

ВИКИПЕДИЯ

# Intel 80386

Материал из Википедии — свободной энциклопедии

**Intel 80386** (также известный как *i386* или просто *386*) — 32-битный микропроцессор с архитектурой x86 третьего поколения фирмы Intel, выпущенный 17 октября 1985 года. Данный процессор был первым 32-разрядным процессором для PC. Применялся, преимущественно, в настольных и портативных ПК. В настоящее время в основном используется в контроллерах.

## Содержание

### Описание

[32 бита](#)

[Страничное преобразование](#)

[Улучшенная поддержка многозадачности и защиты](#)

[Виртуальный режим](#)

[Аппаратная отладка](#)

[Пошаговое выполнение](#)

[Точки останова](#)

[Ловушка переключения задачи](#)

[Другие изменения](#)

[Регистры](#)

[Набор инструкций](#)

[Математический сопроцессор](#)

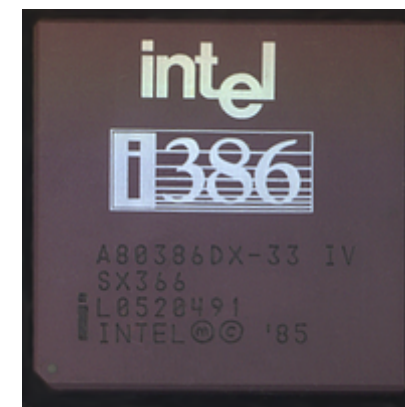
### Модели

[386DX](#)

[386SX](#)

## Intel 80386

*Центральный процессор*



Микропроцессор Intel 80386

<b>Производство</b>	с 17 октября 1985 по сентябрь 2007 <sup>[1][2]</sup>
<b>Разработчик</b>	Intel
<b>Производители</b>	Intel Corp. AMD
<b>Частота ЦП</b>	12—40 МГц
<b>Технология производства</b>	1.5—1.0 мкм
<b>Наборы инструкций</b>	x86



Вся архитектура x86 была расширена до 32 бит — все регистры (за исключением сегментных) стали 32-битными, получив в названии префикс «E» (EAX, EBX, EIP, EFLAGS и т. п.), с сохранением полного набора команд для работы с ними. В том числе:

- регистр флагов, получивший множество новых флагов для управления многозадачностью
- регистр управления процессором MSW процессора 80286, названный в i386 «CR0»

32-битной стала и адресация в защищённом режиме (с возможностью создания 16-битных сегментов для совместимости с 80286). Она позволила впервые со времени появления 8086 забыть о *сегментации*, а точнее, ограничении размера *сегмента* 64 килобайтами (ограничение 16-битного адреса), которое давно перестало устраивать программистов.

До появления i386 программы и операционные системы использовали несколько головомных *моделей организации памяти* (крохотная — tiny, малая — small, большая — large, огромная — huge), различающихся по организации в памяти сегментов кода, стека и данных. 32-битный адрес позволил использовать вместо них одну простую *плоскую* модель (англ. *flat*) — 32-битный вариант *крохотной* модели, в которой все сегменты *задачи* находятся в одном адресном пространстве. Плоская модель обеспечивает размер такого «общего» сегмента до почти 4 гигабайт, которых по тем временам хватало для любой мыслимой задачи.

Плоская модель имеет и недостатки:

- в ней появляются проблемы *перемещаемости машинного кода*, которые ранее легко обходились сегментацией, обеспечение перемещаемости легло на плечи операционных систем с новыми форматами дискового образа программы.
- плоская модель практически сводит на нет управление памятью в защищённом режиме (ограничение доступа и поддержка виртуальной памяти), которое до i386 могло выполняться только на уровне сегментации. Только появление новой модели управления памятью — *страничного преобразования* — обеспечило плоской модели её сегодняшнюю популярность.

Плоская модель вошла в обиход столь широко, что современные программисты зачастую и не подозревают, что программы обращаются в память через сегменты.

## Страничное преобразование

В i386 был введён новый механизм управления памятью — адресное пространство, к которому обращается процессор за данными и кодом, в котором располагаются сегменты (названное *линейным* адресным пространством) может не соответствовать реальной *физической* памяти. Физическая память (включая буфера внешних устройств, например, *видеобуфер*) может быть отображена в линейное адресное пространство произвольным образом — каждая *страница* (размером 4 килобайта) линейного пространства может быть переадресована на любую страницу физической памяти через *каталог страниц*, располагающийся в оперативной памяти (адрес каталога страниц определяется значением нового регистра управления «CR3»).

