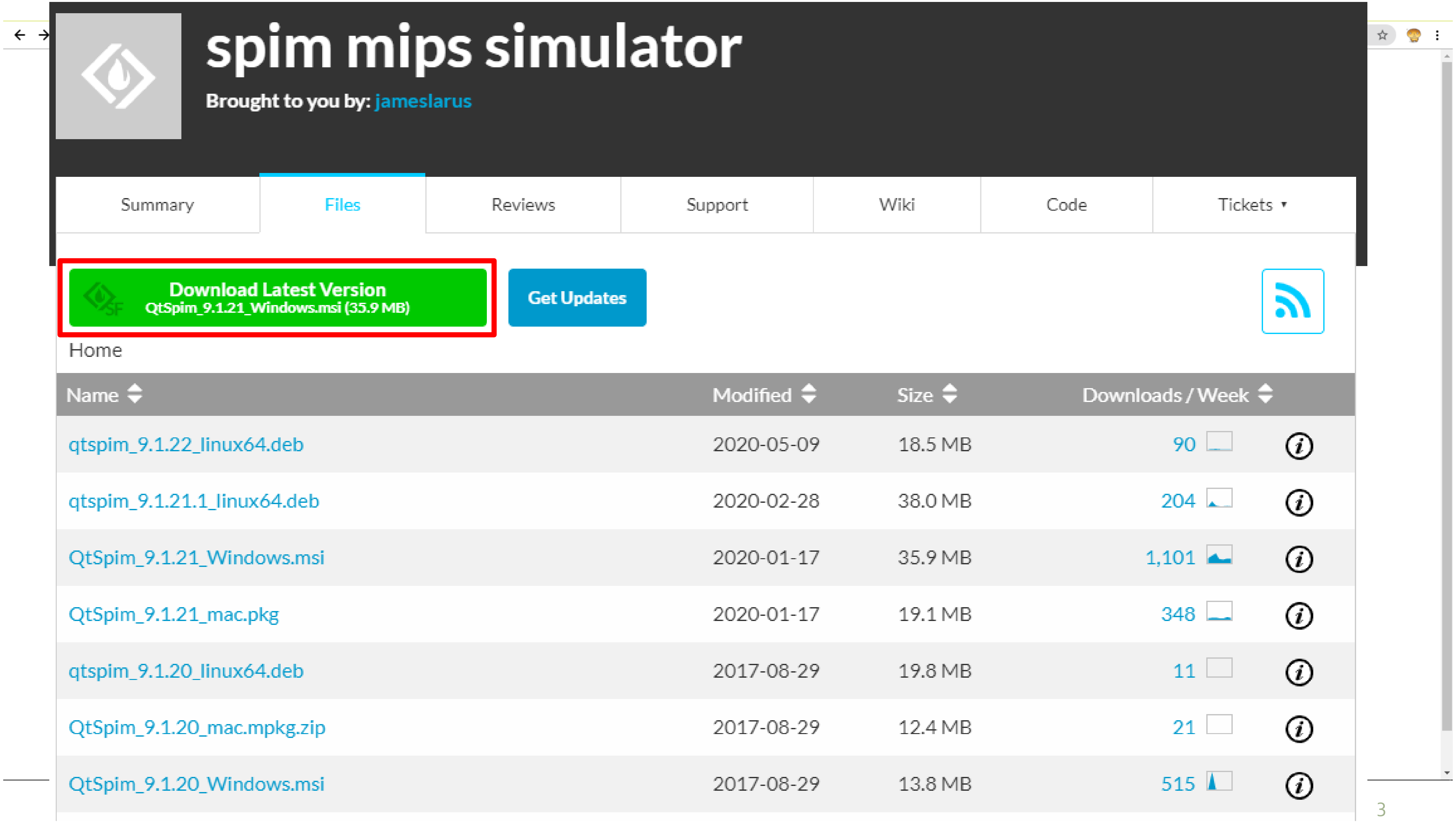
SPIM 教學

NCNU CSIE CA 2023

Introduction

- SPIM : MIPS simulator
  - Read/execute assembly source programs
    - Input: assembly code file
    - Output: results of the program, register value, memory value
  - Debugging your assembly codes

- Download QtSpim (Windows or Linux)
  - https://sourceforge.net/projects/spimsimulator/files/



