реального прототипа сенсорной системы. На основании полученных в опытах данных сформированы интервалы и выполнено их постбинарное кодирование с использованием различных форматов. Определены преимущества и недостатки использования каждого формата. На основании результатов экспериментального анализа предложен гибридный вариант постбинарного кодирования информации.

Введение

На сегодняшний день наблюдается увеличение степени интеграции информационно-компьютерной инфраструктуры во многие сферы человеческого общества. Активно развивается область цифровых услуг и удобств, а промышленное производство характеризуется всё большей степенью автоматизации рутинной работы. Эти процессы невозможно представить без повсеместного увеличения количества используемых датчиков, поскольку именно они и являются «глазами и ушами» для компьютерной техники.

Такую совокупность датчиков можно очертить в рамках единой сенсорной составляющей информационно-компьютерной инфраструктуры. Улучшение её характеристик положительным образом скажется на работе большинства цифровых систем.

Для достижения этой цели в статье [1] рассмотрены различные подходы к модернизации сенсорных устройств. Одним из вариантов является взаимная интеграция большого количества сенсоров в единое периферийное устройство (суперсенсор) [2]. Спектр их применения широк: от сбора статистических параметров окружающей среды до возможности предоставления полной информации обо всех муниципальных аспектах умных городов с автоматическим принятием системой соответствующих решений [3].

В условиях стремительного развития компьютерных систем в мировом масштабе [4] необходимо учитывать качественные характеристики отдельных элементов сенсорной составляющей такой инфраструктуры. В частности, на основе аналитических зависимостей, полученных в статье [5] по

нескольким различным видам датчиков, обнаружены тенденции к снижению размеров, веса, тока питания, а также улучшению шумовых характеристик. Данные параметры позволяют дать благоприятный прогноз для существенного повышения степени взаимной интеграции разнородных датчиков в единую систему.

В рамках поисков путей модернизации сенсорной составляющей в статье [6] разработан модифицированный алгоритм измерения физических величин для сенсорных систем. Блок-схема алгоритма с небольшими доработками представлена на рис. 1. В его основе находятся 2 процедуры:

– формирование и расширение интервала относительно среднего арифметического значения;

– постбинарное кодирование интервала.

Таким образом, в рамках первого этапа происходит рационализация способа хранения информации. Значение, получаемое с датчика, как и полезная информация в виде диапазона погрешности сохраняются в виде единого интервала.

Второй этап связан с использованием современных исследований в области дальнейшей эволюции вычислений. В книге [7] авторами определены закономерности развития компьютеринга и рассмотрен переход к постбинарному представлению информации. Основой такого подхода является логика четырёх состояний – тетралогика, в которой помимо булевых 0 (ложь) и 1 (истина) появились ещё два состояния: А (неопределённость: «ни истина, ни ложь») и М (множественность: «и истина, и ложь»). Эта концепция в определённой степени соответствует квазиквантовому и квазигенетическому компьютерингу.

