



DASpit 23 июля 2008 в 20:13

Язык программирования Форт

Forth, Программирование

Первая реакция у многих людей, с которыми мне приходилось говорить о Форте: «Это, наверное, разновидность Фортрана?». К сожа уровню осведомленности околокомпьютерного сообщества насчет этого совершенно непохожего на остальные языки программир достаточно низок. В этой статье мы попробуем разобраться с тем, что представляет собой Форт, и почему этот язык программир не используется повсеместно.

Окунемся в историю Форта (Forth). В начале 1970х годов Чарльз Мур занялся разработкой программного обеспечения для микроЭВМ, управляющей работой радиотелескопа. Поскольку программирование на Ассемблере привело к неуправляемости разработки довольно сл программы, а для использующейся микроЭВМ из-за аппаратных ограничений не было языка высокого уровня, Чарльз Мур приступил к соз, компактного и мощного языка программирования, позволившего с блеском решить поставленные перед ним задачи. Успех нового языка программирования определился тем, что на ограниченных ресурсах устройства была создана быстрая и функциональная программа, позволившая делать намного больше, чем первоначально планировалось в техническом задании.

Форт (от англ. forth — вперед, получившееся из-за необходимости сократить слово fourth — четвертый, до пяти букв из-за ограничений операционной системы на длину имени файла) в 1976 году стал стандартным языком для программирования для американского астрономического общества. Благодаря высокой степени переносимости и компактности, а также быстройдействию, Форт до 1990х активно применялся во множестве приборов с микроконтроллерным управлением (спектрограф на Шаттле, микромодули управления искусственн спутниками Земли, для системы управления в аэропорту Эр-Рияда, системах компьютерного зрения, автоматизации анализа крови и кардиологического контроля, карманных переводчиках). Форт выступает прародителем такого широко известного языка управления печать PostScript.

По своей структуре Форт представляет собой набор примитивов — слов, которые участвуют в определении других слов. Важная особеннос Форта — использование стека для передачи параметров между словами, такая конструкция позволяет очень гибко и просто реализовывать сложные концепции. Наборы слов, относящиеся к определенной области, могут выделяться в словари. Иерархическая структура словарей позволяет естественно организовывать наследование слов от словаря-родителя. Базовый словарь форта составляет менее четырех десят слов, и уже он позволяет получать полноценные программы расширением исходного словаря.

Форт сам для себя является метасистемой — в нем объединены функции компилятора и интерпретатора, Форт может использоваться без поддержки со стороны операционной системы и использоваться для того, чтобы компилировать самого себя, в том числе и на другие плат

В отличие от большинства языков программирования, которые имеют жесткую структуру, не позволяющую изменять синтаксис и многие ба элементы языка, ничто не мешает на Форте написать модификацию системы, понимающую синтаксис той предметной области, для которо пишется программа. Ярким примером является написание Форт-ассемблера для конкретного процессора, выполняемое за один рабочий д квалифицированным фортером. Мало того, такой ассемблер поддерживает структурное программирование и прозрачную вставку в ассемблерный код слов из словаря самого Форта. Таким же образом реализуется поддержка объектно-ориентированного программирован

До сих пор популярна тема создания Форт-процессоров, система команд которых является машинным представлением наиболее часто используемых слов-примитивов. Благодаря минимализму базовой Форт-системы это легко реализуемо, тем более что программируемые м логических элементов сейчас достигли таких размеров, что Форт-процессор интегрируется на оставшиеся после программирования основ логики 10% вентилях как бонус. Это что-то из разряда такого: «А в углу нашей маленькой и уютной детской комнатки незаметно расположи аквариум с бегемотом».

Легкость написания Форт-систем привела к огромному разнообразию существующих реализаций — не писал свой Форт только ленивый фк Большое количество систем, достаточно плохо совместимых, поскольку имеется три стандарта Форта: Форт-79, Форт-83 и ANSI-Форт-94. Стандарты, хотя и описывают необходимый минимум слов для реализации, но дают огромный простор для самостоятельных изменений, ч повсеместно и происходит. Существуют реализации Форта написанные на Ассемблере, Си, Питоне, Паскале, Яве, а также созданные с пол целевых компиляторов других Форт-систем.

