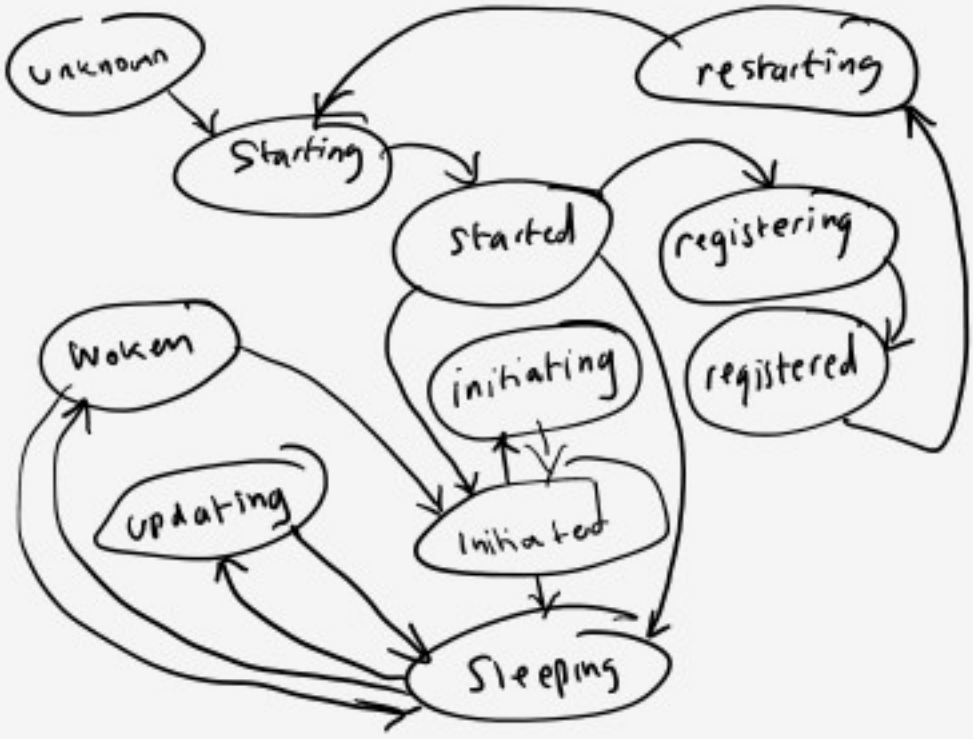
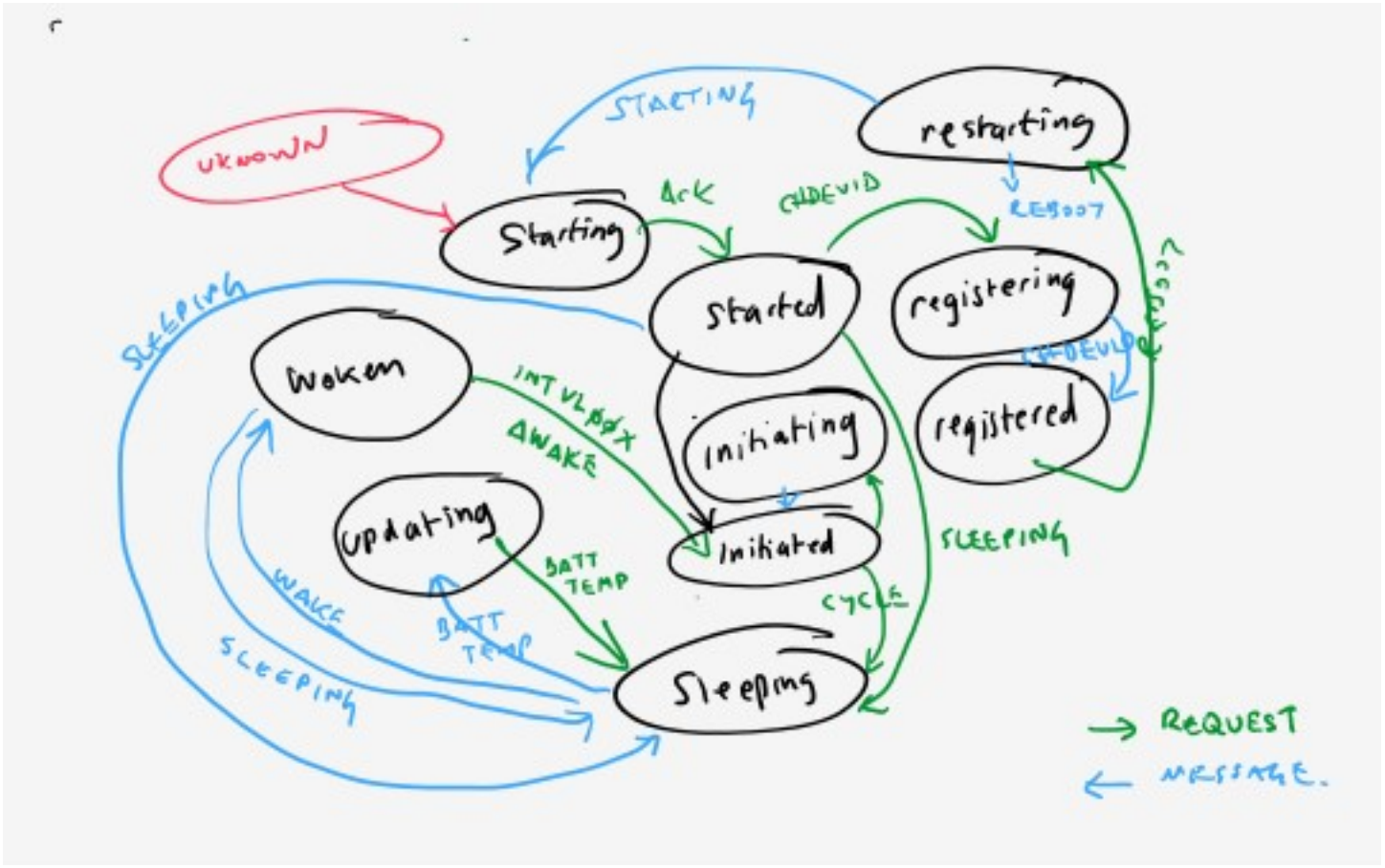