Как и сегменты, страницы линейного адресного пространства могут быть объявлены *неприсутствующими* (обращение к таким страницам вызывает обработчик *страничного нарушения* операционной системы). Неприсутствующие страницы, в первую очередь, используются для организации виртуальной памяти — обработчик страничного нарушения осуществляет *свопинг* страниц памяти со внешними запоминающими устройствами. Также неприсутствующие страницы используются в плоской модели памяти (где размер сегмента обычно имеет размер от 2 до 4 гигабайт, даже если у компьютера не имеется столько физической памяти) для маркировки страниц сегмента, в которые операционная система не выделила памяти. В этом случае страничное нарушение обычно завершает задачу или начинает её отладку.

Через страничное преобразование i386 может адресовать до 4 Гбайт физической памяти и до 64 Гбайт виртуальной памяти.

## Улучшенная поддержка многозадачности и защиты

Поддержка *многозадачности* в процессорах x86 обозначает аппаратную поддержку «прозрачного» переключения с одной обычной программы (*задачи*) на другую. При переключении процессор сохраняет своё состояние (включая адрес следующей команды, *селекторы* сегментов) в *сегменте состояния* (TSS; сегмент памяти, с селектором из регистра TR) одной задачи, после чего восстанавливает состояние другой задачи из её сегмента состояния (селектор сегмента состояния новой задачи загружается из *дескриптора* её сегмента кода).

Переключение между задачами обычно осуществляется:

- прерыванием таймера; время, на которое настроен таймер, называется *квантом* времени для задачи
- системным вызовом (вызовом функции операционной системы)
- исключением — например, при попытке выполнить недопустимую команду или обращении к неприсутствующей памяти
- отладкой

В i386 механизмы защиты и многозадачности были значительно расширены и улучшены. В зависимости от характера нарушений они могут тихо игнорироваться (например, некоторые биты регистра EFLAGS нельзя изменить загрузкой флагов из стека), вызывать обработчик исключения (операционной системы). Серьёзные ошибки на уровне операционной системы (или в реальном режиме) могут привести процессор в режим аварийной остановки (например, при нарушении в обработчике двойного нарушения), из которого можно выйти только аппаратным сбросом (англ. *reset*) процессора.

Например, i386 поддерживает ограничение доступа к портам ввода-вывода и флагу запрещения прерываний через:

- назначение необходимого уровня привилегий для выполнения таких команд (двухбитным полем IOPL регистра флагов)
- разрешением задаче выборочного доступа к портам ввода-вывода через битовую карту в сегменте состояния задачи

Попытка исполнения неразрешённой команды ввода-вывода приводит к исключению, обработчик которого (принадлежащий операционной системе) может завершить задачу по ошибке, игнорировать (возобновить выполнение со следующей команды) или эмулировать ввод-вывод.

Кроме всего, многозадачность i386 полностью поддерживает все новые возможности — для 32-битных задач сегмент состояния содержит все 32-битные и необходимые новые регистры (например, регистр CR3 с адресом каталога страниц для этой задачи).

## Виртуальный режим

В процессоре i386 компания Intel учла необходимость лучшей поддержки реального режима, потому что программное обеспечение времени его появления не было готово полностью работать в защищённом режиме. Поэтому, например, в i386 возможно переключение из защищённого режима обратно в реальный (при разработке 80286 считалось, что это не потребуется, поэтому на компьютерах с процессором 80286 возврат в реальный режим осуществляется схемно — через сброс процессора).

В качестве расширенной поддержки реального режима i386 позволяет одной или нескольким задачам работать в *виртуальном режиме* — режиме эмуляции режима реального адреса.

Важно понимать, что «виртуальный режим», несмотря на похожесть названия, является не «третьим режимом работы процессора» (то есть реальный, защищённый и виртуальный), а лишь режимом работы *задачи* в многозадачном окружении защищённого режима.

Виртуальный режим предназначается для одновременного выполнения программ реального режима (например, программы для DOS) под многозадачной операционной системой защищённого режима.