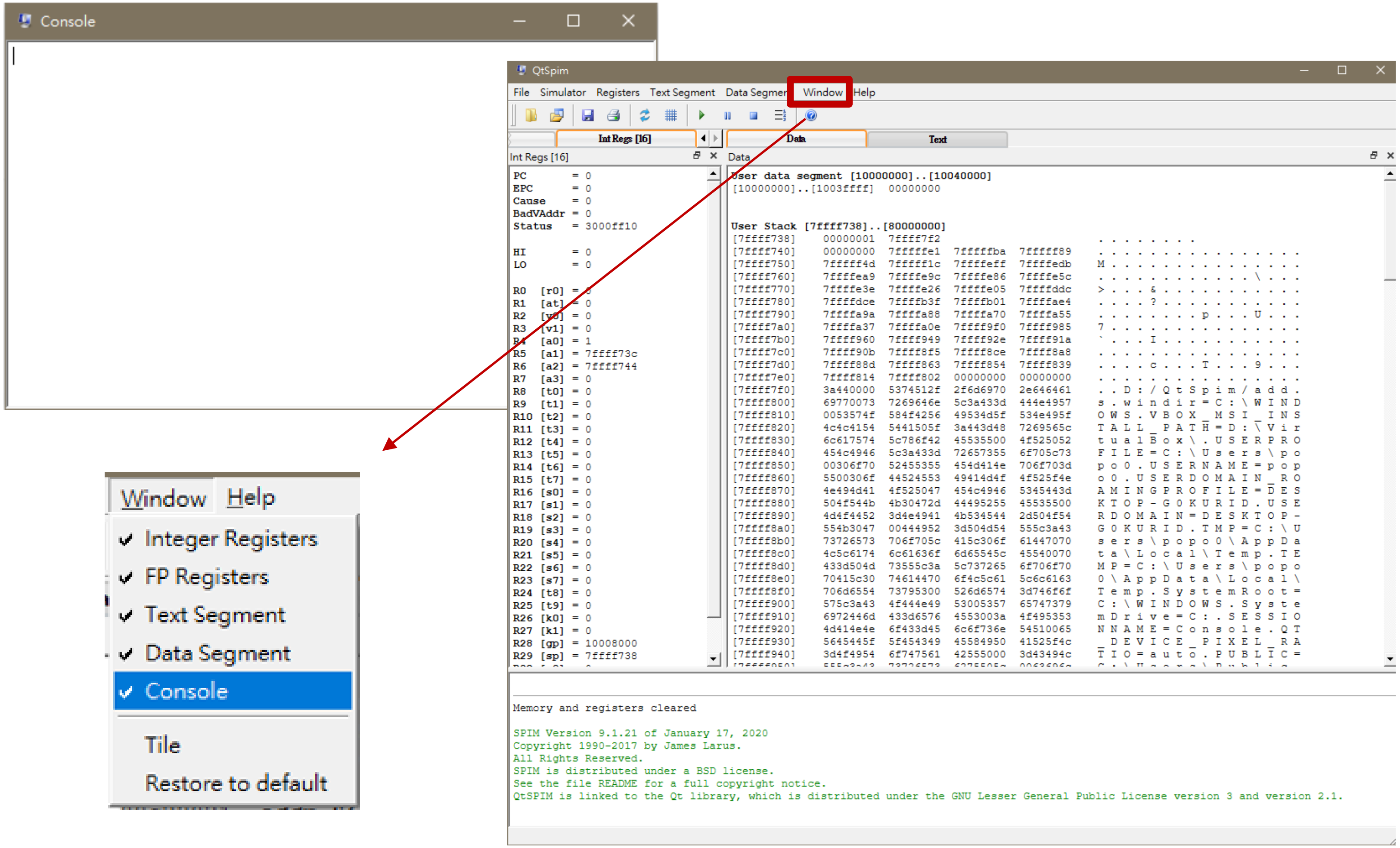
spim mips simulator

Brought to you by: jameslarus

Name	Modified	Size	Downloads / Week
qtspim_9.1.22_linux64.deb	2020-05-09	18.5 MB	90
qtspim_9.1.21.1_linux64.deb	2020-02-28	38.0 MB	204
QtSpim_9.1.21_Windows.msi	2020-01-17	35.9 MB	1,101
QtSpim_9.1.21_mac.pkg	2020-01-17	19.1 MB	348
qtspim_9.1.20_linux64.deb	2017-08-29	19.8 MB	11
QtSpim_9.1.20_mac.mpkg.zip	2017-08-29	12.4 MB	21
QtSpim_9.1.20_Windows.msi	2017-08-29	13.8 MB	515

使用介面

QtSpim



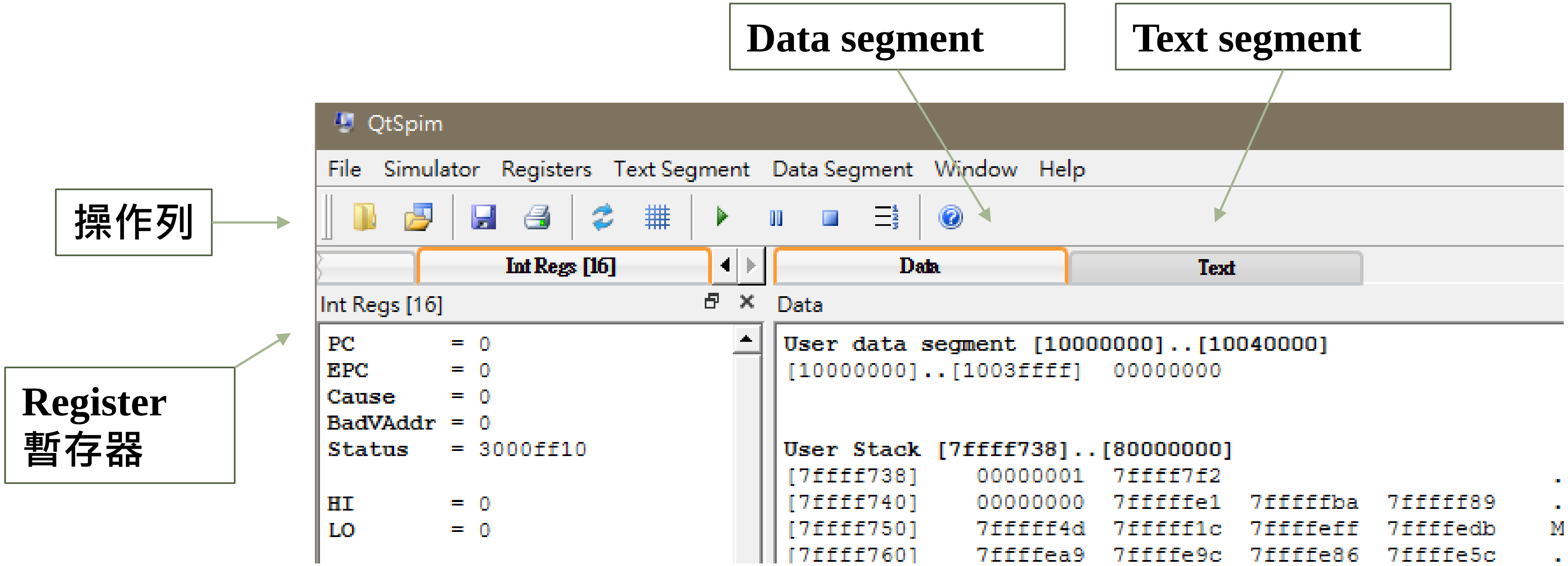
```
PC       = 0
EPC      = 0
Cause    = 0
BadVAddr = 0
Status   = 3000ff10

HI       = 0
LO       = 0

R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 0
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffff73c
R6  [a2] = 7ffff744
R7  [a3] = 0
R8  [t0] = 0
R9  [t1] = 0
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffff738
```

```
User data segment [10000000]..[10040000]
[10000000]..[1003ffff]  00000000


User Stack [7ffff738]..[80000000]
[7ffff738]    00000001  7ffff7f2                        . . . . . . . .
[7ffff740]    00000000  7fffffe1  7fffffba  7fffff89    . . . . . . . . . . . . . . . .
[7ffff750]    7fffff4d  7fffff1c  7ffffeff  7ffffedb    M . . . . . . . . . . . . . . .
[7ffff760]    7ffffea9  7ffffe9c  7ffffe86  7ffffe5c    . . . . . . . . . . . . \ . . .
[7ffff770]    7ffffe3e  7ffffe26  7ffffe05  7ffffddc    > . . . & . . . . . . . . . . .
[7ffff780]    7ffffdce  7ffffb3f  7ffffb01  7ffffae4    . . . . ? . . . . . . . . . . .
[7ffff790]    7ffffa9a  7ffffa88  7ffffa70  7ffffa55    . . . . . . . . p . . . U . . .
[7ffff7a0]    7ffffa37  7ffffa0e  7ffff9f0  7ffff985    7 . . . . . . . . . . . . . . .
[7ffff7b0]    7ffff960  7ffff949  7ffff92e  7ffff91a    ` . . . I . . . . . . . . . . .
[7ffff7c0]    7ffff90b  7ffff8f5  7ffff8ce  7ffff8a8    . . . . . . . . . . . . . . . .
[7ffff7d0]    7ffff88d  7ffff863  7ffff854  7ffff839    . . . . c . . . T . . . 9 . . .
[7ffff7e0]    7ffff814  7ffff802  00000000  00000000    . . . . . . . . . . . . . . . .
[7ffff7f0]    3a440000  5374512f  2f6d6970  2e646461    . . D : / Q t S p i m / a d d .
[7ffff800]    69770073  7269646e  5c3a433d  444e4957    s . w i n d i r = C : \ W I N D
[7ffff810]    0053574f  584f4256  49534d5f  534e495f    O W S . V B O X _ M S I _ I N S
[7ffff820]    4c4c4154  5441505f  3a443d48  7269565c    T A L L _ P A T H = D : \ V i r
[7ffff830]    6c617574  5c786f42  45535500  4f525052    t u a l B o x \ . U S E R P R O
[7ffff840]    454c4946  5c3a433d  72657355  6f705c73    F I L E = C : \ U s e r s \ p o
[7ffff850]    00306f70  52455355  454d414e  706f703d    p o 0 . U S E R N A M E = p o p
[7ffff860]    5500306f  44524553  49414d4f  4f525f4e    o 0 . U S E R D O M A I N _ R O
[7ffff870]    4e494d41  4f525047  454c4946  5345443d    A M I N G P R O F I L E = D E S
[7ffff880]    504f544b  4b30472d  44495255  45535500    K T O P - G 0 K U R I D . U S E
[7ffff890]    4d4f4452  3d4e4941  4b534544  2d504f54    R D O M A I N = D E S K T O P -
[7ffff8a0]    554b3047  00444952  3d504d54  555c3a43    G 0 K U R I D . T M P = C : \ U
[7ffff8b0]    73726573  706f705c  415c306f  61447070    s e r s \ p o p o 0 \ A p p D a
[7ffff8c0]    4c5c6174  6c61636f  6d65545c  45540070    t a \ L o c a l \ T e m p . T E
[7ffff8d0]    433d504d  73555c3a  5c737265  6f706f70    M P = C : \ U s e r s \ p o p o
[7ffff8e0]    70415c30  74614470  6f4c5c61  5c6c6163    0 \ A p p D a t a \ L o c a l \
[7ffff8f0]    706d6554  73795300  526d6574  3d746f6f    T e m p . S y s t e m R o o t =
[7ffff900]    575c3a43  4f444e49  53005357  65747379    C : \ W I N D O W S . S y s t e
[7ffff910]    6972446d  433d6576  4553003a  4f495353    m D r i v e = C : . S E S S I O
[7ffff920]    4d414e4e  6f433d45  6c6f736e  54510065    N N A M E = C o n s o l e . Q T
[7ffff930]    5645445f  5f454349  45584950  41525f4c    _ D E V I C E _ P I X E L _ R A
[7ffff940]    3d4f4954  6f747561  42555000  3d43494c    T I O = a u t o . P U B L I C =
```

```
Memory and registers cleared

SPIM Version 9.1.21 of January 17, 2020
Copyright 1990-2017 by James Larus.
All Rights Reserved.
SPIM is distributed under a BSD license.
See the file README for a full copyright notice.
QtSPIM is linked to the Qt library, which is distributed under the GNU Lesser General Public License version 3 and version 2.1.
```

