

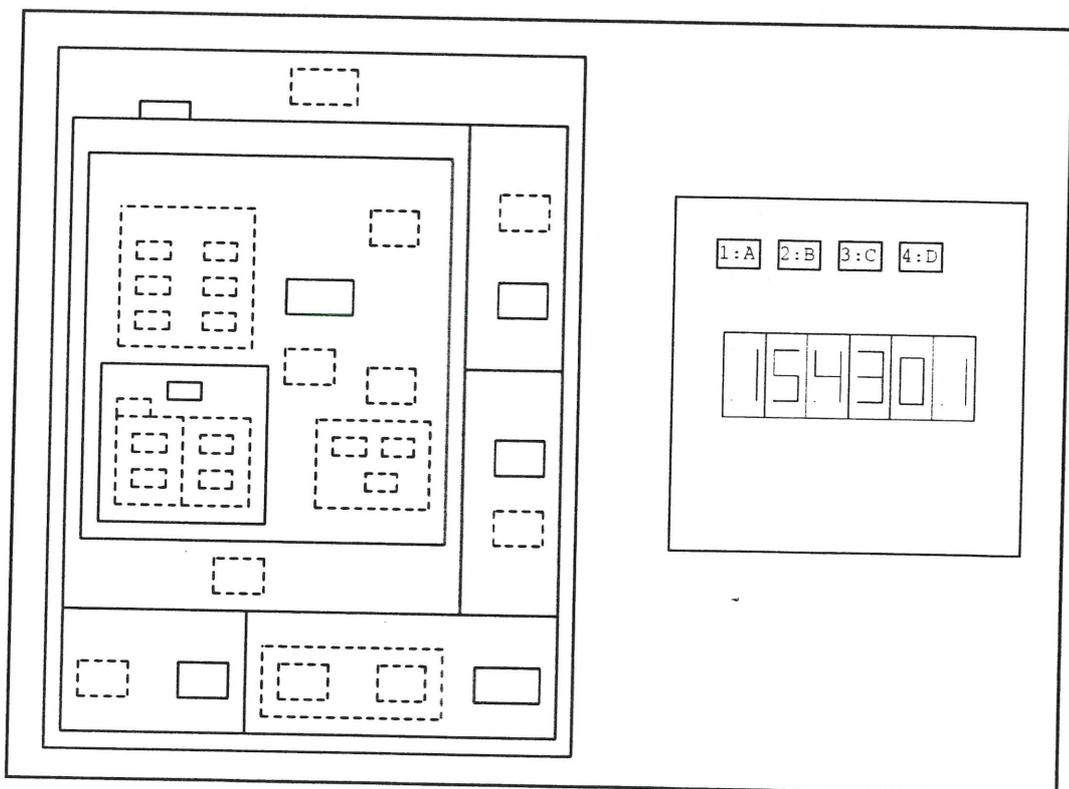
Outline specification for the digital watch, based on Harel's statechart

```

agent power() {
    derivate power_s = battery_charge(time)    // three-valued 2,1,0
}

agent watch() {
    derivate LIVE = (power_s >= 1)
    oracle power_s
    agent main() {          // as specified below
        .....
    }
    agent light() {
        .....
    }
    agent alarm_st() {
        derivate LIVE=LIVEmain
        oracle LIVEmain, displays_s, alarm_s
        state alarm_s = D, set_time = 00.00
        handle alarm_s // 1:Disab, 2:Enab
        protocol displays_s == A & alarm_st == D & !d -> alarm_st = E
                   displays_s == A & alarm_st == E & !d -> alarm_st = D
    }
    agent chime_st() {
        .....
    }
    agent clock() {
        .....
    }
    agent stopwatch() {
        .....
    }
    .....
}

```



```

agent main() {
  derivate LIVE=LIVEwatch
  oracle LIVEwatch, main_s, alarm_s
  state main_s = D // an integer - 1:Displays, 2:Beep | | displays
  handle main_s
  protocol
    (main_s==D) & (time==set_time) & alarm_s==E -> main_s=B
  agent displays() {
    derivate LIVE=LIVEmain
    oracle LIVEmain
    state displays_s = T // 1:Time, 2:Update, 3:Date, etc
    handle displays_s, update_s, upalarm_s
    protocol displays_s==T & !c -> update_s=1,
      displays_s==T & !d -> displays_s=D,
      displays_s==A & !c -> upalarm_s=1,
      .....
      displays_s!=S & displays_s!=T & 2-min -> displays_s=T
  }
  agent disp_date() {
    derivate LIVE = LIVEdisplays & displays_s==D,
      "watch_display = date as of clock()"
    oracle LIVEdisplays, displays_s
  }
  agent disp_time() { .....
  }
  agent disp_upalarm() {
    derivate LIVE = LIVEdisplays & upalarm_s>0,
      displays_s = (upalarm_s==0%4)?A:UA
      "watch_display = time as of alarm setting with
        right digit highlighted"
    oracle LIVEdisplays, upalarm_s
    state upalarm_s = M
      // 1:Min, 2:TenMin, 3:Hr, 4=0(mod 4): Alarm
    handle upalarm_s, set_time
    protocol !b or 2-min -> upalarm_s=A,
      !c -> upalarm++,
      "event -> update set_time so as to
        increment highlighted digit in set_time"
  }
  .....
}
agent beep() {
  derivate LIVE=LIVEmain & main_s==B
  state main_s
  protocol beep_stop -> main_s
}
} // end main()

```