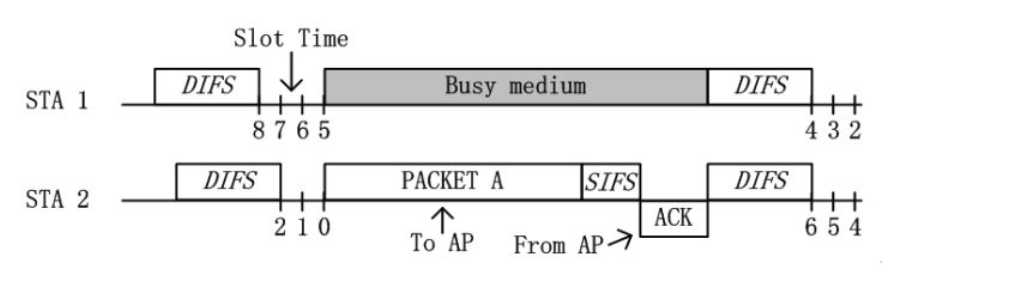
DCF模式以及其核心CSMA/CA机制

**CSMA/CA机制**

所以需要设置一种随机接入机制，以避免多个节点同时访问网络所带来的冲突问题，在WiFi协议中，该随机接入机制即是CSMA/CA。CSMA/CA的全称是Carrier Sense Multiple Access with Collision Avoidance，即载波侦听多路访问／冲突避免。如果熟悉有线网络的可以知道，在集线器与中继器中也会采用一种CSMA/CD的机制。



1. 当STA 1与STA 2相继存在数据，需要在竞争信道进行发送时，其首先需要 **"等待"** DIFS时间，若DIFS时间内，信道保持空闲状态，那么就可以进行backoff过程。
2. 若STA 1与STA 2进入backoff过程时，其首先需要从竞争窗口（Contention window）选择一个随机数，在802.11协议中，默认的初始竞争窗口为31，即随机回退计数值的范围即是[0,31]。在上图中，STA 1则是选择了8，而STA 2选择了2。
3. 在backoff过程中，每经过一个slot time，节点会 **"监听"** 一次信道，若信道空闲，则相应的随机回退计数器的值减1。如上图中，经过3个slot time后，STA 1的随机倒数计数器从8递减至5，而STA 2相应从2递减至0。
4. 当节点的随机倒数计数器倒数至0时，节点竞争获得信道，从而可以发送数据。如上图，STA 2获得信道后，发送PACKET A给AP。在AP接收到数据后，会采用CRC机制对数据进行校验，若校验通过，AP会在SIFS后，反馈ACK确认帧。
5. 当STA 2成功发送完数据， "**等待**" 了SIFS的时间之后，AP会向节点反馈ACK确认帧。当STA 2成功接收到ACK帧之后，这一次传输完成。
6. 当这一次传输完成后，节点需要再次 **"等待"** DIFS的时间后，重新开始backoff过程。若节点刚刚发送完数据，那么在backoff过程开始时，需要重新从竞争窗口中选择一个随机数进行倒数。若节点没有发送数据，那么直接从上一次的倒数结果继续倒数。如上图中，STA 1没有竞争到信道，那么其在第二次的backoff过程中，直接基于上次的5直接进行倒数至4。这样的设计目的是为了保证网络传输的公平性。

## BEB机制

而在正式进入下一次竞争之前，节点需要对竞争窗口（CW）采用BEB机制，按我们之前所述，在初始竞争时，节点的默认CW范围是[0,31]（假设初始窗口是802.11b机制下，即最大31，在802.11a这种，初始窗口就是15）。而如果在节点数较多的情况下，那么就有可能引发之前我们所述的冲突问题，从而我们需要扩大竞争窗口CW。具体在CSMA/CA中，我们则是采用二进制指数退避的方法对竞争窗口CW进行扩展，即发生一次冲突后，那么CW范围就会从[0,31]变化到[0,63]，如图中，在冲突之后，STA 1重新随机选择50，STA 2重新随机选择32。在802.11中，一共允许回退6次，第7次不倍增窗口，再次尝试重发，若再次失败，则丢包。

## RTS/CTS模式

* RTS：Request To Send，即请求发送。RTS帧是一个单播帧，没有加密，其duration字段中填充包含后续发送过程中总体所需要时间。
* CTS：Clear To Send，即信道清除帧。节点在收到CTS后，确认信道是空闲的，可以发送。CTS也是一个单播帧，没有加密，其duration字段包含除去RTS以及一个SIFS后，发送过程总体所需要时间。

## CSMA

CSMA实际上是源于aloha协议。一个aloha节点只要有数据的话，该节点就可以立即发送。当该节点数据发送完之后，其需要等待接收方反馈的ACK。若成功接收到ACK之后，那么这一次传输成功。如果没有收到ACK的话，那么这一次传输失败。该aloha节点会认为网络中还存在另外一个aloha节点也在发送数据，所以造成接收方发生了冲突。最后这些冲突的节点会随机选择一个时间进行回退（backoff），以避免下一次冲突。若冲突节点回退完成，其才可以重新进行发送。CSMA协议对其最大的改进即是引入了LBT机制（Listen Before Talk），在CSMA中的CS（Carrier Sense）即是指Listen监听机制。在LBT机制下CSMA的思想就是：“CSMA节点在每一次发送之前先监听信道是否是空闲的，如果信道不是空闲的话，那么就不发送数据，等待一会再进行尝试。只有确保是空闲的情况下，才可以发送数据，从而避免打断其他节点正在进行的传输过程"。

具体CSMA的工作机制关联着我们之前所叙述的三种CSMA模式，以下我们分别进行叙述：

1-persistentes CSMA：“节点需要持续监听信道，一旦节点发现信道空闲后，则立刻发送数据。”。

0-persistentes CSMA：“节点不连续监听信道，若该时刻节点监听信道为busy，那么等待一段时间后，再次进行监听。若节点该时刻监听信道为空闲，则立刻发送数据。”

p-persistentes CSMA：“节点需要持续监听信道，一旦发现信道空闲后，节点以p的概率立刻发送数据，以1-p的概率不发送数据。若节点该时刻不发送数据，那么等待一段时间后，再次进行监听，并以p概率再次发送”。（注：这里所述的p概率可以理解成抛骰子赌大小，如果抛大，那么就发送，反之不发送。其中抛大的概率就是p，而抛小的概率就是1-p）

## CSMA/CD

节点发送数据之前需要持续监听信道，一旦节点发现信道空闲，则立刻发送数据。在发送数据的同时，节点持续监听信道，"探测" 是否有别的节点也在该时刻发送数据。

若传输过程中没有检测到别的节点的传输，那么成功传输。在成功传输后，节点需要等待帧间间隔IFG（interframe gap）时间后，可以进行下一次传输。

若在传输过程中，探测到别的节点也在传输，那么则检测到冲突。发生冲突后，节点立刻停止当前的传输，并且发送特定的干扰序列（JAM序列），用以加强该次冲突（用以保证其余所有节点都检测到该次冲突），在JAM序列发送完之后，节点随机选择一个时间倒数进行backoff。当backoff完成之后，节点可以尝试再次重传”。