GAS Tutorial - 7. Directives (1)

Young W. Lim

2016-03-09 Wed

Outline

Based on

"Using as", Dean Elsner, Jay Fenlason & friends

In short, it only meaningful to add or subtract the offsets in an address; you can only have a defined section in one of the two arguments.

7.5 .ascii “string “

- expects zero or more string literals separated by commas
- assembles each string into consecutive addresses.
- no automatic trailing zero byte

7.6 .asciz “string “

. . .

- just like .ascii
- but each string is followed by a zero
- ” stands for ero”.

7.8 .byte expressions

- expects zero or more expressions, separated by commas
- each expression is assembled into the next byte

7.35 .double flonums

- expects zero or more flonums, separated by commas
- assembles floating point numbers
- the exact kind of floating point numbers emitted depends on how as is configured

7.99 .single flonums

- expects zero or more flonums, separated by commas
- assembles floating point numbers
- the exact kind of floating point numbers emitted depends on how as is configured

7.53 .float flonums

- assembles zero or more flonums, separated by commas
- has the same effect as .single
- the exact kind of floating point numbers emitted depends on how as is configured

7.63 .int expressions

- expect zero or more expressions, of any section, separated by commas
- For each expression, emit a number that, at run time, is the value of that expression.
- The byte order and bit size of the number depends on what kind of target the assembly is for.

7.76 .long expressions

- the same as '.int'

7.98 .short expressions

- normally the same as '.word'
- In some configurations, however, .short and .word generate numbers of different lengths.

- assembles zero or more flonums, separated by commas
- has the same effect as .float
- The exact kind of floating point numbers emitted depends on how as is configured

7.91 .quad bignums

- expects zero or more bignums, separated by commas.
- emits an 8-byte integer
- If the bignum won't fit in 8 bytes, it prints a warning message;
- and just takes the lowest order 8 bytes of the bignum.
- The term quad comes from the systems where a word is two bytes (intel)
- quad-word for 8 bytes.

7.43 .equ symbol, expression

- sets the value of symbol to expression
- synonymous with .set

7.97 .set

- set the value of symbol to expression
- changes symbol value and type to conform to expression
- for an external symbol, it remains flagged
- may .set a symbol many times
- for a global symbol, the last set takes effect

7.44 .equiv symbol, expression

- .equiv directive is like .equ and .set
- signal an error for an already defined symbol
- a symbol referenced but not actually defined is to be undefined.
- without the error message, this is roughly equivalent to
  - .ifdef SYM
  - .err
  - .endif
  - .equ SYM,VAL

7.45 .eqv symbol, expression

- like .equiv
- no immediate evaluation of the expression
- evaluated when it is used in other expression
- a snapshot of its current value

7.52 .fill repeat, size, value

- all agruments are absolute expressions
- emits 'repeat' copies of 'size' bytes
- 0 <= Repeat
- 0<= Size <= 8
- the repeat bytes are from 8-byte numbers
  - 4 MSB's are zero, only 4 LSB's used
  - in the byte-order of an integer on the computer
- default size = 1
- default value = 0

7.58 .hword expression

- expects zero or more expressions, and emits a 16 bit number for each.
- a synonym for .short

7.101 skip size, fill

- emits size bytes, each of value fill.
- both size and fill are absolute expressions.
- default fill = 0
- the same as .space

7.103 .space size, fill

- emits size bytes, each of value fill.
- both size and fill are absolute expressions.
- default fill = 0
- This is the same as .skip

7.105 .string "str", .string8/16/32/64 "str",

- copy str to the object file.
- can have more than one string to copy, separated by commas
- null terminated (ended with 0 byte)
- can use the escape sequence
- string16 : 8-bit character expanded into 16-bit character
- string32 : 8-bit character expanded into 32-bit character
- string64 : 8-bit character expanded into 64-bit character
- follow the target endianness byte order

```
.string32 "BYE"

expands to:
.string   "B\0\0\0 Y\0\0\0 E\0\0\0"   /* little endian targets.
.string   "\0\0\0B \0\0\0Y \0\0\0E"   /* big endian targets. */
```

7.121 .word expression

- expects zero or more expressions, of any section, separated by commas.
- The size of the number emitted, and its byte order, depend on what target computer

the assembly is for.

TTTT