使用介面

QtSpim



Data segment

Text segment

操作列

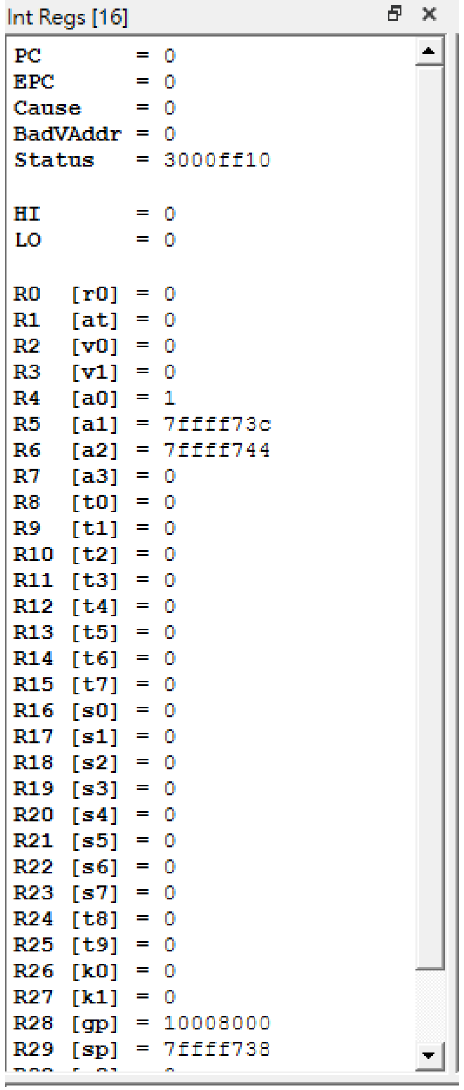
Register
暫存器

```
PC        = 0
EPC       = 0
Cause     = 0
BadVAddr  = 0
Status    = 3000ff10

HI        = 0
LO        = 0
```

```
User data segment [10000000]..[10040000]
[10000000]..[1003ffff]  00000000


User Stack [7ffff738]..[80000000]
[7ffff738]    00000001  7ffff7f2                                .
[7ffff740]    00000000  7fffffe1  7fffffba  7fffff89            .
[7ffff750]    7fffff4d  7fffff1c  7ffffeff  7ffffedb            M
[7ffff760]    7ffffea9  7ffffe9c  7ffffe86  7ffffe5c            .
```

使用介面

- Register 暫存器
  - 可查看暫存器的值
  - 右鍵可選擇別種進位顯示



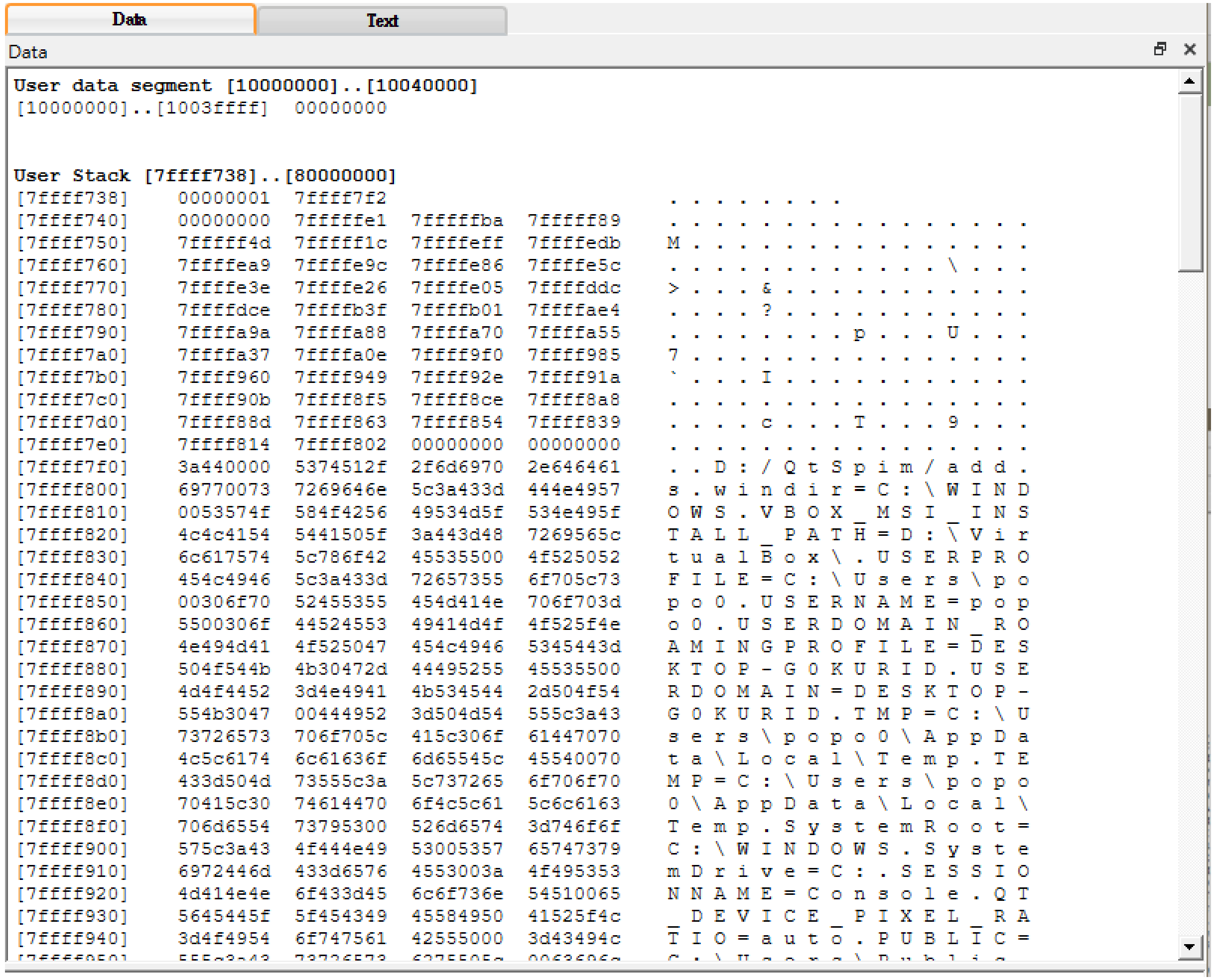
```
PC       = 0
EPC      = 0
Cause    = 0
BadVAddr = 0
Status   = 3000ff10

HI       = 0
LO       = 0

R0  [r0] = 0
R1  [at] = 0
R2  [v0] = 0
R3  [v1] = 0
R4  [a0] = 1
R5  [a1] = 7ffff73c
R6  [a2] = 7ffff744
R7  [a3] = 0
R8  [t0] = 0
R9  [t1] = 0
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0
R26 [k0] = 0
R27 [k1] = 0
R28 [gp] = 10008000
R29 [sp] = 7ffff738
```