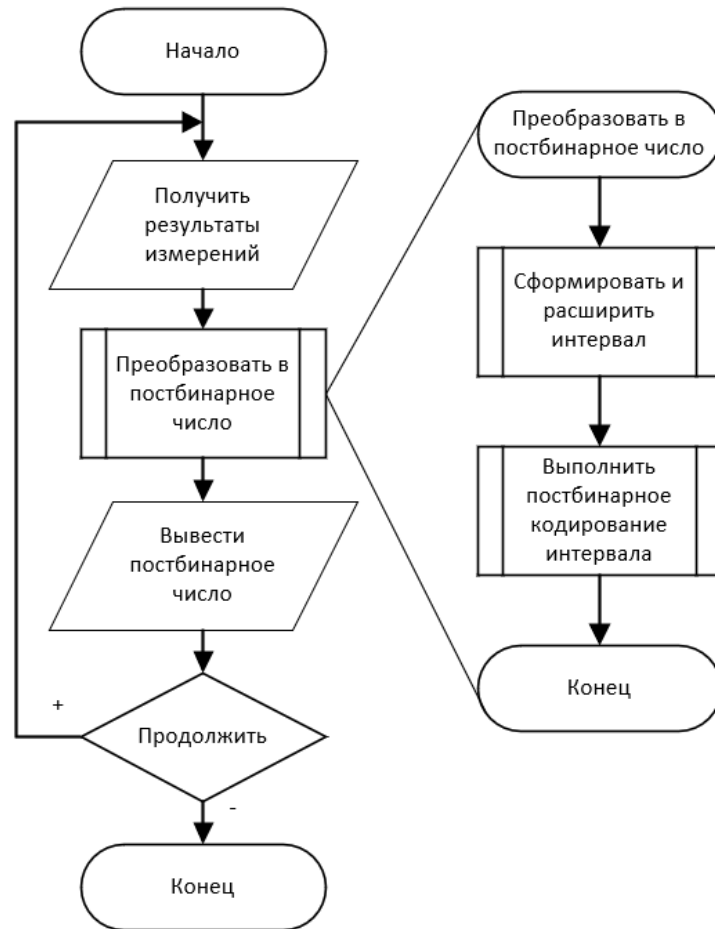


Рисунок 1 – Блок-схема модифицированного алгоритма измерения физических величин для сенсорной системы [6]

Несмотря на то, что развитие квантовых и ДНК-компьютеров находится лишь на начальной стадии, постбинарный компьютеринг уже сейчас позволяет реализовать некоторые особенности таких вычислений [8-9].

В рамках предложенного алгоритма постбинарное кодирование предоставляет множество инструментов эффективной работы с информацией. С целью моделирования работы алгоритма в реальных задачах проведён эксперимент на основе прототипа сенсорной системы.

Проведение эксперимента

Для анализа эффективности алгоритма необходимы реальные значения температуры. Следовательно, нужно расширить набор полученных с сенсора данных, продолжив начатые в статье [6] серии измерений. Схема эксперимента остаётся прежней: за короткий промежуток времени микроконтроллер на плате STM32F407G Discovery получает ряд показаний с подключенного к плате терморезистора. Сам датчик погружён в большой сосуд с теплоёмкой жидкостью, что позволяет минимизировать погрешности при нагревании или остывании до

температуры окружающей среды. Полученный сигнал дискретизируется посредством встроенного 12-разрядного аналогово-цифрового преобразователя (в значениях от 0 до 4095). Одновременно с использованием сконструированной сенсорной системы в жидкость погружён аквариумный термометр с заводской калибровкой, что позволяет достоверно определить реальную температуру жидкости.

Количество измерений за малый промежуток времени было выбрано равным 16. При выборе числа [6] учитывалась не только вероятность попадания результата в доверительный интервал, но и меньшие в сравнении с компьютерами аппаратные ресурсы микроконтроллеров, которые в свою очередь накладывают определённые ограничения на вычисления. Таким образом, при 16 измерениях результат будет расположен:

- с вероятностью 99.993% в доверительном интервале $[-\sigma_x; +\sigma_x]$;
- с вероятностью 95% в доверительном интервале $[-0.5\sigma_x; +0.5\sigma_x]$.

Предыдущие опыты уже подтверждали, что такое количество измерений не будет оказывать существенного влияния на

производительность всей сенсорной системы. В дополнение к проведённым экспериментам сделаны 7 серий по 16 измерений при различных температурах в диапазоне от $\approx 18^\circ\text{C}$ до $\approx 49^\circ\text{C}$. Для каждой серии измерений посчитано среднее арифметическое значение аналогово-цифрового преобразователя. В дополнение ко всему вышеперечисленному представлена температура, полученная с аквариумного электронного

термометра. Результаты измерений представлены в таблице 1.

Из таблицы видно, что присутствуют колебания получаемых с АЦП значений. Тем не менее, среднее арифметическое измерений должно представлять собой достаточно точный результат. На следующем шаге необходимо получить значения в градусах Цельсия ($^\circ\text{C}$).