Выполнение в виртуальном режиме практически идентично реальному, за несколькими исключениями, обусловленными тем, что виртуальная задача выполняется в защищённом режиме:

- виртуальная задача не может выполнять привилегированные команды, потому что имеет низший уровень привилегий
- все прерывания и исключения обрабатываются операционной системой защищённого режима (которая, впрочем, может инициировать обработчик прерывания виртуальной задачи), вместе с тем в задаче виртуального режима можно использовать:
- страничное преобразование, например, для:
  - расширения памяти путём включения страниц в неиспользуемое адресное пространство
  - эмуляции расширений с переключением банков (например, EMS-памяти)
  - виртуальной развёртки или свёртки буферов внешних устройств (видеопамять, аппаратная EMS-память)
- эмуляцию внешних устройств через эмуляцию портов ввода-вывода
- отладку

- при выполнении нескольких задач виртуального режима каждая из них может выполняться совершенно отдельно друг от друга, чего нельзя достигнуть в реальном режиме

Например, начиная с версий 4.01 — 5.0, при использовании менеджера памяти EMM386 (и его аналогов других разработчиков) операционная система MS-DOS работает как задача в виртуальном режиме. EMM386 в этом случае является подобием операционной системы защищённого режима (передавая большинство системных прерываний ядру MS-DOS в виртуальной задаче).

Режим виртуального 8086 поддерживается и в последующих 32-битных процессорах x86 вплоть до режима совместимости в x86-64.

## Аппаратная отладка

Как и в предыдущих процессорах (начиная с 8086), отладка в i386 осуществляется вызовом отладочного прерывания, обработчик которого передаёт управление программе-отладчику. В предыдущих процессорах отладка могла быть вызвана двумя событиями:

- *пошаговое выполнение*
- программные *точки останова*

в i386-отладку также могут начать:

- аппаратные *точки останова*
- *ловушка переключения задачи*

## Пошаговое выполнение

Пошаговое выполнение или трассировка (вызов отладочного прерывания после выполнения каждой команды при установленном флаге TF) в i386 осталось таким же, как и в предыдущих процессорах, работая во всех новых режимах (32-битной адресации и режиме виртуальной задачи).

## Точки останова

Точка останова (англ. *breakpoint*) — прерывание выполнения программы по достижении определённого адреса.

В ранних процессорах x86 для организации точки останова было необходимо заменить первый байт команды на команду вызова отладочного прерывания (int 3), вызов прерывания возвращал управление отладчику, который возвращал исходное значение первого байта команды. Кроме всего, точки останова такого рода было невозможно устанавливать в ПЗУ. i386 поддерживает такие точки

останова для совместимости.

Точки останова i386 определяются адресом в регистрах отладки. Обработчик отладки вызывается, если адрес следующей команды совпадает с адресом в регистре. В связи с тем, что такие точки останова не требуют записи в память, они могут использоваться для отладки программ в ПЗУ. i386 позволяет одновременно устанавливать до четырёх точек останова.

Точки останова i386 также могут быть установлены на обращение (чтение или запись) к переменной по заданному адресу в памяти. Процессор вызывает обработчик отладки при обращении команды к памяти по заданному адресу.

### Ловушка переключения задачи

Отладочное прерывание может быть вызвано при переключении на задачу (для этого в сегменте состояния задачи предусмотрен специальный бит).

### Другие изменения

Линии данных и адресов в процессоре 80386, как и в процессоре 80286, не мультиплексируются: имеется 32 линии данных и 32 независимых от них линии адреса.

### Регистры

В составе микропроцессора имеются 8 32-битных регистров общего назначения (EAX, EBX, ECX, EDX, ESI, EDI, EBP, ESP). Младшая, 16-битная, половина каждого из этих регистров соответствует регистрам AX, BX, CX, DX, SI, DI, BP, SP в предыдущих процессорах семейства x86. Как и раньше, возможна адресация младших и старших 8-битных половин, младшей 16-битной половины 32-разрядных регистров данных (AL и AH, BL и BH, CL и CH, DL и DH). Регистр указателя команды и регистр флагов также стали 32-битными (EIP и EFLAGS соответственно), причём в регистре флагов добавлена новая группа флажков. К 4 16-битным сегментным регистрам (ES, CS, SS, DS) добавились ещё 2 16-битных регистра (FS и GS), теперь стало 6 сегментных регистров. Добавлены несколько новых групп регистров (каждый регистр размером 32 бит): 3 регистра управления (CR0 (MSW), CR2, CR3), 8 регистров отладки (DR0, DR1, DR2, DR3, DR6, DR7), 2 тестовых регистра (TR6, TR7).

