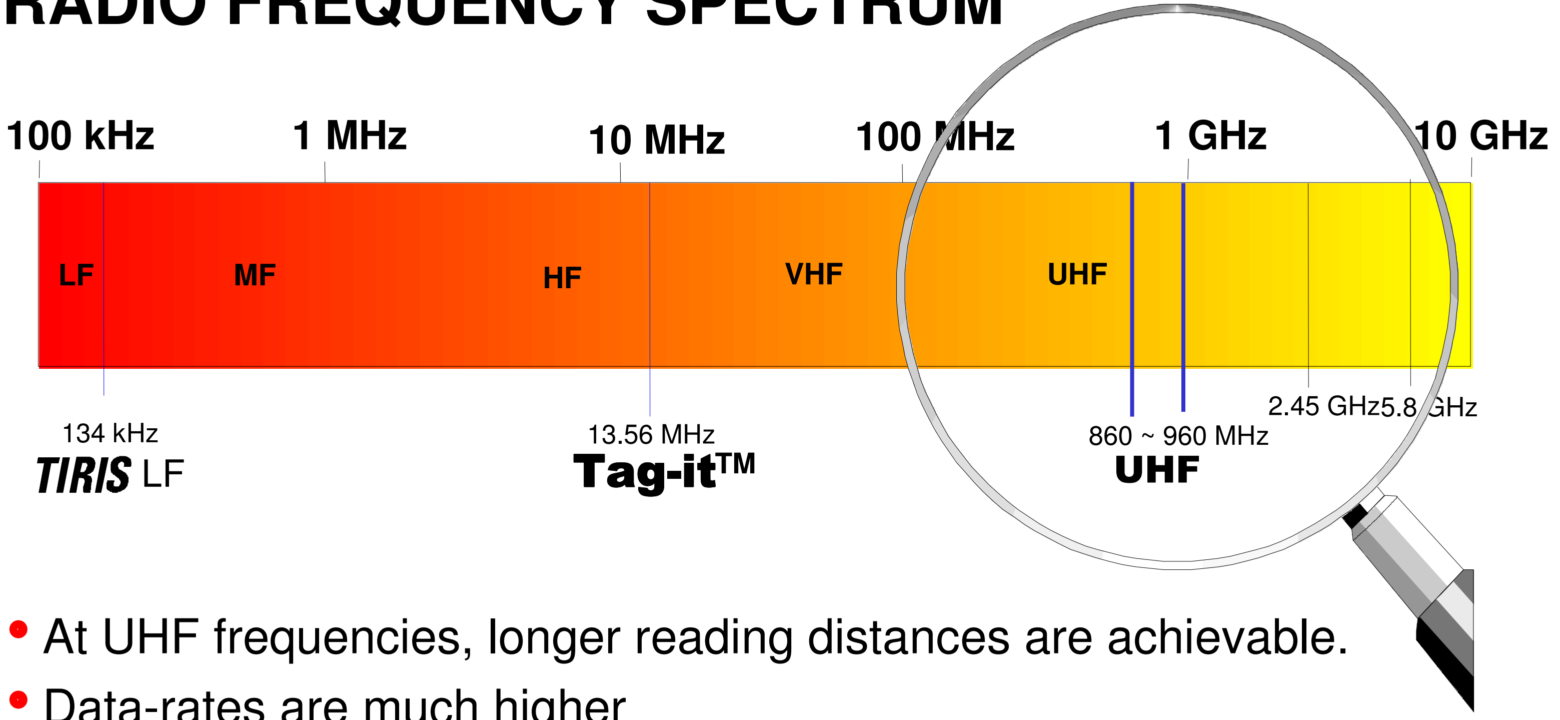
TI-RFid™
A TEXAS INSTRUMENTS TECHNOLOGY

# UHF Gen 2
## System Overview

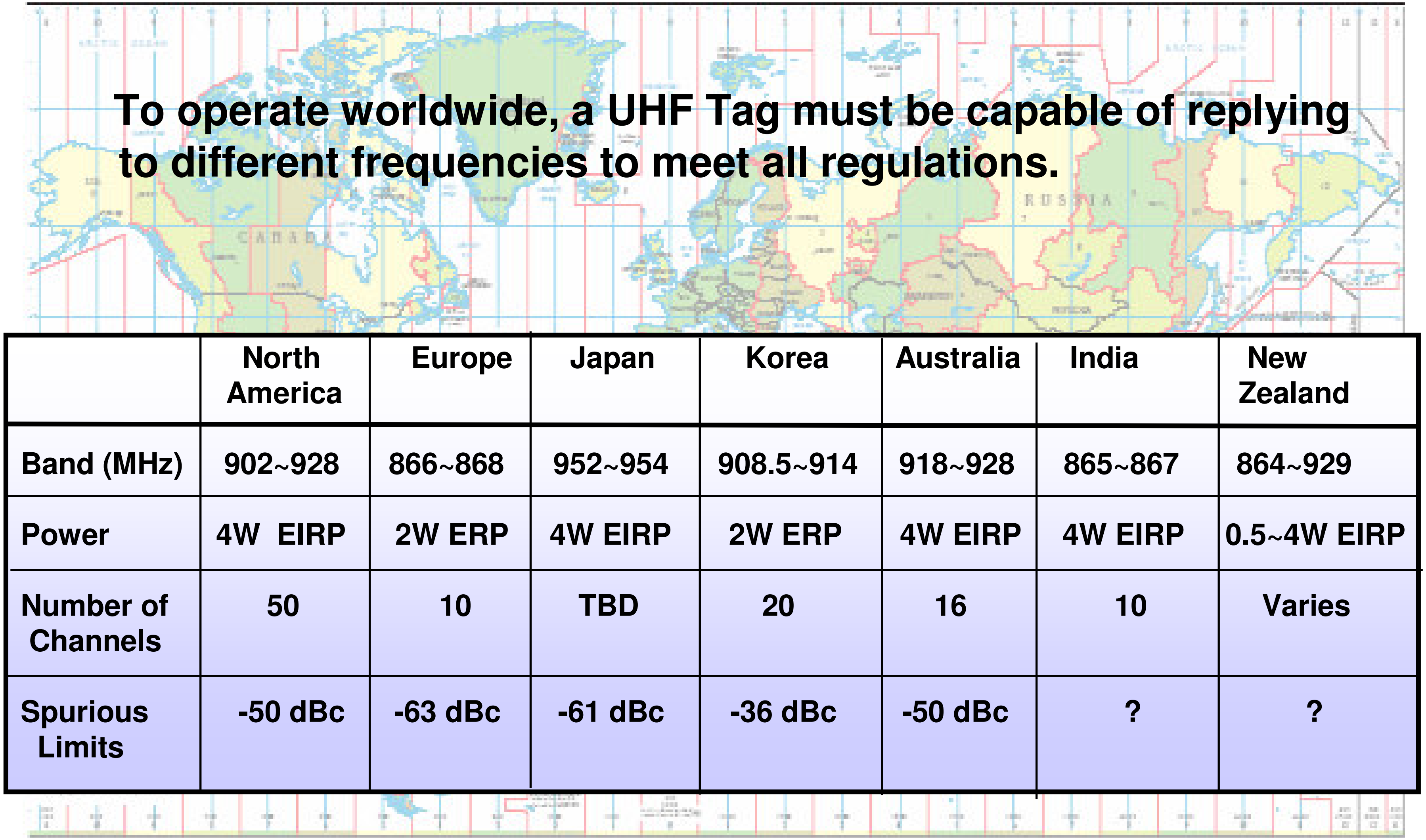
# Principles of Operation

## RADIO FREQUENCY SPECTRUM



- At UHF frequencies, longer reading distances are achievable.
- Data-rates are much higher
- Signals don't pass through materials as well as lower frequencies.
- Reflections can extend the read range, but make the reading zone less well defined. (Ghost readings from labels thought to be out-of-range)

# Global Regulatory Situation



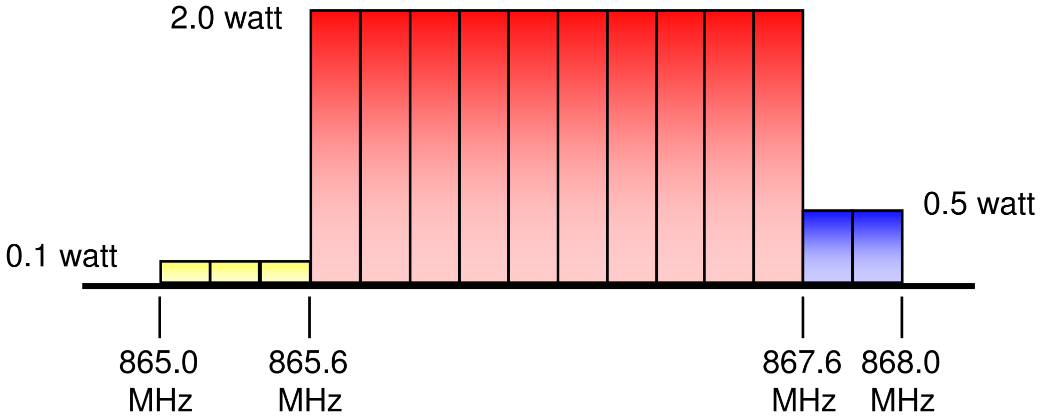
**To operate worldwide, a UHF Tag must be capable of replying to different frequencies to meet all regulations.**

| | North America | Europe | Japan | Korea | Australia | India | New Zealand |
|---|---|---|---|---|---|---|---|
| Band (MHz) | 902~928 | 866~868 | 952~954 | 908.5~914 | 918~928 | 865~867 | 864~929 |
| Power | 4W EIRP | 2W ERP | 4W EIRP | 2W ERP | 4W EIRP | 4W EIRP | 0.5~4W EIRP |
| Number of Channels | 50 | 10 | TBD | 20 | 16 | 10 | Varies |
| Spurious Limits | -50 dBc | -63 dBc | -61 dBc | -36 dBc | -50 dBc | ? | ? |

# Global Regulatory Situation

## Europe

- The new ETSI 302 208 regulations define 3 sub-bands
  - Band 1: 865.0 ~ 865.6 MHz, 0.1 watt ERP, LBT* level -83 dBm
  - Band 2: 865.6 ~ 867.6 MHz, 2.0 watt ERP, LBT level -96 dBm
  - Band 3: 867.6 ~ 868.0 MHz, 0.5 watt ERP, LBT level -90 dBm

