RFID

Is a system composed by:

- TAGS -> cheap devices to store the ID
- READER -> queries tags to get their IDs
- SERVERS -> handles data by readers and process it

with the goal of OBJECT IDENTIFICATION

Generally the communication is wireless and is done by the tags (which have NO POWER /BATTERY) that receive energy by the reader and then reflect the high-power signal to send their IDs.

An RFID channel is wireless and shared so we can have COLLISIONS that must be handled

Several MAC protocols to do it:

SEQUENTIAL

CONCURRENT

Tree based

- BS
- Query Tree

Aloha based

- FSA
- EPC
- TSA
- BSTSA

Binary Splitting

Recursively split the tree in 2 subgraups depending on random 0/1.

Each tag has a counter (0)

while all readed:

Read the tag with $c=0$

If collision:

every tag generates a random value and sum to c

If silent tag → $c=c+1$

If 0 or 1 reply → all tags: $c=c-1$

Query Tree Protocol

Query tags according to the binary structure of their ID

S= NULL

while all readed:

if prefix match s:

sends query

if collision:

len (s)+=1

add 0/1

PERFORMANCE OF TREE BASED

If we have a uniform ID distribution $\rightarrow PF(BS) = PF(QT)$

System Efficiency: $SE = \frac{n}{q}$ $n \rightarrow$ # of tags, $q \rightarrow$ # of queries

$$BS(n) = \begin{cases} 1 & \text{if } n \leq 1 \\ 1 + \sum_{K=0}^{n} \binom{n}{K} \left(\frac{1}{2}\right)^{K} \left(1 - \frac{1}{2}\right)^{n-K} (BS(K) + BS(n-K)) & \text{if } n > 1 \end{cases}$$

$\downarrow$

# of queries in binary splitting

$SE_{BS} = 0{,}38$ (LOW)

ALOHA

- Time slotted
- Slot duration = tag's ID transmission time
- Slot are grouped in frames
- Each tag choose a slot to respond

Framed Slotted Aloha (FSA)

In the 1st query the reader includes the # of slots
The tag then randomly pick a slot to reply
If collision -> wait for an other request

Repeat the process untill all are identified.

EFF: 37% = $\frac{\text{\# IDENTIFICATIONS}}{\text{\# SLOTS}}$

EPC GEN 2

↓

Based on FSA, it adopts the frame len according to the # of collisions and empty slots

IDLE RESPONSES last less then ID/collision responses

In FSA protocols in which n tags and N slots:

- p (r tags answ in same slot)

$$= S(r) = N \times \binom{n}{r}\left(\frac{1}{N}\right)^r\left(1-\frac{1}{N}\right)^{n-r}$$

R_{ID} = # of ID $\quad$ R_C = # of coll $\quad$ R_{IDLE} = # of idle rounds

EFF with round of same duration $= \dfrac{R_{ID}}{R_{ID}+R_C+R_{IDLE}} = 36\%$

EFF with IDLE rounds last a β fraction of ID/coll round $= \dfrac{R_{ID}}{R_{ID}+R_C+(R_{IDLE}\cdot\beta)}$

40% of time is wasted in IDLE and collision slots.

Tree Slotted Aloha

Create a new child frame ∀ collision slot;
the tags that reply to the same frame slot partecipate in new slot

EFF = 43%

MAIN ISSUES with FSA e TSA:

- How to estimate frame size?
- How many tags in environment?

The initial frame size → predefined value

$$\text{tagsPerCollSlot} = \frac{\text{estim \# of tags} - \text{identified tags}}{\text{collision slot}}$$

We can estimate it:

let - N = size of completed frame

- $\langle c_0, c_1, c_k \rangle$ = obs values
- $\langle a_0, a_1, a_k \rangle$ = estim. values

empty, ID, collision

If we compute the bounds ($^{SL}50$) we can see that the upper bound is inaccurate (can be ∞!)

Can we have a better solution? Get a good initial estimation

BSTSA : Binary Splitting and Tree Slotted Aloha

Any large group of tags randomly split into 2 groups of almost the same size

- BS -> devide tags into groups
- TSA -> identify tags

When the splitting process reaches a single-tag group
-> start the tag identification

EFF: 70/80 % (n tag $= 0$ to 5000)