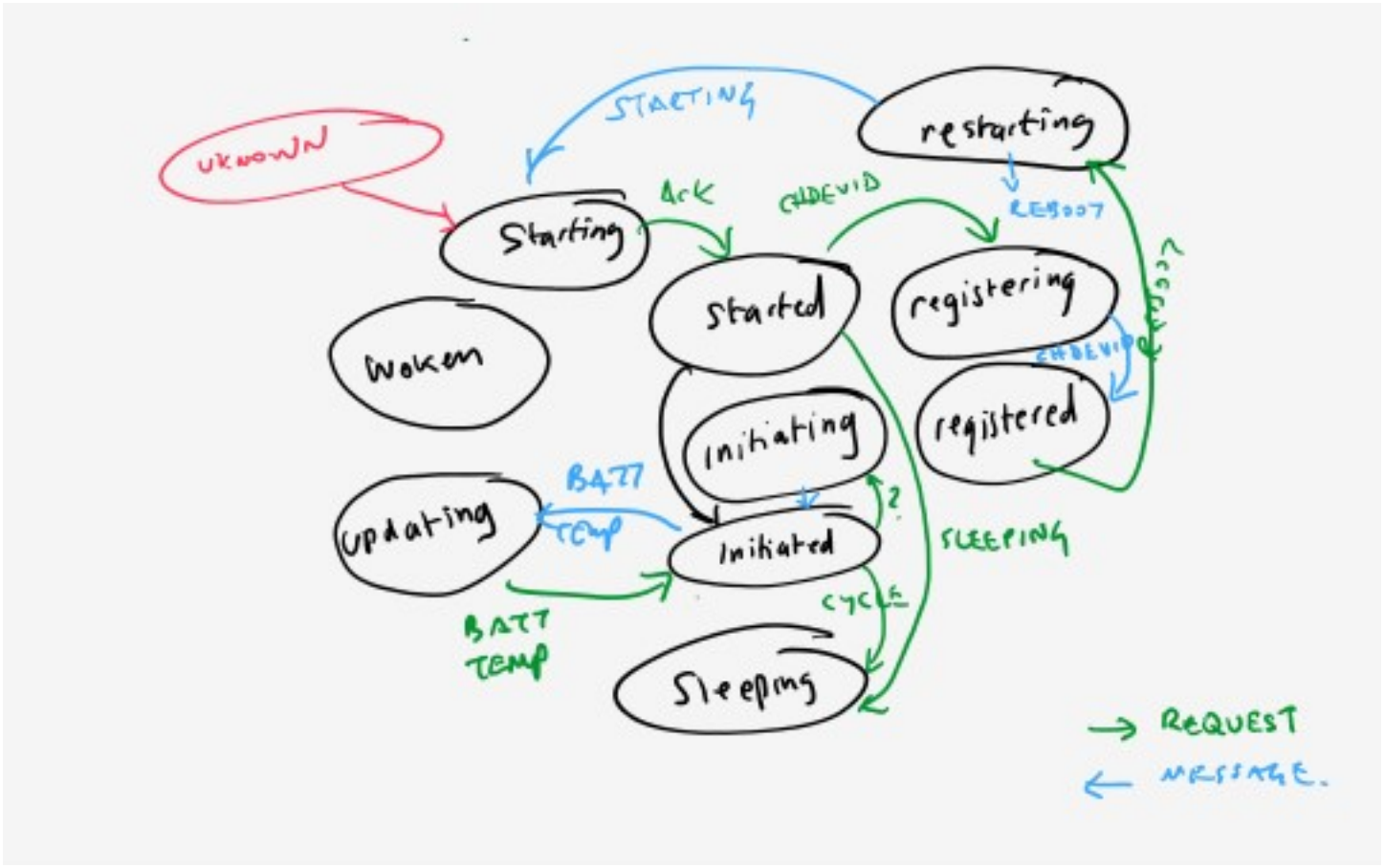
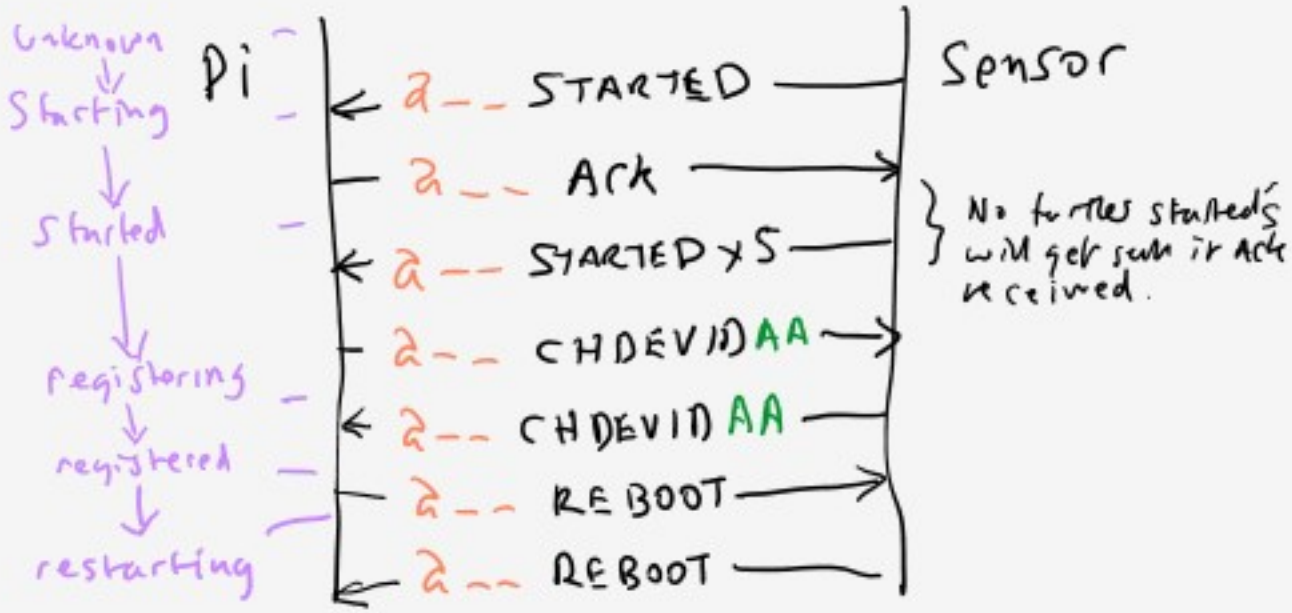