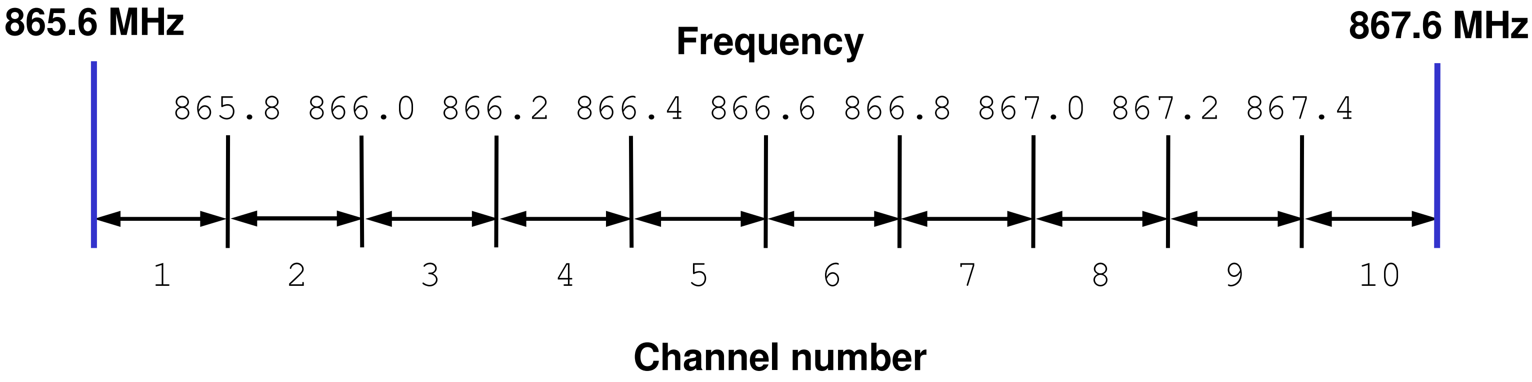


* LBT = Listen Before Talk

# Global Regulatory Situation

## Europe

- Supply chain Tags will mostly operate in the band 2:
    - 865.6 ~ 867.6 MHz  (ETSI EN 302 208 regulations)
    - Regulations come into effect when published in EU Journal

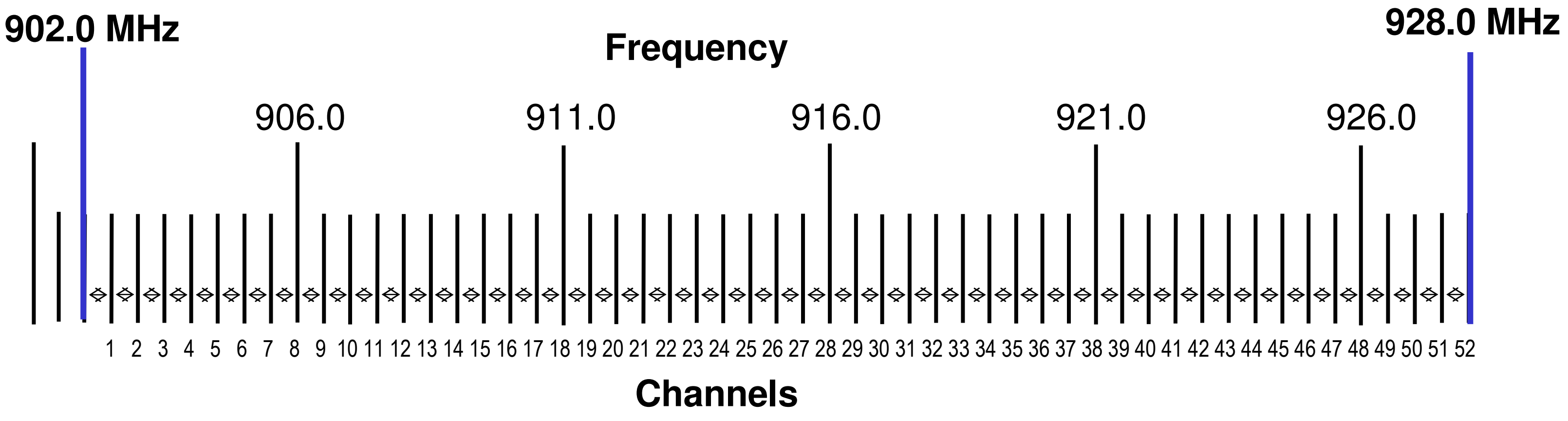


- 10 channels of 200 kHz @ 2W ERP (3.2W EIRP)

# Global Regulatory Situation

## North America

- Tags are approved to operate in the following band:



- 902 ~ 928 MHz (FCC Part 15.247 regulations)
- Frequency Hopping – 52 channels × 500 kHz @ 4W EIRP

# Reader Operating Environments

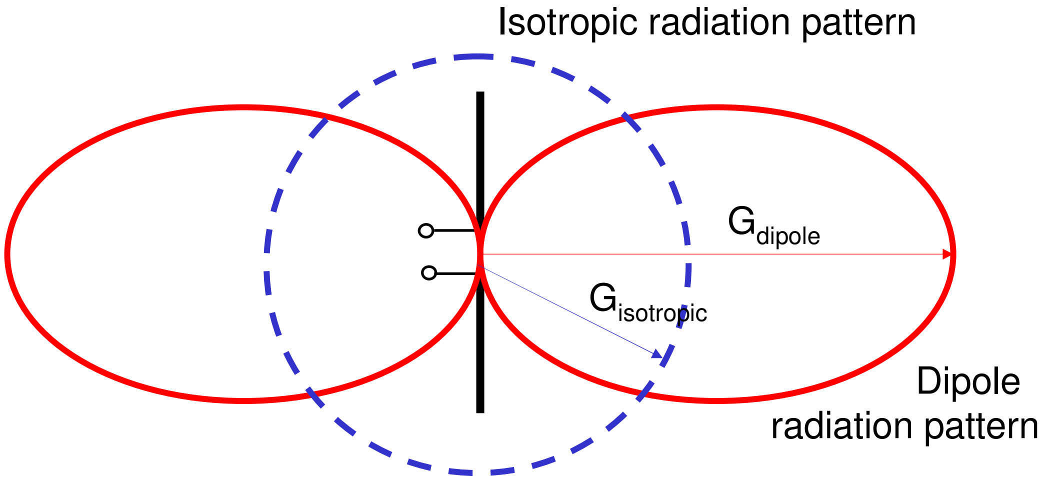
- **Listen Before Talk** (LBT)
  - Part of the European regulations, is Listen Before Talk. If a reader detects a signal in its environment, on the channel it intends transmitting, it must switch to another free channel. After 4 seconds it must turn its transmitter off for 0.1 seconds to allow other readers access to that channel.
- **Operating Environment**
  - This is defined are the zone within which the reader's RF signal is greater than -90 dB (a radius of approximately 1 Km).
- **Single Reader Environment**
  - When only a single reader is operating in an Environment.
- **Multiple reader Environment**
  - In such an Environment, the number of simultaneously operating readers, will be less than the available number of channels.

# Reader Operating Environments

- **Dense Reader Mode**
  - When the number of readers operating is large when compared to the number of available channels, then this is defined as a Dense Reader Environment, e.g. > 20 readers operating in 20 available channels.
  - In such an environment, certified readers must incorporate the schemes defined in the Gen 2 Specification to minimise mutual interference.
    - With the time synchronized technique, the readers all transmit together, then, while maintaining their CW, listen for the tag responses.
    - In the frequency separated method, Readers transmit on even numbered channels, while tags respond on odd numbered channels.
      - In this method the powerful reader signals (100dB greater than the backscattered signal) do not mask the tag signals
    - Tags have no frequency selection but respond to the strongest signal

# ERP and EIRP compared

Regulations expressed in EIRP (equivalent isotropic radiated power) are based on the spherical radiation pattern of an isotropic emitter



Real antennas such as dipoles, do not radiate uniformly in all directions (e.g. no power is radiated along the axis).

ERP power levels relate to the dipole antenna, and the relationship between the gain of an isotropic and a dipole antenna is given by:

$$\mathbf{P_{EIRP} = P_{ERP} \times 1.64}$$

Thus the European limit of 2 W ERP is equivalent to 3.28 W EIRP (USA = 4 W EIRP)

# Global Comparison

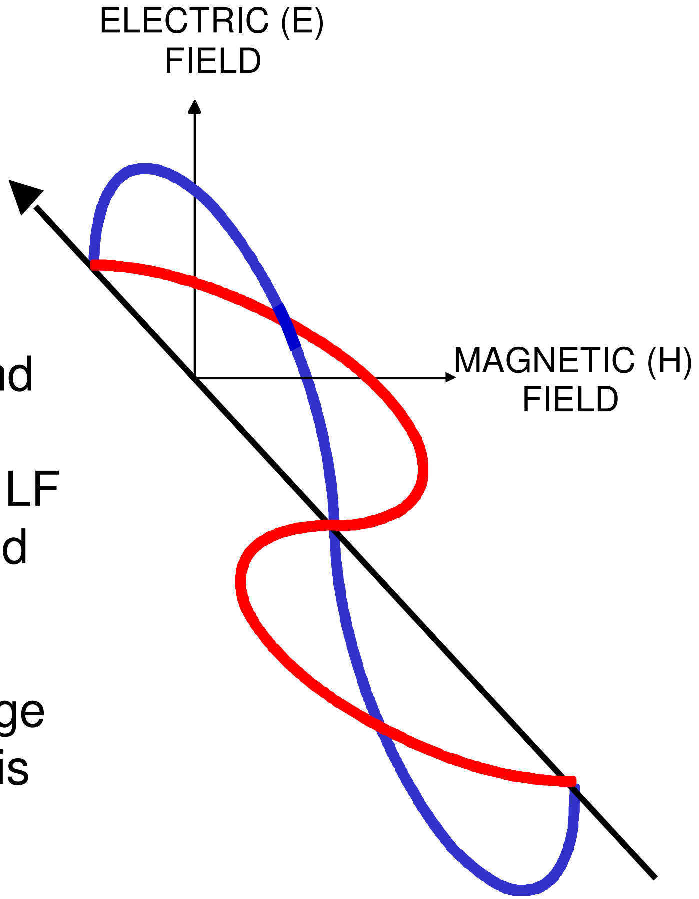
## US, European and Japanese regulations compared