Как не хотелось бы радостно заявлять, что Форт живет полноценной жизнью языка программирования, однако это не совсем так. Форт про занял нишу как язык программирования микроконтроллеров, для которых надо организовать довольно сложную схему функционирования минимуме ресурсов. Однако, примеры написания в современных условиях больших программ немногочисленны, поскольку скорость разр на системах RAD превышает возможности Форт-систем.

Таким образом, по частоте использования, высокоуровневый язык программирования Форт находится даже ниже уровня Ассемблера. При решении проблем, требующих написания ассемблерного кода Форт имеет много преимуществ: компактность кода, структурный ассемблер позволяющий легко внедрять в тело ассемблерных слов высокоуровневые определения. Для тех, кто хочет изучать и использовать Ассемб

бы порекомендовал начать это с изучения Форта, как я в свое время и сделал. Результатом такого метода обучения будет намного более быстрое и качественное освоение Ассемблера, тем более, что после изучения основ вам обязательно захочется сделать свою Форт-систему (как ни говори, практика — кузница умения).

Для меня Форт дал много — после освоения языка я стал замечать, что код в Дельфи стал более коротким и понятным, появилась привычка документировать основные методы и поля объектов (поскольку в Форте очень желательно описывать стековые диаграммы для каждого слайда).

Благодаря изучению книги «Проектирование и программная реализация экспертных систем на персональных ЭВМ» Таусенда и Фохта, я почерпнул много информации о внутренней логике работы таких языков как Лисп и Пролог. Книга Келли и Спайса «Язык программирования: ФОРТ» привела меня к четкому пониманию того, как можно написать свой язык программирования и что такое Форт-ассемблер. Для совершенствования стиля программирования оказалась совершенно неоценимой по влиянию на меня книга Лео Броуди «Способ мышления: Форт».

Форт может дать программисту много полезного, даже если он будет в дальнейшем программировать на другом языке. Форт дает то понятие дисциплины программирования, которое необходимо для создания рабочей и легко сопровождаемой программы. Плохому программисту Форт прощает ошибок, а хорошего стимулирует к тому, чтобы стать великим.

Дмитрий Спицын

Полезные ссылки по теме

- <http://www.forth.org.ru/> — Russian FORTH Interest Group — на сайте размещена большая подборка книг и статей по Форту, много ссылок на существующие Форт-системы, интересующимся очень рекомендуется посетить
- http://www.nncron.ru/index_ru.shtml — продвинутый планировщик задач под Windows, пример коммерчески успешной программы на языке Форт
- <http://winglion.ru/> — самодельный Форт-процессор, сайт содержит достаточно толковый форум по Форту
- <http://www.msyst.ru/quark.htm> — Форт-система с поддержкой OpenGL в виде .dll
- <http://ronware.org/> — кроссплатформенный Форт Windows 32bit (x86) — Linux 32bit (x86) — WinCE (ARM)
- <http://forth.gsfc.nasa.gov/> — список систем NASA, управляемых Фортом

Метки: язык, Форт, Forth, программирование, sdisle.com


 +41



 46

 15,4k

 46



58,4

Карма

0,0

Рейтинг

7

Подписчики

Дмитрий Спицын

@DASpit

Пользователь

Поделиться публикацией

ПОХОЖИЕ ПУБЛИКАЦИИ

19 июня 2015 в 18:42

Go как язык для обучения программированию

 +20

 24,4k

 100

 96

19 ноября 2012 в 21:23

Разомнем мозг при помощи Forth?