Таблица 1 – Результаты серий измерений датчика температуры

№	1	2	3	4	5	6	7	8	9
АЦП	1415	1426	1608	1727	1868	2043	2166	2192	2575
	1382	1412	1635	1723	1877	2053	2147	2196	2559
	1391	1421	1610	1753	1868	2039	2156	2189	2596
	1409	1424	1628	1746	1876	2028	2145	2206	2581
	1415	1421	1629	1723	1875	2042	2159	2196	2581
	1399	1425	1613	1723	1886	2050	2164	2206	2588
	1404	1422	1630	1720	1869	2030	2156	2196	2596
	1390	1420	1629	1729	1862	2052	2152	2205	2576
	1406	1428	1615	1728	1864	2042	2162	2180	2598
	1415	1430	1636	1725	1866	2054	2173	2187	2590
	1395	1423	1611	1721	1877	2035	2158	2198	2575
	1401	1428	1625	1725	1873	2033	2165	2185	2571
	1409	1416	1628	1730	1865	2047	2149	2199	2579
	1397	1425	1633	1748	1869	2051	2151	2196	2588
	1403	1419	1618	1731	1870	2043	2170	2201	2595
	1411	1427	1625	1723	1873	2051	2155	2193	2596
АЦП _{ср}	1402.6	1422.9	1623.3	1729.7	1871.1	2043.3	2158	2195.3	2584
t _{акв} , $^\circ\text{C}$	17.8	18.6	23.3	26.1	29.7	33.3	37.5	38.9	49.0

При анализе алгоритма прецизионность датчика температуры (повторяемость, или, другими словами, соответствие показаний датчика реальной температуре) не является целевой задачей, поскольку работа алгоритма нацелена именно на точность (математическое ожидание). Поэтому в данном эксперименте прецизионностью можно пренебречь и выполнить «грубую» калибровку датчика температуры.

График на рис. 2 оси абсцисс соответствуют серии значений среднего арифметического числа, полученного после дискретизации сигнала с терморезистора. Ось ординат соответствует температуре в градусах Цельсия ($^\circ\text{C}$), полученной с аквариумного термометра. По результатам сформирована

линия тренда и получена линейная зависимость температуры от полученного с АЦП дискретного значения.

Вследствие этого получена формула:

$$y = 0,0262x - 19,95. \quad (1)$$

Данные из таблицы 1 пересчитаны с использованием формулы (1): определены минимальные и максимальные значения, полученные с АЦП (АЦП_{min} и АЦП_{max}), а также вычислено среднее арифметическое (АЦП_μ).

С использованием формулы эти числа преобразованы в градусы Цельсия ($^\circ\text{C}$). Исходные данные и результаты расчётов представлены в таблице 2

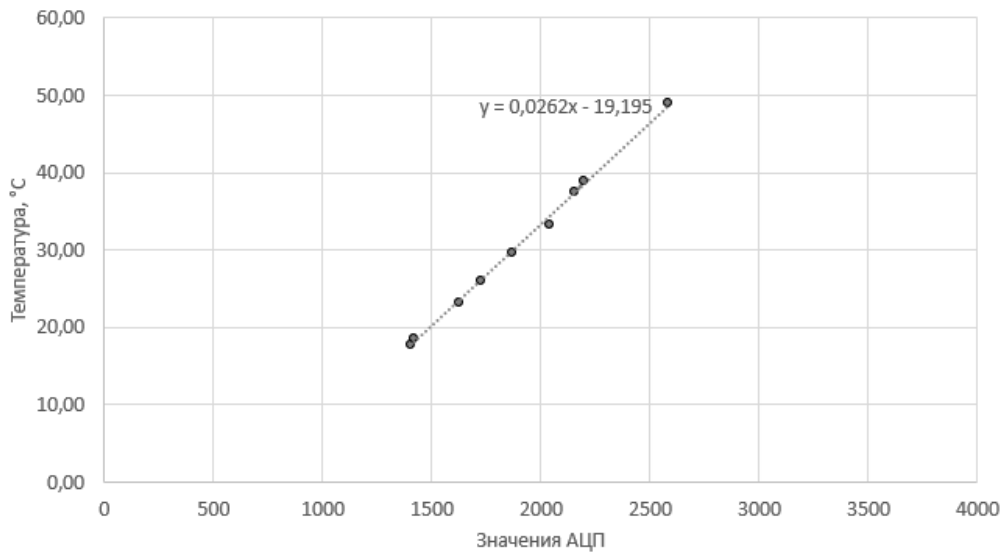


Рисунок 2 – Градуировка терморезистора

Таблица 2 – Данные пересчёта дискретных значений температуры в градусы Цельсия (°C)

