

ENADS 10 11-19-08

11

REVIEW -

ODD PARITY CIRCUIT



TRUTH TABLE

a	b	c	d
0	0	0	1
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

[2]

Logical Expression

$$X \equiv [\bar{a} \bar{b} \bar{c} + (\bar{a} \cdot b) \cdot c] + [a \cdot \bar{b} \cdot c + (a \cdot b) \cdot \bar{c}]$$

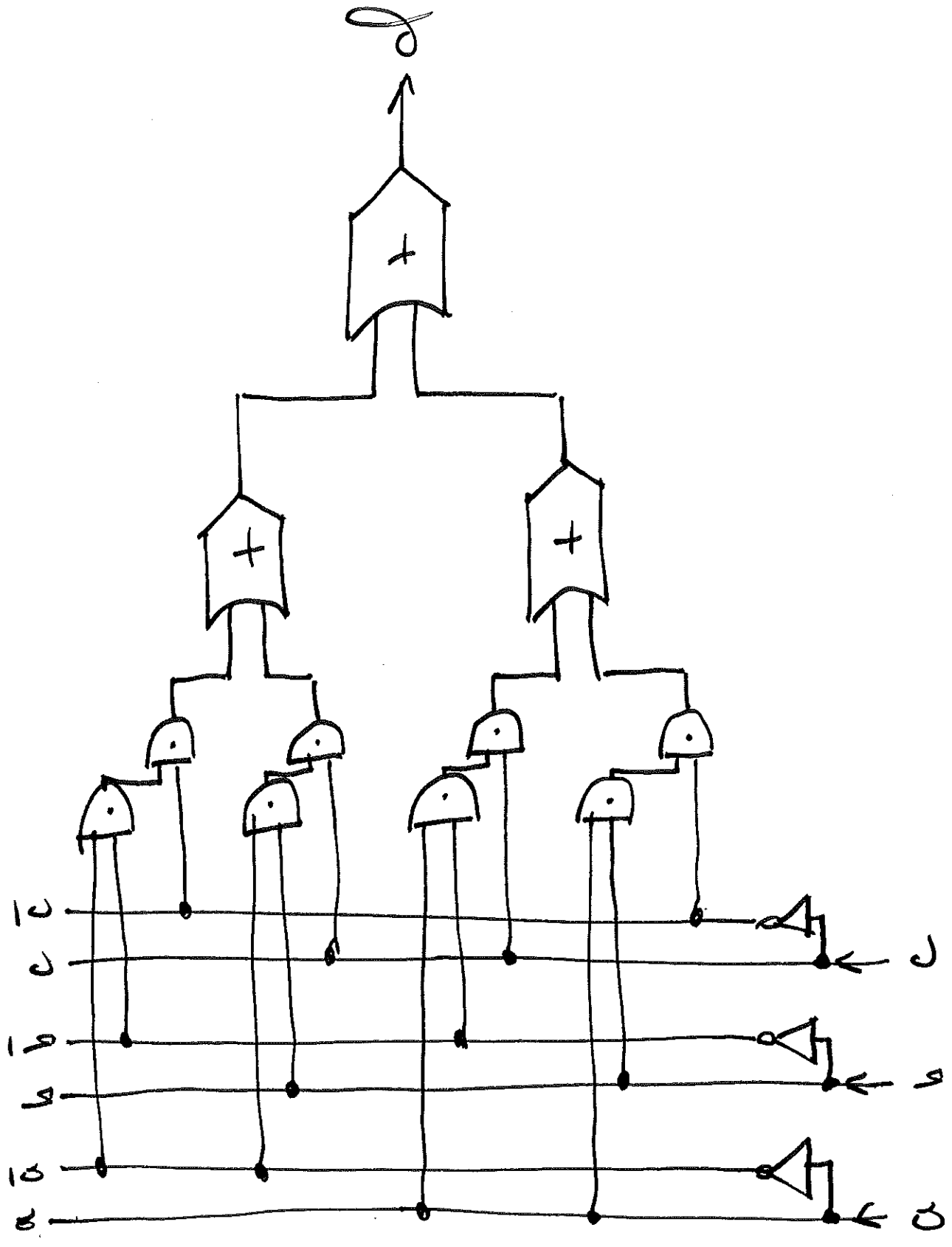
$$X \equiv [(\bar{a} \cdot \bar{b}) \cdot \bar{c} + (\bar{a} \cdot b) \cdot c] + [(a \cdot \bar{b}) \cdot c + (a \cdot b) \cdot \bar{c}]$$

Axioms: $\bar{a} \bar{b} \bar{c} \neq \overline{abc}$, $\bar{a} \cdot \bar{b} \neq \overline{a \cdot b}$

DeMorgan's
laws $\left\{ \begin{array}{l} \overline{a \cdot b} \equiv \bar{a} + \bar{b} \\ \overline{a + b} \equiv \bar{a} \cdot \bar{b} \end{array} \right.$

M

Circuit

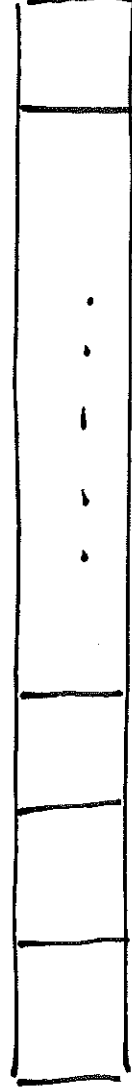


ARRAYS IN C++

14

AN ARRAY IS A CONTIGUOUS SET OF MEMORY LOCATIONS, ALL SHARING THE SAME DATA TYPE, ALL REFERS TO BY THE SAME IDENTIFIER.