 +63

 38,3k

 141

 117

21 февраля 2012 в 20:31

Стековые языки программирования

 +46

 9,3k

 110

 27

РАБОТА В IT-ИНДУСТРИИ

Мой к

Программист PHP (Symfony)

от 80 000


10/02/2018

Язык программирования Форт / Хабрахабр



PlayGround.ru · Москва · Полный рабочий день	
Ведущий full-stack / frontend разработчик	от 160 000
EXERICA · Москва	
Разработчик алгоритмов анализа данных	от 90 000
EXERICA · Москва · Полный рабочий день	
Все вакансии	Создать резюме Разместить вака

Реклама


Комментарии 46

- miros



23.07.08 в 20:44



↑

спасибо! очень интересно.
- Delsian


23.07.08 в 20:45





↑

Когда-то на заре моего программизма (лет 20 назад :) довольно много повозился с Фортом (повезло, были хорошие учителя), но впоследствии этот с

остался невостребованным и забылся. Не так давно попытался работать в своей новой предметной области (микроконтроллеры) с Фортом отсюда <http://krue.net/avrforth/> - но уже мозги в другую сторону повернуты, не то.

ИМХО, если человек уже испорчен сями и джавами - то вряд ли из него получится хороший Форт-программер.
- DASpit





23.07.08 в 20:55




↑

Изучение Форта может быть полезно любому программисту - появляется новый взгляд на многие вещи, тем более если человек "испорчен" С и Яв
- nerezus


23.07.08 в 23:45



↑

С и Ява - это все равно, что каменный топор и перфоратор. Разница все же очень большая будет. Как по мне - то больше, чем между С и форт
- kibitzer




23.07.08 в 21:15




↑

Для тех, кто хочет попробовать свои силы, могу посоветовать sp-forth (spf.sourceforge.net). Как раз ncscop написан на нем, если не ошибаюсь, также и ряд других успешных софтин. Много примеров, есть версия, как для виндоус, так и для линукс. Еще интересно, что он сам по сути написан на фор проект опенсорсный, так что полезно и его код поизучать, чтобы лучше понять форт. Под винду есть примеры с гуи, с опенжл и работы с винапи. Огромное кол-во дополнительных либ(словарей), примеров работы с фортом(практических). Сам когда-то делал словарь для гтк(правда тоже под ви тогда у них еще не было линуксовой версии).
- DASpit



23.07.08 в 21:36



↑

Великолепный проект под девизом "лучшая документация к программе - исходный код самой программы" - сам в свое время провел немало вече изучая исходники СП-Форты, узнал много нового. СП-Форт помог мне разобраться с WinAPI. Андрею Черезову респект за продолжающуюся работ своим детищем.
- ayc


18.06.09 в 21:35



↑

Служу Советскому Союзу :)
- Tagire




23.07.08 в 21:19




↑

Единственное, что я помню про Форт, так это шутку про Йоду)
- wolf




23.07.08 в 22:14



↑

Эта шутка применима не только к Форту но и к Лиспу и к любому другому языку где используется обратная польская нотация (да хоть язык микрокалькулятора БЗ-23)
- Recluse


24.07.08 в 00:38



↑

<http://bash.org.ru/quote/5592>
- IZENfire

23.07.08 в 22:07






↑

Мечта: написать Java Virtual Machine на Forth.

 **DASpit** 24.07.08 в 11:08    


Часть в девичестве Форт-процессоров волшебным образом превратились в Ява-процессоры, когда пришло время свататься (потребовалось найти широкий рынок сбыта).

 **iZENfire** 23.07.08 в 22:20  






Для FreeBSD нашёл порты:
fcl-4.0.31_1 — Forth Inspired Command Language
gforth-0.6.2_2 — Fast and portable Forth system
pfe-0.32.94_1 — Implementation of ANSI Forth
С чего начать?

 **DASpit** 24.07.08 в 11:14    

С теории. Большинство нужных книжек и тексты стандартов Форты вы сможете найти на <http://www.forth.org.ru/>

 **TMC** 23.07.08 в 23:23  

Дома книжка валяется про Форт. Помню в детстве что-то кодил на реализации для ZX-Spectrum-совместимого компьютера. А сейчас, кстати, Форт используется в загрузчике FreeBSD.

 **andrewmaster** 24.07.08 в 00:09    

Более того, он использовался в firmware на powerpc маках (эпплдрочеры, а вы это знали?)

 **chs** 23.07.08 в 23:28  

странно, но <http://www.forth.org.ru/> не отвечает

 **DASpit** 24.07.08 в 11:18    

<http://www.forth.org.ru/> живее всех живых. А насчет недоступности - это, видно, был хабраэфект - слишком много хабраюзеров захотело разобрать Форт.