### Набор инструкций

Набор инструкций i386 был расширен, в основном, за счёт добавления 32-битных вариантов существующих инструкций (образованных 32-битными префиксами), включая те, у которых 32-битные варианты имеют другие мнемоники (pushad/popad, pushfd/popfd, cwd\*/cdq, movsd/cmppsd/scasd/lodsd/insd/outsd, iretd), а также команды mov для новых регистров. Другие новые инструкции:

- установка байта по условию (set\*)
- команды переходов (jmp, j\*) с относительным 16-битным смещением
- команды загрузки полного адреса для новых сегментных регистров (lfs, lgs)
- логические команды (and, or, xor) при работе с 16-разрядными непосредственными данным получили возможность использовать знаковые расширения байта, что позволяет сократить такие команды на байт, когда данные помещаются в байт (эта возможность впервые была описана в фирменной документации на 80386, но, возможно, имелась и в более ранних процессорах)

Средняя длина инструкций составляет 3,2 байта.

## Математический сопроцессор

Так же, как и в случае связки 80286-80287, обмен данными процессора 80386 с его сопроцессором шёл через порты с адресами oF0H-oFFH, и, в принципе, небольшим дополнением схемы можно было подключить сопроцессор 80287 к процессору 80386DX<sup>[3][4]</sup>, это предлагалось в качестве дешёвой альтернативы использованию сопроцессора 80387.

Специально для микропроцессоров Intel 80386 были разработаны математические сопроцессоры 387SX и 387DX, объединяемые единым наименованием Intel 80387. Эти сопроцессоры совместимы с процессорами 386SX и 386DX соответственно; так, сопроцессор 387DX возможно подключить к 32-разрядной шине процессора. В остальном, обе модели аналогичны математическому сопроцессору 80287, который также можно использовать вместе с процессором 80386.

## Модели

---

С 1985 года было выпущено множество модификаций процессора i386, различающихся между собой производительностью, потребляемой мощностью, разъёмами, корпусами и прочими характеристиками.

### 386DX

Первый процессор семейства 386 выпущен 13 октября 1985 года и имел тактовую частоту 16 МГц. После выпуска процессоров 386SX процессоры этой серии получили индекс “DX” — “D”ouble-word e”X”ternal, что указывало на его 32-разрядную внешнюю шину. Процессор позиционировался как производительное решение для настольных систем. Производился процессор по СНМОS IV технологии и потреблял 400 мА, что значительно меньше, чем Intel 8086. 16 февраля 1987 года анонсирована модель с частотой 20

МГц; 4 апреля 1988 года с частотой 25 МГц и 10 апреля 1989 года с частотой 33 МГц. Процессор выпускался в корпусах PQFP-132 (такие процессоры имели литеру «NG» в начале названия, например, NG80386DX25) или в керамическом PGA-132 (такие процессоры имели литеру «A» в начале названия, например, A80386DX25)

Первые процессоры 386DX имели ошибку, которая иногда приводила к неверным результатам при работе с 32-разрядными числами в таких программах, как OS/2 2.x, UNIX/386 или Windows в расширенном режиме. Ошибка приводила к тому, что система зависала. Вследствие некоторых проблем, в том числе из-за отсутствия 32-битных операционных систем, устранить ошибку удалось лишь в апреле 1987 года. Уже вышедшие процессоры прошли проверку, в результате которой процессоры, не имеющие ошибки, были промаркированы двойным символом «сигма» и/или одним символом «IV», а с ошибкой — 16 BIT S/W ONLY.

## 386SX

Первая модель этого семейства была представлена 16 июня 1988 года и имела частоту 16 МГц, позже были представлены и более быстрые модели: 20 МГц (25 января 1989 года), 25 МГц и 33 МГц (оба процессора представлены 26 октября 1992 года). Процессоры позиционировались как решения для настольных ПК начального уровня и портативных ПК. Семейство SX отличалось от семейства DX тем, что у него разрядность внешней шины данных составляла 16 бит, а разрядность внешней шины адреса — 24 бит, в результате чего процессор мог адресовать только 16 Мбайт физической памяти, что делало вычислительные системы, построенные на его основе, аппаратно совместимыми с предыдущими процессорами Intel 80286. В то же время процессор 386SX мог выполнять все программы, написанные для 386DX, и это сделало его популярным для изготовления «Турбо-плат», например, Cumulus 386SX, Intel InBoard или Orchid Tiny Turbo.

В рамках серии SX были выпущены процессоры с маркировками 80386SXTA, 80386SXSA, 80386SXLP, которые представляли собой встраиваемые (embedded) процессоры (серия SXSA), процессоры низкого потребления (Low Power), а также использовались в других целях.