- 2 W ERP is equivalent to 3.28 W EIRP so the power levels in Europe, Japan are slightly down on North America limits.
- Unfortunately, in India, Japan and Europe only 2 MHz of spectrum is available, whereas in the USA it is 26 MHz.
  - This means the data rate between readers and Tags will be much less in India, Europe and Japan.
  - The spectral mask imposed by the EU/parts of Asia, limits data transfer rates to 30% of those possible in North America. (500 vs. 1500 reads/sec)
  - .
  - This may limit the speed of pallet loads on fork lifts passing through dock doors

# System Overview

## Power Transfer

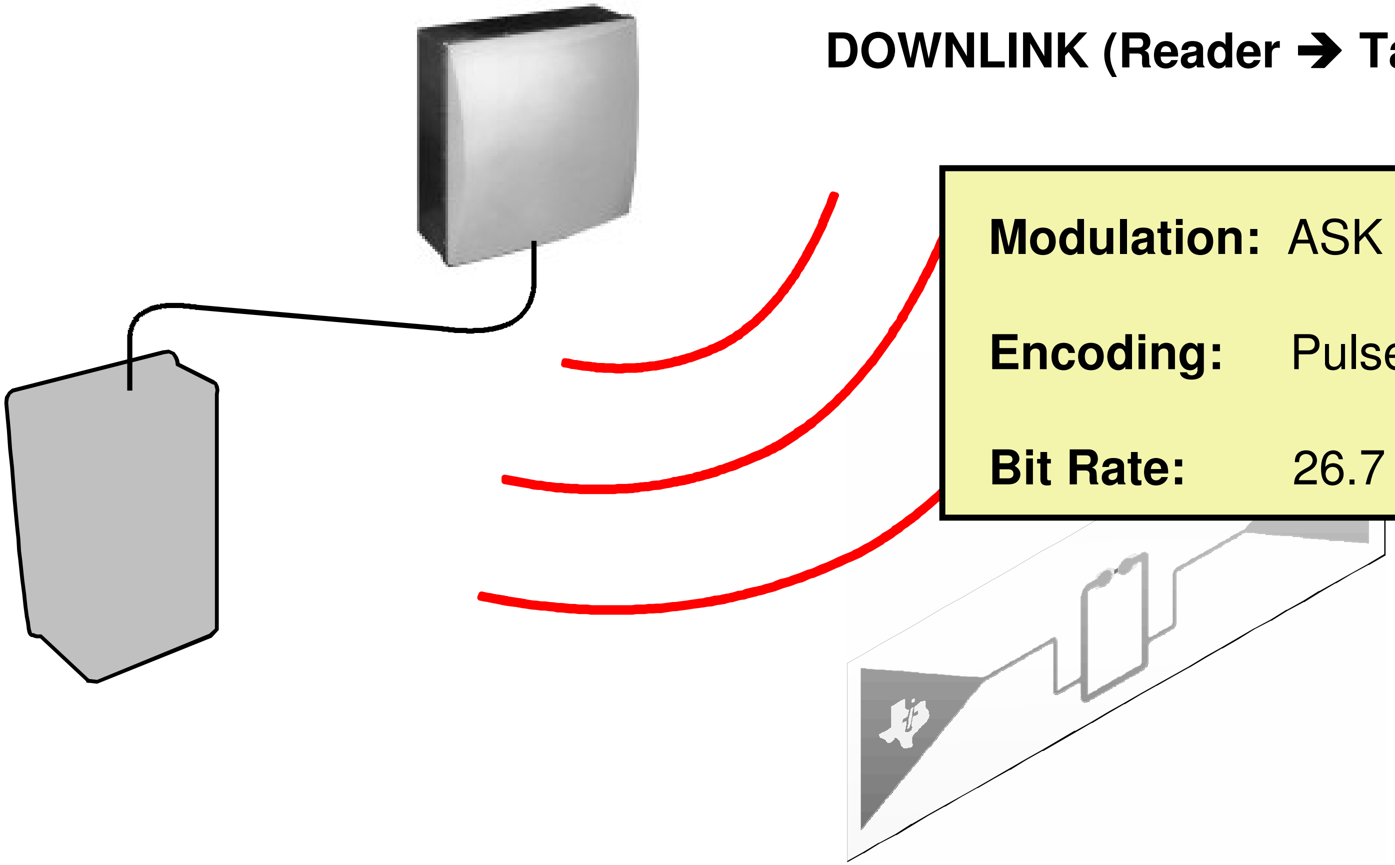
- Radio signals are electromagnetic waves, having a magnetic component (H-Field) and an electric component (E-Field)
- UHF systems use the **Electric field** and transfer power by capacitive coupling, achieving greater reading ranges than LF & HF Tags which use the magnetic field (inductive coupling)
- The Electric field results from the voltage changes occurring in the antenna and is measured in V/m or more commonly dBµV/m



# Principles of Operation

## Communication between Reader and Tag

**DOWNLINK (Reader → Tag)**



**Modulation:** ASK

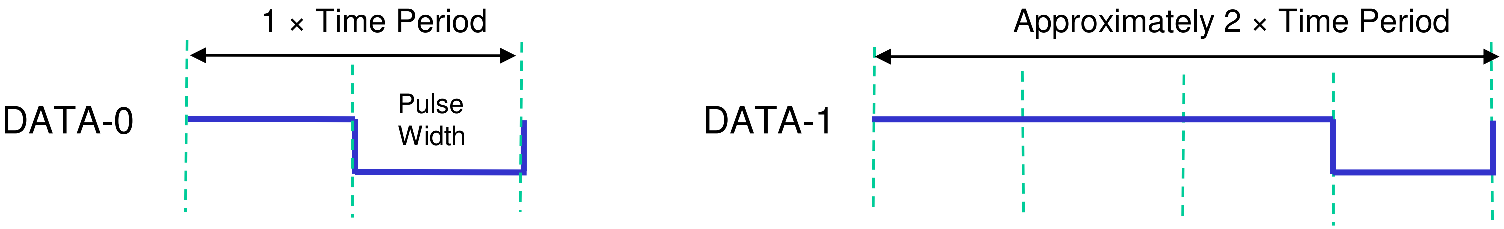
**Encoding:** Pulse Interval

**Bit Rate:** 26.7 to 128 Kbits/s

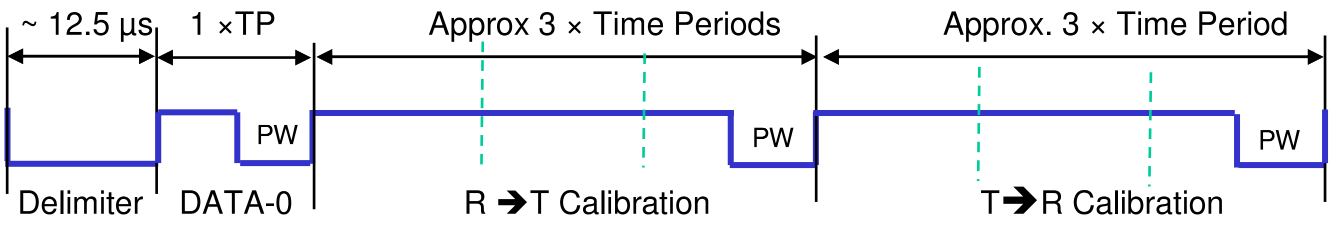
# Reader → Tag Modulation

## **Pulse Interval Encoding** (PIE)

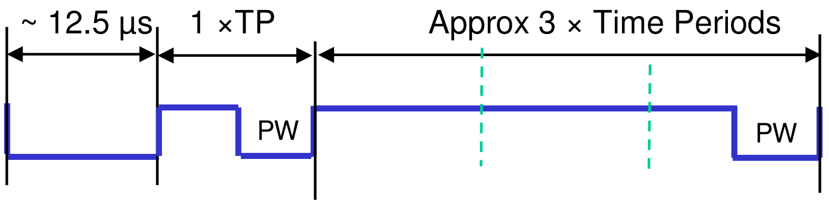
- Data is passed to the Tag by pulsing the carrier wave (CW) at different time intervals to indicate the 1 & 0 bits.



- All Reader to Tag communication must start with a Preamble:



- Subsequent commands can use a Frame-Synch:



**Note**

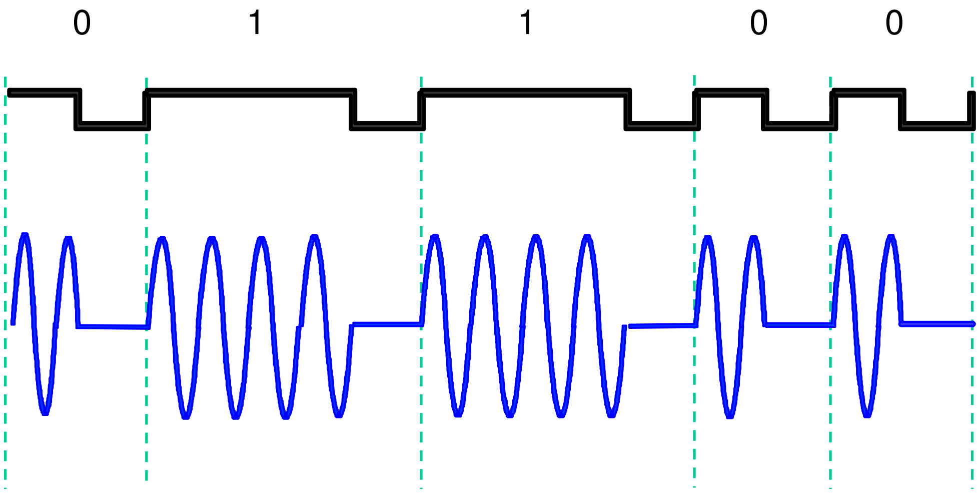
No EOF is necessary

# Principles of Operation

## Reader → Tag Modulation

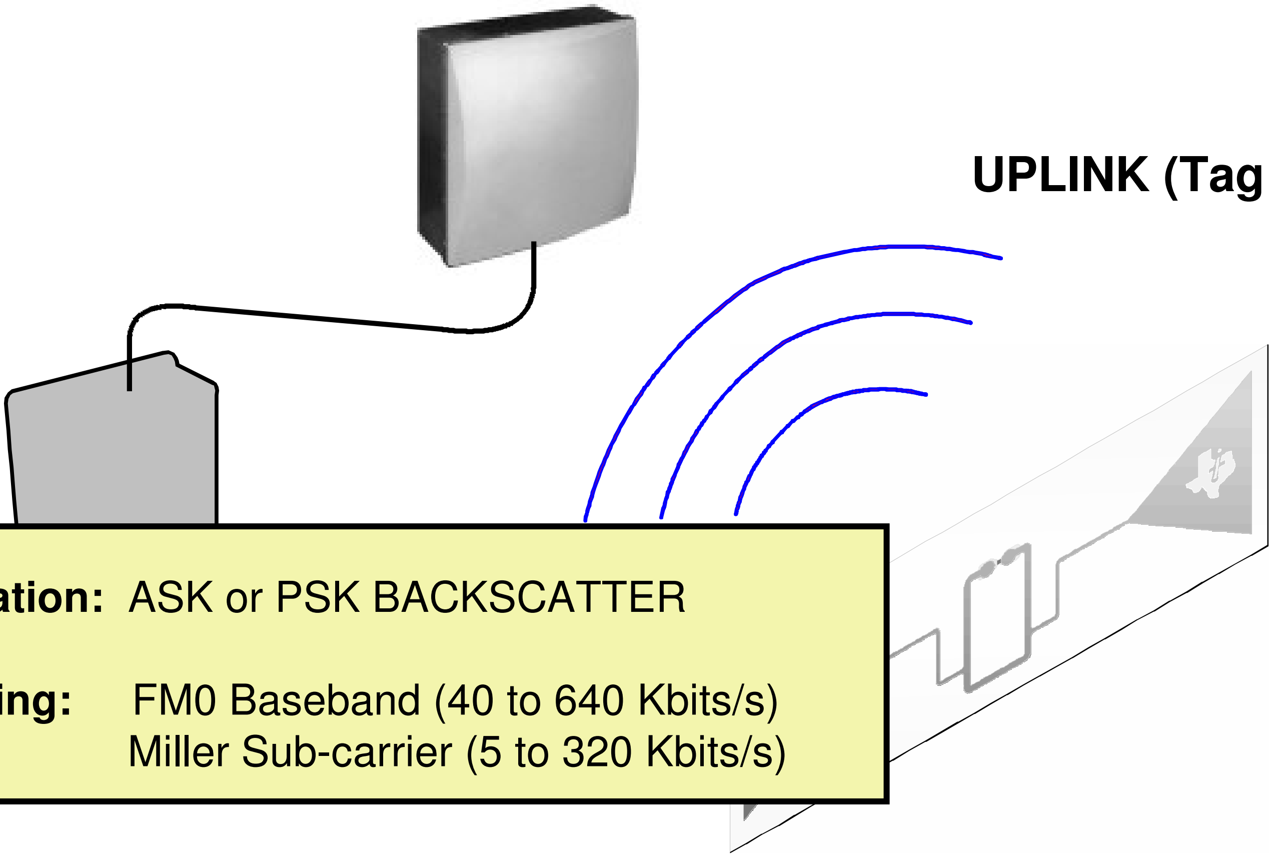
PIE bits
(Pulse Interval Encoding)

ASK Modulation
(Amplitude Shift Keying)



# Principles of Operation

## Communication between Tag and Reader



**UPLINK (Tag ➔ Reader)**

**Modulation:** ASK or PSK BACKSCATTER

**Encoding:** FM0 Baseband (40 to 640 Kbits/s)
Miller Sub-carrier (5 to 320 Kbits/s)

# Reflected Electromagnetic Waves

Backscattering: energy reflected in a direction opposite to that of the incident E-field waves.

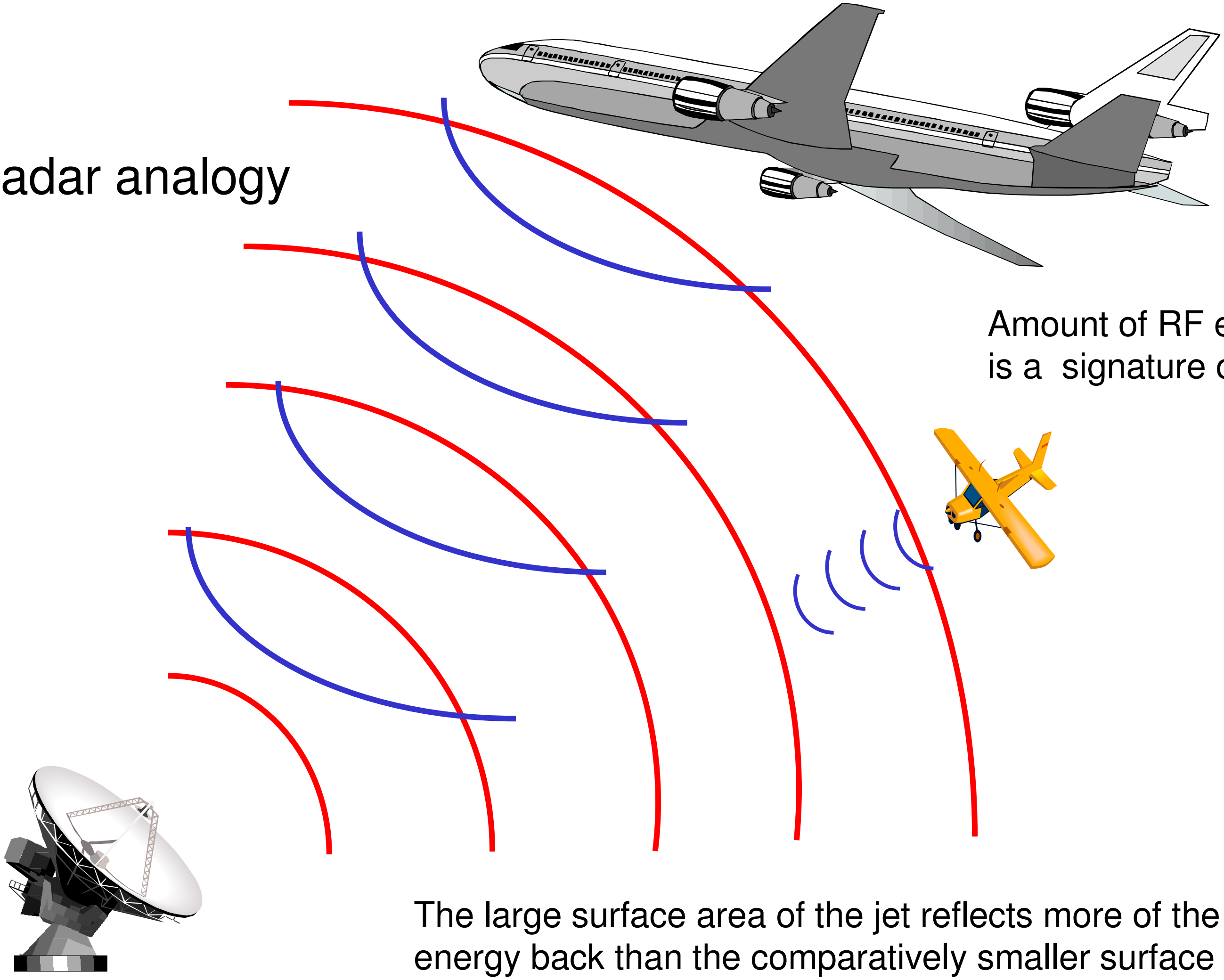
Think about a radar system analogy:

Radar on the ground is tracking an airplane:

- radar transmits waves of RF energy toward the target
- the target reflects these waves of energy back.
- amount of energy reflected back is dependent upon the surface area of the target that is exposed to the incident RF waves.
- more surface area (larger airplane) equates to more energy being reflected back.
- determination of target size is determined indirectly through the amount of signal reflected back.