 **calx** 23.07.08 в 23:29  

Ох! Вспомнилось как забивал из приложения к "Юному Технику" Форт-транслятор напечатанный, как тогда водилось, в шестнадцатичном коде. Это испортили мне немало нервов. Потом еще неделю искал ошибки и опечатки... Хорошее было время.

 **watabou** 23.07.08 в 23:46  

Читал книги по Форту в школе, когда постоянного доступа к компьютеру еще не было, а интерес к теме уже был :) В то время идеи этого языка меня впечатлили сильнее, чем других "нестандартных языков" - Лиспа и Пролога.

НЛО прилетело и опубликовало эту надпись здесь

 **romychs** 24.07.08 в 10:16    

Буквально два дня назад перерывал свои архивы и наткнулся на свой интерпретатор форты, я его на 2-м курсе (1991г.) на турбо-паскале написал полную реализацию, но что-то там работало). Эх... Ностальгия...




 **kibitzer** 24.07.08 в 22:48    

Как-то поспорил, что сделаю мини ОС, которая будет иметь примитивный шелл с языком программирования, на котором можно писать проги и работать с диском(дискетами). Условие было уложиться в 1кБ, включая загрузчик(+простой редактор командной строки) :) Победил, по сути на асме Форт с набором нужных слов :)

 **DASpit** 24.07.08 в 11:27    

Предлагаю вам, как знающему человеку, просветить народ. Благо теперь вам кармы хватает для написания в личный блог. Если людям понравится топик, ваша карма быстро превысит 5 и вы сможете перейти топик в тематический блог.

НЛО прилетело и опубликовало эту надпись здесь

 **SCoon** 24.07.08 в 00:58  

Thanx. Читал с удовольствием. Но есть нюансы. Например:

Важная особенность Форты — использование стека для передачи параметров между словами, такая конструкция позволяет очень гибко и просто реализовывать сложные концепции.

Практически все распространенные современные архитектуры используют стек для передачи параметров между подпрограммами. Особенность форт в данном случае в другом: он использует *два независимых стека*: стек возвратов и стек данных. Именно это лежит в основе его элегантной схемы.

P.S. Странно было не услышать ни слова про Color Forth. И жаль, что не раскрыта тема DSL.

 **DASpit** 24.07.08 в 11:49    

В маленькой статье невозможно описать все, а писать книгу-баян не хочется. Для более подробного изучения проблемы я дал достаточно ревалс теме ссылки.

Пожалуйста, поделитесь в комментариях ссылками на интересные вещи, благо хабр такое позволяет.

 **SCoon** 24.07.08 в 12:49    


Делюсь: [Color Forth](#), последняя версия языка лично и персонально от Чака Мура.

В частности, по этой ссылке есть замечательная фраза об особенностях Форты:






Distinctive for its use of 2 push-down stacks. The Return stack is used for subroutine return addresses, as usual. The Data stack holds parameters for results of subroutine calls. This distinction between control and data minimizes the cost of subroutine calls. As a result, Forth code is typically highly factored into many tiny subroutines, called words. Referencing a word causes its code to be executed.

 **Koc** 24.07.08 в 01:10  

а можно программку, которая выводит на экран "Hello, World!"?

 **fatal** 24.07.08 в 01:36    

```
<<hello_world.fs>>=
." Hello World!" CR
(взято отсюда)
```

 **fatal** 24.07.08 в 01:44    

Так же смотрите урок по Форту для OLPC, с примером "Hello, world!"

Кстати, встроенная программа (firmware) для этого устройства написана на Форте, и является открытой.