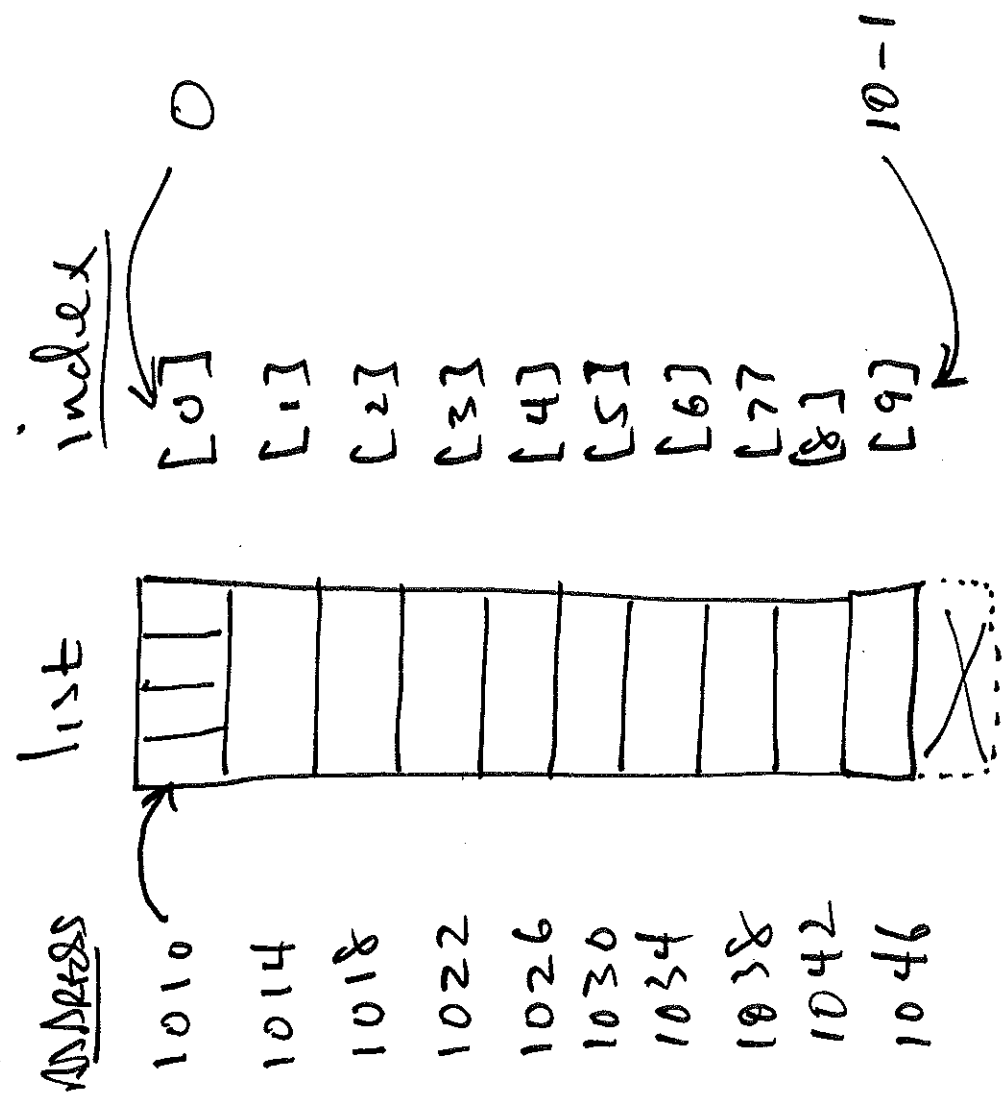
AN ARRAY OF INTS



DECLARATIONS :

EX int list[10];
double weight[20];
char word[30];

For Example: `int list[10];`



Programmer uses symbolic names:

`list[0], list[1], list[2], ..., list[9]`

NOTE: list[10] DOES NOT REFER TO
AN INIT IN THIS ARRAY.

Char Arrays are also called strings.

C++ string library: `<string>`

`#include <string>`

• Refs: 2-dim & 3-dim ARRAYS
IN C++ &.

```
int i, j;  
int list[10];
```

```
list[0] = 5;
```

```
cout >> list[1];
```

```
cout >> list[2];
```

```
cout << list[0] << endl;
```

```
list[5] = list[7] = 12;
```

```
i = 2;
```

```
j = 3;
```

```
list[i] = -4;
```

```
list[j] = -10;
```

```
list[i+j] = -50;
```

```
list[8] = list[i-i];
```

