

# IoT Project: ”Keep your distance”

Prof. Matteo Cesana - Year 2020/2021

*Gabba Rohit [codice persona: 10706944]*

*Tortorelli Giuseppe [codice persona: 10582962]*

# 1 TinyOS

Al fine di identificare i messaggi consecutivi abbiamo scelto di associare ad ogni messaggio un id incrementale. La struttura è la seguente:

- *id*: contenente l'id di ogni mote
- *msg\_number*: l'id del messaggio

Abbiamo scelto di associare ad ogni mote 3 memorie di grandezza fissata:

- *near\_motes\_ids*: per salvare l'id dei nodi vicini
- *near\_motes\_counters*: per contare quanti messaggi consecutivi sono stati ricevuti
- *near\_motes\_msgs\_number*: per salvare l'id del messaggio che ci si aspetta di ricevere

Ogni volta che un mote invia un messaggio, incrementa l'id da associare al messaggio e lo invia.

Alla ricezione di un messaggio, si scorre l'array *near\_motes\_ids* in cerca dell'id del mote da cui è stato ricevuto il messaggio.

Se l'id non viene trovato, viene aggiunto nella prima cella disponibile. Altrimenti si procede con la verifica dell'id del messaggio.

Quindi si legge il valore aspettato dall'array *near\_motes\_msgs\_number* e si confronta con quello ricevuto. Se i due valori non corrispondono, il contatore associato al mote viene azzerato.

Si procede quindi alla verifica del valore contenuto nell'array *near\_motes\_counters*: nel caso sono stati ricevuti 10 messaggi consecutivi, si lancia l'allarme e si azzerava il contatore, altrimenti il contatore viene incrementato.

## 2 Cooja

Abbiamo testato il funzionamento in *Cooja* inserendo 5 *Sky mote* e provando varie configurazioni di questi nella Network.

Nell'implementazione abbiamo assunto che la ricezione dei messaggi avvenga senza interferenze, per questo abbiamo ignorato i casi in cui il contatore viene resettato a causa di un messaggio non consegnato correttamente.

Di seguito gli screenshots delle configurazioni testate (sono ordinati secondo il log file)



Ad ogni nodo abbiamo associato un socket verso l'ip di *Node-RED* ognuno con una porta differente (da 6001 a 6005)

## 3 Node-RED

Abbiamo creato 5 (numero di mote scelti per il testing) nodi *tcp in* ognuno ascoltante sulla porta del nodo associato.

Abbiamo quindi inserito un nodo *function* per filtrare i messaggi di allarme ed estrarne gli id dei mote interessati. Infine abbiamo utilizzato un nodo *http request* per mandare una POST request verso *IFTTT* specificando la nostra API key

## 4 IFTTT

Ricevuti i dati tramite un applet *Webhooks* ascoltante sull'evento *ALARM*, inviamo una notifica sull'app *IFTTT* per cellulare con il seguente testo:

*"ALARM mote x: I'm close to mote y! COVID"*

Dove  $x$  è l'id del mote che ha inviato il messaggio.