 **ksurent** 24.07.08 в 02:00  

Неплохо было бы привести хоть какие-то примеры кода сюда

 **stager** 24.07.08 в 06:15    

ipcalc для DragonForth - это который на PalmOS:

```
( ipcalc stager 29.04.05 )

: .bin ( word - ) 10 swap 2 base ! u. base ! ;

: / ( a b c d - word ) swap 256 * + swap 256 256 * * + swap 256 256 256 * * * + ;

: (hosts) ( /mask - hosts ) 32 swap - 2 swap ** 2 - ;

: mask ( /mask - word ) 32 swap - 1- 1 swap 0 do 2* 1+ loop not ;

: (naddr) ( word /mask - word ) mask and ;

: (faddr) ( word /mask - word ) mask and 1+ ;

: (baddr) ( word /mask - word ) dup (hosts) rot rot (faddr) + ;

: (laddr) ( word /mask - word ) (baddr) 1- ;

: .addr ( word - ) 256 u/mod 256 /mod 256 /mod . ." . " . ." . ." . ." u. ;

: bin.addr ( word - ) 256 u/mod 256 /mod 256 /mod

.bin ." . " .bin ." . " .bin ." . " .bin ;
```

```
: net over over (naddr) ." net work: " .addr ;

: first over over (faddr) ." first addr: " .addr ;

: last over over (laddr) ." last addr: " .addr ;

: bcast over over (baddr) ." broadcast: " .addr ;

: hosts over over (hosts) ." hosts: " u. drop ;

: bin dup cr bin.addr ;

: ipcalc cr net cr first cr last cr bcast cr hosts ;
```

а вот и результат работы:

```
10 17 28 16 / 28 ipcalc

net work: 10 .17 .28 .16

first addr: 10 .17 .28 .17

last addr: 10 .17 .28 .30

broadcast: 10 .17 .28 .31

hosts: 14 ok

192 168 233 91 / 30 ipcalc

net work: 192 .168 .233 .88

first addr: 192 .168 .233 .89

last addr: 192 .168 .233 .90

broadcast: 192 .168 .233 .91

hosts: 2 ok

10 233 16 40 / 10 ipcalc

net work: 10 .192 .0 .0

first addr: 10 .192 .0 .1

last addr: 10 .255 .255 .254

broadcast: 10 .255 .255 .255

hosts: 4194302 ok

255 255 255 240 / bin

11111111 .11111111 .11111111 .11110000 ok

10 17 28 16 / bin

1010 .10001 .11100 .10000 ok
```



klimin007 24.07.08 в 05:57



Когда-то в начале девяностых заказывал кассету с Фортом для Хобби ПК8000 на КР580ВМ80А... Но кассета затерялась на почте... Вот так я и не познакомился с этим языком, а ушел в ассемблер...



wolf 25.07.08 в 08:41



А набить из двоичных кодов в "Мониторе"? Там всего 4К кода было. ПК8000 это ж галимый Радио РК-86, ну может контроллер дисплея немного другой...

 mikhanoid 24.07.08 в 08:06


Занятно читать о возможности добавлять собственный синтакс в язык, у которого никакого синтаксиса нет. Плюс оценка непонятная какая-то сложная реализации интерпретатора в железе. По идее, нужен полноценный стековый процессор, которые действительно являются очень простыми, но по сложности процессоростроения. То есть, это как минимум, несколько десятков тысяч транзисторов на ядро, плюс интерфейс к памяти или сама память. В итоге так уж и мало получается.

Вот. И ещё. Не понятно, зачем людей призывать тратить время на такие вот решения из прошлого, которые вообще-то могут и подпортить мозг в контексте требований к решениям из настоящего: стековые машины не параллелятся и dataflow в них, грубо говоря, тупой и линейный. Нужно ли развивать мозг в этом направлении в эпоху всеобщей многоядерности?

 DASpit 24.07.08 в 12:04

Приветствую ваш скептический взгляд - это немного отрезвит тех, кто захочет попытаться применять Форт не понимая баланса его хороших и не совсем хороших сторон.

Попробуйте форум на <http://winglion.ru/> - возможно, там найдется кому ответить на вашу критику.

 Abaza 24.07.08 в 08:12

Познакомился с этим языком программирования при освоении планировщика pncron. Сначала чисто психологически было неудобно работать из-за:

В языке программирования Форт принята т. н. "**постфиксная нотация**" (ее еще иногда называют "обратной польской записью"). Это означает, что операции (логической или арифметической) в нем располагается **после** чисел

И приведу пару ссылок из хэлпа к программе:

Специально для тех, кто хочет побольше узнать о Форте, автор pncron перевел в "электронный" вид книжку Leo Broudie Starting Forth (Лео Броуди "Начальный курс программирования на языке Форт"). Это очень доступный текст на русском языке - <http://www.nncron.ru/download/sf.chm> (1,5Mb).