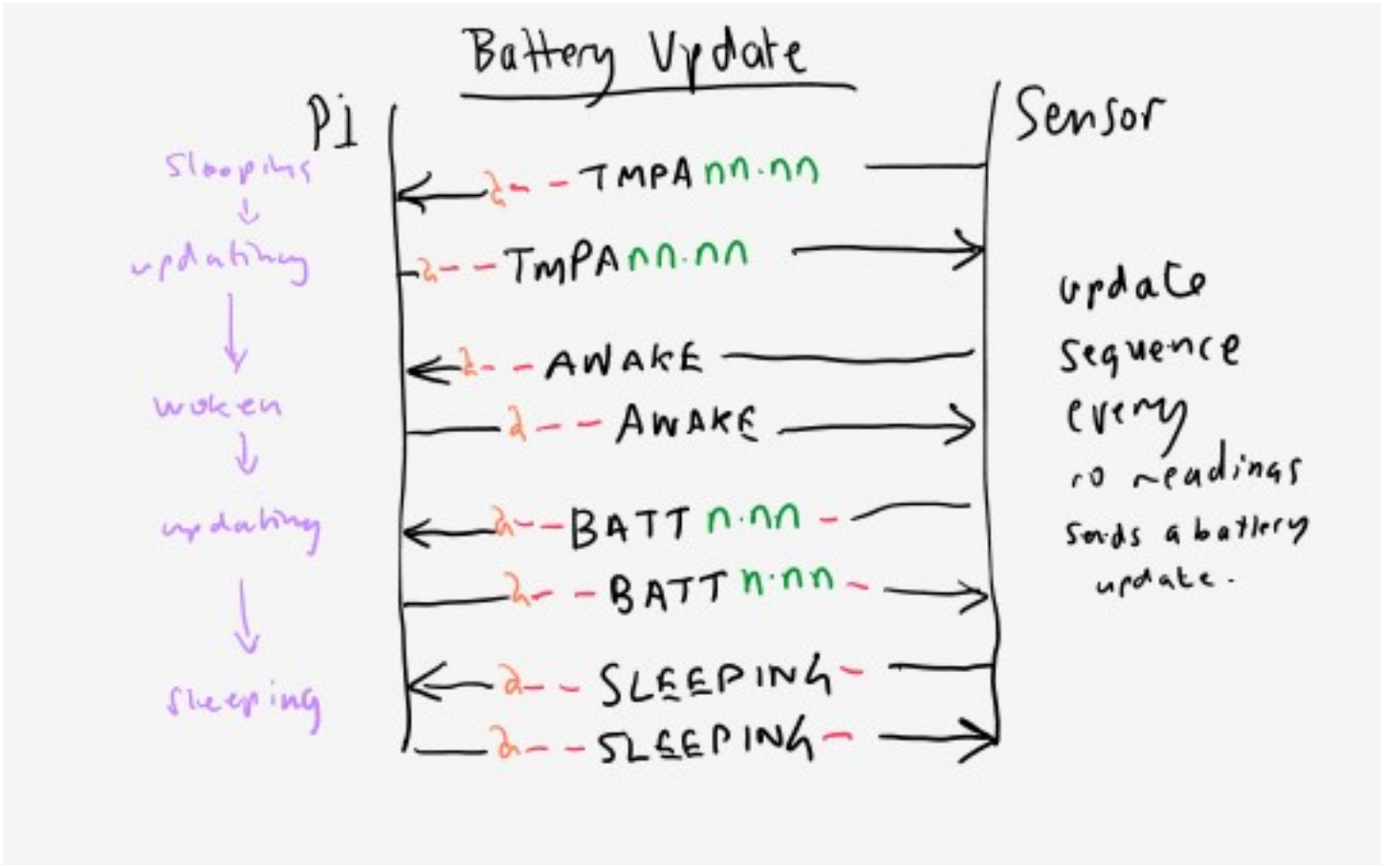
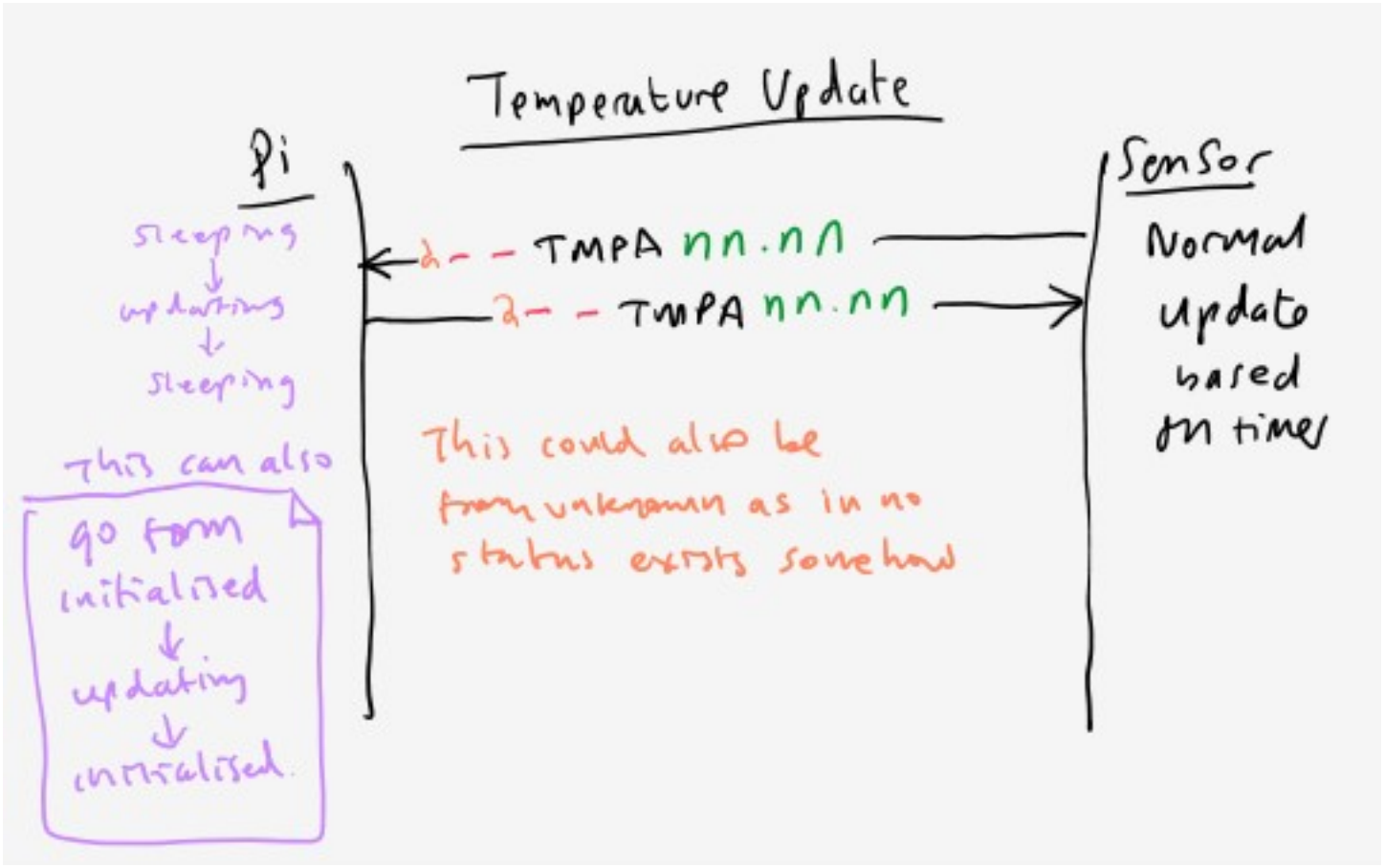

