No. 1

Date

Project 1

Puzzle alarm clock

Clock that requires solving some simple puzzles before turning alarm off

Objective: Very easy to go back to sleep after stopping the alarm. Possible to go back to sleep without waking up fully. Making me perform some cognitive activity before turning the alarm off will probably wake me up.

Requirements

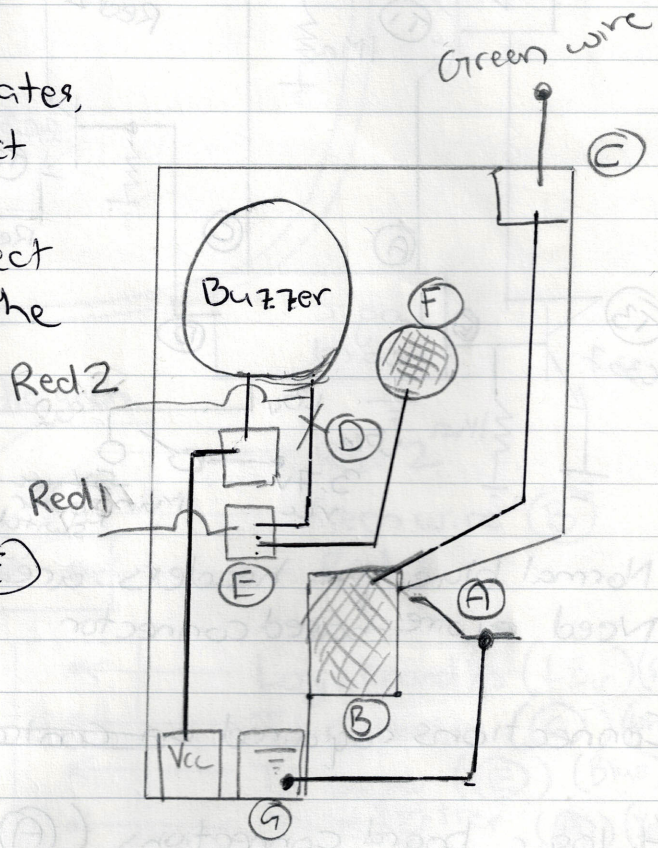
1. Basic (Need)
   a) Can set time & alarm time
   b) Cannot turn off alarm while it is ringing
   c) Need some puzzles-type interface to turn alarm off

2. Advanced (Want)
   a) Be able to choose alarm tone
   b) Alarm clock tries to fun away

Clock hand order (in order closest to face)
1) Alarm hand (silver)
2) hour hand
3) Minute hand
4) Second hand

(Bearing ~~points~~ burr points upward)

Circuitry notes:
1) When the alarm activates, Swith (A) ~~Go~~ makes contact with pad (B).
2) This makes pad (B) connect to ground, and makes the buzzter buzz



Plan:
1) Disconnect wire (D)
2) Add connection point to (E)
3) Add wire to (C)
4) Add wire to gnd pad (G)

