Федеральное государственное бюджетное образовательное учреждение высшего образования «Чувашский государственный университет им. И.Н. Ульянова»

Кафедра вычислительной техники

Расчётно-графическая работа №1

по дисциплине ЭВМ и периферийные устройства

Определение характеристик и возможностей   
процессора с использованием команды CPUID

Вариант №7

Выполнил: Кузнецов Даниил Андреевич студент группы ИВТ-42-23

Проверила: доцент Андреева А.А.

Чебоксары, 2025

**Цель работы**

Получить практические навыки по определению характеристик и возможностей процессора с использованием команды CPUID.

**Задание к лабораторной работе**

Написать программу с использованием команды CPUID для вывода в виде строки наименования производителя процессора в MessageBoxA. Следующей строкой вывести семейство процессоров, наименование компании, марку процессора, номер процессора и тактовую частоту, а также максимальный объём памяти, поддерживаемый процессором.

**Описание команды CPUID**

|  |  |
| --- | --- |
| Вход | Выход |
| EAX = 80000008H | **EAX**: размер линейного (физического) адреса.  Биты 07-00: количество битов физического адреса.  Биты 15-08 количество битов линейного адреса.  Биты 31-16: зарезервировано = 0.  **EBX, ECX, EDX**: зарезервированы, равны 0. |

**Граф схема алгоритма получения информации о бренде**

EAX = 80000000H

EAX = 80000000H

Строка бренда не поддерживается

Строка бренда поддерживается частично

EAX ≥ 80000004H

Строка бренда поддерживается

**Текст программы**

                includelib      ../kernel32.lib

                includelib      ../user32.lib

                extern  ExitProcess: proc

                extern  MessageBoxA: proc

                extern  wsprintfA: proc

.data

MsgCaption      db      "cpuid", 0

vendorId        db      13 dup(0)

brandString     db      "Brand string not supported", 24 dup(0)

fmt             db      "Vendor ID: %s", 13, 10

                db      "CPU: %s", 13, 10

                db      "Physical memory: 2^%d", 13, 10

                db      "Linear memory: 2^%d", 0

MsgBoxText      db      256 dup(0)

.code

WinMain         proc

                sub     rsp, 7 \* 8      ; выделить стек для передачи параметров + выравнивание

                lea     rdi, vendorId

                xor     rax, rax        ; eax = 0

                cpuid                   ; получить Vendor ID

                mov     [rdi], ebx      ; сохранить

                mov     [rdi + 4], edx  ; в строку

                mov     [rdi + 8], ecx

                mov     eax, 80000000h  ; проверить поддержку

                cpuid                   ; строки бренда

                test    eax, 80000000h  ; если не поддерживается

                jz      fin             ; то перейти

                cmp     eax, 80000004h  ; или поддерживается не полностью

                jb      fin

                lea     rdi, brandString

                mov     eax, 80000002h  ; получить 1-ю часть

                cpuid                   ; строки бренда

                mov     [rdi], eax      ; сохранение в строку

                mov     [rdi + 4], ebx

                mov     [rdi + 8], ecx

                mov     [rdi + 12], edx

                mov     eax, 80000003h  ; получить 2-ю часть

                cpuid                   ; строки бренда

                mov     [rdi + 16], eax ; сохранение в строку

                mov     [rdi + 20], ebx

                mov     [rdi + 24], ecx

                mov     [rdi + 28], edx

                mov     eax, 80000004h  ; получить 3-ю часть

                cpuid                   ; строки бренда

                mov     [rdi + 32], eax ; сохранение в строку

                mov     [rdi + 36], ebx

                mov     [rdi + 40], ecx

                mov     [rdi + 44], edx

fin:            xor     rdi, rdi        ; обнуляем ответ (кол-во бит физ. адреса)

                xor     rsi, rsi        ; обнуляем ответ (кол-во бит лин. адреса)

                mov     eax, 80000008h  ; получить размер линейного (физического) адреса

                cpuid

                mov     dil, al         ; ответ в rdi (кол-во бит физ. адреса)

                shr     rax, 8

                mov     sil, al         ; ответ в rsi (кол-во бит лин. адреса)

                ; формирование строки с результатом

                mov     rcx, offset MsgBoxText

                mov     rdx, offset fmt

                mov     r8, offset vendorId

                mov     r9, offset brandString

                mov     [rsp + 32], rdi

                mov     [rsp + 40], rsi

                call    wsprintfA

                ; вывод результата

                xor     rcx, rcx

                mov     rdx, offset MsgBoxText

                mov     r8, offset MsgCaption

                xor     r9, r9          ; MB\_OK

                call    MessageBoxA

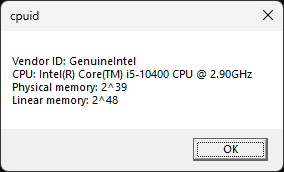
                xor     rcx, rcx

                call    ExitProcess     ; выход

WinMain         endp

                end

**Результаты работы программы**



| **Состояние регистров до вызова CPUID** | **Состояние регистров после вызова CPUID** | **Расшифровка выходных значений команды** |
| --- | --- | --- |
|  |  | **EAX: максимально допустимое входное значение EAX для команды CPUID.**  **EBX\_EDX\_ECX: производитель (ASCII-строка)** |
|  |  | **EAX: максимально допустимое входное значение EAX для команды CPUID.**  **EBX, EDX, ECX: зарезервировано** |
|  |  | **EAX: строка идентификации процессора (строка бренда)**  **EBX, EDX, ECX: продолжение строки бренда**  **“Intel(R) Core (TM”** |
|  |  | **EAX, EBX, EDX, ECX: продолжение строки бренда**  **“) i5-10400 CPU @”** |
|  |  | **EAX, EBX, EDX, ECX: продолжение строки бренда**  **“ 2.90GHz”** |
|  |  | **EAX**: размер линейного (физического) адреса.  Биты 07-00: количество битов физического адреса.  Биты 15-08 количество битов линейного адреса.  Биты 31-16: зарезервировано = 0.  **EBX, ECX, EDX**: зарезервированы, равны 0. |

**Выводы**

Получены практические навыки по определению характеристик и возможностей процессора с использованием команды CPUID.