addi $s0, $zero, 0x7ff8

addi $s0, $s0, 0x7ff8 - $s0 store IO address 0xfff0

lw $t0, 0xc($s0) main

andi $t1, $t0, 0x1

andi $t2, $t0, 0x2

andi $t3, $t0, 0x4

srl $t2, $t2, 0x1

srl $t3, $t3, 0x2

bne $t1, $zero, 0x3 --> branch to INPUT\_PAT

bne $t2, $zero, 0x5 --> branch to INPUT\_SRC

bne $t3, $zero, 0x7 --> branch to KMP

j 0x2 --> jump to main

jal 21 --> jump to input\_pat INPUT\_PAT

NOP

j 0x2 --> return to main

jal 42 --> jump to input\_src INPUT\_SRC

NOP

j 0x2 --> return to main

jal 63 --> jump to kmp KMP

NOP

j 0x2 --> return to main

addi $t8, $zero, 0x1 } input\_pat

sw $t8, 0x4($s0) - set $t8 = cpuvalid = 1 }

addi $t9, $zero, 0x4 - set $t9 = 4 as str index }

sw $zero, 0x4($s0) - set cpuvalid = 0 first } } A

lw $t0, 0xc($s0) } } }

srl $t0, $t0, 0x18 } } }

andi $t0, $t0, 0x1 } } }

beq $t0, $zero, -4 - loop to wait keyboardvalid } } }

sw $t8, 0x4($s0) - set cpuvalid = 1 } }

lw $t0, 0xc($s0) } }

srl $t0, $t0, 0x10 } }

andi $t1, $t0, 0xff - $t1 : fresh keyboard data } }

sw $t1, 0x0($s0) - $t0 : display data to led } }

sw $t1, 0x1000($t9) } }

addi $t9, $t9, 0x4 - index++ } }

bne $t1, $zero, -13 --> branch to A } }

sw $zero, 0x0($s0) }

sw $zero, 0x4($s0) }

addi $t9, $t9, -8 }

sw $t9, 0x1000($zero) }

jr $ra } return

addi $t8, $zero, 0x1 } input\_src

sw $t8, 0x4($s0) - set $t8 = cpuvalid = 1 }

addi $t9, $zero, 0x4 - set $t9 = 4 as str index }

sw $zero, 0x4($s0) - set cpuvalid = 0 first } } B

lw $t0, 0xc($s0) } } }

srl $t0, $t0, 0x18 } } }

andi $t0, $t0, 0x1 } } }

beq $t0, $zero, -4 - loop to wait keyboardvalid } } }

sw $t8, 0x4($s0) - set cpuvalid = 1 } }

lw $t0, 0xc($s0) } }

srl $t0, $t0, 0x10 } }

andi $t1, $t0, 0xff - $t1 : fresh keyboard data } }

sw $t1, 0x0($s0) - $t0 : display data to led } }

sw $t1, 0x1800($t9) } }

addi $t9, $t9, 0x4 - index++ } }

bne $t1, $zero, -13 --> branch to B } }

sw $zero, 0x0($s0) }

sw $zero, 0x4($s0) }

addi $t9, $t9, -8 }

sw $t9, 0x1800($zero) }

jr $ra } return

lw $k0, 0x1000($zero) } - store pat len in $k0

lw $k1, 0x1800($zero) } - store src len in $k1

addi $at, $zero, 0x0 - store result cnt in $at

addi $t1, $zero, 0x0 } - $t1 as src index

beq $t1, $k1, 0xf } - Loop1, branch to A

addi $t0, $zero, 0x0 } } - $t0 as pat index

beq $t0, $k0, 0x7 } } - Loop2, branch to B

add $t2, $t0, $t1 } }

beq $t2, $k1, 0x9 } } - branch to C

lw $t8, 0x1004($t0) } }

lw $t9, 0x1804($t2) } }

bne $t8, $t9, 0x6 } } - branch to C

addi $t0, $t0, 0x4 } }

beq $zero, $zero, -8 } } - branch to Loop2

srl $v0, $t1, 0x2 } - B

addi $v0, $v0, 0x1 }

sw $v0, 0x1f00($at) }

addi $at, $at, 0x4 }

addi $t1, $t1, 0x4 } - C

beq $zero, $zero, -16 } - branch to Loop1

sw $zero, 0x1f00($at) - A

addi $a2,$zero,0

addi $s1,$zero,1

addi $s2,$zero,2

addi $s3,$zero,3

addi $s4,$zero,4

addi $s5,$zero,5

addi $s6,$zero,6

addi $s7,$zero,7

addi $t0,$zero,8

addi $t9,$zero,9

addi $t1,$zero,0

addi $t7,$zero,100

DEPART:

lw $t2,0x1000($zero) #/取出显示数 \*\*\*\*\*\*\*\*

addi $a3,$zero,0x1000

add $t3,$zero,$zero #累加数

add $t4,$zero,$zero #十位

add $t5,$zero,$zero #个位

addi $t6,$zero,10 #存10，作比较

beq $t2,$t3,SHOW\_FAIL #若为0则直接显示

WHILE:

addi $t3,$t3,1 #累加数加1

addi $t5,$t5,1 #个位加1

bne $t5,$t6,ElSE1 #若个位等于10

add $t5,$zero,$zero #个位清零

addi $t4,$t4,1 #十位进1

ElSE1:

beq $t3,$t2,ELSE2 #个位不等于10，判断累计数是否等于显示数

beq $zero,$zero,WHILE #不等于则返回循环

ELSE2:

beq $zero,$zero,SHOW #等于则开始显示

DEPART\_1:

lw $t2,0x4($a3)