QtSpim

使用介面

QtSpim

- Data segment
  - machine code

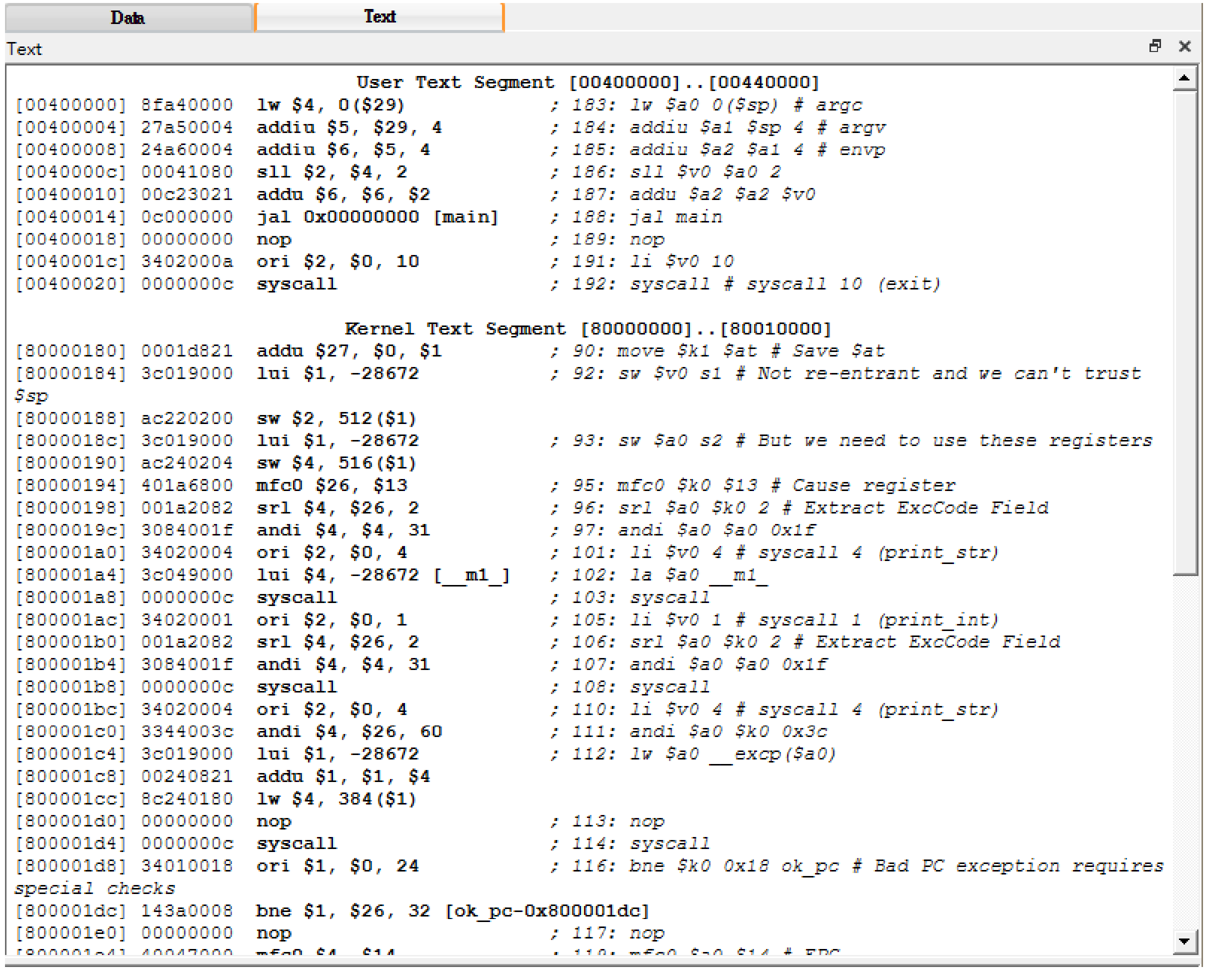


```
User data segment [10000000]..[10040000]
[10000000]..[1003ffff]  00000000


User Stack [7ffff738]..[80000000]
[7ffff738]    00000001  7ffff7f2                        . . . . . . . .
[7ffff740]    00000000  7fffffe1  7fffffba  7fffff89    . . . . . . . . . . . . . . . .
[7ffff750]    7fffff4d  7fffff1c  7ffffeff  7ffffedb    M . . . . . . . . . . . . . . .
[7ffff760]    7ffffea9  7ffffe9c  7ffffe86  7ffffe5c    . . . . . . . . . . . . \ . . .
[7ffff770]    7ffffe3e  7ffffe26  7ffffe05  7ffffddc    > . . . . & . . . . . . . . . .
[7ffff780]    7ffffdce  7ffffb3f  7ffffb01  7ffffae4    . . . . . ? . . . . . . . . . .
[7ffff790]    7ffffa9a  7ffffa88  7ffffa70  7ffffa55    . . . . . . . . p . . . U . . .
[7ffff7a0]    7ffffa37  7ffffa0e  7ffff9f0  7ffff985    7 . . . . . . . . . . . . . . .
[7ffff7b0]    7ffff960  7ffff949  7ffff92e  7ffff91a    ` . . . . I . . . . . . . . . .
[7ffff7c0]    7ffff90b  7ffff8f5  7ffff8ce  7ffff8a8    . . . . . . . . . . . . . . . .
[7ffff7d0]    7ffff88d  7ffff863  7ffff854  7ffff839    . . . . . c . . . T . . . 9 . . .
[7ffff7e0]    7ffff814  7ffff802  00000000  00000000    . . . . . . . . . . . . . . . .
[7ffff7f0]    3a440000  5374512f  2f6d6970  2e646461    . . D : / Q t S p i m / a d d .
[7ffff800]    69770073  7269646e  5c3a433d  444e4957    s . w i n d i r = C : \ W I N D
[7ffff810]    0053574f  584f4256  49534d5f  534e495f    O W S . V B O X _ M S I _ I N S
[7ffff820]    4c4c4154  5441505f  3a443d48  7269565c    T A L L _ P A T H = D : \ V i r
[7ffff830]    6c617574  5c786f42  45535500  4f525052    t u a l B o x \ . U S E R P R O
[7ffff840]    454c4946  5c3a433d  72657355  6f705c73    F I L E = C : \ U s e r s \ p o
[7ffff850]    00306f70  52455355  454d414e  706f703d    p o 0 . U S E R N A M E = p o p
[7ffff860]    5500306f  44524553  49414d4f  4f525f4e    o 0 . U S E R D O M A I N _ R O
[7ffff870]    4e494d41  4f525047  454c4946  5345443d    A M I N G P R O F I L E = D E S
[7ffff880]    504f544b  4b30472d  44495255  45535500    K T O P - G 0 K U R I D . U S E
[7ffff890]    4d4f4452  3d4e4941  4b534544  2d504f54    R D O M A I N = D E S K T O P -
[7ffff8a0]    554b3047  00444952  3d504d54  555c3a43    G 0 K U R I D . T M P = C : \ U
[7ffff8b0]    73726573  706f705c  415c306f  61447070    s e r s \ p o p o 0 \ A p p D a
[7ffff8c0]    4c5c6174  6c61636f  6d65545c  45540070    t a \ L o c a l \ T e m p . T E
[7ffff8d0]    433d504d  73555c3a  5c737265  6f706f70    M P = C : \ U s e r s \ p o p o
[7ffff8e0]    70415c30  74614470  6f4c5c61  5c6c6163    0 \ A p p D a t a \ L o c a l \
[7ffff8f0]    706d6554  73795300  526d6574  3d746f6f    T e m p . S y s t e m R o o t =
[7ffff900]    575c3a43  4f444e49  53005357  65747379    C : \ W I N D O W S . S y s t e
[7ffff910]    6972446d  433d6576  4553003a  4f495353    m D r i v e = C : . S E S S I O
[7ffff920]    4d414e4e  6f433d45  6c6f736e  54510065    N N A M E = C o n s o l e . Q T
[7ffff930]    5645445f  5f454349  45584950  41525f4c    _ D E V I C E _ P I X E L _ R A
[7ffff940]    3d4f4954  6f747561  42555000  3d43494c    T I O = a u t o . P U B L I C =
[7ffff950]    555c3a43  73726573  6275505c  0063696c    C : \ U s e r s \ P u b l i c
```