# Backscatter

A radar analogy

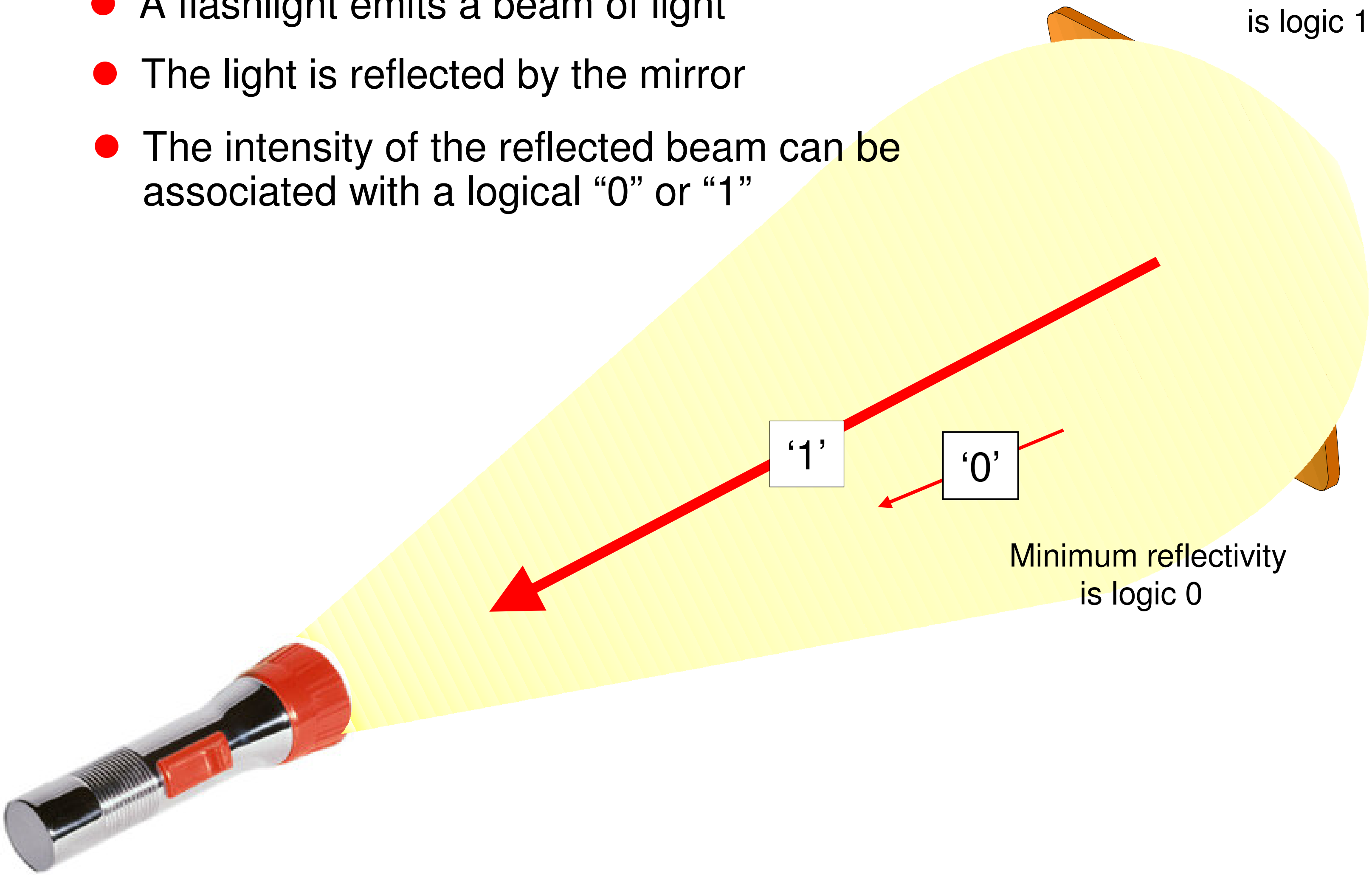


Amount of RF energy reflected back is a signature of the target.

The large surface area of the jet reflects more of the incident RF energy back than the comparatively smaller surface area of the orange bi-plane.

# Backscatter Modulation

- A flashlight emits a beam of light
- The light is reflected by the mirror
- The intensity of the reflected beam can be associated with a logical “0” or “1”



Maximum reflectivity is logic 1
‘1’
‘0’
Minimum reflectivity is logic 0

# Backscatter

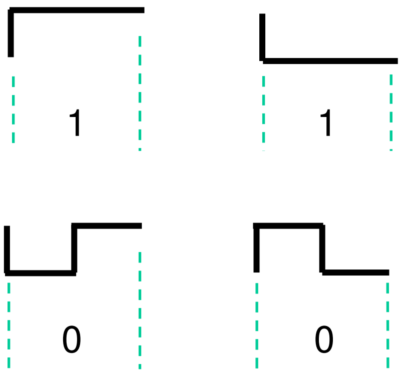
## Tag to Reader Modulation

- The tag uses Backscatter modulation to respond to a reader. It does this by switching the reflection coefficient of its antenna (using a shunt circuit) from a matched load where the incident RF signal is absorbed, to a short at the antenna terminals where the maximum reflected RF signal is created.
- The reader instructs the tag which method of data encoding to use when sending its data back:
  - Miller Subcarrier encoding
  - FM0 Baseband encoding
- The tag can use either/or two modulation formats - the tag manufacturer selects:
  - ASK (Amplitude Shift Keyed)
  - PSK (Phase Shift Keyed)

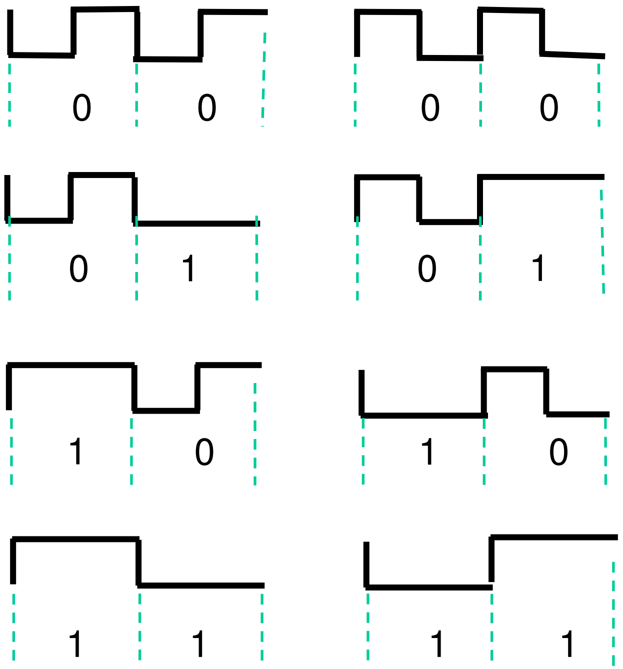
# Principles of Operation - Uplink

## Tag ➔Reader - FM0 Encoding

- In FM0 encoding, a transition has to occur at the end of each bit period, but  for a zero bit, an addition transition in the middle is required.
  - FM0 Symbols



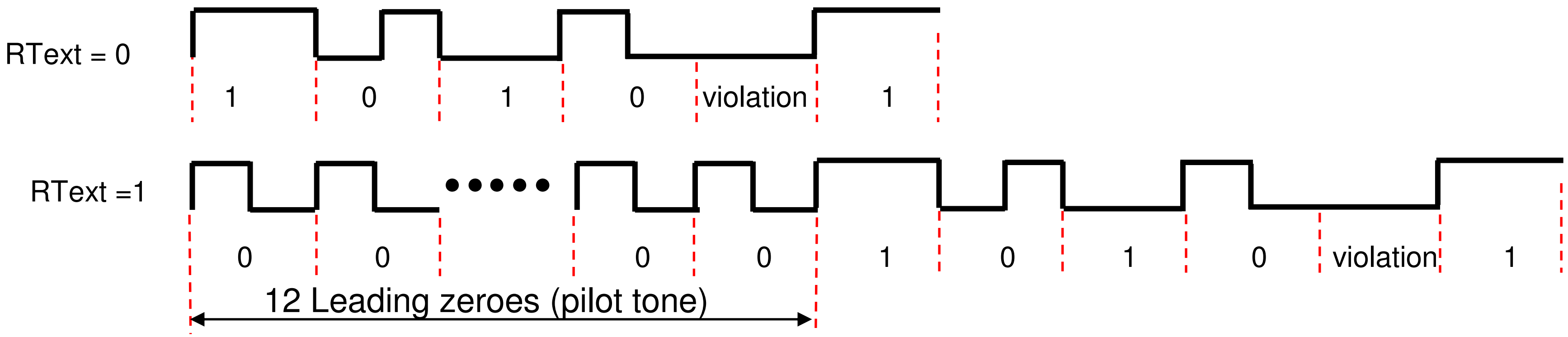
  - FM0 2-bit Sequences



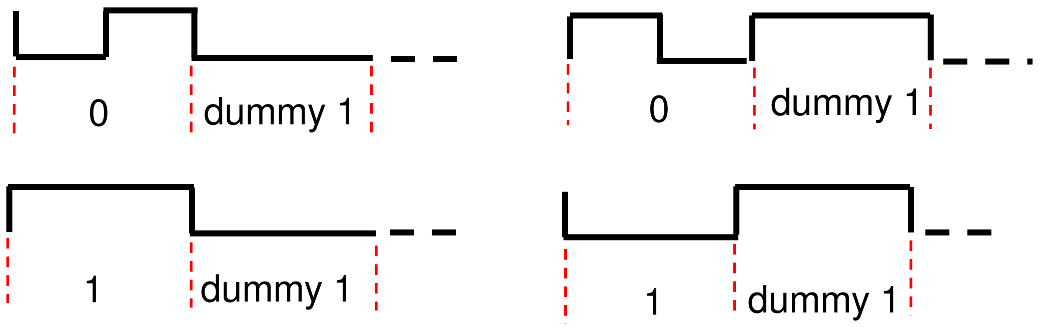
# Principles of Operation - Uplink

## FM0 Bit Encoding

- A FM0 message begins with one of these Tag to reader pre-ambles.



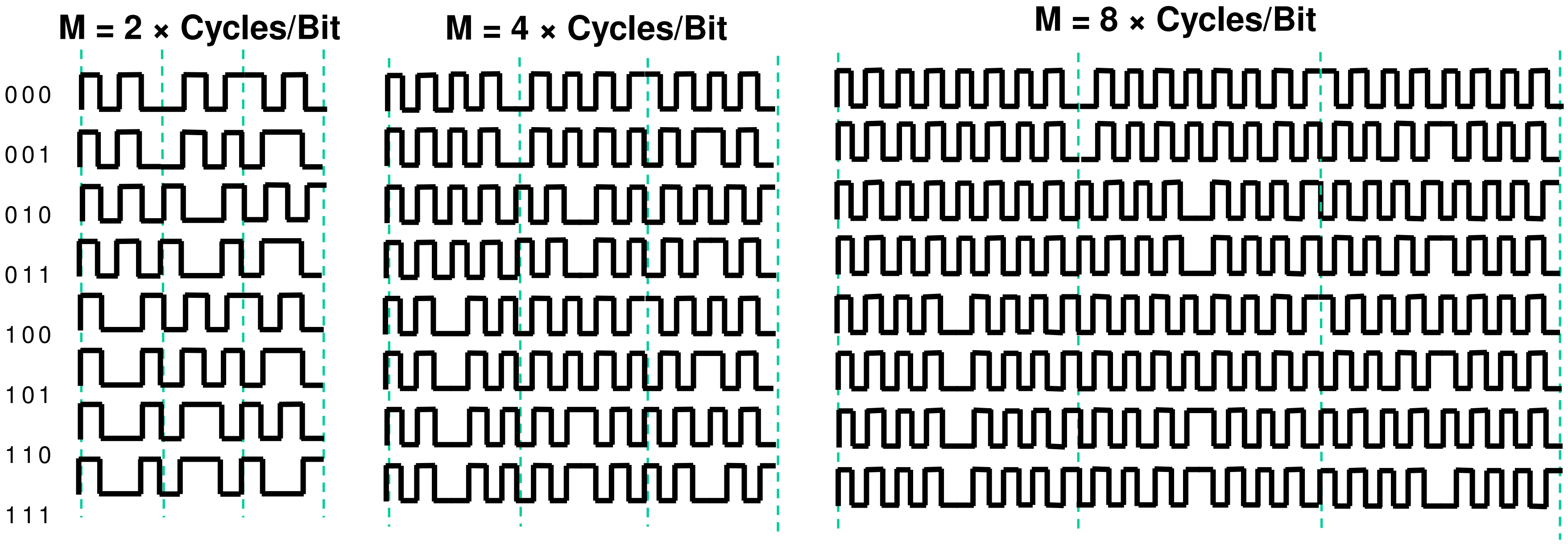
- and ends with one of these terminating sequences