## 386SL

Первая модель этого семейства была представлена 15 октября 1990 года и имела частоту 20 МГц, позже была представлена модель 25 МГц (30 сентября 1991 года). Процессоры позиционировались как первые энергоэффективные микропроцессоры, специально предназначенные для портативных ПК. Семейство SL отличалось от семейства SX тем, что имело на кристалле также контроллер оперативной памяти, контроллер внешней кэш-памяти объёмом от 16 до 64 Кбайт и контроллер шины.

## 386EX

Представляет собой модификацию процессора 386SX. Процессор предназначался для встраиваемых приложений с высокой интеграцией и малой потребляемой мощностью. Ключевые особенности этого процессора — низкое энергопотребление, пониженное напряжение питания, расположенные на кристалле контроллер прерываний, микросхема выбора чипа, счётчики и таймеры, логика тестирования JTAG. Эта серия процессоров имела несколько модификаций: EXSA, EXTA, EXTB, EXTC. Максимальный ток, потребляемый процессорами, составляет 320 мА для процессоров серии EXTC и 140 мА для процессоров серии EXTB.

Использовался на борту различных орбитальных спутников и микроспутников и в NASA'овском проекте FlightLinux.



Intel i386DX, 25 МГц

Intel i386EXTC, 25 МГц

Intel i386CXSA, 25 МГц

Intel i386, упакованный фирмой IBM

Процессоры для upgrade (Cyrix и Texas Instruments)

## Компьютеры на базе процессора Intel 80386

На основе микропроцессоров 80386 фирмой IBM были созданы персональные компьютеры IBM AT 386 (семейство PC) и IBM PS/2-80 (семейство PS/2). В первом применялась ОС PC-DOS, а во втором — OS/2.

## Команда разработчиков

- John Crawford, главный архитектор
- Jim Slager, главный инженер
- Patrick Gelsinger
- Khaled El-Ayat
- Riaz Haq
- Gene Hill

- Jan Wiliam L. Prak
- David Vannier
- Joseph «Chip» Krauskopf
- Ken Shoemaker
- Paul Ries
- Saurabha Radhika

## Технические характеристики различных моделей

### Данные, относящиеся ко всем моделям

- Разрядность регистров: 32
- Объем виртуальной адресуемой памяти: 64 Тбайт<sup>[5]</sup>
- Максимальный объем сегмента: 4 Гбайт

Характеристика, параметр	386DX	386SX	386SL	386EX
Дата анонса первой модели	<u>13 октября 1985 года</u>	<u>16 июня 1988 года</u>	<u>15 октября 1990 года</u>	<u>август 1994 года</u>
Тактовые частоты (МГц)	12, 16, 20, 25, 33, 40	16, 20, 25, 33, 40 (серии SXSA и SXTA)	20, 25	16, 20, 25, 33 (возможно существовали и другие модели)
Разрядность внешней шины данных	32 бит	16 бит	16 бит	16 бит
Разрядность внешней шины адреса	32 бит	24 бит	24 бит	26 бит
Объем физической адресуемой памяти	4 <u>Гбайт</u> <sup>[5]</sup>	16 Мбайт	16 Мбайт	64 Мбайт (для серий EXTB, EXTC)
Напряжение питания, вольт	5	5	5	2,7~3,6 В (серия EXTB) 4,5~5,5В (серия EXTC)
Количество транзисторов	275 000	275 000	855 000	275 000
Техпроцесс (нм)	1500 и 1000 (модель 33 МГц)	1500 и 1000	1000	800

Площадь кристалла	42 мм² (для модели 33 МГц)	Нет данных	Нет данных	Нет данных
Максимально потребляемый ток	0,39 А (модель 33 МГц)	0,38 А (модель 33 МГц)	Нет данных	Нет данных
Максимально потребляемая мощность, Вт	1,95 (модель 33 МГц)	1,9 (модель 33 МГц)	Нет данных	Нет данных
Разъём	Гнездо типа «Socket» или припаивался к плате	Процессор припаивался к плате	Процессор припаивался к плате	Процессор припаивался к плате
Корпус	132-контактный керамический PGA, или 132-контактный PQFP	100-контактный PQFP	196-контактный PQFP	132-контактный PQFP, 144-контактный TQFP
Набор инструкций	x86 (150 инструкций, не считая модификаций)	x86 (150 инструкций, не считая модификаций)	x86	x86

