

1. Polling takes more time than interrupts. However, if we know that certain even will last certain amount of time, polling can be ideal.

4.

(a) ISR first.

(b) ISR first.

(c) CPU first.

(d) CPU first.

5.

(a) Software context save and restore is performed.

(b) Shadow register set reduces ISR entry and exit time

8.

(a)

IEC0SET = 0b1<<8; // enable the 8th bit

IFS0CLR = 0b1<<8;

IPC2CLR = 0x001F;

IPC2SET = 0b00010110;

(b)

IEC1SET = 0b1<<15;

IFS1CLR = 0b1<<15;

IPC8CLR = 0x1F<<24;

IPC8SET = 0b11001<<24;

(c)

IEC2SET = 0b1<<4;

IFS2CLR = 0b1<<4;

IPC12CLR = 0x1F<<8;

IPC12SET = 0b11111<<8;

(d)

IECOSET = 0b1<<11;

IFS0CLR = 0b1<<11;

IPC2CLR = 0x1F<<24;

IPC2SET = 0b1110<<8;

INCONSET = 1;

3.

(a) Float and long double have jumps.

```
f3 = f1/f2;
9d008720: 8fc50034      lw      a1,52(s8)
9d008724: 8fc40030      lw      a0,48(s8)
9d008728: 0f402345      jal     9d008d14 <__divsf3>
9d00872c: 00000000      nop
9d008730: afc20058      sw      v0,88(s8)

d3 = d1+d2;
9d008734: 8fc60040      lw      a2,64(s8)
9d008738: 8fc70044      lw      a3,68(s8)
9d00873c: 8fc40038      lw      a0,56(s8)
9d008740: 8fc5003c      lw      a1,60(s8)
9d008744: 0f401f65      jal     9d007d94 <__adddf3>
9d008748: 00000000      nop
9d00874c: afc20060      sw      v0,96(s8)
9d008750: afc30064      sw      v1,100(s8)
```

(b) Char, int, long long int do not have jumps. Char is not involved in the smallest data type. Char has an extra andi command which will remove the most significant bit.

(c)

Lines					
	char	int	long long	float	long double
+	5	4	11	5	8
-	5	4	11	5	8
*	5	4	18	5	8
/	7	7	8	5	8
Ratio					
	char	int	long long	float	long double
+	1.25	1	2.75	1.25	2
-	1.25	1	2.75	1.25	2
*	1.25	1	4.5	1.25	2
/	1.4	1.4	1.6	1	1.6

(d)

According to the screenshot, subadd is one of the subroutines that uses 0x430 bytes of memory.

kseg0 section	address	length [bytes]	(dec)	Description
.text.dp32mul	0x9d007490	0x4b8	1208	
.text	0x9d007948	0x444	1092	App's exec code
.text.dp32subadd	0x9d007d8c	0x430	1072	
.text.dp32mul	0x9d0081bc	0x32c	812	
.text	0x9d0084e8	0x5b4	1460	App's exec code
.text.fpsubadd	0x9d008a9c	0x278	632	
.text.fp32div	0x9d008d14	0x230	560	
.text.fp32mul	0x9d008f44	0x1bc	444	
.text.general_exception	0x9d009100	0xdc	220	
.text.dinit.startup	0x9d0091dc	0x80	128	
.text.main_entry	0x9d00925c	0x54	84	
.text._bootstrap_except	0x9d0092b0	0x48	72	
.text._general_exception	0x9d0092f8	0x48	72	
.vector_default	0x9d009340	0x48	72	
.text	0x9d009388	0x44	68	App's exec code
.text.dinit.startup	0x9d0093cc	0x20	32	
.MIPS.abiflags	0x9d0093f0	0x18	24	
.rodata	0x9d009408	0x18	24	Read-only const
.text	0x9d009420	0x18	24	App's exec code
.dinit	0x9d009438	0x10	16	
.text._on_reset	0x9d009448	0x8	8	
.text._on_bootstrap	0x9d009450	0x8	8	

4.

&	4
	4
>>	3
<<	3