# Principles of Operation

## Tag → Reader - Miller Encoding

- In Miller sub-carrier encoding, a transition occurs between two data-0s in sequence and also in the middle of a data-1. A Miller sequence can contain 2, 4 or 8 sub-carrier cycles/bit



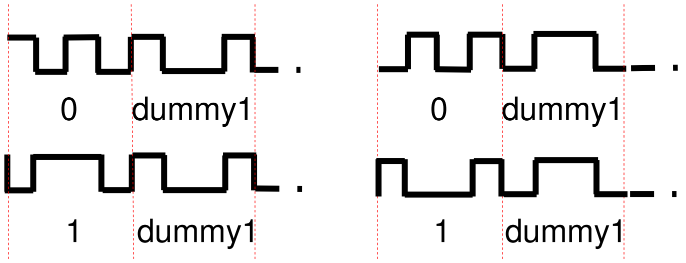
**Note** 'M' is a parameter in the Query command

# Principles of Operation

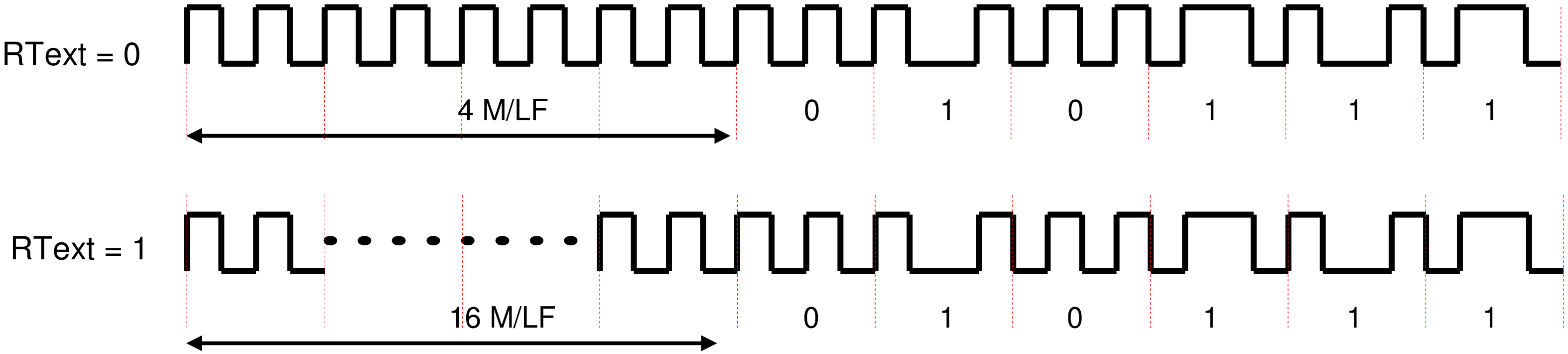
## Miller Encoding – Preambles and endings

- A Miller sequence terminates with a dummy 1

  (only 2 cycles/bit are shown in the examples on this page)



- There are 2 Miller Sub-carrier preambles. The *Query* command tells the Tag which to use.

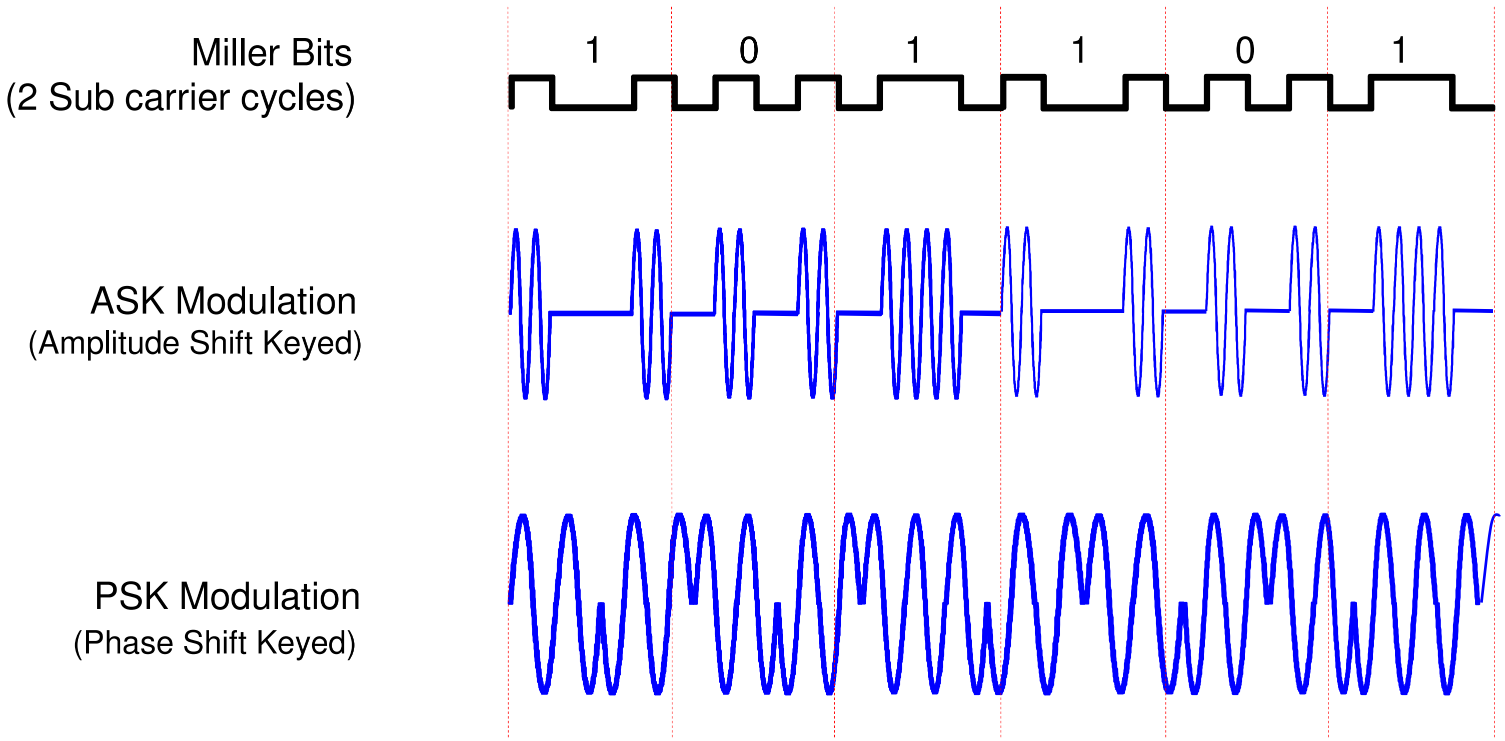


# Principles of Operation

## Tag ➔ Reader Modulation

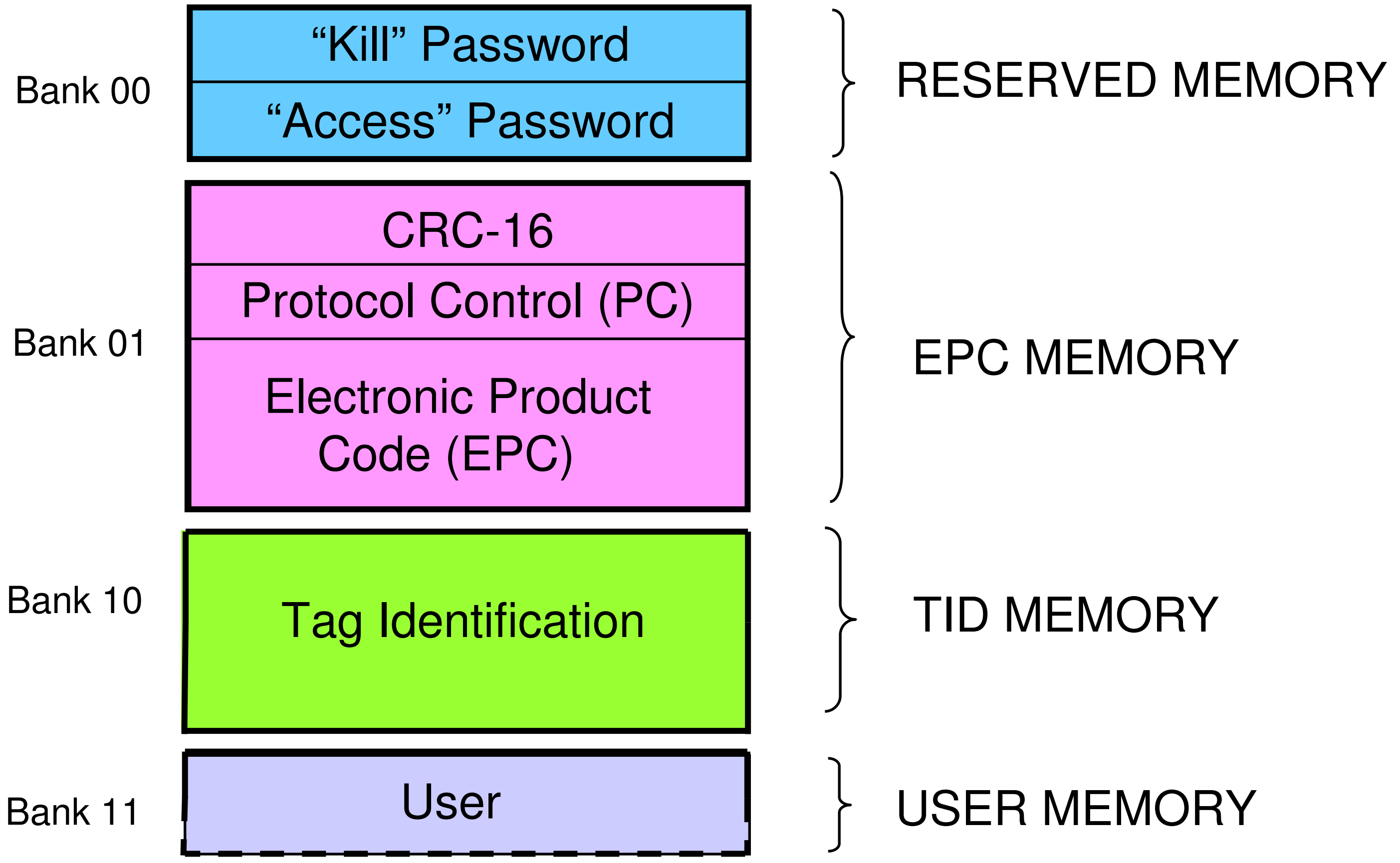
- The Tag uses either ASK or PSK modulation to return its data:

  (Miller encoding shown in example)



# UHF Gen 2 Memory

## Tags can have 4 banks of non-volatile memory



# UHF Gen 2 Memory

## Reserved Memory

- This area of memory holds the tag's passwords:
  - A 32-bit "Kill" password that allows a Tag to be permanently silenced.
    - The default Kill password value is zero
    - The *Kill* command will only execute if the password has been set, i.e. is non-zero
  - A 32-bit "Access" password that allows the Tag to transition to the *Secured* state
    - A Tag in the *Secured* state can execute all *Access* commands, including writing to locked blocks.
- Reserved memory can be read-locked.

