

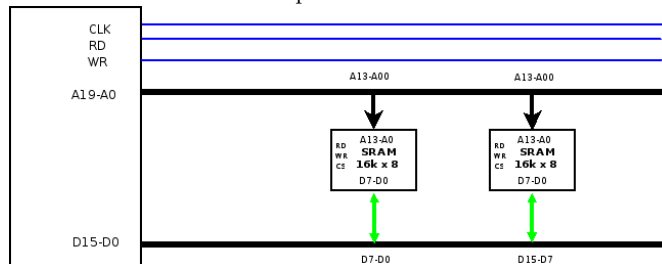
Pauta auxiliar 5
Rodrigo Cánovas
10 de Octubre del 2008

1. Problema 1

```
// Busqueda binaria
int p ( int x , int *a, int n )
{
    int j = n - 1;
    int i = 0;
    do
    {
        unsigned int k = i + j ;
        k /= 2 ;
        int e = a[k];
        if ( x == e )
            return k ;
        else if ( e >= x )
            j = k - 1 ;
        else
            i = k + 1;
    } while ( i <= j ) ;
    return -1;
}
```

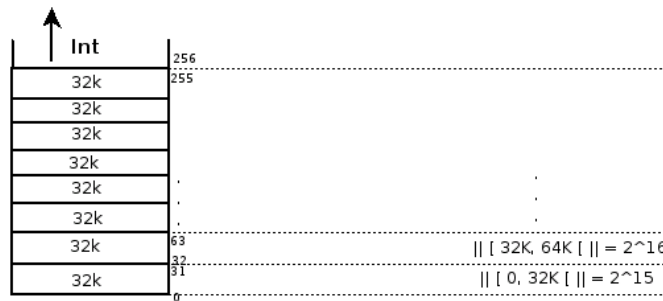
2. Problema 2

- Lo primero sera rescatar los datos:
 $|BusdeDatos| = 16bits$
 $|BusdeDireccion| = 20bits$
- El primer problema con el cual nos encontramos es que el ancho de memoria de los SRAM (8bits) no coinciden con el bus de datos. Para solucionar esto basta con colocar dos SRAM en paralelo.



(Notar que ahora tengo en total 32Kb en memoria)

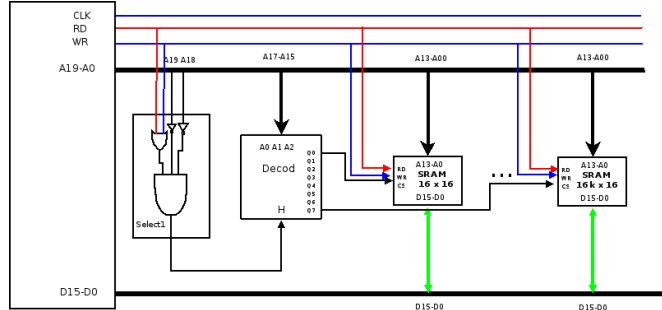
- Ahora necesitamos diseñar la interfaz para los 256K (es decir entre $[0, 256K]$).



entonces tenemos lo siguiente:

A_{19}	A_{18}	A_{17}	A_{16}	A_{15}	A_{14}	$A_{13} - A_{00}$	
0	0	0	0	0	*	*	$[0, 32k[$
0	0	0	0	1	*	*	$[32k, 64k[$
0	0	0	1	0	*	*	$[64k, 96k[$
0	0	0	1	1	*	*	$[96k, 128k[$
0	0	1	0	0	*	*	$[128k, 160k[$
0	0	1	0	1	*	*	$[160k, 192k[$
0	0	1	1	0	*	*	$[192k, 224k[$
0	0	1	1	1	*	*	$[224k, 256k[$

- finalmente solo falta el diseño de la interfaz de memoria:



Con el selector nos aseguramos que estemos dentro del rango de los 256k y con el decodificador avisamos al chip SRAM que deba seleccionarse segun lo que venga en la direccion de memoria.