

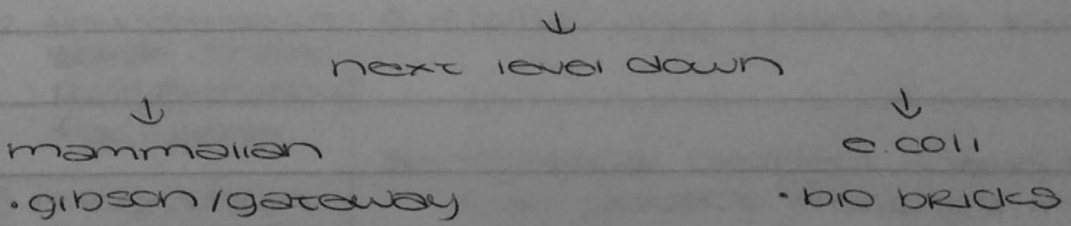
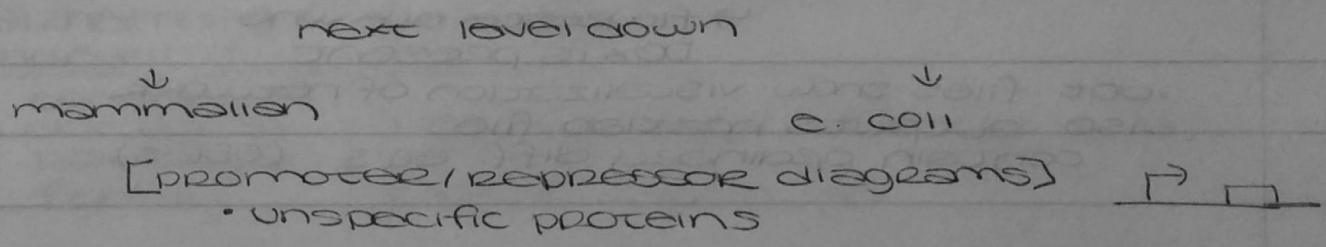
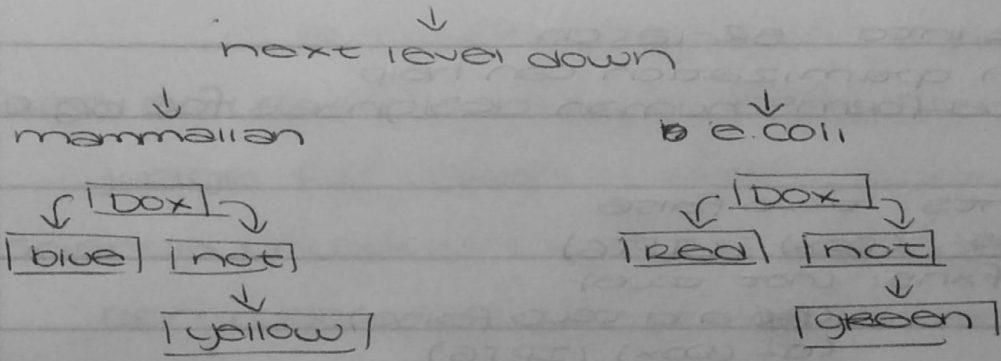
High Level Genetic Circuit Design: The Proto Biocompiler

the vision of the biocompiler

```

(def simple-sensor-actuator ()
  (let (x (test-sensor)))
    (debug (x))
    (debug-2 (not x)))
  
```

we make an actuator  
 assign a test sensor (ex. dox)  
 ← if it's present, do this  
 ← else do this



next level down  
 [glowing cells (done)]

motif based compilation

```

(primitive not (boolean) boolean)
:grn-motif ((P+high R- arg0 value T)))
  
```

translates to

classic example

(green (not (IPTG)))

$\boxed{\text{IPTG}} \rightarrow \boxed{\text{not}} \rightarrow \boxed{\text{Green}}$

~~W~~

~~W~~

\*NOTE look into SR-latch

design optimization can help

→ outperforms human designers for big circuits

Syntax

constants: true false

Sensors: (DOX) (IPTG)

basic fns: (not true)

(~~not~~ and true false)

(or (DOX) (IPTG))

(blue (DOX))

→ fluoresce blue when

DOX is present

•dot files show visualization of network

also outputs matlab files

contain ordinary diff. eq's (ODE's)