## Примечания

- Intel caches in ancient chips ([https://www.webcitation.org/618aFir9z?url=http://www.reghardware.com/2006/05/18/intel\\_cans\\_386\\_486\\_960\\_cpu/](https://www.webcitation.org/618aFir9z?url=http://www.reghardware.com/2006/05/18/intel_cans_386_486_960_cpu/)) (англ.). Дата обращения: 10 апреля 2010. Архивировано из оригинала ([http://www.reghardware.co.uk/2006/05/18/intel\\_cans\\_386\\_486\\_960\\_cpus/](http://www.reghardware.co.uk/2006/05/18/intel_cans_386_486_960_cpus/)) 22 августа 2011 года.
- Product Change Notification #106013 - 00 (<http://www.yeralan.org/agen/pdf/PCN106013.pdf>) . Дата обращения: 24 августа 2011. Архивировано (<https://web.archive.org/web/20120111065626/http://www.yeralan.org/agen/pdf/PCN106013.pdf>) 11 января 2012 года.
- краткое описание 80287 на cadcenter (<http://cadcenter.ru/content/view/478/33/>) Архивировано (<https://web.archive.org/web/20080426095158/http://cadcenter.ru/content/view/478/33/>) 26 апреля 2008 года.
- BootPrompt-HOWTO в Linux (<http://www.linuxrsp.ru/docs/BootPrompt-HOWTO-3.html>) Архивная копия (<https://web.archive.org/web/20100412100853/http://www.linuxrsp.ru/docs/BootPrompt-HOWTO-3.html>) от 12 апреля 2010 на Wayback Machine: «Параметр загрузки ядра 'по387': В некоторых сопроцессорах i387 есть ошибки, проявляющиеся в 32-битовом защищённом режиме. Например, некоторые ранние чипы ULSI-387 вызывают жёсткое зависание при вычислениях с плавающей запятой, очевидно, из-за ошибки в командах FRSAV/FRRESTOR. Параметр загрузки `по387` позволяет Linux игнорировать мат. сопроцессор, даже если он имеется. Конечно, у вас тогда должно быть ядро с поддержкой математической эмуляции. Это может быть также полезным, если у вас действительно одна из старых 386 машин с 80287 FPU, так как Linux не может использовать 80287.»; оригинал на английском (<http://www.faqs.org/docs/Linux-HOWTO/BootPrompt-HOWTO.html>) Архивная копия (<https://web.archive.org/web/20090830033703/http://www.faqs.org/docs/Linux-HOWTO/BootPrompt-HOWTO.html>) от 30 августа 2009 на Wayback Machine
- P3 (386) Third-Generation Processors | Microprocessor Types and Specifications | InformIT (<http://www.informit.com/articles/article.aspx?p=130978&seqNum=27>). Дата обращения: 12 ноября 2017. Архивировано (<https://web.archive.org/web/20171023042958/http://www.informit.com/articles/a>

[rticle.aspx?p=130978&seqNum=27](#)) 23 октября 2017 года.

## Литература

---

- Intel 80386 Programmer’s Reference Manual
- Intel 80387 Programmer’s Reference Manual (1987 год)
- *Брамм П., Брамм Д.* Микропроцессор 80386 и его программирование. — М.: Мир, 1990. — 448 с. — ISBN 5-03-001441-1.
- *Паппас К., Мари У.* Микропроцессор 80386. — М.: Радио и связь, 1993. — 320 с. — ISBN 5-256-00826-9.

## Ссылки

---

- Описание и фотографии процессоров семейства i386 (<http://www.cpu-collection.de/?tn=0&l0=co&l1=Intel&l2=80386>) (англ.)
  - Документация на процессор (<https://web.archive.org/web/20140728052748/http://bitsavers.informatik.uni-stuttgart.de/pdf/intel/80386/>) на сайте bitsavers.org (англ.)
  - Intel 80386 processor family (<http://www.cpu-world.com/CPUs/80386/>) (англ.)
  - Краткое описание микропроцессора Intel 80386 ([http://cs.usu.edu.ru/study/chips.html#\\_Toc9352965](http://cs.usu.edu.ru/study/chips.html#_Toc9352965)) (рус.)
- 

Источник — [https://ru.wikipedia.org/w/index.php?title=Intel\\_80386&oldid=133090963](https://ru.wikipedia.org/w/index.php?title=Intel_80386&oldid=133090963)

---

**Эта страница в последний раз была отредактирована 18 сентября 2023 в 11:49.**

Текст доступен по лицензии Creative Commons «С указанием авторства — С сохранением условий» (CC BY-SA); в отдельных случаях могут действовать дополнительные условия.

Wikipedia® — зарегистрированный товарный знак некоммерческой организации Фонд Викимедиа (Wikimedia Foundation, Inc.)