Accademic Year: 2023-2024

POLITECNICO
MILANO 1863

IoT Challenge #1

Wokwi and Power Consumption

INTERNET OF THINGS

PROFESSOR CESANA MATTEO

CONTI ALESSANDRO PERSONAL CODE: 10710583
CASALI SARA PERSONAL CODE: 10727955
LAZZERI MATTEO PERSONAL CODE: 10710962

Contents

1. EXPLANATION OF CODE LOGIC

First, the pins of the HC-SR04 ultrasonic sensor are configured. Next, the sensor is activated to detect the presence of a car. This functionality is implemented in the String readOccupancy() function, which returns a string indicating whether the parking space is free or occupied.

Next, the Wi-Fi is activated with the void enableWiFi() function, which not only turns on the module, but also records the node information with the network sink. At this point, the message representing the parking spot state is transmitted to the sink node.

Finally, the system enters deep sleep mode via a call to the void goToDeepSleep() function, which sets a time interrupt that allows the node to wake up and recheck for the presence or absence of a car.

2. ENERGY CONSUMPTION ESTIMATION

- Duty cycle period: $10710583 \rightarrow 83\ mod 50 + 5 = 33 + 5 = 38\ s$
- Battery energy $= 10710583 \rightarrow 0583 + 5 = 588\ J$
- Important data
  - ESP-NOW transmission rate $= 1\ Mbps$
  - ESP-NOW packet header $= 24B + 1B + 3B + 4B + 7B + 4B = 43B$
    - MAC header $= 24\ B$
    - Category code $= 1\ B$
    - Organization identifier $=\ 3\ B$
    - Random value $= 4\ B$
    - Vendor specific header $= 7\ B$
    - FCS $= 4\ B$
  - Max ESP-NOW packet body $= 250\ B$
  - Max transmission time $= \frac{(43B+250B)\cdot 8bit}{1Mbps} = 0.002344\ s$
- Average power consumption:
  - deep_sleep.csv
    - Deep Sleep: $59.62\ mW$
    - Wake up: $313.41\ mW$
    - Wi-Fi On: $776.63\ mW$
    - Wi-Fi Off: $308.27\ mW$
  - send_different_TX.csv
    - Transmission 19,5 dBm: $1221.76\ mW$
    - Transmission 2 dBm: $797.29\ mW$
    - Wi-Fi on but not transmitting: $704.18\ mW$
  - read_sensor.csv
    - Sensor reading: $466.74\ mW$
    - Idle: $331.59\ mW$

- **Energy consumption** $= 2.263766481J + 0.000000331J + 0.012359275J + 0.00155326J + 0.002863805\,J = 2.280543134J = \mathbf{2.2805\,J}$

  Calculations are performed by considering the maximum possible values, to underestimate battery life and be able to make it last at most the estimated time.

  - Deep sleep
    - Time $= 38s - 1\mu s - 2ms - 0.002344s = 37.9699175\,s$
    - Power $= 59.62\,mW$
    - Energy $= 59.62mW \cdot 37.9699175s = 2.263766481\,J$
  - Idle

    This state occurs before the real reading from the sensor, when the pins of HC-SR04 are put low.
    - Time $= 1\,\mu s$
    - Power $= 331.59\,mW$
    - Energy $= 331.59mW \cdot 1\mu s = 0.000000331\,J$
  - Sensor reading

    The sensor reading is composed of two phases, each of which has a time associated with it. The first time is the lowest time needed to send the ultrasonic wave in order to perform a best reading, the other one is the maximum waiting time to receive back a wave that travel up to 450cm of distance.
    - Time $= 10\mu s + 26470\mu s = 26480\,\mu s$
    - Power $= 466.74\,mW$
    - Energy $= 466.74mW \cdot 26480\mu s = 0.012359275\,J$
  - Wi-Fi on

    The sensor only stays in this state if it is to receive the ACK message from the sink node.
    - Time $= 2\,ms$
    - Power $= 776.63\,mW$
    - Energy $= 776.63mW \cdot 2ms = 0.00155326\,J$
  - Transmission

    The sensor stays in this state for the time it takes to send a complete packet to the sink node.
    - Time $= 0.002344\,s$
    - Power $= 1221.76\,mW$
    - Energy $= 1221.76mW \cdot 0.002344s = 0.002863805\,J$

- **Battery lifetime** $= 257 cycle \cdot 38 s = 9766 s = \mathbf{2\ h\ 42\ min\ 45\ s}$
  - Number or cycle $= \frac{588 J}{2.2805} = 257\ cycle$
  - Cycle time $= 38\ s$

3. COMMENTS RESULTS AND IMPROVEMENTS

3.1. Comment on the implemented system

Most of the battery consumption is due to the deep sleep state of ESP-32, because the node spends most of its time there, so all the timing improvement implementations suggested below are not as sufficient as one would expect.

Due to the high-power consumption of ESP32 in deep sleep mode and to the low battery capacity, we estimate that the sensor node could run less than 3 hours, making totally unfeasible a realization of this kind, with this technology.

A small comment out of the scope of this challenge, looking at the datasheet of the ESP32 we have noticed that the power consumption in deep sleep mode is about 150 μA at 3.7 V that's lead to a power lower than 1mW. With this type of power consumption an application like the one required from this challenge it's a lot more feasible.

3.2. Possible improvements in terms of energy consumption

To optimize power consumption, these strategies can be implemented.

First, it is possible to implement the node without the need to receive an ACK after each sending of the message to the sink node.

Secondly, exploit the RTC memory of the node's ESP32 board to store a Boolean variable. This variable serves as an indicator of any change in the state of the parking spot. As a result, the node triggers the transmission of a message to the sink node only when there is an actual alteration in the parking state. This approach significantly reduces the power consumption associated with activating and maintaining the Wi-Fi connection, improving overall efficiency and dramatically reducing power consumption to allow the battery present in the node to last much longer.

One more is to extend the duty cycle period from the currently implemented seconds to several minutes. By lengthening the intervals spent in deep sleep mode, the node effectively reduces battery usage. This adjustment ensures that the node remains in a low-power state for longer periods, thus conserving energy resources.

Another strategy could be to reduce the amount of data transmitted. Instead of transmitting a string to describe the state of the parking spot, could be transmitted just a Boolean variable that is set to 1 if the spot is occupied by a car and 0 if it's free.