Alarm is engaged when green wire is connected to ground

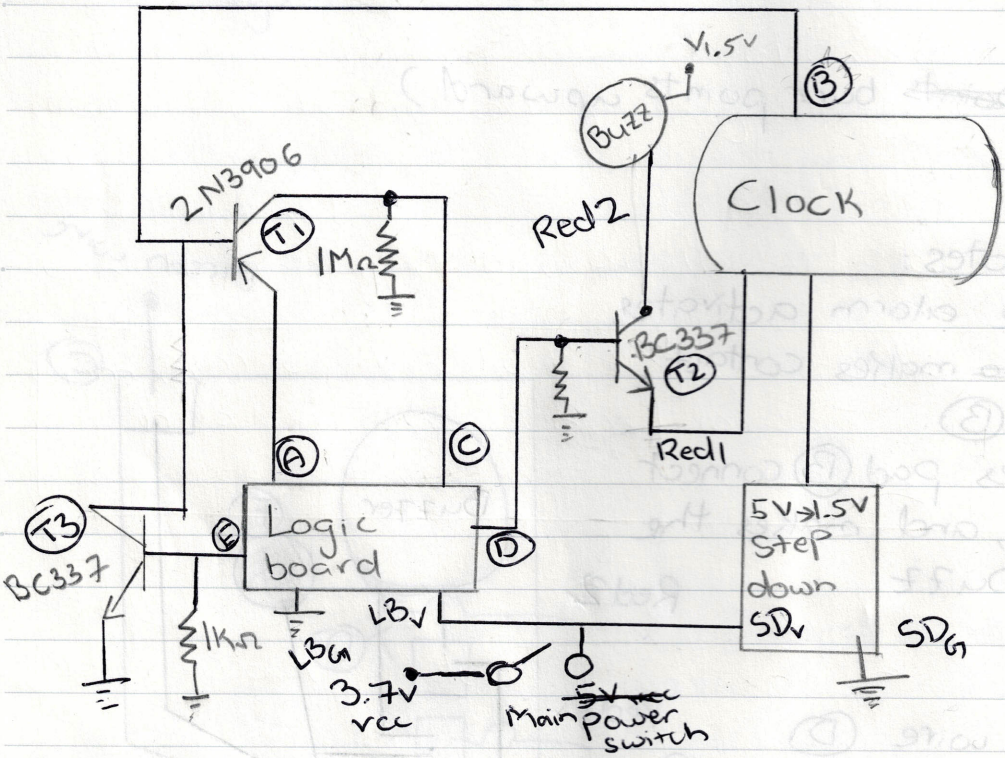
No. 4

Date

New circuitry

Logic circuit input => Green wire is grounded

Circuit action => Short red wires together



Normal blue pin headers are too big for this project

Need wire-based connector

Connections required ~~on control board + (CB1)~~

4 logic board connections (A, C, D, E) - 1 6pin connect

1 logic board ground + logic board power

2 Battery Power connector } - 2pin connector

2 Step down input - 2pin connector

2 Red wire from clock, - 1 3pin connector

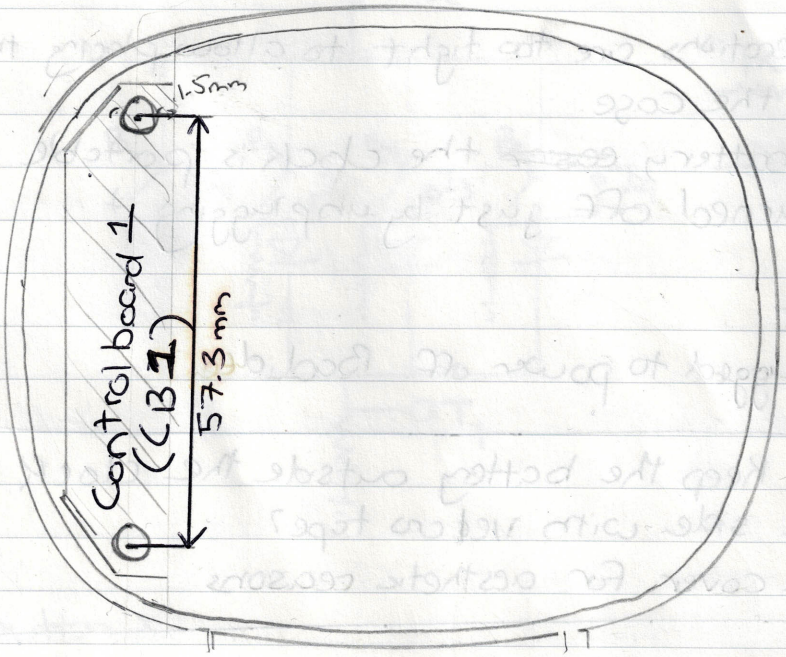
1 Green wire from Clock

No. 5