Настоятельно советую всем, кто интересуется Фортом, скачать краткий справочник по основным словам SP-Forth (http://www.nncron.ru/download/spf_help.z...) исходники pncron (<http://www.nncron.ru/download/src.rar>) и SP-Forth 3.75 (<http://www.enet.ru/win/cherezov.forth...>) - в них хорошо документированы большинство слов, которые вам могут понадобиться.

 Inskin 24.07.08 в 09:44

Да, помню, был в шоке, когда узнал, что двум можно присвоить значение три, и при умножении двух на два получать девять :) Потому что в Форте именами переменных могут быть цифры :)

 krysanov 24.07.08 в 10:15

Помнится много-много лет назад, когда компьютер у меня дома был большой и неподъемный, наткнулся в одном из журналов Радио на распечатку и интерпретатора Форты для Z80 (если не ошибаюсь). Перевел коды в ассемблер и даже пытался переписать под x86. Что-то даже работало :) Но вид юношеского задора и опыта не хватило на завершение работы до конца. Но язык оставил приятные впечатления, когда позже я немного "потрогал" и действии.

 amduscias 24.07.08 в 10:51

Спасибо за интересную статью! Добавил в избранное

 Krovosos 24.07.08 в 11:58

Я написал на Форт пейджинг-систему. Форт (SMAL32) был написан моим знакомым. Система успешно продавалась. Форт работал под DOS в защищенном режиме. Система была довольно устойчива.

Основная проблема при программировании больших систем на форте - необходимость постоянно контролировать состояние стека после работы сл. Поскольку сигнатура стека для отдельных слов может различаться - это непросто. Если что-то вдруг оставляет что-то случайно на стеке - привет программе! Мой знакомый делал DROPALL в главном цикле :-))) Я так не могу - я все-таки старался аккуратно писать.

 DASpit 24.07.08 в 12:08


"Форт дает то понятие дисциплины программирования, которое необходимо для создания рабочей и легко сопровождаемой программы. Плохому программисту Форт не прощает ошибок, а хорошего стимулирует к тому, чтобы стать великим."

 Krovosos 24.07.08 в 13:08

Вот тут есть версии SMAL32 под WIN2K, если кто-то хочет поиграться...

<http://www.forth.ru/smal32.html>

 24.07.08 в 14:14

 **Burchik** 29.07.00 в 14:14

Будучу студентом прочитал книжку по Форт. Очень заинтересовал язык, но толком попробовать было не на чем.

 **rw6hrm** 31.12.15 в 23:39

Жаль, что не вспомнили про однокристалльные ФОРТ-контроллеры от Rockwell 65F11 / 65F12. Весьма приличные чипы с ядром от 6502.

Только полноправные пользователи могут оставлять комментарии. Войдите, пожалуйста.

ИНТЕРЕСНЫЕ ПУБЛИКАЦИИ

Bloomberg: как акция Илона Маска по продаже огнеметов изменит финансирование стартапов

 +5


 44

 5

 0

Как в реальности патч от уязвимостей Meltdown и Spectre влияет на производительность GT

 +6

 3,3k

 5


 30

Как я переехал в ЕС: легализация, изучение языка, поиск жилья и работы

 +6

 2,3k

 21

 10

Выпуск#10: ITренировка — актуальные вопросы и задачи от ведущих компаний

 +5


 628

 12

 5



Тutorial по Unreal Engine: C++

 +12

 1,2k

 27

 4

Аккаунт	Разделы	Информация	Услуги	Приложения
Войти	Публикации	О сайте	Реклама	<div> </div>
Регистрация	Хабы	Правила	Тарифы	
	Компании	Помощь	Контент	
	Пользователи	Соглашение	Семинары	
	Песочница	Конфиденциальность		
<div> © 2006 – 2018 «TM»</div>		Служба поддержки	Мобильная версия	