Unknown
restarting
Starting
started
registering
Woken
initiating
registered
updating
Initiated
Sleeping

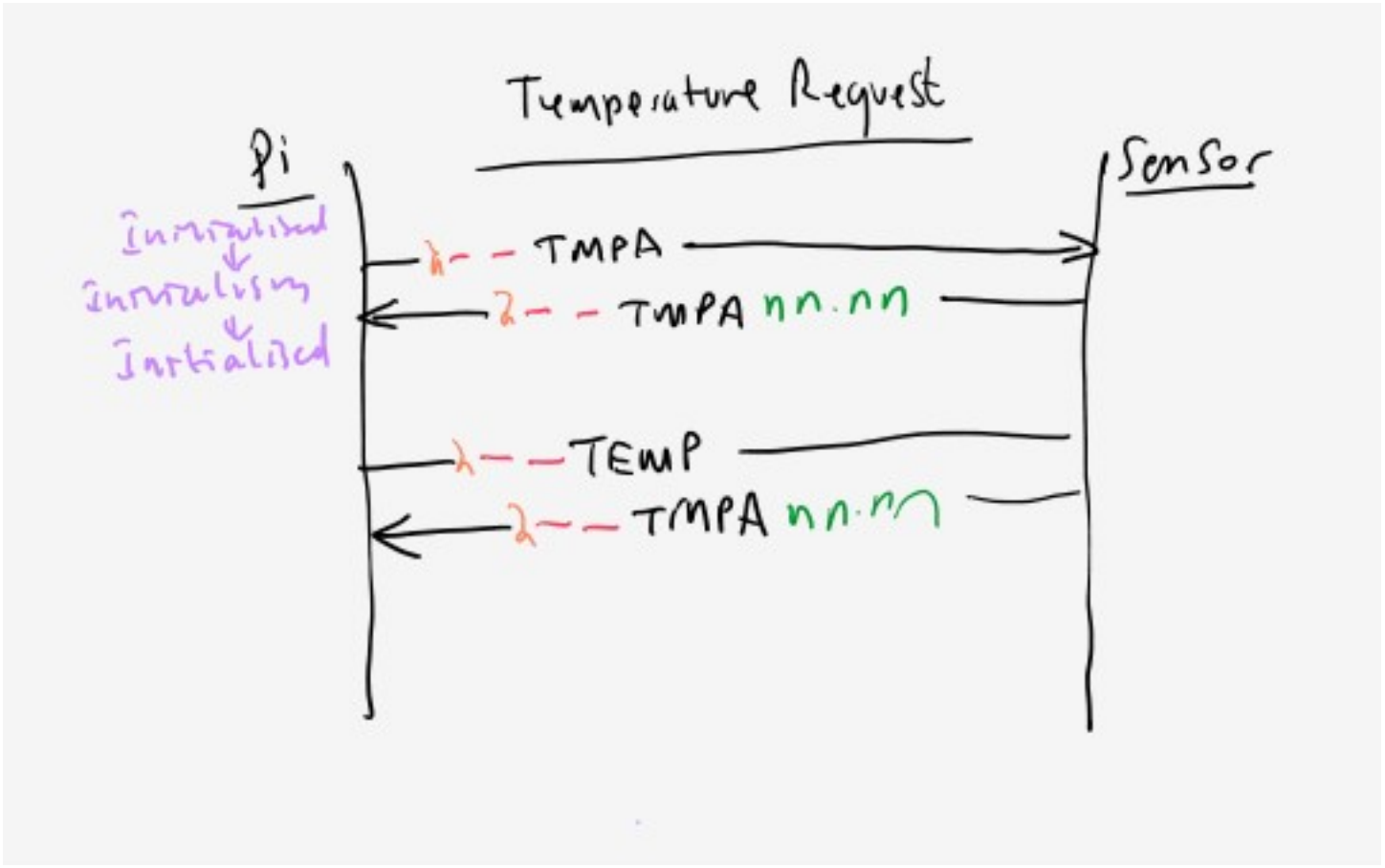

UNKNOWN
STARTING
restarting
Starting
Ack
CMDEVID
REBOOT
started
registering
registered
Woken
INTVLPDX
AWAKE
initiating
Initiated
Updating
BATT TEMP
WAKE
SLEEPING
Sleeping
CYCLE
→ REQUEST
← MESSAGE.

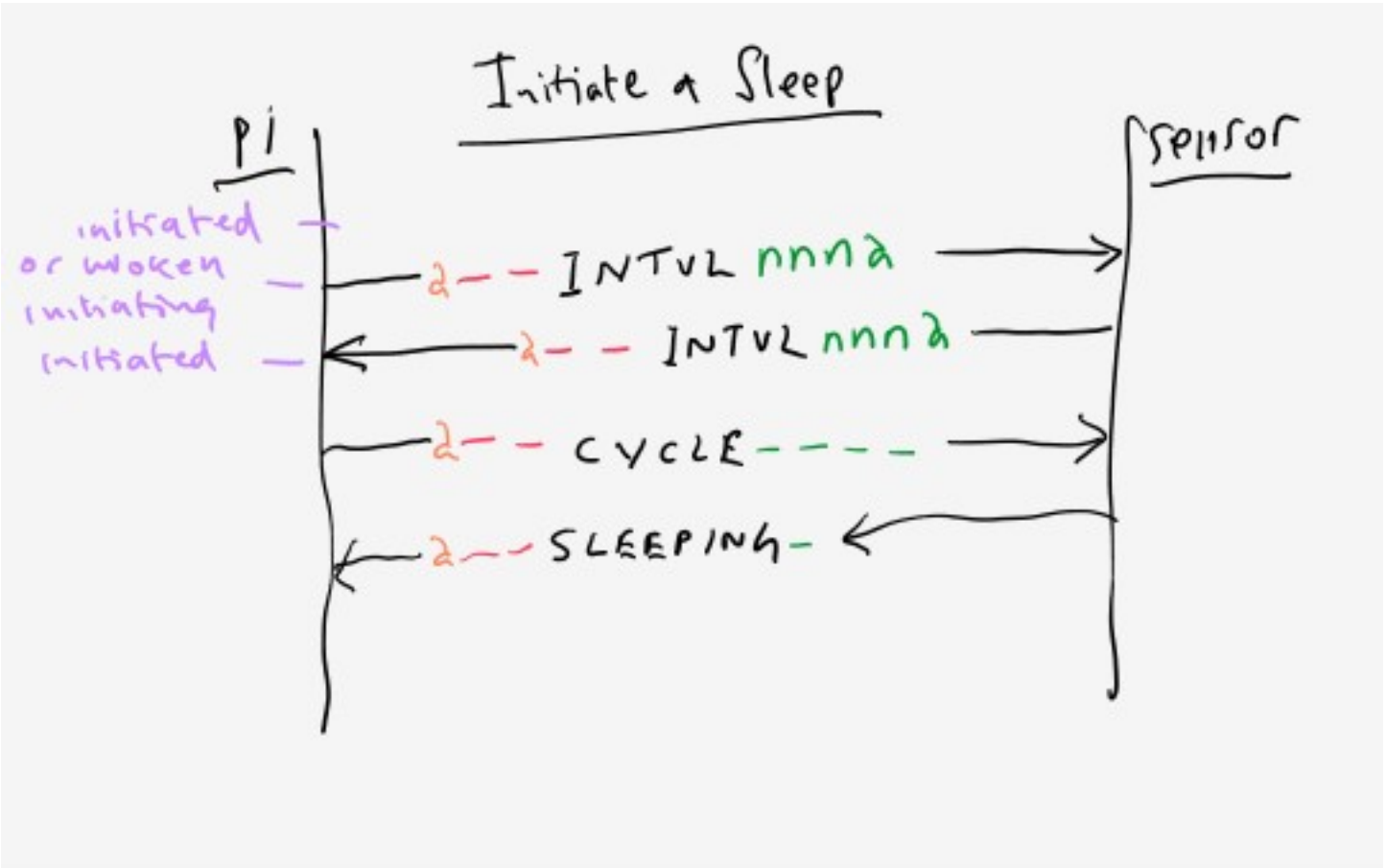

UKNOWN
STARTING
restarting
Starting
Ack
CHDEVID
REBOOT
started
registering
Woken
initiating
registered
BATT
TEMP
updating
Initiated
SLEEPING
CYCLE
BATT
TEMP
Sleeping
→ REQUEST
← MESSAGE.