使用介面

- Text segment
  - assembly code

QtSpim



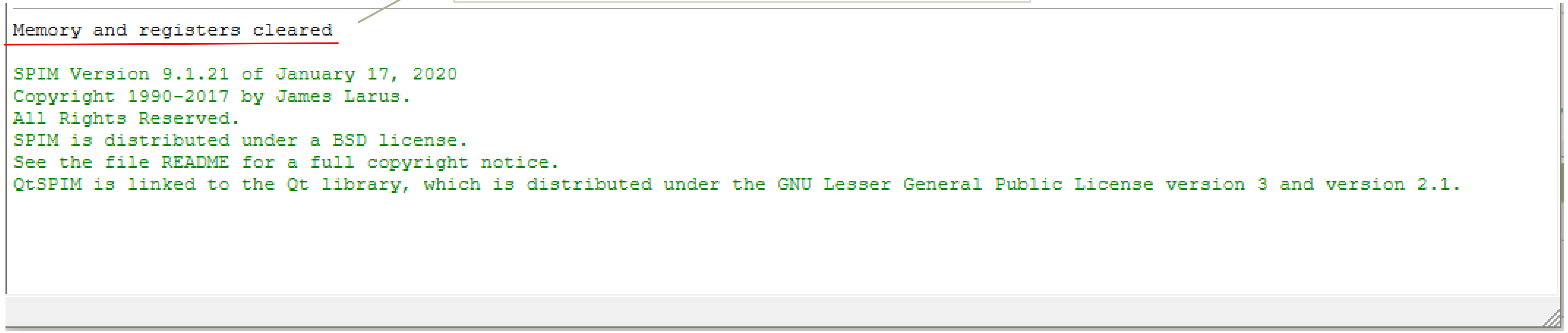
```
                       User Text Segment [00400000]..[00440000]
[00400000] 8fa40000  lw $4, 0($29)             ; 183: lw $a0 0($sp) # argc
[00400004] 27a50004  addiu $5, $29, 4          ; 184: addiu $a1 $sp 4 # argv
[00400008] 24a60004  addiu $6, $5, 4           ; 185: addiu $a2 $a1 4 # envp
[0040000c] 00041080  sll $2, $4, 2             ; 186: sll $v0 $a0 2
[00400010] 00c23021  addu $6, $6, $2           ; 187: addu $a2 $a2 $v0
[00400014] 0c000000  jal 0x00000000 [main]     ; 188: jal main
[00400018] 00000000  nop                       ; 189: nop
[0040001c] 3402000a  ori $2, $0, 10            ; 191: li $v0 10
[00400020] 0000000c  syscall                   ; 192: syscall # syscall 10 (exit)

                      Kernel Text Segment [80000000]..[80010000]
[80000180] 0001d821  addu $27, $0, $1          ; 90: move $k1 $at # Save $at
[80000184] 3c019000  lui $1, -28672            ; 92: sw $v0 s1 # Not re-entrant and we can't trust
$sp
[80000188] ac220200  sw $2, 512($1)
[8000018c] 3c019000  lui $1, -28672            ; 93: sw $a0 s2 # But we need to use these registers
[80000190] ac240204  sw $4, 516($1)
[80000194] 401a6800  mfc0 $26, $13             ; 95: mfc0 $k0 $13 # Cause register
[80000198] 001a2082  srl $4, $26, 2            ; 96: srl $a0 $k0 2 # Extract ExcCode Field
[8000019c] 3084001f  andi $4, $4, 31           ; 97: andi $a0 $a0 0x1f
[800001a0] 34020004  ori $2, $0, 4             ; 101: li $v0 4 # syscall 4 (print_str)
[800001a4] 3c049000  lui $4, -28672 [__m1_]    ; 102: la $a0 __m1_
[800001a8] 0000000c  syscall                   ; 103: syscall
[800001ac] 34020001  ori $2, $0, 1             ; 105: li $v0 1 # syscall 1 (print_int)
[800001b0] 001a2082  srl $4, $26, 2            ; 106: srl $a0 $k0 2 # Extract ExcCode Field
[800001b4] 3084001f  andi $4, $4, 31           ; 107: andi $a0 $a0 0x1f
[800001b8] 0000000c  syscall                   ; 108: syscall
[800001bc] 34020004  ori $2, $0, 4             ; 110: li $v0 4 # syscall 4 (print_str)
[800001c0] 3344003c  andi $4, $26, 60          ; 111: andi $a0 $k0 0x3c
[800001c4] 3c019000  lui $1, -28672            ; 112: lw $a0 __excp($a0)
[800001c8] 00240821  addu $1, $1, $4
[800001cc] 8c240180  lw $4, 384($1)
[800001d0] 00000000  nop                       ; 113: nop
[800001d4] 0000000c  syscall                   ; 114: syscall
[800001d8] 34010018  ori $1, $0, 24            ; 116: bne $k0 0x18 ok_pc # Bad PC exception requires
special checks
[800001dc] 143a0008  bne $1, $26, 32 [ok_pc-0x800001dc]
[800001e0] 00000000  nop                       ; 117: nop
[800001e4] 40047000  mfc0 $4, $14              ; 118: mfc0 $a0 $14 # EPC
```

