

Taller #001 - Febrero de 2016 - v2

1. Escriba los 4 principios de diseño de hardware aprendidos en clase.

- 1. La simplicidad favorece la regularidad
- 2. Entre más pequeño, más rápido
- 3. Hacer el caso común más rápido
- 4. Buenos diseños demandan buenos compromisos

2. Convertir a instrucciones de bajo nivel.

```
int x=0;
int y =8;
int z = 1;
# Inicializo las variables
or %g0, 0, %l0; # x = 0
or %g0, 8, %l1; # y = 8r %g0, 1, %l2; # z = 1
y=x+3;
z=z+3;
x=(x-z)+(3+y);
add %l0, 3, %l1; # y = x + 3
add %l2, 3, %l2; # z = z + 3
sub %l0, %l2, %l3; # %l3 = x - z (tmp)
add %l1, 3, %l0; # x = 3 + y
add %l0, %l3, %l0; # x = (x - z) + (3 + y)
```

3. Usar el ld, y st.

```
a[4]= a[2]+x;
y = y[40]+13;
# Presuponemos a[] = %l0, y x = %l1
ld [%l0 + (2 * 4)], %l2; # tmp1 = a[2]
add %l2, %l1, %l2; # tmp1 = a[2] + x
st %l2, [%l0 + (4 * 4)]; # a[4] = a[2] + x
```

4. Convertir a lenguaje de máquina.

```
a.
int main(){
  int i =3; p=2;
  return i+3;
}
or %g0, 3, %l0; # i = 3
or %g0, 2, %l1; # p = 2
add %l0, 3, %o0; # return i + 3
```

opcode	op	rd	op3	rs1	i	simm13 / asi + rs2	hex
or	10	10000	000010	00000	1	00000000000011	A0102003
or	10	10001	000010	00000	1	00000000000010	A2102002
add	10	01000	000000	10000	1	00000000000011	90042003

```
b.
int main(){
  int p=3; x=1; z=4;
  int w=0;
  w=(p+40)+(x-z);
  return 0;
}
```

```
# Inicializo las variables
or %g0, 3, %l0; # p = 3
or %g0, 1, %l1; # x = 1
or %g0, 4, %l2; # z = 4
or %g0, 0, %l3; # w = 0
```

opcode	op	rd	op3	rs1	i	simm13 / asi + rs2	hex
or	10	10000	000010	00000	1	00000000000011	A0102003
or	10	10001	000010	00000	1	00000000000001	A2102001
or	10	10010	000010	00000	1	00000000000100	A4102004
or	10	10011	000010	00000	1	00000000000000	A6102000

```
# Procedo con las operaciones
add %l0, 40, %l4; tmp1 = p + 40
sub %l1, %l2, %l5; tmp2 = x + z
add %l4, %l5, %l3; w = (p+40) + (x-z)
```

opcode	op	rd	op3	rs1	i	simm13 / asi + rs2		hex
add	10	10100	000000	10000	1	0000000101000		A8042028
sub	10	10101	000100	10001	0	00000000	10010	AA244012
add	10	10011	000000	10100	0	00000000	10101	A6050015

5. Inicializar las siguientes variables negativas usando OR.

```
n=-12,
a=-11,
b=-14.
or %g0, -12, %l0; # n = -12
or %g0, -11, %l1; # a = -11
or %g0, -14, %l2; # b = -14
```