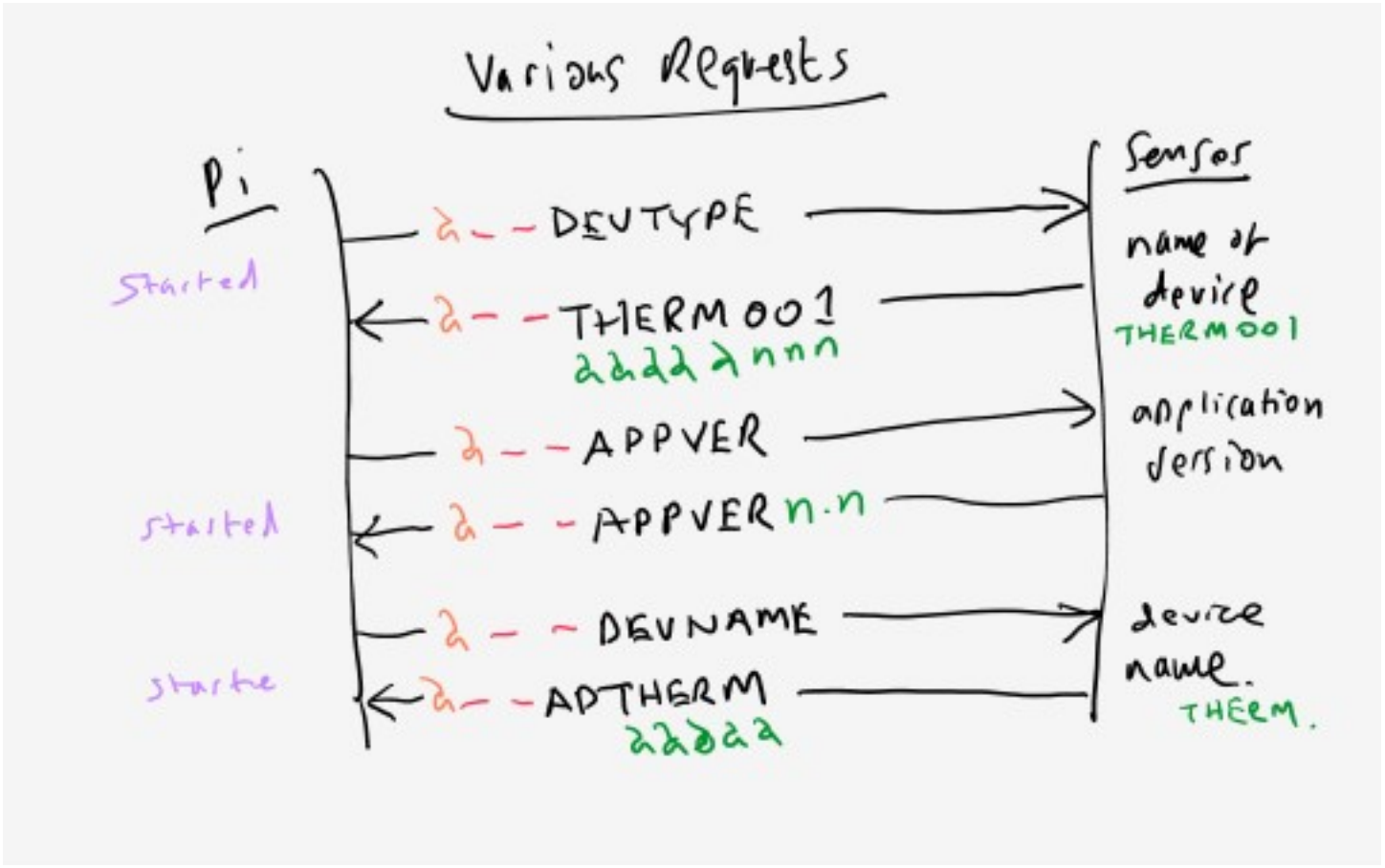

Unknown
Starting
started
registering
registered
restarting
Pi
Sensor
STARTED
ACK
STARTED x 5
CHDEVID AA
CHDEVID AA
REBOOT
REBOOT
No further started's will get sent if ACK received.