使用介面

QtSpim

- Spim messages

初始狀態：記憶體與暫存器為清空



```
Memory and registers cleared

SPIM Version 9.1.21 of January 17, 2020
Copyright 1990-2017 by James Larus.
All Rights Reserved.
SPIM is distributed under a BSD license.
See the file README for a full copyright notice.
QtSPIM is linked to the Qt library, which is distributed under the GNU Lesser General Public License version 3 and version 2.1.
```

錯誤訊息也會顯示在這

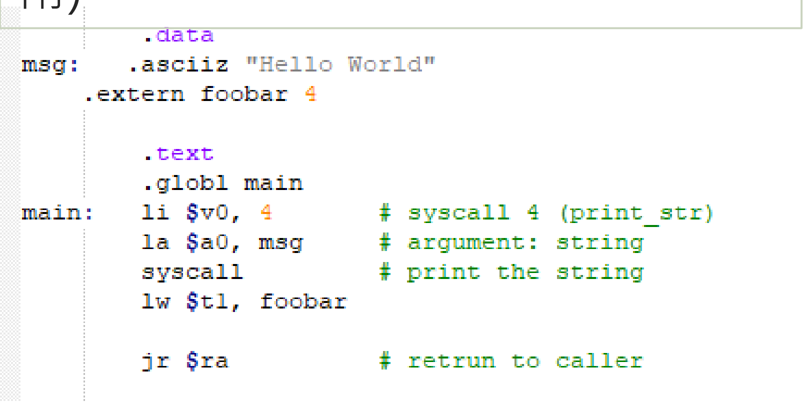
操作流程 - 編寫程式碼

QtSpim

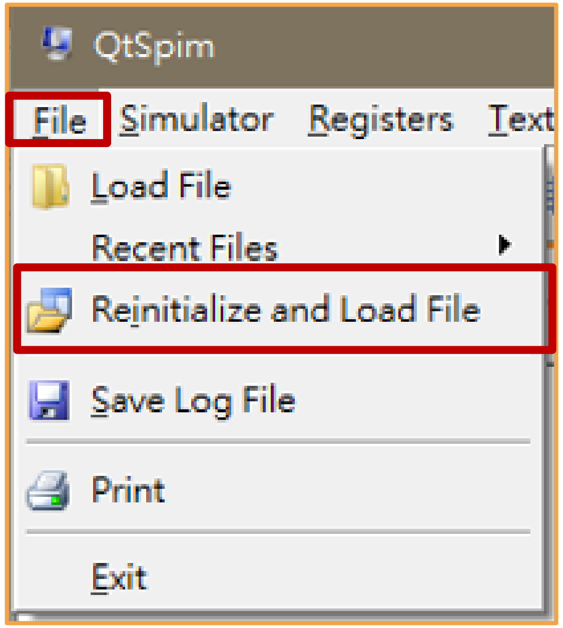
- 下載文字編輯器寫程式
- Notepad++
  - https://notepad-plus-plus.org/downloads/
- Subline text
  - https://www.sublimetext.com/

操作流程 - 讀檔

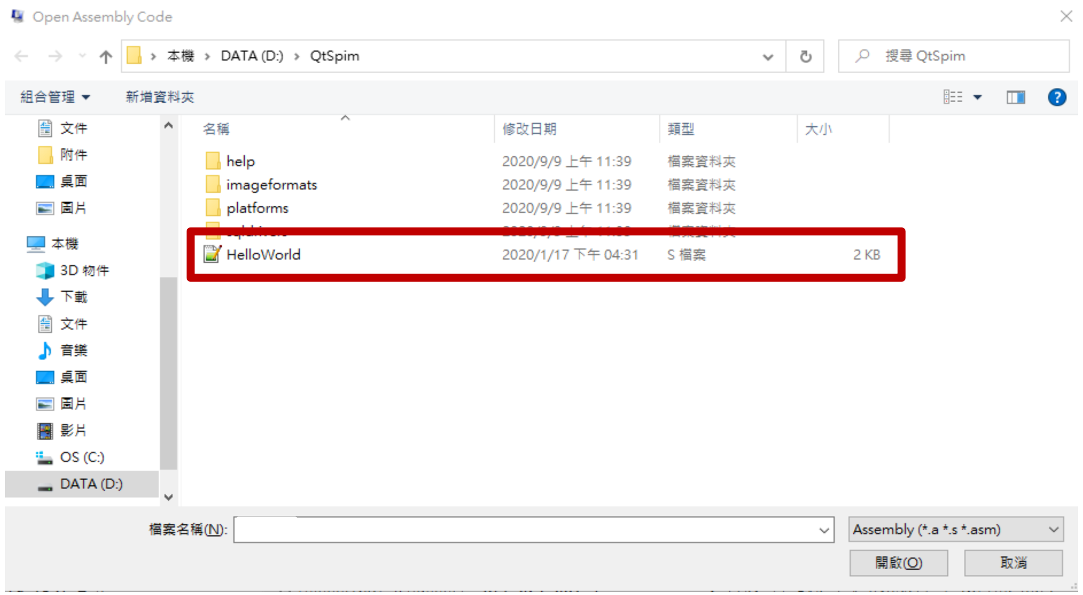
Hello World.s 檔案內容 (資料夾內附)



```
        .data
msg:   .asciiz "Hello World"
     .extern foobar 4

        .text
        .globl main
main:   li $v0, 4          # syscall 4 (print_str)
        la $a0, msg        # argument: string
        syscall            # print the string
        lw $t1, foobar

        jr $ra             # retrun to caller
```



