



Об история СПО

Pascal, C, TSS Unix v6, v7, Unix BSD 4.2

Терабайты кода OS/360, RSX-11M, RT-11, ОС Диспак, Диапак, МС Дубна...

Сейчас никого не волнуют эти терабайты, они исчезли вместе с сотнями тысяч программных комплексов, практически без следа.

PDP-11 (СМ-3, СМ-4, Электроника-60, м-400...)

+

Наличие и качество документация по использованию и по реализации очень важны для жизненного цикла СПО (и КПО, и просто ПО).



40 лет спустя...

**В прошлом остались перфоленты, перфокарты, магнитные ленты и барабаны, а также гибкие диски от 3,5 до 8 дюймов. Жёсткие диски супер на 100 Мб и не супер на 2,4Мб. Их размеры.
Телетайпы и видеотоны.**

Изучении естественных дисциплин начинают с далёкого прошлого, геометрию вообще чуть ли не с греков начинают изучать - Пифагор, Евклид, Герон...



Пролог, но не как язык программирования

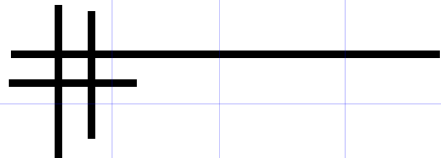
Про C и Pascal.

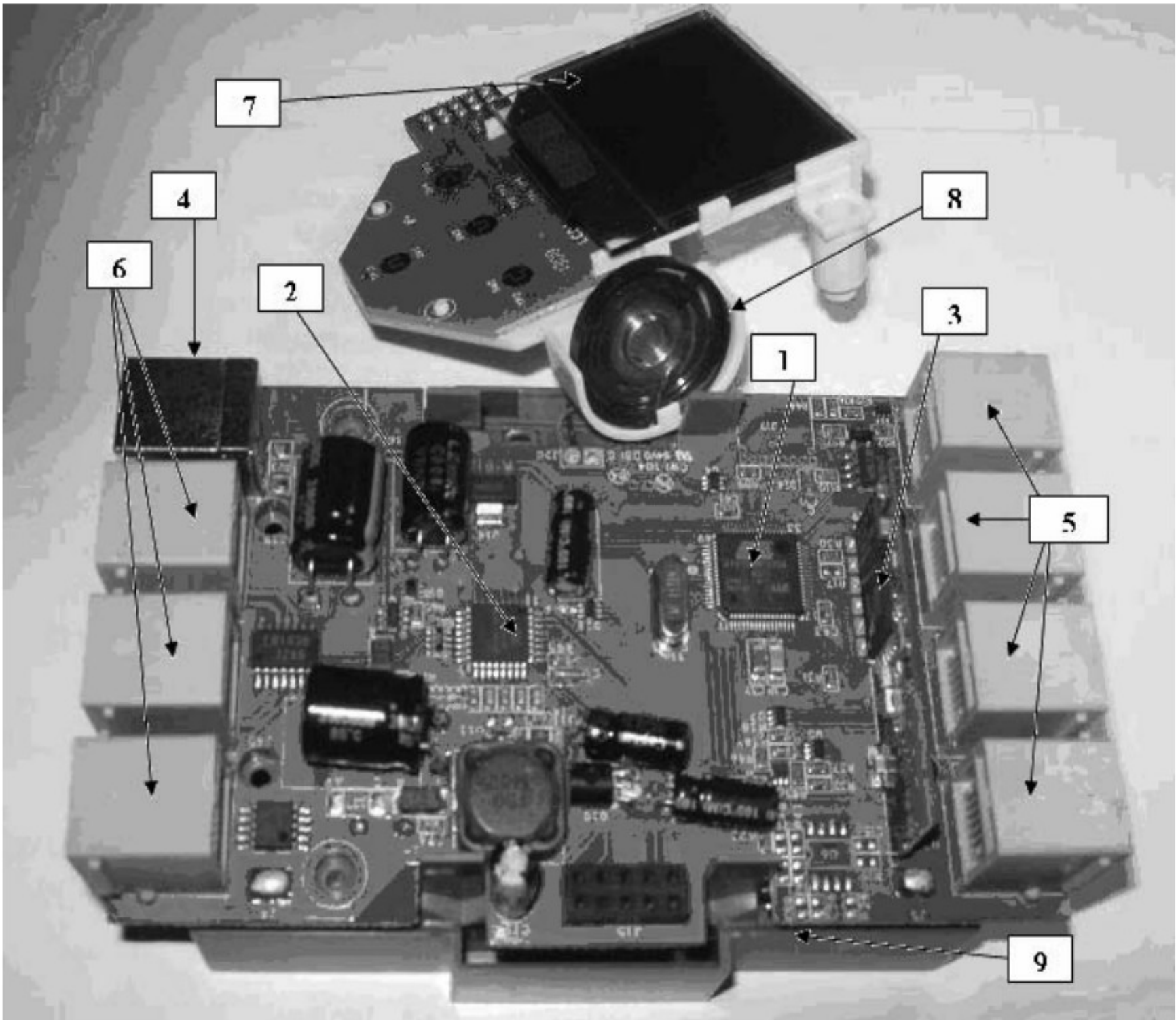
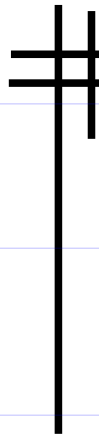
C — очень успешный язык, возможно благодаря супер успеху «Реализации системы Unix». При этом переносимость этой системы в некоторых случаях сводилась к переносу компилятора этого языка.