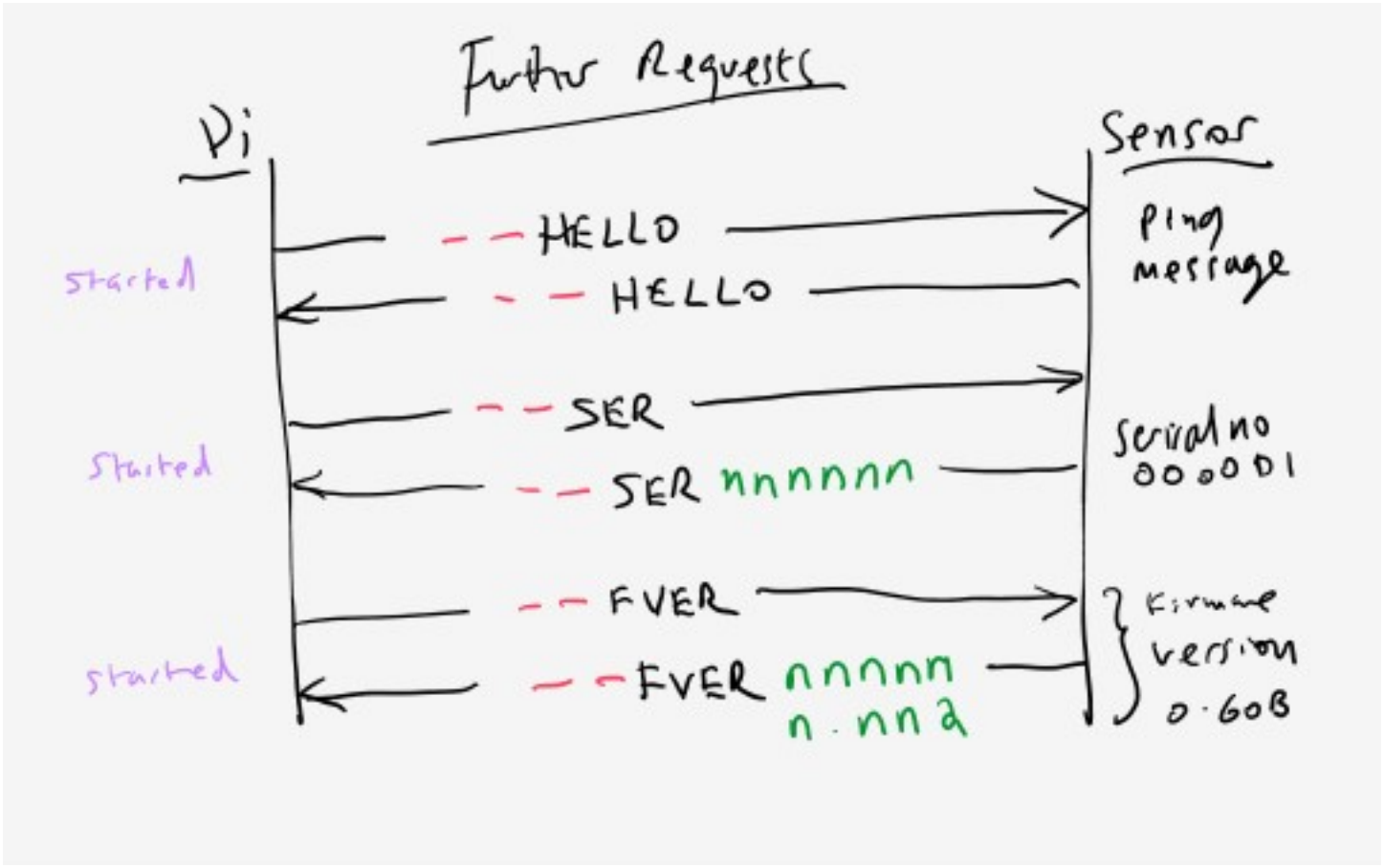

Battery Update
Pi
Sensor
sleeping
↓
updating
↓
woken
↓
updating
↓
sleeping
TMPA nn.nn
TmPA nn.nn
AWAKE
AWAKE
BATT n.nn
BATT n.nn
SLEEPING
SLEEPING
update sequence every ~0 readings sends a battery update.

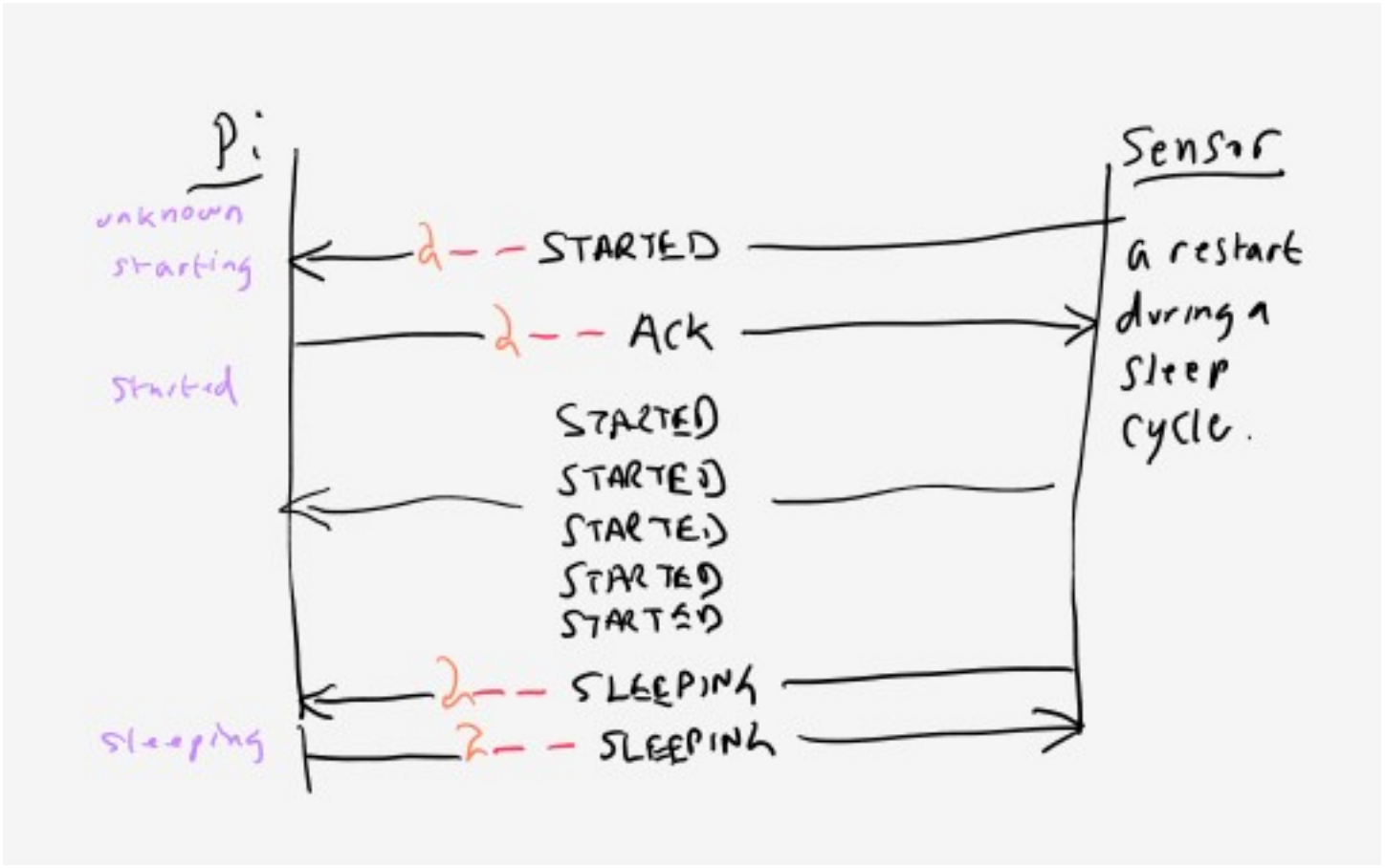

Temperature Update
Pi
Sensor
Sleeping
updating
Sleeping
2- - TMPA nn.nn
2- - TMPA nn.nn
Normal
Update
based
on time
This could also be
from unknown as in no
status exists somehow
This can also
go form
initialised
updating
initialised.