№	АЦП _{min}	АЦП _{max}	$\overline{\text{АЦП}}$	$t_{\min}, ^\circ\text{C}$	$t_{\max}, ^\circ\text{C}$	$\bar{t}, ^\circ\text{C}$
1	1382	1415	1402,6	17,0134	17,8780	17,55312
2	1412	1430	1422,9	17,7994	18,2710	18,08498
3	1608	1636	1623,3	22,9346	23,6682	23,33546
4	1720	1753	1729,7	25,8690	26,7336	26,12314
5	1862	1886	1871,1	29,5894	30,2182	29,82782
6	2028	2054	2043,3	33,9386	34,6198	34,33946
7	2145	2173	2158	37,0040	37,7376	37,34460
8	2180	2206	2195,3	37,9210	38,6022	38,32186
9	2559	2598	2584	47,8508	48,8726	48,50580

Из таблицы 2 можно сделать вывод, что погрешность в данном случае достаточно велика и разница между минимальным и максимальным значениями достигает величины вплоть до одного градуса. Тем не менее, благодаря заранее определённому количеству измерений результат в виде среднего арифметического с высокой вероятностью имеет необходимую точность.

Выравнивание интервала

Поскольку входные данные для первой процедуры алгоритма готовы, в рамках эксперимента каждый интервал прошёл процедуру выравнивания. Согласно статье [6] данный этап выполняется в три шага:

– вычисление расстояний до границ интервала (формулы 1, 2):

$$t_{d_1} = |\bar{t} - t_{\min}| \quad (1)$$

$$t_{d_2} = |\bar{t} - t_{\max}| \quad (2)$$

– поиск максимального расстояния из предыдущего шага (формула 3):

$$t_{d_{\max}} = \max \{t_{d_1}, t_{d_2}\}. \quad (3)$$

4): – расширение границ интервала (формула 4):

$$t \in [\bar{t} - t_{d_{\max}}, \bar{t} + t_{d_{\max}}]. \quad (4)$$

Эти шаги и формирование нового интервала для всех серий измерений отражены в таблице 3. Итоговый расширенный интервал отражён в столбцах $t_{\min}$ и $t_{\max}$ ($t \in [t_{\min}; t_{\max}]$).

Таблица 3 – Данные процесса выравнивания интервала

№	$t_{\min}, ^\circ\text{C}$	$t_{\max}, ^\circ\text{C}$	$\bar{t}, ^\circ\text{C}$	$t_{d1}, ^\circ\text{C}$	$t_{d2}, ^\circ\text{C}$	$t_{d\max}, ^\circ\text{C}$	$t_{n\min}, ^\circ\text{C}$	$t_{n\max}, ^\circ\text{C}$
1	17.0134	17.8780	17.5531	0.5397	0.3249	0.5397	17.0134	18.0928
2	17.7994	18.2710	18.0850	0.2856	0.1860	0.2856	17.7994	18.3706
3	22.9346	23.6682	23.3355	0.4009	0.3327	0.4009	22.9346	23.7364
4	25.8690	26.7336	26.1231	0.2541	0.6105	0.6105	25.5126	26.7336
5	29.5894	30.2182	29.8278	0.2384	0.3904	0.3904	29.4374	30.2182
6	33.9386	34.6198	34.3395	0.4009	0.2803	0.4809	33.9386	34.7404
7	37.0040	37.7376	37.3446	0.3406	0.3930	0.3930	36.9516	37.7376
8	37.9210	38.6022	38.3219	0.4009	0.2803	0.4009	37.921	38.7228
9	47.8508	48.8726	48.5058	0.6550	0.3668	0.6550	47.8508	49.1608

Постбинарное кодирование информации

Вторая процедура алгоритма предполагает постбинарное кодирование интервала. Для интервала, границы которого представлены вещественными числами в двоичном формате float, наибольшим образом подходят следующие два формата [10]:

- PostBinary 64/32 p, позволяющий хранить одно вещественное число одинарной точности в виде тетракода;
- PostBinary 128/32 ip, предназначенный для хранения двух вещественных чисел одинарной точности в виде тетракода.