# UHF Gen 2 Memory

## EPC Memory

- This memory area contains:
  - A 16-bit CRC calculated on the PC and EPC
    - The actual data is the 1's complement of the published CRC-16 definition.
  - A 16-bit Protocol Control (PC):
    - 5-bits giving the length of the PC + EPC
    - 2-bits RFU ($00_2$)
    - 9-bits for a Numbering System Id (NSI)
      - Which may contain an EPCglobal™ header
      - or an AFI as defined in ISO 15961
  - An EPC
    - The electronic product code of the object to which the Tag is attached

# UHF Gen 2 Memory

## TID Memory

- This area of memory contains:
  - An 8-bit ISO 15963 allocation class identifier
    - For EPCglobal Tags it is 0xE2
  - A 12-bit Tag mask-designer ID
  - A 12-bit Tag model number.
  - Manufacturers can also include other information if required e.g. Tag serial number

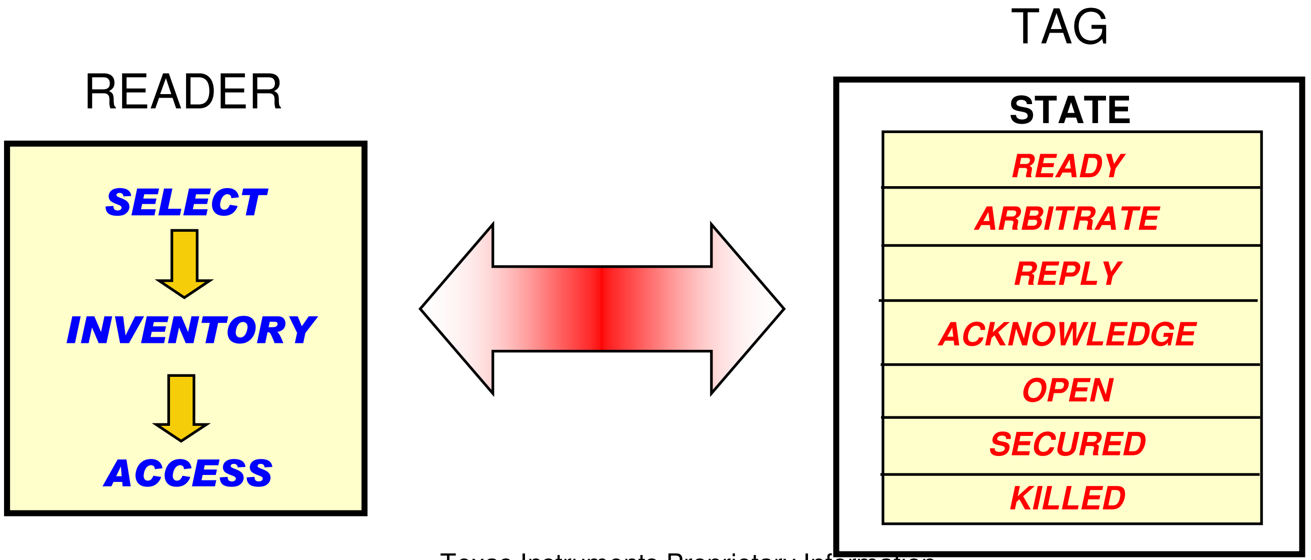
## User Memory

- This optional area of memory contains user specific data:
  - The memory organization is user defined.

# UHF Gen 2 Commands

## Three basic operations manage Tag populations

- ***Select*** is used to determine which groups of Tags will respond.
- ***Inventory*** is used to identify (singulate) individual Tags from a group
- ***Access*** is used once Tags have been singulated and individual commands can now be addressed to them



# Select Command

## *Tag Selection*

- All Tags support four sessions (**S0**, **S1**, **S2** and **S3**).
  - A session is the inventory process between the reader and a population of tags. The reader chooses one session and inventories the Tags associated with that session.
  - Two or more readers can use sessions, to independently inventory a common Tag population.
  - For each of the four possible time-interleaved inventory sessions, Tags maintain an independent **inventoried** flag to keep track of their status.
  - Each of the four **Inventoried** flags has two values (**A** or **B**).
  - Sessions take place in sequence **NOT** simultaneously

# Select Command

## *Select*

- This command allows the reader to select those Tags that will take part in the next **Inventory** round:
- Included in the ***Select*** command string are the following parameters:
  - **Target** ……. The ***SL*** or ***Inventoried*** flag to select and if ***Inventoried*** which of the four sessions [**S0, S1, S2** and **S3**] to choose
  - **Action** ……. How matching Tags set [e.g. A ←→B] the flags
  - **Mask** …….. A bit string that the Tag compares to a memory location
  - **MemBank** .. The memory bank that Mask refers too [EPC, TID, User]
  - **Pointer** ……. A memory start location for Mask
  - **Length** …… The number of bits of memory for Mask
  - **Truncate** …. Instructs Tag to return whole EPC or part following Mask
- If ‘Length’ is zero, all Tags are considered matching
- By building up multiple ***Select*** commands the reader can define the exact Tag population that is to take part in the **Inventory**

# Select Command

## *Select*

- Tags must maintain **inventoried** and **SL** flag values (persistence times) even when power is lost, as shown in the table below:

| Flag | Tag energised | Tag not energised |
|---|---|---|
| S0 **inventoried** flag | indefinite | none |
| S1 **inventoried** flag | 500ms < persistence < 5s | 500ms <persistence < 5s |
| S2 **inventoried** flag | indefinite | 2s < persistence |
| S3 **inventoried** flag | indefinite | 2s < persistence |
| Selected (**SL**) flag | indefinite | 2s < persistence |

- A reader can choose to inventory Tags with **SL** set or not set (**SL** or **~SL**) or ignore it.
- A ***Select*** command uses Frame-Synch
- Tags don't reply to ***Select*** commands

# Inventory Commands

## *Inventory*

- The **Inventory** process uses a slotted random anti-collision algorithm to determine which Tags are present and its command set includes *Query*, *QueryAdjust*, *QueryRep*, *Ack* and *Nak*.
  - *Query* is used to select Tags for the interrogation process and contains a slot-counter value (Q = 0 to 15)
  - *QueryAdjust* is used to decrement the Tag's slot-counter without changing any other parameters.
  - *QueryRep* is used to repeat the last *Query*. This is shorter (quicker) that issuing another complete *Query* command.
  - *Ack* is used to acknowledge a Tag response.
  - *Nak* is used to force a change of state back to *Arbitrate*

# Inventory Commands

## *Query*

- This **Inventory** command specifies and initiates the singulation process.  It has the following parameters.
  - **DR** ….. (Divide Ratio) Sets the Tag to Reader frequency
  - **M** ……. Sets the Tag to Reader data rate and modulation format.
  - **TRext**  . Determines whether or not, the Tag send the 'pilot tone'
  - **Sel** …..  Chooses which Tags respond to the *Query*.
- This command must send the encoding preamble, subsequent commands (***QueryAdjust, QueryRep, Ack, Nak)*** use the frame delimiter

# Inventory Commands

## The Tag is a State machine.

- Once in an RF field, it will change to the ***Ready*** state, and on receiving a ***Query*** command will:
- Verify it is in the selected group and if so, roll a $2^Q$ -1 sided die.

  (Q is an integer in the range 0 ~ 15 passed with ***Query***)