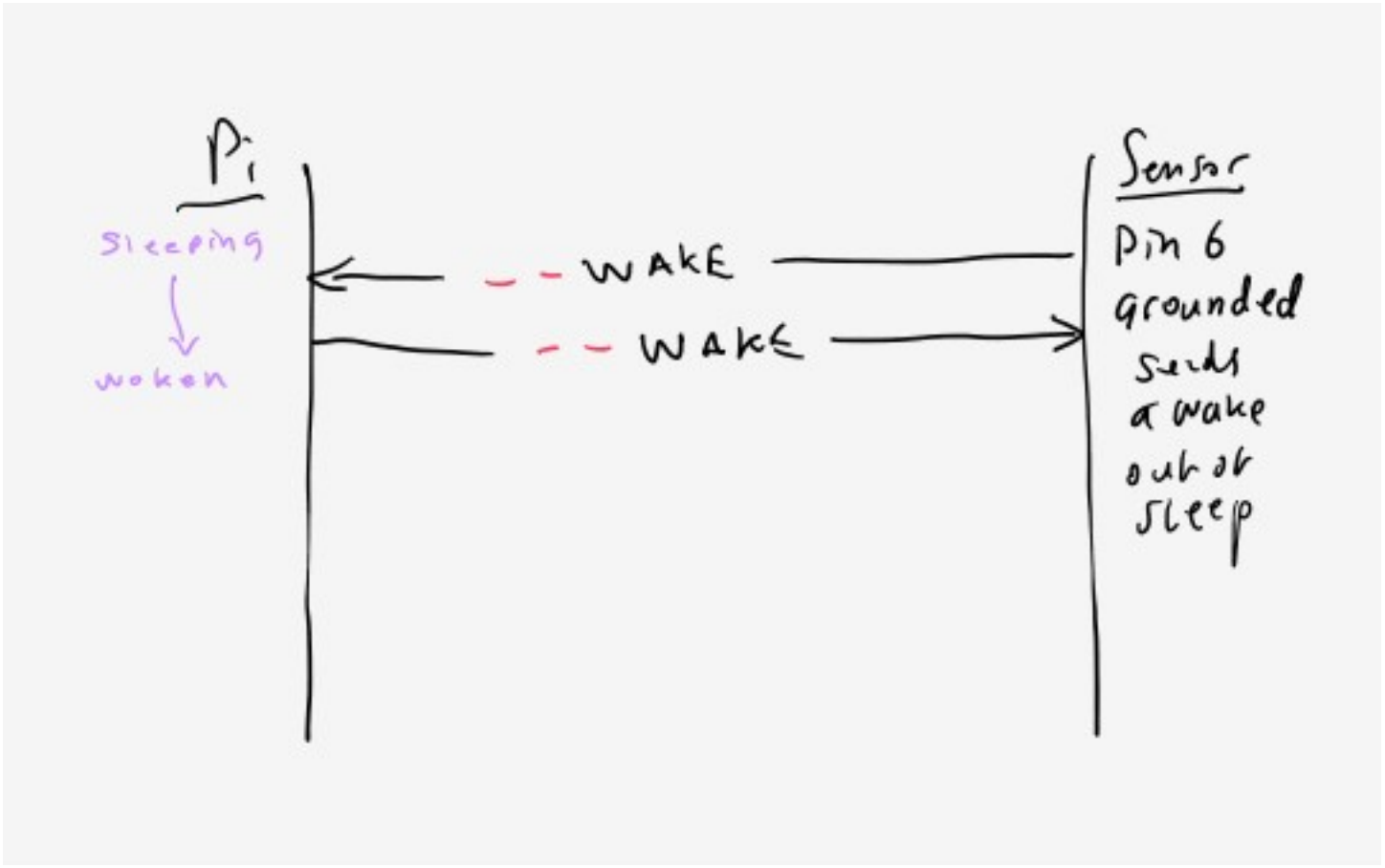

Temperature Request
Pi
Sensor
Uninitialised
Initialising
Initialised
1 - - TMPA
2 - - TMPA nn.nn
1 - - TEMP
2 - - TMPA nn.nn

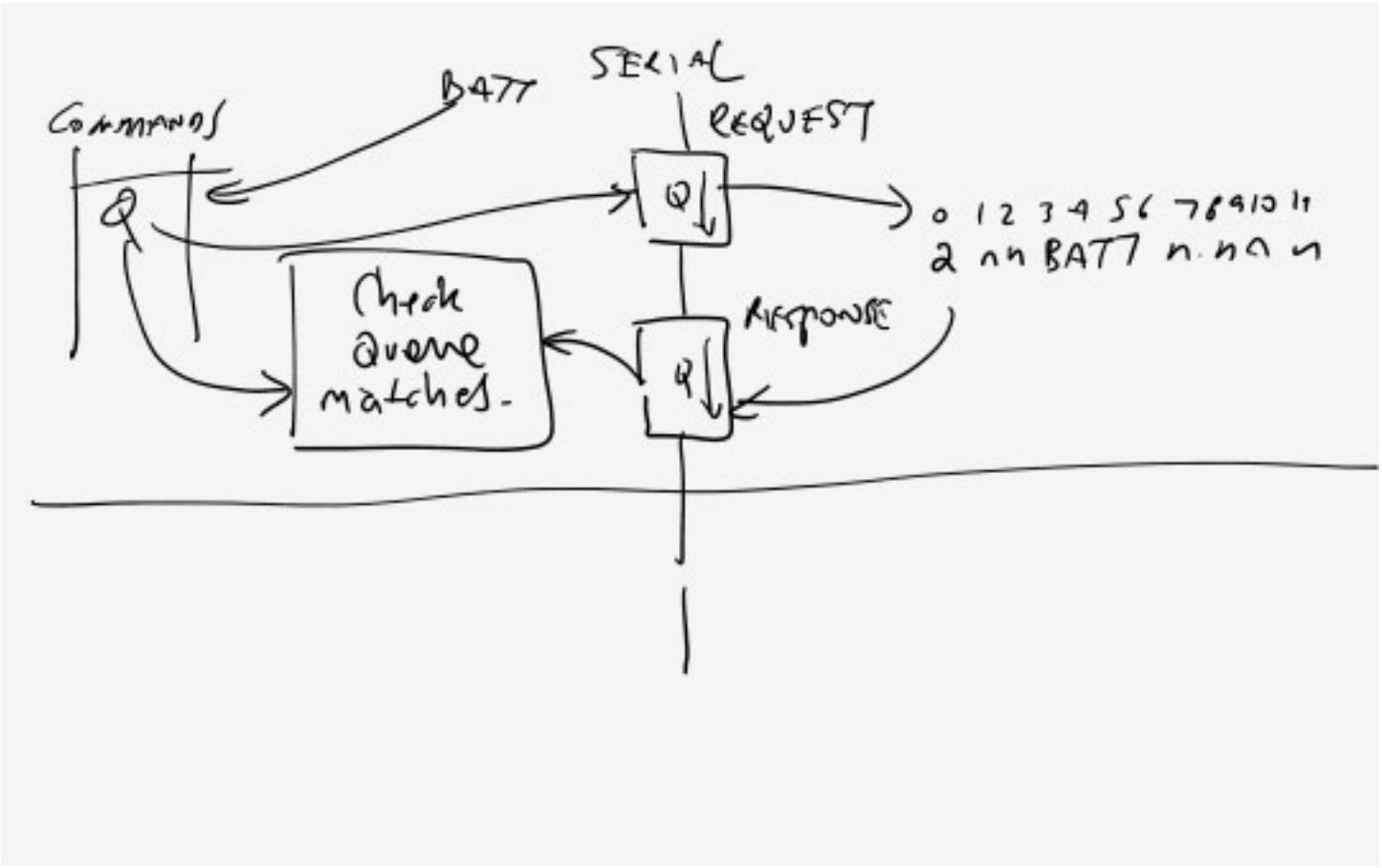

Initiate a Sleep
Pi
Sensor
initiated
or woken
initiating
initiated
2-- INTVL nnn a
2-- INTVL nnn a
2-- CYCLE----
2-- SLEEPING-


Various Requests
Pi
Sensor
started
DEVTYPE
name of device
THERM001
THERM001
aaaaannn
APPVER
application version
started
APPVERn.n
DEVNAME
device name.
THERM.
starte
APTHERM
aaaaa


Further Requests
Pi
Sensor
started
HELLO
HELLO
Ping message
started
SER
SER nnnnnn
Serial no 00001
started
FVER
FVER nnnnn n.nn2
Firmware version 0.608


Pi
Sensor
unknown
starting
STARTED
a restart
during a
sleep
cycle.
ACK
started
STARTED
STARTED
STARTED
STARTED
STARTED
SLEEPING
sleeping
SLEEPING


Pi
Sleeping
woken
WAKE
WAKE
Sensor
Pin 6
grounded
sends
a wake
out of
sleep


SERIAL
BATT
COMMANDS
REQUEST
Q
Q
0 1 2 3 4 5 6 7 8 9 10 11
2 nn BATT n.nn n
Check
queue
matches.
RESPONSE
Q

THERMISTOR

The Thermistor firmware is capable of operating as a permanently powered "polled" device or a cyclic "sleeping" device more suited to battery applications.