Хотя известна платформа на которой C не выдержал конкуренции с Pascal. Это Apple-2. UCSD Pascal против Aztek C.

Рассмотрим небольшую проблему: выбор программных средств для NXT (NXT-2). Компилятора с языка Pascal не оказалось, что не так уж плохо.

Что из себя представляет комплект NXT?







Внутри Базового узла расположены ниже перечисленные элементы:

- 1 32-bit ARM7 AT91SAM7S256 микроконтроллер 256 Kbytes FLASH, 64 Kbytes RAM
- 2 8-bit AVR микроконтроллер Atmega 48 4 Kbytes FLASH, 512 Byte RAM память
- 3 Беспроводной канал Bluetooth (Bluetooth Class II V2.0 compliant)
- 4 USB скоростной порт (12 Mbit/s)
- 5 4 порта входа, 6-wire cable digital platform (Один порт включает IEC 61158 Type 4/EN 50 170 порт расширения для использования в дальнейшем)
- 6 3 порта вывода, 6-wire cable digital platform
- 7 100 × 64 pixel LCD графический дисплей
- 8 Динамик – 8 kHz качество
- 9 Питание: 6 AA батареек



Что хотелось бы от ПО для NXT? Задачи более чем скромные:

- заинтересовать программированием,
- познакомить с робототехникой,
- использовать оборудование.

Сразу выяснилось, что фирменное ПО на базе LabView NXT-G вызывает некоторые затруднения. Массивы, списки, арифметика не очень удобны в этом визуальном языке.

Сама система LabView — коммерческая, и довольно объёмная.

Остановились на двух вариантах. С и Java (NXC и NXJ). В обеих реализациях присутствует многопоточность, без которой работы вряд ли можно программировать эффективно.

Пока был один системный блок и программировал один школьник. Использовали NXC. Возникали проблемы с отладкой, и программирование продвигалось с трудом.