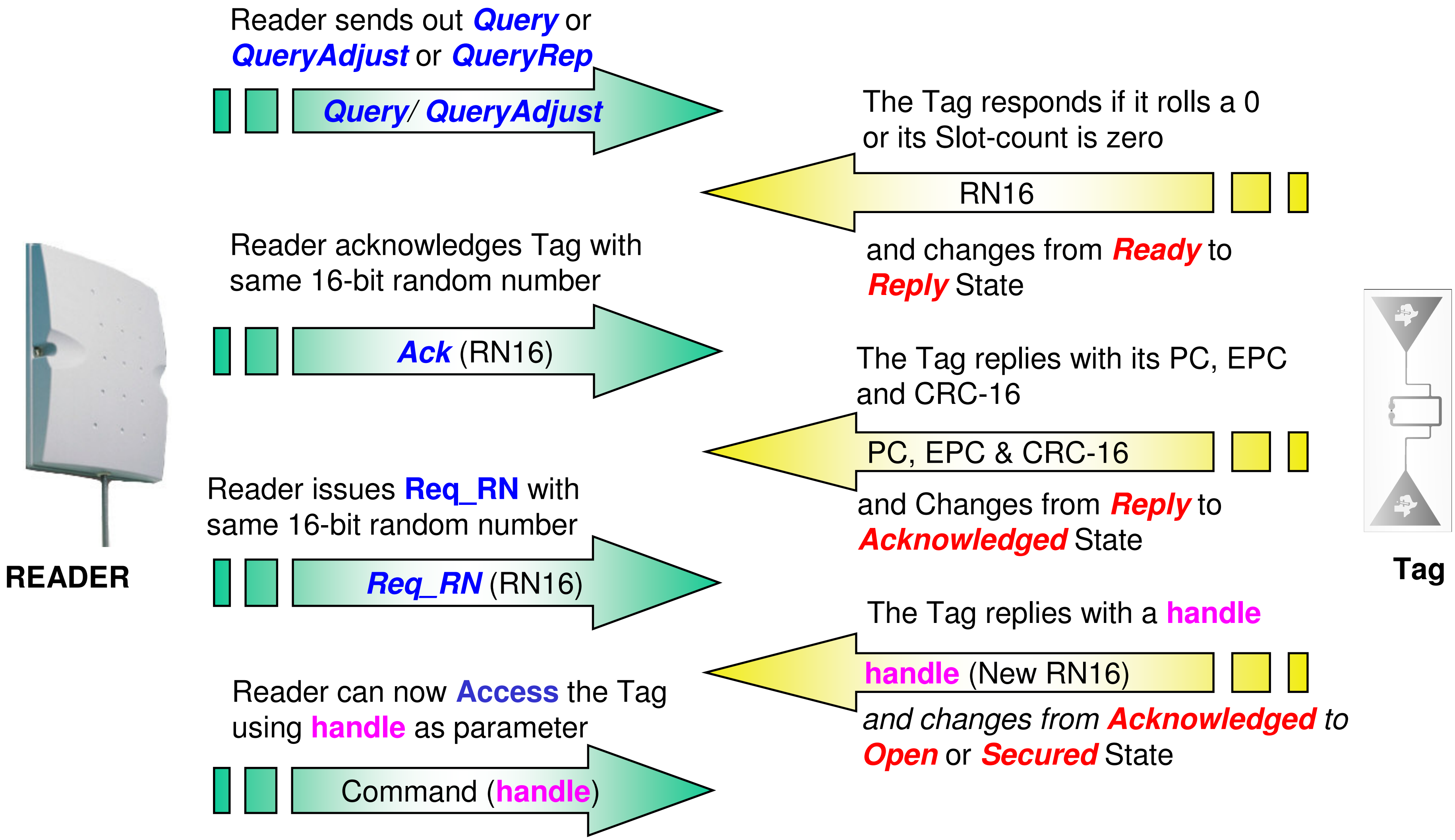
- If a ‘0’ is rolled, the Tag will immediately transition to the ***Reply*** state, backscattering a 16-bit (RN16) random number.
- The reader acknowledges with an ***Ack*** (containing the same 16-bit random number).
- This Tag now changes state to ***Acknowledged*** and backscatters its PC, EPC and the 16-bit CRC.
- A reader now sends a ***QueryAdjust*** causing the identified Tag to invert its **Inventoried** flag ( A➔ B, or B🡐A) and to transition to ***Ready*** state.

# Inventory Commands

  - If a non-zero value is rolled, the Tag will store that number in its slot-counter and change its state to *Arbitrate* and await further commands

- If more than one Tag responds, unless the reader can resolve the collision and send a valid *Ack*, each Tag will return to *Arbitrate.* These un-acknowledged Tags will again roll their dice and store the result.
- The reader now issues a *QueryAdjust* command which causes each unresolved Tag to decrement its slot counter
- Tags will reply when their slot counters get to zero
- At any point the reader can issue a *Nak* which forces all Tags back to *Arbitrate*.

# Inventory Command



Reader sends out ***Query*** or ***QueryAdjust*** or ***QueryRep***

***Query/ QueryAdjust***

The Tag responds if it rolls a 0 or its Slot-count is zero

RN16

and changes from ***Ready*** to ***Reply*** State

Reader acknowledges Tag with same 16-bit random number

***Ack*** (RN16)

The Tag replies with its PC, EPC and CRC-16

PC, EPC & CRC-16

and Changes from ***Reply*** to ***Acknowledged*** State

Reader issues **Req_RN** with same 16-bit random number

***Req_RN*** (RN16)

The Tag replies with a **handle**

**handle** (New RN16)

*and changes from **Acknowledged** to*
***Open*** or ***Secured*** State

Reader can now **Access** the Tag using **handle** as parameter

Command (**handle**)

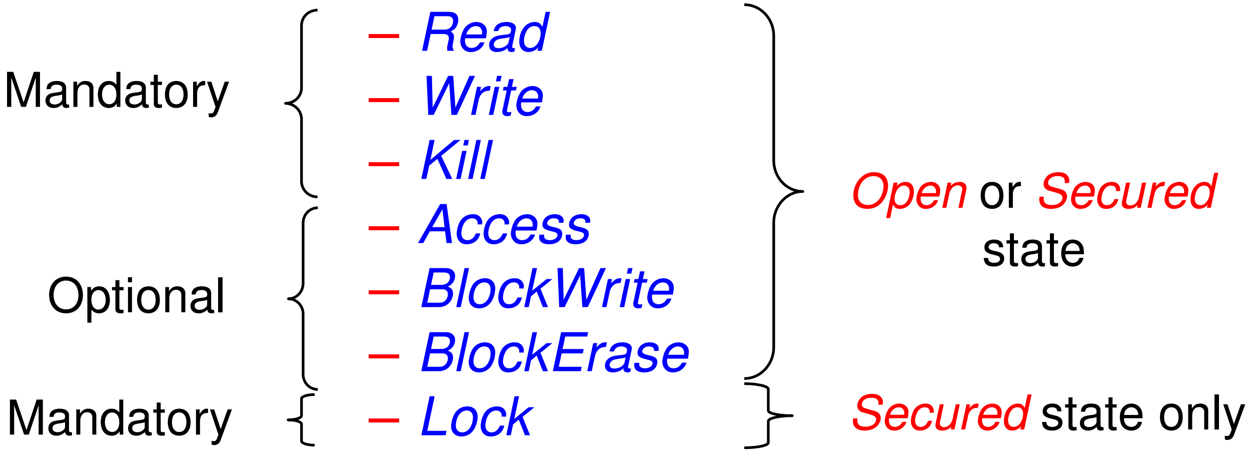
**READER**

**Tag**

# Access Commands

## *Access*

- Before Access commands can be used, a Req_RN (request random number) command is sent, to cause the Tag to transition from *Acknowledge* to *Open* (or *Secured* if its password is zero)
- The Tag will return a new authorizing random number (RN16) called the handle. The handle is a required part of the command string for the following *Access* commands.



Mandatory
- *Read*
- *Write*
- *Kill*

Optional
- *Access*
- *BlockWrite*
- *BlockErase*

*Open* or *Secured* state

Mandatory
- *Lock*

*Secured* state only

- Access commands *Write*, *Kill* and *Access* use encrypted data

# Access Commands

## *Read*

- This **Access** command allows the reader to access part or all of a Tag's Reserved, EPC, TID and User memory
- Included in the ***Read*** command string is the Tag's **handle** and:
  - MemBank ……. Memory type (Reserved, EPC, TID, User)
  - WordPtr ……… The starting address
  - WordCount ….. The number of words to read
  - CRC-16 ……… Checksum
- The Tag will indicate success, errors or failure (timeout) in its response

# Access Commands

## *Write*

- This **Access** command allows Tag memory locations to be changed. This command accesses Reserved, EPC, TID and User memory.

- As well as **handle**, included in the ***Write*** command string are
  - MemBank ….. Specifying the memory to access
  - WordPtr ……. The address to be accessed
  - Data ………… The 16-bit word to write
  - CRC-16 …….. Checksum

- A new **handle** has to be requested for each ***Write*** command

- Data is sent encrypted (link cover coding)

- The Tag's response will indicate success, error or failure

# Access Commands

## *Kill*

- This **Access** command will permanently disable a Tag.
- This is a multi-stage process; two ***Kill*** commands are sent:
  1. Containing the encrypted 16-MSBs of the kill password
  2. Containing the encrypted 16-LSBs of the kill password
- Before **each** ***Kill*** command a new **handle** is requested
- In response to the command, the Tag backscatters its **handle** and then never responds again.
- No response indicates the command failed
- If the ***Kill*** password is zero, the Tag cannot be 'Killed'

# Access Commands

## *Lock*

- This **Access** command allows a reader to:
  - Lock individual passwords, preventing subsequent reads or writes.
  - Lock individual memory banks, preventing subsequent writes.
  - Permalock (permanently lock) the lock status of passwords or memory banks
- Permalock bits, once set, cannot be changed
- The lock bits cannot be read directly but inferred by other memory operations
- The Tag will indicate success, error or failure (timeout)

# Access Commands

## *Lock ….*

- As well as ***handle***, the command has the following parameters:
  - A 20-bit **Payload** comprising **Mask** and **Action** bits
    - **MASK** …. Which memory areas to select
    - **Action** …. What action to perform on the selected memory

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| KILL | | ACCESS | | EPC | | TID | | USER | | KILL | | ACCESS | | EPC | | TID | | USER | |
| MASK-BITS | | | | | | | | | | ACTION-BITS | | | | | | | | | |

  - A CRC-16 checksum
- The Tag has to be in ***Secured*** state for the command to be accepted

# Access Commands

## Access

- This optional command will allow a reader to transition a Tag with a non-zero access password, from an *Open* to a *Secured* state.
- This is a multi-stage process; two *Access* commands are sent:
  1. Containing the encrypted 16-MSBs of the access password
  2. Containing the encrypted 16-LSBs of the access password
- In response to the command, the Tag will indicate success, error or failure (timeout)

# Access Commands

## BlockWrite

- This optional command will allow a reader to write multiple blocks to a Tag's Reserved, EPC, TID or User memory.
- As well as handle, included in the BlockWrite command are
  - MemBank ….. Specifying the memory to access
  - WordPtr ……. The address to be accessed
  - WordCount … The number of 16-bit words to write
  - Data ………… The 16-bit words to write
  - CRC-16 …….. Checksum
- Data is **Not** sent encrypted
- The tag's response will indicate success, error or failure

# Access Commands

## BlockErase

- This optional command will allow a reader to erase multiple blocks to a Tag’s Reserved, EPC, TID or User memory.
- As well as handle, included in the BlockErase command are
  - MemBank ….. Specifying the memory to access
  - WordPtr ……. The address to be accessed
  - WordCount … The number of 16-bit words to erase
  - CRC-16 …….. Checksum
- The tag’s response will indicate success, error or failure

# Security Features

## Security

- A number of features work together to enhance the security of EPC Gen 2 tags
  - An Access password, is required before the tag can be transitioned to the *Secured* State.  Only in this state can the *Lock* command be activated
  - The *Lock* command allows passwords and data to be Read and Re-Write protected
  - Link Cover-coding is used to scramble the data passed to the tag with the *Kill*, *Write* and *Access* Commands.