The suported commands for the temp firmware are:

Generic commands

- APVER - LLAP version
- DEVTYPE - Device type
- DEVNAME - Device name
- HELLO - Hello (PING)
- SER - Serial number
- FVER - Firmware version
- CHDEVID - Change device ID
- PANID - Change PANID
- REBOOT - Restart the device
- INTVLxxxx - Sets the interval for reading the temperature - 9935 would be 993 seconds - T=ms, S=secs, M=mins, H=hours, D=Days - Once the command is sent the device will begin to send readings at the frequency requested. Whilst you are testing it's useful not to yet issue CYCLE, once asleep it's not an easy task to wake it.
- CYCLE - Activate cyclic sleeping, the device will wake every INTVL - Once this command has been issued the device will remain asleep for the majority of the time. Whilst asleep everything is turned off, that includes the radio.
- WAKE

Device specific commands

- TMPA - Reply with the temperature
- TEMP - Reply with the temperature
- BVALxxxxx sets the B coefficient for the simplified Steinhart equation (default aXXBVAL3977-)
- RN0Mxxxxx Sets the Thermistor resistance at 25 degrees c (default aXXRN0M10000)
- SRESxxxxx Sets the series resister value (default aXXSRES10000)
- IRxxxxxxx Sets the Analog input resistance (default aXXIR185000-) you should not normally change this.

Battery voltage

In cyclic sleeping mode, the battery voltage is broadcast every 10 readings. The voltage is read at the point of trasmission so is always lower than the battery "at rest".

DALLAS

The Dallas firmware is capable of operating as a permanently powered "polled" device with 1 Dallas DS18B20 sensors connected. Looking at the datasheet for the DS18B20 and the CC1110 a voltage of between 3v and 3.6v should suffice. A nominal voltage of 3.3v would be best.

The supported commands for the temp firmware are:

Generic commands

- APVER - LLAP version
- DEVTYPE - Device type
- DEVNAME - Device name
- HELLO - Hello (PING)
- SER - Serial number
- FVER - Firmware version
- CHDEVID - Change device ID
- PANID - Change PANID
- REBOOT - Restart the device

Device specific commands

- TEMP - Reply with the temperature an channel A
- TMPA - Reply with the temperature on channel A

Please note:

Dallas temperature sensors do not lend themselves to low power broadcasts so cyclic sleep is not implemented on the Dallas firmware

However like most device the Dallas does support the polled Sleep command (SLEEP999P) deatials can be found here

Trouble shooting

If the reading comes back as 127 degrees then the XRF is talking to the sensor but the actual sensor has failed to do a temp conversion, this can be because the supply voltage is not high enough

LIGHT DEPENDENT RESISTOR

The LDR firmware is capable of operating as a permanently powered "polled" device or a cyclic "sleeping" device more suited to battery applications. Readings are taken from a Light Dependent Resistor and vary from 0 (no light) to 99 (full bright light)

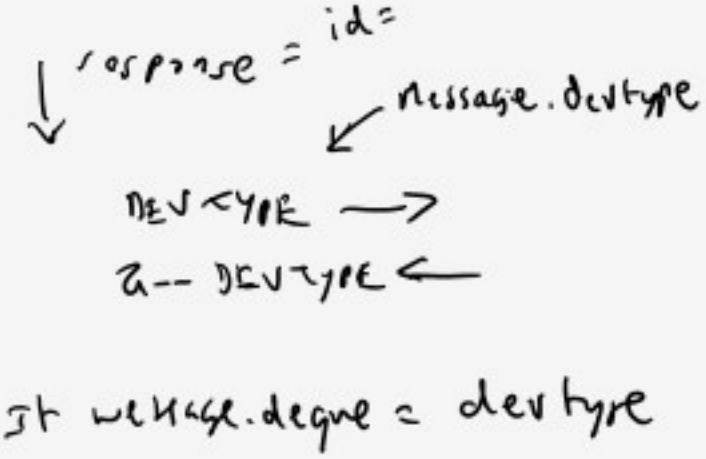
The supported commands for the temp firmware are:

Generic commands

- APVER - LLAP version
- DEVTYPE - Device type
- DEVNAME - Device name
- HELLO - Hello (PING)
- SER - Serial number
- FVER - Firmware version
- CHDEVID - Change device ID
- PANID - Change PANID
- REBOOT - Restart the device
- INTVLxxxx - Sets the interval for reading the temperature - 9995 would be 999 seconds - T=ms, S=secs, M=mins, H=hours, D=Days - Once the command is sent the device will begin to send readings at the frequency requested. Whilst you are testing it's useful not to yet issue CYCLE, once asleep it's not an easy task to wake it
- CYCLE - Activate cyclic sleeping, the device will wake every INTVL - Once this command has been issued the device will remain asleep for the majority of the time. Whilst asleep everything is turned off, that includes the radio

Device specific commands

- LVAL - Reply with the light level from 0 to 99



If message.deque = devtype

Personality 1 AΔ 7

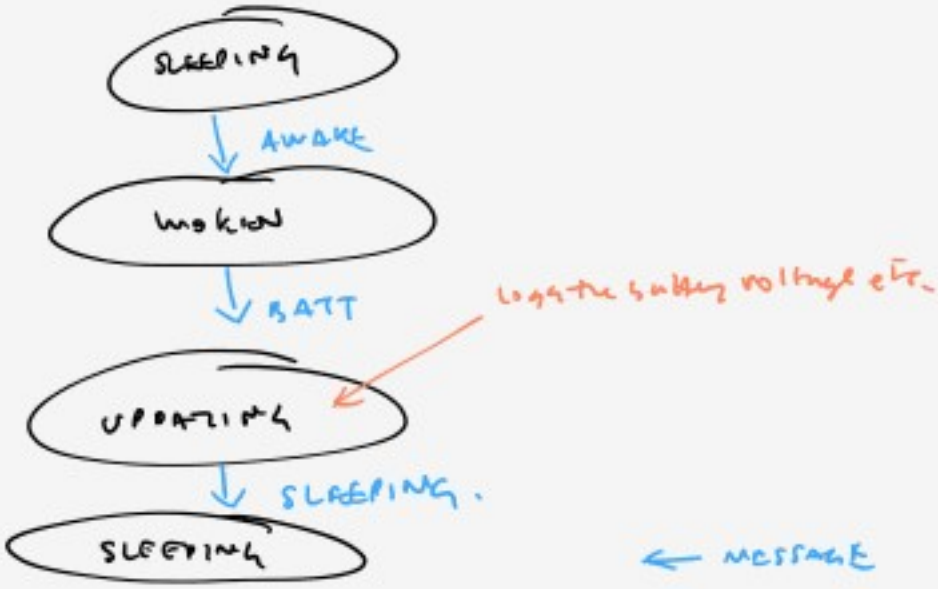
Personality 2 AB 7

registar Regis

registrum


Sleeping
NO
yes
wait for BATT
sent INTERVAL

So the question is
can I map this
sequence with an
ethtool command.



store device specific data.

use KVP's

could be another lookup.

deviceid, keyId, value.

Keys