NXJ - LeJOS

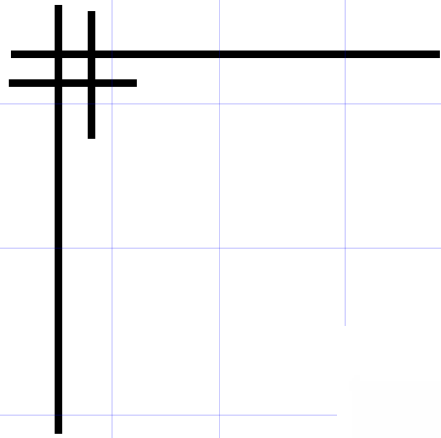
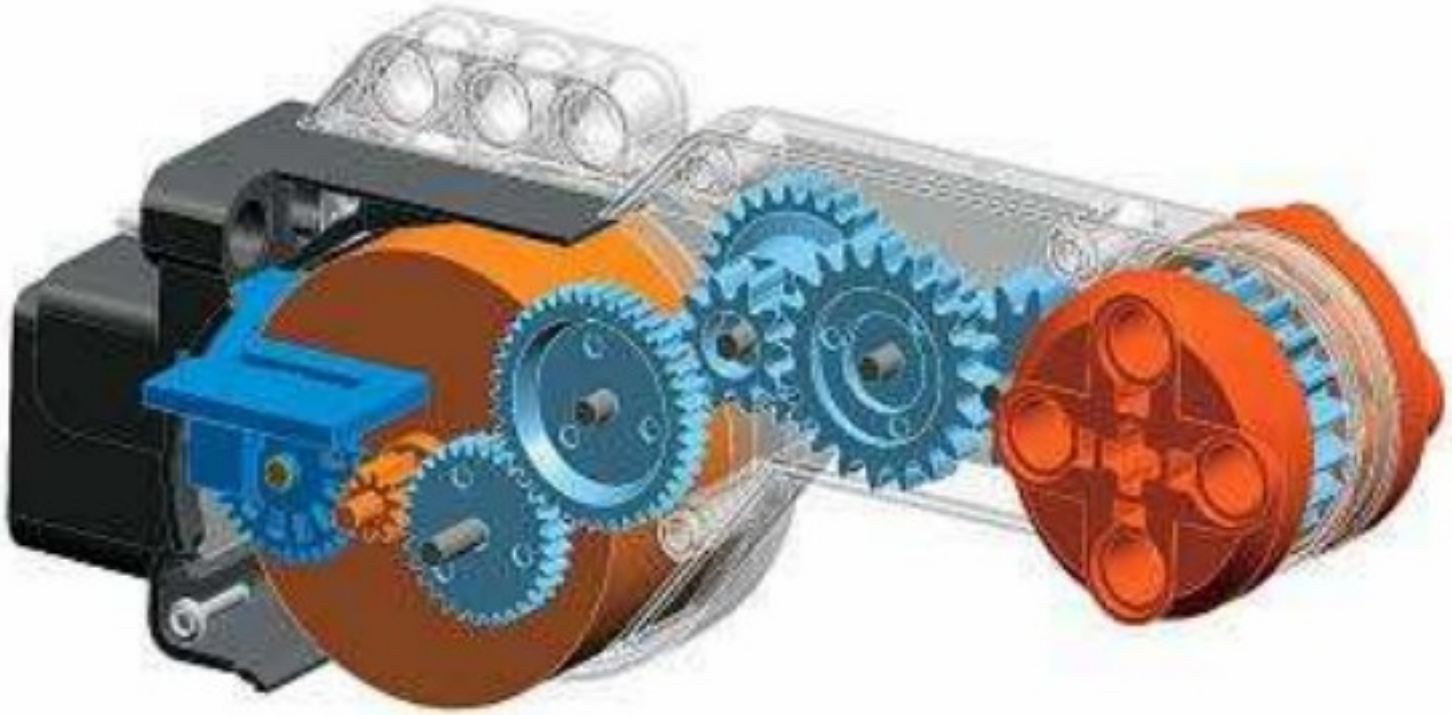
Появление трёх дополнительных блоков позволило расширить количество участников. Решили попробовать на Java. Виртуальная машина содержится в прошивке. И в принципе она удобнее для занятий, чем фирменная прошивка.

Имеется возможность программировать из командной строки, без проблем можно использовать Eclipse.

Отладка программ примерно так же проблемна, как и для NXC.

Но ощущается довольно сильная методическая поддержка со стороны сообщества разработчиков.

Подводить итоги рано, хотелось бы только отметить, что при знакомстве с программированием на Java, интерес учащихся 4-6 класса был значительно выше с NXT, чем без него. В небольших программах они осуществляли управление моторами.







NXT Brick & LeJOS



4 NXTe
16 LSC



160 RC
Servos