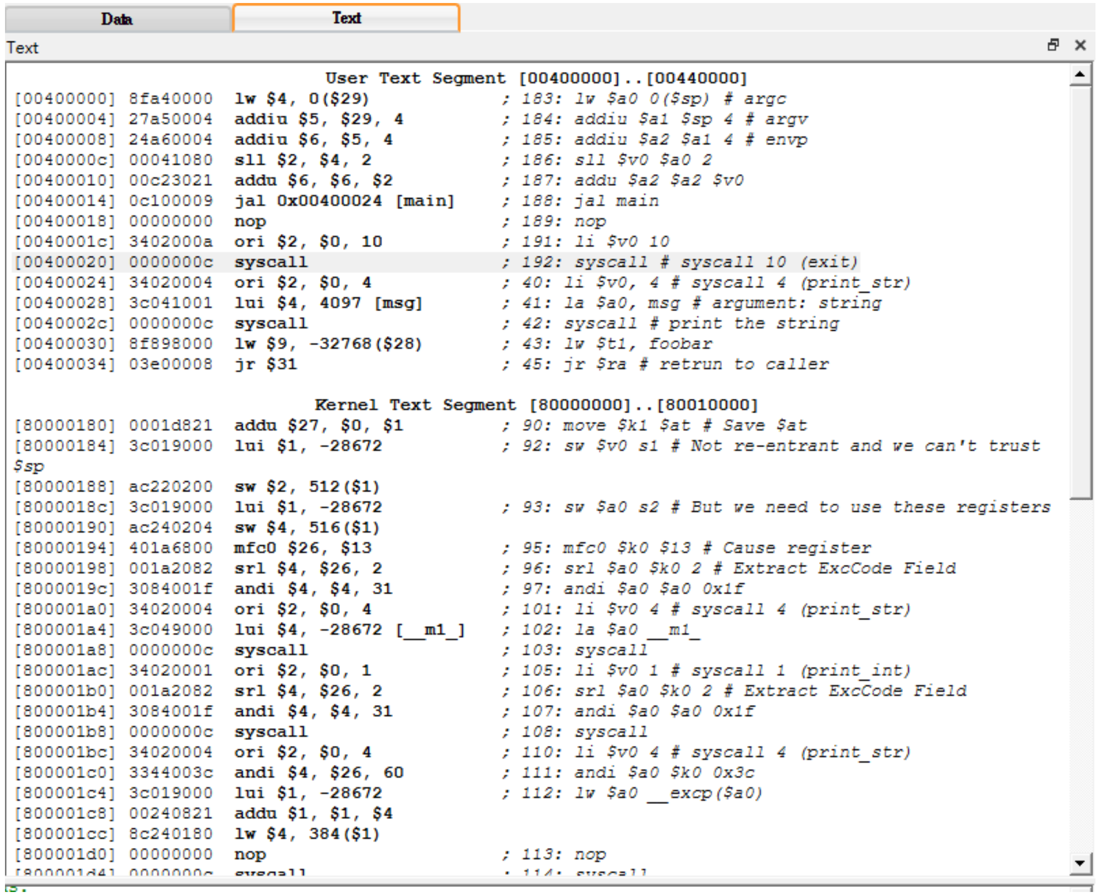
→



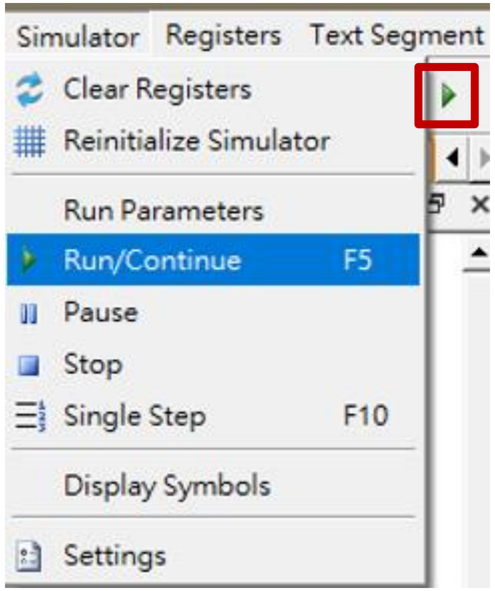
名稱	修改日期	類型	大小
help	2020/9/9 上午 11:39	檔案資料夾	
imageformats	2020/9/9 上午 11:39	檔案資料夾	
platforms	2020/9/9 上午 11:39	檔案資料夾	
HelloWorld	2020/1/17 下午 04:31	S 檔案	2 KB

操作流程 – 執行

QtSpim

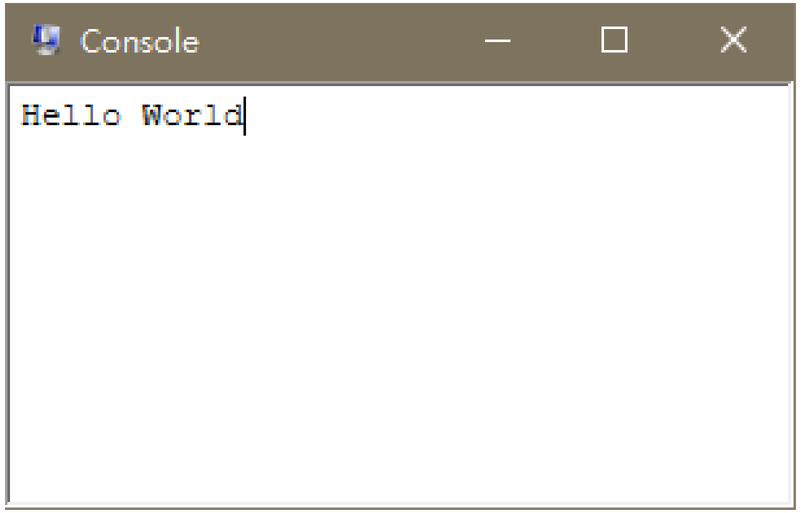


```
                    User Text Segment [00400000]..[00440000]
[00400000] 8fa40000  lw $4, 0($29)            ; 183: lw $a0 0($sp) # argc
[00400004] 27a50004  addiu $5, $29, 4         ; 184: addiu $a1 $sp 4 # argv
[00400008] 24a60004  addiu $6, $5, 4          ; 185: addiu $a2 $a1 4 # envp
[0040000c] 00041080  sll $2, $4, 2            ; 186: sll $v0 $a0 2
[00400010] 00c23021  addu $6, $6, $2          ; 187: addu $a2 $a2 $v0
[00400014] 0c100009  jal 0x00400024 [main]    ; 188: jal main
[00400018] 00000000  nop                      ; 189: nop
[0040001c] 3402000a  ori $2, $0, 10           ; 191: li $v0 10
[00400020] 0000000c  syscall                  ; 192: syscall # syscall 10 (exit)
[00400024] 34020004  ori $2, $0, 4            ; 40: li $v0, 4 # syscall 4 (print_str)
[00400028] 3c041001  lui $4, 4097 [msg]       ; 41: la $a0, msg # argument: string
[0040002c] 0000000c  syscall                  ; 42: syscall # print the string
[00400030] 8f898000  lw $9, -32768($28)       ; 43: lw $t1, foobar
[00400034] 03e00008  jr $31                   ; 45: jr $ra # retrun to caller

                   Kernel Text Segment [80000000]..[80010000]
[80000180] 0001d821  addu $27, $0, $1         ; 90: move $k1 $at # Save $at
[80000184] 3c019000  lui $1, -28672           ; 92: sw $v0 s1 # Not re-entrant and we can't trust
$sp
[80000188] ac220200  sw $2, 512($1)
[8000018c] 3c019000  lui $1, -28672           ; 93: sw $a0 s2 # But we need to use these registers
[80000190] ac240204  sw $4, 516($1)
[80000194] 401a6800  mfc0 $26, $13            ; 95: mfc0 $k0 $13 # Cause register
[80000198] 001a2082  srl $4, $26, 2           ; 96: srl $a0 $k0 2 # Extract ExcCode Field
[8000019c] 3084001f  andi $4, $4, 31          ; 97: andi $a0 $a0 0x1f
[800001a0] 34020004  ori $2, $0, 4            ; 101: li $v0 4 # syscall 4 (print_str)
[800001a4] 3c049000  lui $4, -28672 [__m1_]   ; 102: la $a0 __m1_
[800001a8] 0000000c  syscall                  ; 103: syscall
[800001ac] 34020001  ori $2, $0, 1            ; 105: li $v0 1 # syscall 1 (print_int)
[800001b0] 001a2082  srl $4, $26, 2           ; 106: srl $a0 $k0 2 # Extract ExcCode Field
[800001b4] 3084001f  andi $4, $4, 31          ; 107: andi $a0 $a0 0x1f
[800001b8] 0000000c  syscall                  ; 108: syscall
[800001bc] 34020004  ori $2, $0, 4            ; 110: li $v0 4 # syscall 4 (print_str)
[800001c0] 3344003c  andi $4, $26, 60         ; 111: andi $a0 $k0 0x3c
[800001c4] 3c019000  lui $1, -28672           ; 112: lw $a0 __excp($a0)
[800001c8] 00240821  addu $1, $1, $4
[800001cc] 8c240180  lw $4, 384($1)
[800001d0] 00000000  nop                      ; 113: nop
[800001d4] 0000000c  syscall                  ; 114: syscall
```

