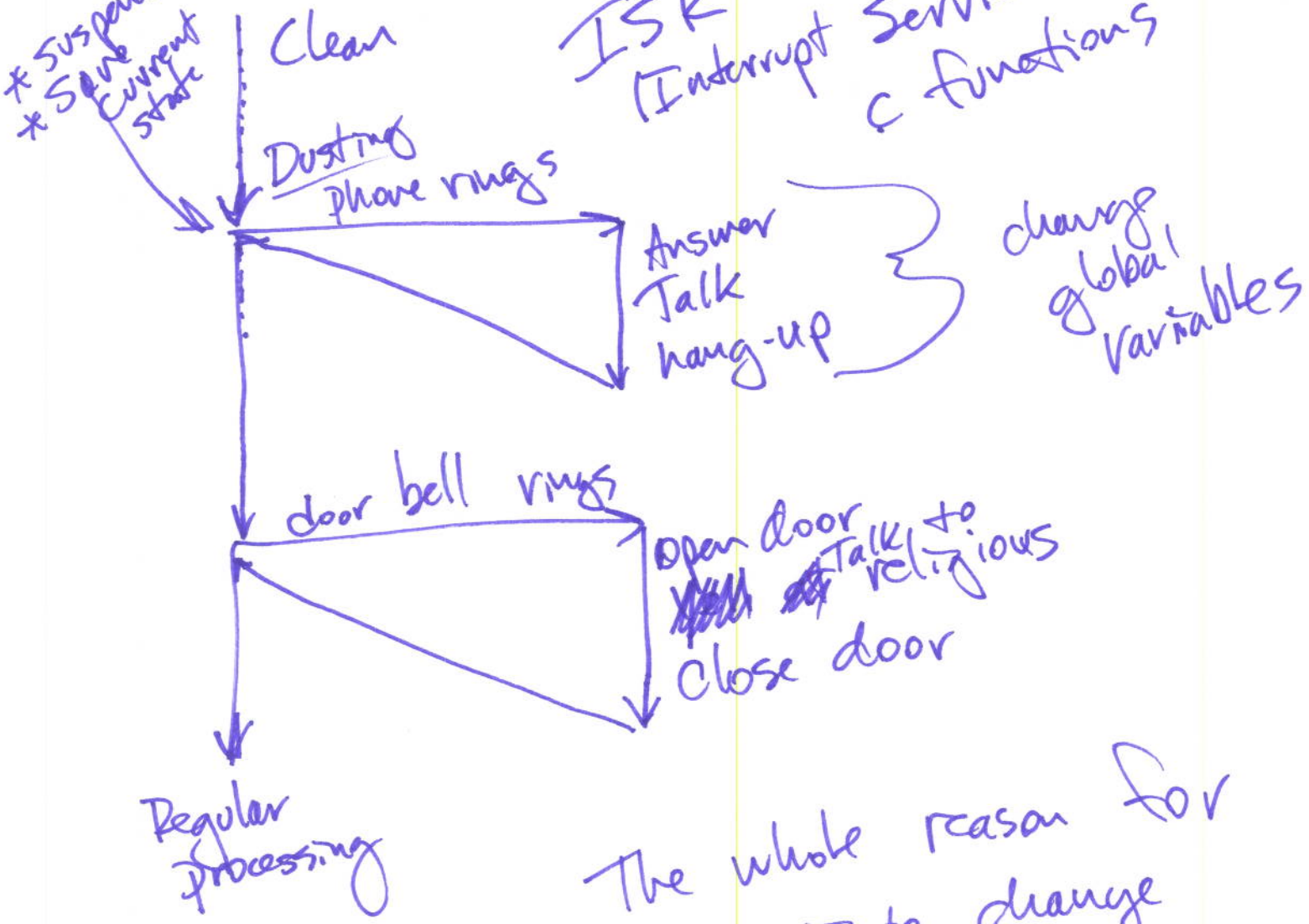
Embedded Systems Lecture 20

~~[illegible]~~

Interrupt Concepts

ISR (Interrupt Service Routines)
C functions



The whole reason for ISR is to change global variables
i.e. State Variable

Embedded Systems

Lecture 20

Arduature → functions & ISR are needed (2)

---

theft_possibility_ISR (void)

state = alarm_lock_up

start_theft_timer

disable button interrupts

---

return_button_ISR (void)

if state = money_inserted

then state = change_return

Coin_inserted_ISR (void)

if not a valid coin, return coin, return

state = money_inserted

Save coin value in coin_inserted (value)

---

theft_timer_ISR (void)

state = idle

---

button_selection_ISR (Void)

if state = money_inserted {

state = Validate

Save buttonpress in button_value (variable)

```
Sensor_of_drink_ISR (Void)
    state = idle
```

So now we have addressed ISR - Next?

```
while (1) {
    switch (state)
        idle:  idle_state_function
        money_inserted:  money_inserted_function
        validate:
        dispense_and_change:
        change_return:
        alarm_lock_up:
    end switch
}
```

```
alarm_lockup_function
    lock bottle drawer, dispenser, coins
    sound alarm
    activate taser
```