addi $a3,$a3,4

addi $t1,$zero,0

add $t3,$zero,$zero #累加数

add $t4,$zero,$zero #十位

add $t5,$zero,$zero #个位

addi $t6,$zero,10 #存10，作比较

beq $t2,$t3,SHOW\_NONE #若为0则直接显示

WHILE1:

addi $t3,$t3,1 #累加数加1

addi $t5,$t5,1 #个位加1

bne $t5,$t6,ElSE11 #若个位等于10

add $t5,$zero,$zero #个位清零

addi $t4,$t4,1 #十位进1

ElSE11:

beq $t3,$t2,ELSE21 #个位不等于10，判断累计数是否等于显示数

beq $zero,$zero,WHILE1 #不等于则返回循环

ELSE21:

beq $zero,$zero,SHOW #等于则开始显示

SHOW:

addi $t1,$t1,1 #t1做循环变量

beq $t1,$t7,DEPART\_1

SWITCH:

addi $t8,$zero,0

beq $a2,$zero,CHANGE1 # 位选为0 就置1，为1就置0

bne $a2,$zero,CHANGE2

CHANGE1:

addi $a2,$zero,1

addi $t8,$t8,1

addi $a0,$zero,3328 #显示十位数 1101

bne $t8,$t7,CHANGE1

beq,$zero,$zero,CONTINUE1

CHANGE2:

add $a2,$zero,$zero

addi $t8,$t8,1

addi $a0,$zero,3584 #显示个位数 1110

bne $t8,$t7,CHANGE2

beq,$zero,$zero,CONTINUE2

CONTINUE1: #段选信号实现，t4为十位

beq $t4,$zero,SHOW\_0

beq $t4,$s1,SHOW\_1

beq $t4,$s2,SHOW\_2

beq $t4,$s3,SHOW\_3

beq $t4,$s4,SHOW\_4

beq $t4,$s5,SHOW\_5

beq $t4,$s6,SHOW\_6

beq $t4,$s7,SHOW\_7

beq $t4,$t0,SHOW\_8

beq $t4,$t9,SHOW\_9

CONTINUE2: #段选信号实现，t5为个位

beq $t5,$zero,SHOW\_0

beq $t5,$s1,SHOW\_1

beq $t5,$s2,SHOW\_2

beq $t5,$s3,SHOW\_3

beq $t5,$s4,SHOW\_4

beq $t5,$s5,SHOW\_5

beq $t5,$s6,SHOW\_6

beq $t5,$s7,SHOW\_7

beq $t5,$t0,SHOW\_8

beq $t5,$t9,SHOW\_9

SHOW\_0:

addi $a0,$a0,0xc0

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_1:

addi $a0,$a0,0xf9

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_2:

addi $a0,$a0,0xa4

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_3:

addi $a0,$a0,0xb0

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_4:

addi $a0,$a0,0x99

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_5:

addi $a0,$a0,0x92

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_6:

addi $a0,$a0,0x82

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_7:

addi $a0,$a0,0xf8

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_8:

addi $a0,$a0,0x80

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_9:

addi $a0,$a0,0x90

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW

SHOW\_FAIL:

addi $t8,$zero,0

beq $a2,$zero,TO1 # 位选为0 就置1，为1就置0

beq $a2,$s1,TO2

beq $a2,$s2,TO3

beq $a2,$s3,TO4

TO1:

addi $a2,$zero,1

addi $t8,$t8,1

addi $a0,$zero,1792 #显示千位 0111

bne $t8,$t7,TO1

beq,$zero,$zero,FFFF

TO2:

addi $a2,$zero,2

addi $t8,$t8,1

addi $a0,$zero,2816 #显示百位数 1011

bne $t8,$t7,TO2

beq,$zero,$zero,AAAA

TO3:

addi $a2,$zero,3

addi $t8,$t8,1

addi $a0,$zero,3328 #显示十位数 1101

bne $t8,$t7,TO3

beq,$zero,$zero,IIII

TO4:

add $a2,$zero,$zero

addi $t8,$t8,1

addi $a0,$zero,3584 #显示个位数 1110

bne $t8,$t7,TO4

beq,$zero,$zero,LLLL

FFFF:

addi $a0,$a0,0x8e

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_FAIL

AAAA:

addi $a0,$a0,0x88

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_FAIL

IIII:

addi $a0,$a0,0xf9

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_FAIL

LLLL:

addi $a0,$a0,0xc7

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_FAIL

SHOW\_NONE:

addi $t8,$zero,0

beq $a2,$zero,TO11 # 位选为0 就置1，为1就置0

beq $a2,$s1,TO22

beq $a2,$s2,TO33

beq $a2,$s3,TO44

TO11:

addi $a2,$zero,1

addi $t8,$t8,1

addi $a0,$zero,1792 #显示千位 0111

bne $t8,$t7,TO11

beq,$zero,$zero,FFFF1

TO22:

addi $a2,$zero,2

addi $t8,$t8,1

addi $a0,$zero,2816 #显示百位数 1011

bne $t8,$t7,TO22

beq,$zero,$zero,AAAA1

TO33:

addi $a2,$zero,3

addi $t8,$t8,1

addi $a0,$zero,3328 #显示十位数 1101

bne $t8,$t7,TO33

beq,$zero,$zero,IIII1

TO44:

add $a2,$zero,$zero

addi $t8,$t8,1

addi $a0,$zero,3584 #显示个位数 1110

bne $t8,$t7,TO44

beq,$zero,$zero,LLLL1

FFFF1:

addi $a0,$a0,0xc8

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_NONE

AAAA1:

addi $a0,$a0,0xc0

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_NONE

IIII1:

addi $a0,$a0,0xc8

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_NONE

LLLL1:

addi $a0,$a0,0x86

sw $a0,0xfff8($zero)

beq,$zero,$zero, SHOW\_NONE