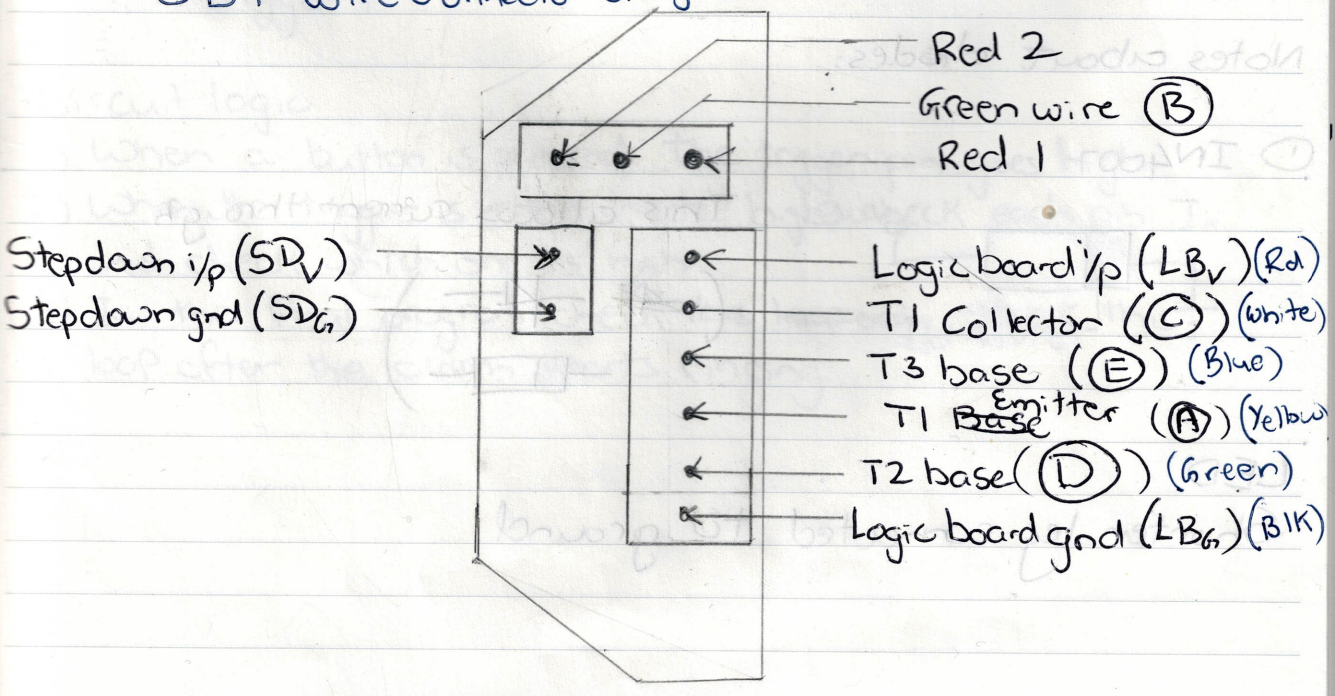
Date

Physical design

57.3mm bet^n holes



CB1 wire connector diagram



No. 6

Date

Power source considerations

1) Battery
Space considerations are too tight to allow placing the battery inside the case.
If there is a battery, ~~easier~~ the clock is portable and cannot be turned off just by unplugging it

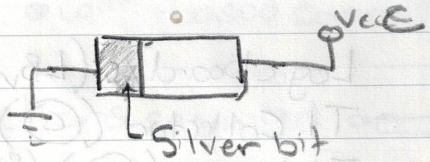
2) Wall socket
Can be unplugged to power off. Bad idea

Compromise: Keep the battery outside the clock casing
Attach to the side with velcro tape?
Also create a cover for aesthetic reasons

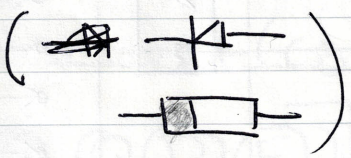
Control circuitry

Notes about diodes:

① IN4007



This allows current through

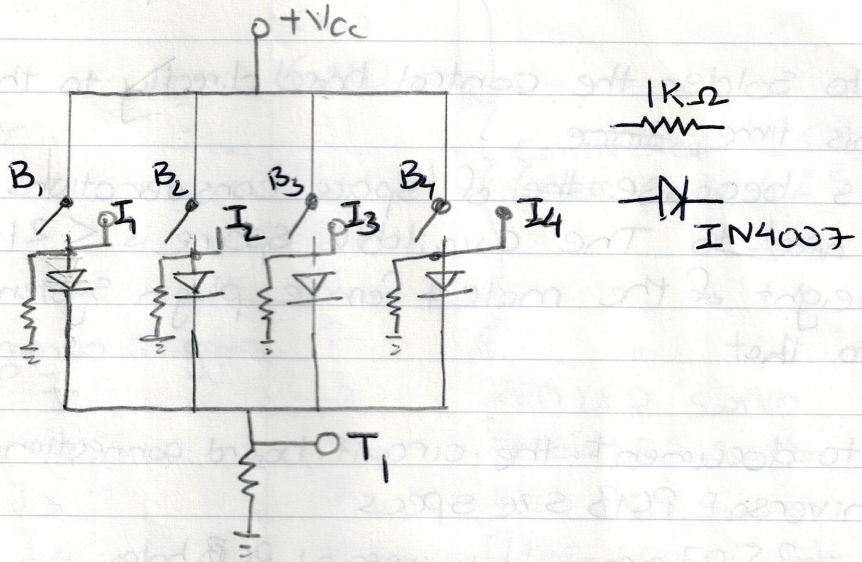


② LED
Shorter leg connected to ground

No. 7

Date

Button press circuit



B_x = Button x

~~B_x~~ ≡ I_x = Input x

T_1 = Trigger 1

Circuit logic

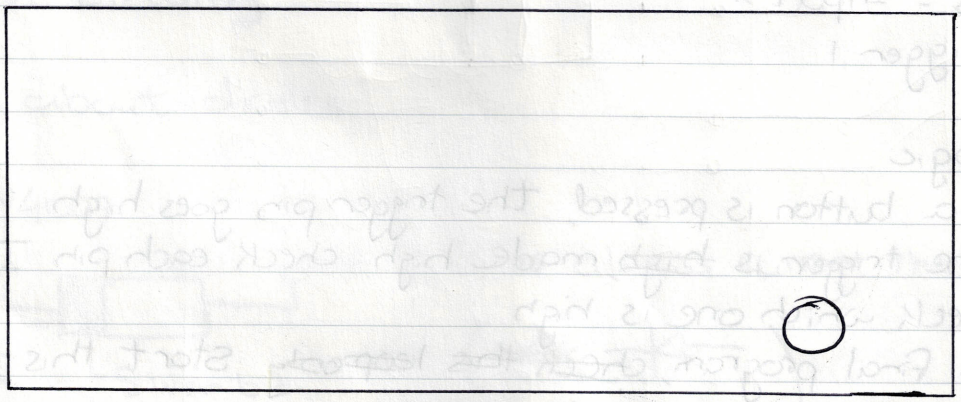
1) When a button is pressed, the trigger pin goes high
2) When the trigger is ~~high~~ made high, check each pin I_x and check which one is high
3) In the final program, ~~check this loop~~ start this loop after the alarm starts ringing.

Normally it is better to have a plug-in connection for the control board in case it needs to be replaced

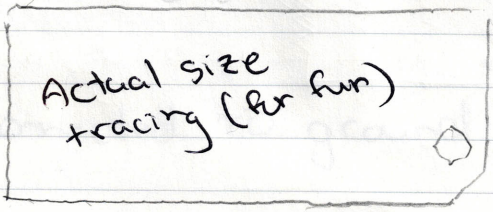
Need to solder the control board directly to the universal PCB this time, ~~since~~
This is because ~~the~~ of space considerations.
~~Might need to~~ The available space is < 21 mm, and the height of the male + female plug-in system is very close to that

Need to document the circuit board connections carefully
Universal PCB size specs
Width = ~25.07 mm = 11 universal PCB holes
Length = ~63 mm = 26 universal PCB holes

CB2



Actual size tracing (for fun)