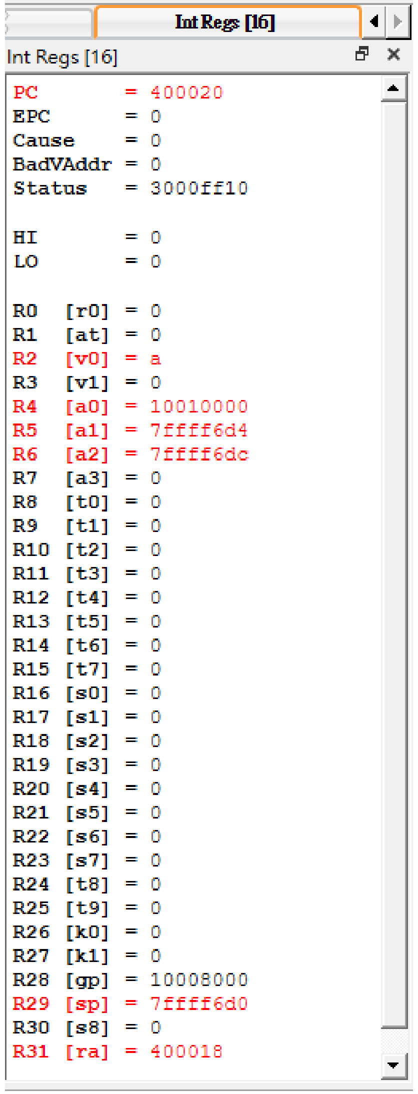


執行結果

- 查看結果並確認暫存器值



```
Hello World
```



```
PC         = 400020
EPC        = 0
Cause      = 0
BadVAddr   = 0
Status     = 3000ff10

HI         = 0
LO         = 0

R0   [r0]  = 0
R1   [at]  = 0
R2   [v0]  = a
R3   [v1]  = 0
R4   [a0]  = 10010000
R5   [a1]  = 7ffff6d4
R6   [a2]  = 7ffff6dc
R7   [a3]  = 0
R8   [t0]  = 0
R9   [t1]  = 0
R10  [t2]  = 0
R11  [t3]  = 0
R12  [t4]  = 0
R13  [t5]  = 0
R14  [t6]  = 0
R15  [t7]  = 0
R16  [s0]  = 0
R17  [s1]  = 0
R18  [s2]  = 0
R19  [s3]  = 0
R20  [s4]  = 0
R21  [s5]  = 0
R22  [s6]  = 0
R23  [s7]  = 0
R24  [t8]  = 0
R25  [t9]  = 0
R26  [k0]  = 0
R27  [k1]  = 0
R28  [gp]  = 10008000
R29  [sp]  = 7ffff6d0
R30  [s8]  = 0
R31  [ra]  = 400018
```

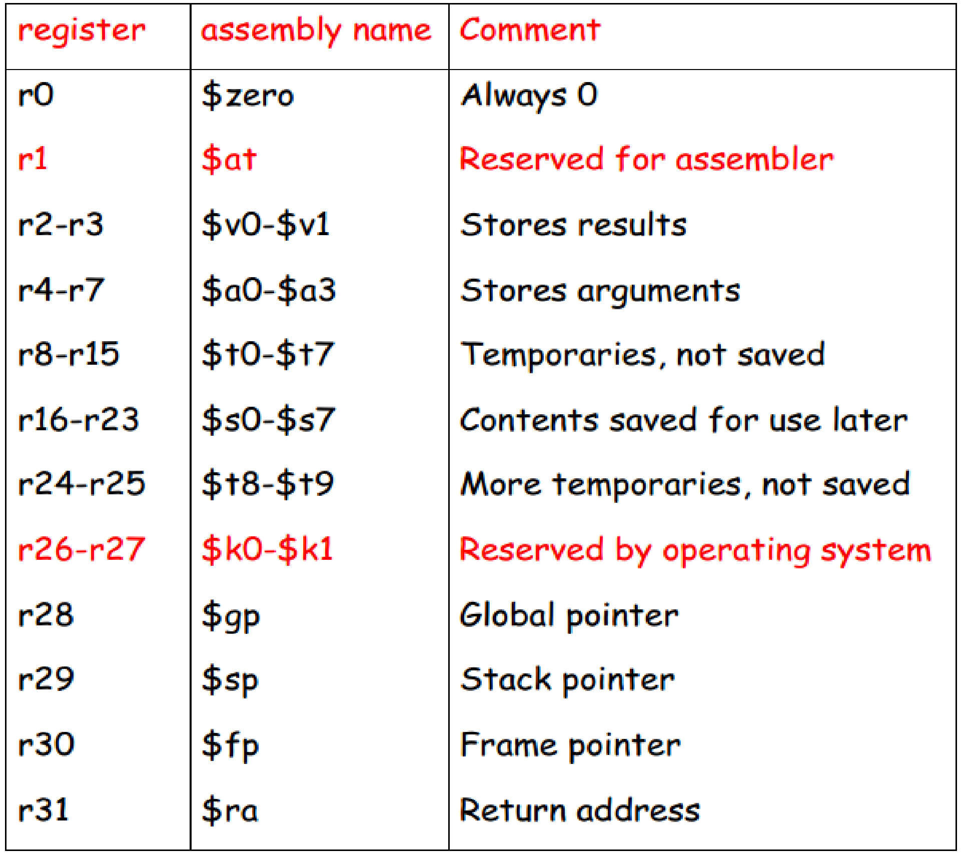
紅字代表暫存器值有改變

DEMO

System Call Codes

Service	Code (put in $v0)	Arguments	Result
print_int	1	$a0 = integer	
print_float	2	$f12 = float	
print_double	3	$f12 = double	
print_string	4	$a0 = addr. of string	
read_int	5		int in $v0
read_float	6		float in $f0
read_double	7		double in $f0
read_string	8	$a0 = buffer, $a1=length	
sbrk	9	$a0 = amount	addr in $v0
exit	10		

Overview of MIPS ISA: registers



register	assembly name	Comment
r0	$zero	Always 0
r1	$at	Reserved for assembler
r2-r3	$v0-$v1	Stores results
r4-r7	$a0-$a3	Stores arguments
r8-r15	$t0-$t7	Temporaries, not saved
r16-r23	$s0-$s7	Contents saved for use later
r24-r25	$t8-$t9	More temporaries, not saved
r26-r27	$k0-$k1	Reserved by operating system
r28	$gp	Global pointer
r29	$sp	Stack pointer
r30	$fp	Frame pointer
r31	$ra	Return address

作業一 - 四則運算

- *A*為$S0，*B*為$S1，*ans*為$S3
- 計算 $ans = (A * 6 + 8) * (B / 2)$

- *A*、*B* 先設定任一值並驗證是否計算正確，demo時助教會改值
- 不可使用mul、mult
- 不限制暫存器使用個數

作業二 - MIPS for loop

- $A = [n_0, n_1, n_2, n_3, n_4]$ ($n_{0\text{-}4}$皆為正整數)
- $Target = n_a + n_b$ (n_a, n_b為A中的數字, a不會等於b)
- 算出*Target*分別為那兩個在A中數字做相加

- A需由array的方式做存取
- A中不會出現相同的數字，*Target*必定為A中兩個數字相加
- 不限制暫存器使用個數
- 要print出a, b分別為兩個數字(a, b之間以空白符隔開)

Example:
Input:
$A = [2, 4, 7, 8, 10]$
$Target = 9$

Output : 0 2

評分標準

- 線上demo應完成事項
  - 解釋code
  - 以QtSpim展示結果
    - 有無error
    - 觀察register的值 (對應code)
- 本次只接受線上demo