В статье [6] дан обзор первого формата и разобран пример кодирования/декодирования на одном из диапазонов (без выравнивания интервала). Данный формат характеризуется меньшими затратами по памяти, чем второй, а также хорошо показывает себя на интервалах, между границами которых разница не слишком велика.

Таблица 4 – Результаты кодирования и декодирования интервала (формат PostBinary 64/32 p)

№	$[t_{\min}; t_{\max}], ^\circ\text{C}$	$\bar{t}, ^\circ\text{C}$	$t_{\text{postbinary64/32p}}$	$[t_{\text{decmin}}; t_{\text{decmax}}], ^\circ\text{C}$	$\bar{t}_{\text{dec}}, ^\circ\text{C}$
1	[17.0134; 18.0928]	17,5531	0 10000011 00MMAAAAAAAAAAAAAAAAAA M0	[17; 19]	18
2	[17.7994; 18.3706]	18,0850	0 10000011 00MMAAAAAAAAAAAAAAAAAA M0	[16.5; 19.5]	18
3	[22.9346; 23.7364]	23,3355	0 10000011 011MMAAAAAAAAAAAAAAAAAA M0	[22.25; 23.75]	23
4	[25.5126; 26.7336]	26,1231	0 10000011 10MMAAAAAAAAAAAAAAAAAA M0	[25; 27]	26
5	[29.4374; 30.2182]	29,8278	0 10000011 11MMAAAAAAAAAAAAAAAAAA M0	[29; 31]	30
6	[33.9386; 34.7404]	34,3395	0 10000100 000MMAAAAAAAAAAAAAAAAAA M0	[33; 35]	34
7	[36.9516; 37.7376]	37,3446	0 10000100 0010MMAAAAAAAAAAAAAAAAAA M0	[36.25; 37.75]	37
8	[37.9210; 38.7228]	38,3219	0 10000100 001MMAAAAAAAAAAAAAAAAAA M0	[37; 39]	38
9	[47.8508; 49.1608]	48,5058	0 10000100 MMMMAAAAAAAAAAAAAAAAAA M0	[34; 62]	48

Тем не менее, в [10] автором указано, что при конвертировании чисел в постбинарный формат не должен быть затронут порядок вещественного числа, что может иметь место в реальных ситуациях.

Для того, чтобы постбинарное число полностью охватывало интервал, при этом не слишком его расширило, введено дополнительное условие для битов, переходящих в процессе кодирования в серию множественности (M):

- меньшее число интервала должно содержать по крайней мере один бит со значением 1 (истина);
- большее число интервала должно содержать по крайней мере один бит со значением 0 (ложь).

Постбинарное кодирование интервала, а также обратное декодирование с дальнейшим расчётом среднего арифметического отображено в таблице 4.

Из таблицы видно, что порядок вещественного числа не затронут сериями множественности или неопределённости. Тем не менее, в большинстве чисел множественность проявляется в старших разрядах мантиссы. Это обусловлено особенностями представления двоичного числа с плавающей запятой. Данный факт имеет определённое влияние на точность среднего арифметического значения, и в ряде случаев способствует сильному расширению интервала.

Для подобных диапазонов подходит другой формат, отображенный на рисунке 3. Он изначально нацелен на независимое хранение значений обеих границ интервала. Данный формат занимает в памяти 64 тетрита (8 тетрабайт) или, в совместимости с двоичными системами, 128 бит (16 байт). Таким образом, как и в формате PostBinary 64/32 p, каждый разряд представлен двумя двоичными разрядами: 01 – тетраноль; 10 – тетраединица; 00 – неопределённое значение; 11 – множественное значение.



Рисунок 3 – Спецификация формата PostBinary 128/32 ip [10]

В старших разрядах содержится информация о левой границе интервала, в младших – о правой границе. Последние биты представляют собой идентификатор формата числа.

Для экспериментального исследования целесообразности использования данного формата согласно его спецификации

произведено кодирование расширенных интервалов температур. После этого числа в постбинарном формате были декодированы обратно в интервал двоичных вещественных чисел с плавающей запятой. Результаты кодирования интервала в формат PostBinary 128/32 ip с последующим декодированием отображены в таблице 5.