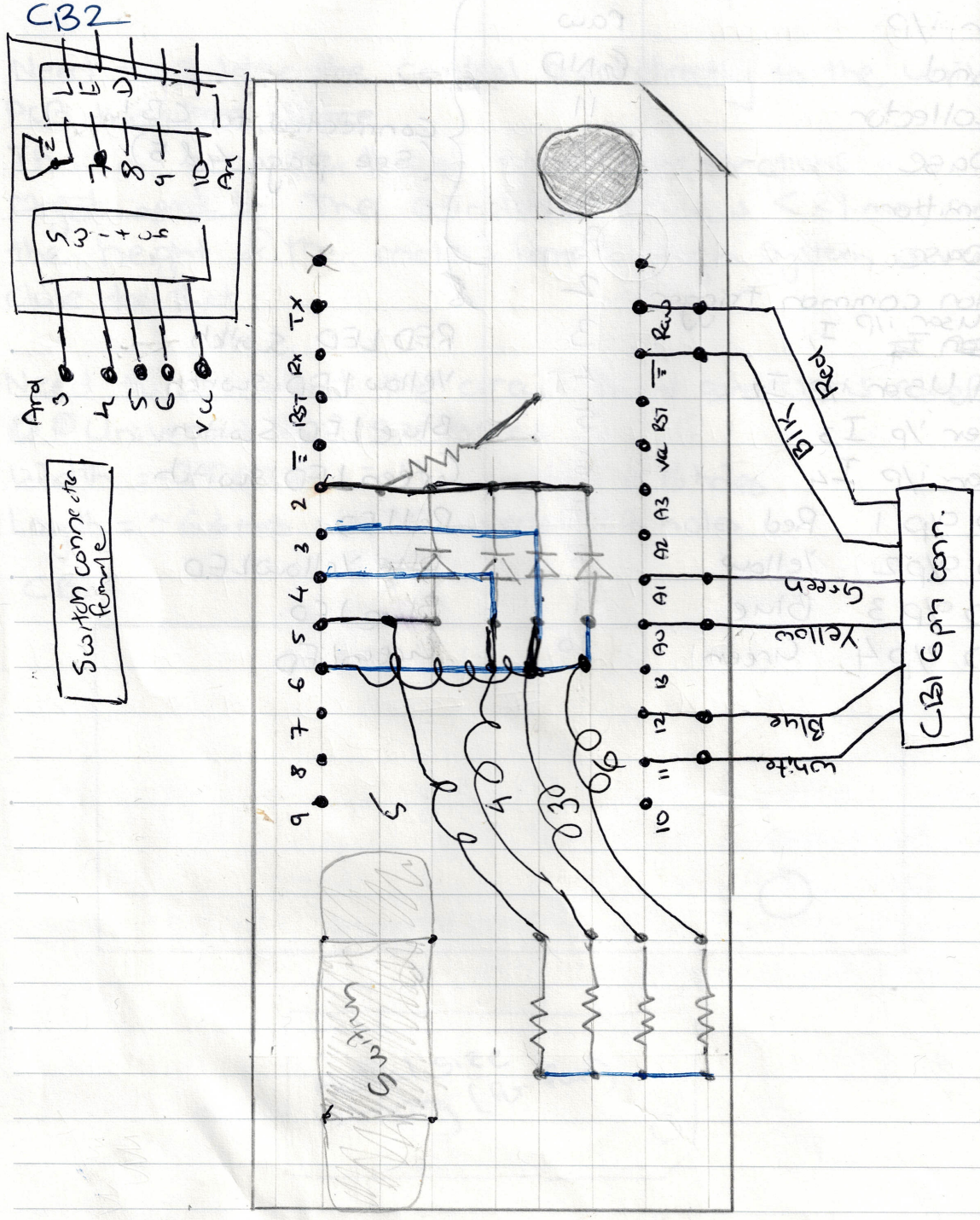
List of required connections and corresponding pins

