|  |  |  |  |
| --- | --- | --- | --- |
| Cell Address | Cell Content | Mnemonics | комментарии |
| 238 | 056E | ADDR | Начальный адрес, в котором хранятся строки. |
| 239 | 0000 | NOW | Текущий адрес, в котором хранится строка. |
| 23A | 000A | NL | СИМБОЛ СТОП |
| 23B | 0004 | CONST | Const число для получения первого симбола с помощью and операторо |
| 23C | AEFB | LD IP-5 | Нормализовать начальный адрес, в котором хранится строку |
| 23D | EEFB | ST IP-5 |  |
| 23E | 0200 | CLA |  |
| 23F | 1205 | IN 5 | Чтение статус регистра ВУ-2 |
| 240 | 2F40 | AND #40 | Проверка на открытость |
| 241 | F0FD | BEQ IP-3 |  |
| 242 | 1204 | IN 4 | Чтение данных регистра ВУ-2 |
| 243 | 0680 | SWAB | Своп для достижения правильного формата строки |
| 244 | E8F4 | ST (IP-12) | Сохранение второго симбола в строке |
| 245 | 0680 | SWAB |  |
| 246 | 7EF3 | CMP IP-13 | Проверка наличия симбола остановки |
| 247 | F00A | BEQ IP+10 |  |
| 248 | 0200 | CLA | Очистить АС и будем читпть первый симбол |
| 249 | 1205 | IN 5 | Чтение статус с регистра ВУ-2 |
| 24A | 2F40 | AND #40 | Проверка на открытость |
| 24B | F0FD | BEQ IP-3 |  |
| 24C | A8EC | LD (IP-20) | Соединить первый симбол с вторым симболом |
| 24D | 1204 | IN 4 |  |
| 24E | EAEA | ST (IP-22) | Сохранить строковый результат |
| 24F | 2EEB | AND IP-21 |  |
| 250 | 7EE9 | CMP IP-23 | Проверка наличия симбола остановки |
| 251 | F1EC | BNE IP-20 |  |
| 252 | 0100 | HLT | STOP. |

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Код команды** | **Мнемоника** | **Комментарий** |
| 068 |  |  | кол-во символов |
| 069 | 057C |  | начало массива |
| 06A |  |  | R |
| 06B | 0200 | CLA | 0 ->AC |
| 06C | 1205 | IN 5 | проверка готовности |
| 06D | 2F40 | AND (40)16 | AC AND 40 |
| 06E | F0FD | BEQ (FD)16 | IF Z==1 THEN IP+1-3 ->IP |
| 06F | 1204 | IN 4 | Ввод кол-ва символов |
| 070 | E068 | ST 068 | AC ->068 |
| 071 | 0200 | CLA | 0 ->AC |
| 072 | 1205 | IN 5 | проверка готовности |
| 073 | 2F40 | AND (40)16 | AC AND 40 |
| 074 | F0FD | BEQ (FD)16 | IF Z==1 THEN IP+1-3 ->IP |
| 075 | 1204 | IN 4 | Ввод  символа |
| 076 | E06A | ST 06A | Сохранение в буфер |
| 077 | A068 | LD 068 | Загрузка кол-во символов |
| 078 | 0740 | DEC | AC-1->AC |
| 079 | E068 | ST 068 | Сохранение кол-ва символов |
| 07A | F006 | BEQ 081 | IF Z==1 THEN IP+1+6 ->IP |
| 07B | 0200 | CLA | 0->AC |
| 07C | 1205 | IN 5 | проверка готовности |
| 07D | 2F40 | AND (40)16 | AC AND 40 |
| 07E | F0FD | BEQ (FD)16 | IF Z==1 THEN IP+1-3 ->IP |
| 07F | 1204 | IN 4 | Ввод  символа |
| 080 | 0680 | SWAB | Обмен старших байтов |
| 081 | 406A | ADD 06A | добавление буфера |
| 082 | EAE6 | ST (069)+ | Сохранение результата |
| 083 | 8068 | LOOP 068 | Проверка на кол-во символов |
| 084 | C071 | JUMP 071 | Возврат в начало цикла |
| 085 | 0100 | HLT | Остановка |

ORG 0x276

START: CLA

S1: IN 7

AND #0x40

BEQ S1

IN 6

SWAB

ST (STR)

CMP STOP

BEQ DONE

S2: CLA

IN 7

AND #0x40

BEQ S2

IN 6

ADD (STR)

ST (STR)+

AND MASK

CMP STOP

BEQ DONE

BR S1

DONE: HLT

STR: WORD 0x55A

STOP: WORD 0x0

MASK: WORD 0x00FF

|  |  |  |
| --- | --- | --- |
| ORG | 0 x235 | |
| ADDR: | WORD $STRING | |
| ADRINC | WORD 0 | |
| LEN | WORD 0 Кол-во оставшихся значений | |
| START: | LD ADR | |
| ST | ADRINC | |
| **PAUSE0** |  | |
| IN 0x3 | Ожидание готовности к выводу числа символов | |
| AND | #0x40 Проверка 6 бита SR | |
| BEQ | PAUSED 0 | |
| LD | $LENGTH | |
| OUT | 0x2 | |
| **COUPLE** |  | |
| ST | LEN | |
| BEQ | STOP | |
| **PAUSE1:** |  | |
| IN 0x3 |  | |
| AND | #0 x40 | |
| BEQ | PAUSE1: | |
| LD | | (ADRINC) |
| OUT | | 0 x2 |
| LD | | LEN |
| DEC | |  |
| ST | | LEN |
| BEQ | | STOP |
| **PAUSE2:** | |  |
| IN 0x3 | |  |
| AND | | #0x40 |
| BEQ | | PAUSED2 |
| LD | | (ADRINC)+ |
| SWAB | |  |
| OUT | | 0x2 |
| LD | | LEN |
| DEC | |  |
| BR | | COUPLE |
| STOP | | HLT |
| ORG | | 0x56D |
| LENGTH | | WORD 5 |
| STR | | WORD 0x???? |
|  | | WORD 0x???? |
|  | | WORD 0x???? |