ریزپردازنده ۱

دانشکده مهندسی کامپیوتر و فناوری اطلاعات دانشگاه صنعتی امیرکبیر

AVR ASSEMBLER SYNTAX

سرفصل مطالب

- كلمات كليدى
- دایرکتیو های پیشپردازنده
 - ه Comment •
 - ادامه دادن خط کد
 - ثابتهای عدد صحیح
- و شتهها و ثابتهای کاراکتری
- قرارگیری چند دستور در یک خط
 - عملوندها

كلمات كليدي

شناساگرهای از پیش تعریف شده (کلمات کلیدی) کلماتی رزرو شده هستند و نمی توان از نام آنها به عنوان نام متغیر جدید استفاده کرد. کلمات کلیدی شمال علائم دستوری و توابع است.

کلمات کلیدی اسمبلر صرف نظر از بزرگ یا کوچک بودن حروف نوشتاری قابل شناسایی هستند مگر آنکه از گزینه « c » استفاده شود که در این صورت کلمه کلیدی با حروف کوچک شناخته می شود. (یعنی در این حالت به عنوان مثال add رزرو شده محسوب می شود اما ADD نه)

دایرکتیو های پیشپردازنده

• AVRASM2 تمام خطوطی که با # شروع میشوند را به عنوان یک دایرکتیو پیشپردازنده در نظر میگیرد.

La COMMENT

علاوه بر روش اصلی و قدیمی قرار دادن comment که با ; شروع میشود، AVRASM2 روشهای comment گذاری مشابه زبان C را نیز پشتیبانی میکند. مثال:

```
; The rest of the line is a comment (classic assembler comment)
// Like ';', the rest of the line is a comment
/* Block comment; the enclosed text is a comment, may span
multiple lines. This style of comments cannot be nested. */
```

ادامه دادن خط کد

• مانند C خطوط کد را می توان با قرار دادن کاراکتر « \ » در انتهای خط ادامه داد. این قابلیت به ویژه در تعریف ماکروهای بلند پیشپردازنده کاربرد دارد و همچنین تعاریف طولانی با استفاده از دایر کتیو تعریف متغیر db.

مثال:

.db 0, 1, "This is a long string", '\n', 0, 2, \
"Here is another one", '\n', 0, 3, 0

اثابتهای عدد صحیح

AVRASM2 اجازه استفاده از کاراکتر «_ » به عنوان جداساز برای ایجاد خوانایی بیشتر را میدهد. این کاراکتر مشخص کاراکتر می تواند هرجایی در عدد ظاهر شود به جز به عنوان کاراکتر اول یا پیش از کاراکتر مشخص کننده ی مبنا (b,x,...)

به عنوان مثال دو عبارت زیر مجاز هستند:

0b1100_1010,0b_11_00_10_10_

اما دو عبارت زیر غیرمجاز هستند:

_0b11001010,0_b11001010

رشتهای که درون " " قرار گیرد تنها میتواند در کنار DB و دایرکتیو های MESSAGE، WARNING و WARNING استفاده شود. رشته مورد نظر عیناً در نظر گرفته می شود و دنباله کاراکترهای گریز (escape sequence) شناخته نمی شوند. همچنین رشته تعریف شده منتهی به کاراکتر null نیست.

رشتههای درون " " براساس قرارداد ANSI C امکان پیوست به یکدیگر را دارند. به عنوان "This is a long string" برابر با رشته "This is a long string" است. از این قابلیت می توان برای تعریف یک رشته طولانی که چند خط را پوشش می دهد استفاده کرد.

ثابتهای کاراکتری در '' قرار می گیرند و هرجا که عبارت صحیح مجاز باشد حضور این کاراکترها نیز مجاز است.

علاوه بر کاراکترهایی که در صفحه بعد آمده، دنبالههای کاراکتری 000\ به عنوان 000 یعنی عدد در مبنای هشت، و xhh\ به عنوان hh یعنی عدد در مبنای هشت شناخته می شوند.

کاراکترهای گریز زیر که مشابه زبان C هستند شناخته میشوند:

| | معنى | دنباله کاراکتر گریز |
|---|--|---------------------|
| | خط جدید معادل اسکی : LF 0x0a | \n |
| | بازگشت به سر خط معادل اسکی: CR 0x0d | \r |
| | بوق هشدار معادل اسکی: BEL 0x07 | \a |
|) | بازگشت به عقب (backspace) معادل اسکی: BS 0x08 | \b |
| | آغاز صفحه جدید (formfeed) معادل اسکی: FF 0x0c | \f |

| معنی | دنباله کاراکتر گریز |
|--|---------------------|
| ایجاد فاصله (tab) افقی معادل اسکی : HT 0x09 | \t |
| ایجاد فاصله (tab) عمودی معادل اسکی: VT 0x0b | \ v |
| کاراکتر \ | \\ |
| کاراکتر null معادل اسکی: NUL | \0 |

مثالها:

```
.db "Hello\n" // is equivalent to:
.db 'H', 'e', 'l', 'l', 'o', '\\', 'n'
.db '\0', '\177', '\xff'
```

برای ساخت رشتهای که در زبان C به صورت "Hello, world\n" است باید به صورت زیر عمل کرد:

.db "Hello, world", '\n', 0

قرارگیری چند دستور در یک خط

AVRASM2 اجازه قرار گیری چند دستور و دایر کتیو در یک خط را میدهد اما استفاده از این قابلیت توصیه نمی شود.

چنین قابلیتی همانطور که گفته شد برای پشتیبانی از ماکروهای چند خطی پیشپردازنده لازم است.

عملوندها

AVRASM2 از عملوندهای صحیح پشتیبانی می کند و نیز پشتیبانی محدودی از عبارتهای ثابت اعشاری ممیز شناور دارد.

همه عملوندها در بخش operands در اسلایدهای expression ها اراده شدهاند.