	Pin/no	Notes
1) Power i/p	raw	
2) Ground	GND	
3) T_1 collector	11	Connected to CB1 (See page 4 & 5)
4) T_3 base	12	
5) T_1 emitter	A_0	
6) T_2 base	A_1	
7) Button common trigger	2	
8) ~~Button~~ User i/p I_1	3	RED LED switch
9) ~~Button~~ User i/p I_2	4	Yellow LED switch
10) User i/p I_3	5	Blue LED switch
11) User i/p I_4	6	Green LED switch
12) LED o/p 1 Red	7	Red LED
13) LED o/p 2 Yellow	8	~~Blue~~ Yellow LED
14) LED o/p 3 Blue	9	Blue LED
15) LED o/p 4 Green	10	Green LED

No. 10

Date

Control board layout



No. 11

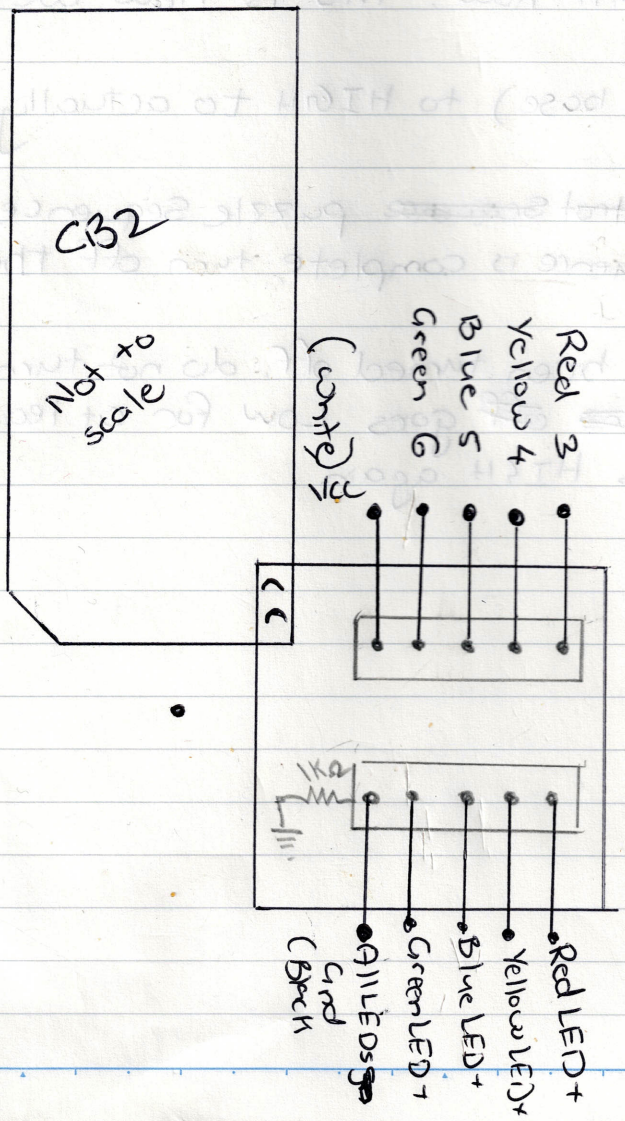
Date

LED and button selection

5mm diameter LEDs

LED, switch wiring notes

(LED shorter leg to ground)



No. 12

Date

Working

Initially (A) (T_1 emitter) ~~and~~ is set HIGH ~~& (C) (T_1 collector) is set LOW.~~

(B) (Alarm on/off indicator) is ~~initially~~ HIGH when the alarm is off so (C) (T_1 collector) should read LOW.

When the alarm is triggered, (B) goes LOW. This forward-biases the emitter-base junction of ~~the~~ T_1 and it turns on (C) should read HIGH now. This is how we know the alarm is on.

Now set (D) ~~to~~ (T_2 base) to HIGH to actually turn on the alarm.

Now activate the ~~control sequence~~ puzzle sequence.

When the puzzle sequence is complete, turn off the alarm

Once the alarm has been turned off, do not turn it back on until ~~(B)~~ (C) ~~turns off~~ goes LOW for at least 5 seconds before it goes HIGH again.