Таблица 5 – Результаты кодирования и декодирования интервала (формат PostBinary 128/32 ip)

№	$t_{\text{postbinary}}$	$[t_{\text{decmin}}; t_{\text{decmax}}], ^\circ\text{C}$
1	0_10000011_000100000011011011100_0_10000011_001000010111110000011_A11M	[17.0134; 18.0928]
2	0_10000011_001000010111110000011_0_10000011_00100101111011011111_A11M	[17.7994; 18.3706]
3	0_10000011_011011101111010000100_0_10000011_011110111100100001001_A11M	[22.9346; 23.7364]
4	0_10000011_100110000011001110011_0_10000011_101010111011110011010_A11M	[25.5126; 26.7336]
5	0_10000011_110101101111111110011_0_10000011_111000110111110111000_A11M	[29.4374; 30.2182]
6	0_10000100_000011111000001001000_0_10000100_000101011110110001010_A11M	[33.9386; 34.7404]
7	0_10000100_001001111001110011100_0_10000100_001011011110011010011_A11M	[36.9516; 37.7376]
8	0_10000100_001011110101111000110_0_10000100_001101011100100001001_A11M	[37.9210; 38.7228]
9	0_10000100_011111101100111001110_0_10000100_100010010100100101010_A11M	[47.8508; 49.1608]

Полученные после декодирования интервалы наглядно показывают, что значения их границ в процессе преобразований практически не претерпевает изменений. Соответственно, сам результат измерений также сохраняет свою точность, полученную после первого этапа алгоритма.

Однако у данного подхода присутствуют и недостатки: для хранения полноценного интервала в тетракоде необходимо задействовать

в два раза больший объём памяти по сравнению с единичным постбинарным числом в формате PostBinary 64/32 p.

Подытоживая результаты экспериментов, можно выделить особенности применения формата PostBinary 64/32 p в контексте анализируемого алгоритма:

- положительные стороны:
- меньший расход памяти;
- отрицательные стороны:

- потеря точности границ интервала и конечного результата при значениях множественности в старших разрядах мантиссы;
- необходимость отслеживания некорректных значения множественности в порядке вещественного числа.

Аналогично и для PostBinary 128/32 ip:

положительные стороны:

- высокая точность границ интервала;
- высокая точность результата вычисления;

отрицательные стороны:

- большой расход памяти.

Из сравнения можно сделать вывод, что целесообразнее использовать второй формат. Однако, несмотря на более высокую точность хранения информации, в контексте огромного количества вычислений всей сенсорной составляющей информационно-компьютерной инфраструктуры повышенный объём занимаемой памяти приведёт не только к необходимости задействовать множество дополнительных хранилищ, но и увеличит нагрузку на сетевые системы.

В связи с этим предлагается использование гибридного варианта использования форматов:

- постбинарный формат PostBinary 64/32 р рационально использовать в тех случаях, когда множественность не затрагивает порядок и старшие разряды мантиссы;

- в случаях, когда под угрозой точность хранения информации, целесообразно пожертвовать дополнительной памятью и использовать формат PostBinary 128/32 ip.

Выводы

В статье рассмотрены отдельные детали реализации алгоритма измерения физических величин. С целью анализа его эффективности проведены новые серии измерений. На основании полученных данных сформированы интервалы, которые в дальнейшем прошли через процедуру постбинарного кодирования.

После декодирования интервала и сравнения с изначальными данными выявлено, что в ряде случаев формат PostBinary 64/32 р может снижать точность результата. Предложено использовать формат PostBinary 128/32 ip в случаях, когда формат PostBinary 64/32 р может приводить к ухудшению качества представления информации.

Литература

1. Койбаш, А. А. Пути повышения эффективности устройств сенсорной составляющей информационно-компьютерной инфраструктуры / А. А. Койбаш // Информатика

и кибернетика. – Донецк: ДонНТУ, 2019. – № 2(16). – С. 51-57.

2. Аноприенко, А. Я. Разработка прототипа суперсенсорного компьютера: особенности реализации и визуализации результатов измерений / А. Я. Аноприенко, Р. Л. Варзар // Материалы пятой международной научно-технической конференции «Моделирование и компьютерная графика», 24-27 сентября 2013 г., Донецк. – Донецк: ДонНТУ, 2013. – С. 218-229.

3. Аноприенко, А. Я. Анализ эффективности устройств сенсорной составляющей образовательной системы умного города / А. Я. Аноприенко, А. А. Койбаш, Е. И. Приходченко // Информатика и кибернетика. – Донецк: ДонНТУ, 2020. – № 1(19). – С. 5-11.

4. Аноприенко, А. Я. Системный анализ закономерностей мирового развития компьютерных систем / А. Я. Аноприенко, К. А. Сидоров, Н. С. Максименко, А. А. Койбаш // Информатика и кибернетика. – Донецк: ДонНТУ, 2019. – № 4(18). – С. 41-48.

5. Аноприенко, А. Я. Закономерности развития инерциальных датчиков информационно-компьютерной инфраструктуры / А. Я. Аноприенко, А. А. Койбаш, Н. С. Максименко, К. А. Сидоров // Информатика и кибернетика. – Донецк: ДонНТУ, 2021. – № 1 - 2 (23 - 24). – С. 14-21.

6. Койбаш, А. А. Разработка модифицированного алгоритма измерения физических величин для сенсорных систем / А. А. Койбаш // Информатика и кибернетика. – Донецк: ДонНТУ, 2021. – № 4(26). – С. 63-69.

7. Аноприенко, А. Я. Введение в постбинарный компьютинг. Арифметико-логические основы и программно-аппаратная реализация / А. Я. Аноприенко, С. В. Иваница. — Донецк: ДонНТУ- УНИТЕХ, 2017. — 308 с.

8. Аноприенко, А. Я. Развитие многомерной логики / А. Я. Аноприенко, С. Г. Джуря, С. В. Иваница // Дельфис, 2016. — № 4(88). – С. 85-91.

9. Аноприенко, А. Я. Пример Румпа в контексте традиционных, интервальных и постбинарных вычислений / А. Я. Аноприенко, В. А. Гранковский, С. В. Иваница // Научные труды Донецкого национального технического университета. Серия «Проблемы моделирования и автоматизации проектирования» (МАП-2011). – Донецк: ДонНТУ, 2011. – Вып. 9 (179). – 356 с.

10. Иваница, С. В. Обоснование закономерностей, арифметико-логических алгоритмов и структур систем компьютерной обработки информации / С. В. Иваница // Диссертация на соискание ученой степени кандидата технических наук. – Донецк: ДонНТУ, 2019. – 226 с.

Койбаш А. А. Анализ эффективности модифицированного алгоритма измерения физических величин для сенсорных систем. Рассмотрены детали реализации модифицированного алгоритма измерения физических величин для сенсорных систем, в основе которого лежит постбинарное кодирование информации. С целью проверки работоспособности алгоритма выполнен анализ его эффективности с использованием реального прототипа сенсорной системы. На основании полученных в опытах данных сформированы интервалы и выполнено их постбинарное кодирование с использованием различных форматов. Определены преимущества и недостатки использования каждого формата. На основании результатов экспериментального анализа предложен гибридный вариант постбинарного кодирования информации.

Ключевые слова: сенсоры, сенсорные системы, алгоритм измерения физических величин, микроконтроллер, постбинарный компьютеринг.

Koibash A. Analysis of the effectiveness of the modified algorithm for measuring physical quantities for sensor systems. The details of implementation are considered for the modified algorithm for measuring physical quantities for sensor systems based on post-binary coding of information. Analysis of effectiveness of the algorithm was carried out using a real prototype of the sensor system in order to check its performance. Intervals were formed based on the data obtained in the experiments and their post-binary coding was performed with the usage of various formats. Advantages and disadvantages of using each format are identified. Based on the results of the experimental analysis, a hybrid version of post-binary information coding is proposed.

Keywords: sensors, sensor systems, algorithm for measuring physical quantities, microcontroller, post-binary computing.

Статья поступила в редакцию 09.03.2023
Рекомендована к публикации профессором Мальчевой Р.В.