the SCO slots in the two piconets do not overlap, but as the clocks of the two piconets' Masters drift, the two SCO links will move until they overlap one another.

## 5.7 MASTER / SLAVE ROLE SWITCHING

Bluetooth allows any device to request a switch in roles with respect to another device it is communicating with. For example, a Master in an existing piconet might allow itself to be paged and connected to a new device and then switch between Slave/Master (temporarily imposed by the paging procedure) and Master/Slave to integrate the new Slave into its piconet. This is accomplished with a Master/Slave switch and is particularly useful in situations where a connection has just been established by a device which normally wishes to be a Slave, such as where a mobile computing device enters a piconet controlled by a LAN access point.

The mechanism essentially involves the slave sending its FHS packet to the Master; the Master takes on a CLK offset to match the Slave's CLKN, while the Slave switches to using its own CLKN, and each device swaps access codes. The new Master also sends an LMP message, which contains the lower part of the Bluetooth CLK not contained in the FHS together with the sub-slot offset information in μs to allow the new Slave to fully synchronise its timing.

### 5.7.1 Messaging

Figure 5–13 shows the sequence of messages exchanged when a Slave becomes a Master by initiating a Master / Slave switch. The Master side can also request the role switch.

In version 1.0B of the Bluetooth core specification, there is a contradiction in the description of this process in the baseband and link manager sections. The Slave must give the Master detailed information on its clock, so that the Master can move onto the Slave's timing. An LMP_slot_offset message is used by the Slave to pass this information to the Master. The version 1.0B LMP section specifies that a Slave requesting a Master/Slave switch will send the message before requesting the switch; the version 1.0B baseband section specifies that the message will be sent later, after both sides have agreed to perform the Master/Slave switch.

The baseband also implies that a Master which is becoming a Slave could hand over its old Slaves to the new Master; however, LMP does not provide any messages to tell the new Master the active member addresses of the old Slaves, or to pass on information about Slaves in Hold, Park or Sniff modes. For the new Master to attempt to acquire the Slaves of the old Master, it would have to try polling all seven active member addresses using the old Master's timings, and see if any respond. However, if they do respond, there is no mechanism defined to move them straight onto the new Master's timing and frequency hop sequence. The only way it could be done would be via two Master/Slave switches. So the new Master would become a Slave again, then switch roles to acquire the old Master's Slave as its Slave. It would probably be simpler to let supervision timeouts elapse, then inquire and page to connect with the old Master's Slaves.

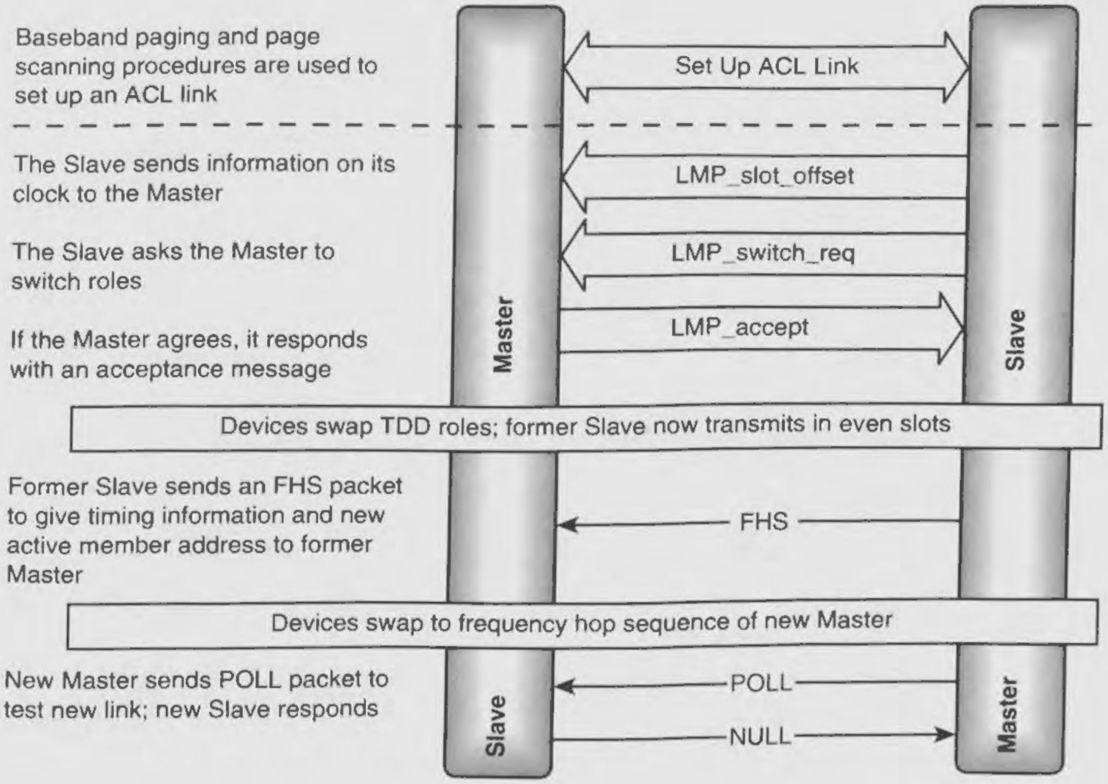


**Figure 5–13** Message sequence chart for Master / Slave switch.

In practice, the acquisition of Slaves from a Master is unlikely to be a problem, as the main use for a Master/Slave switch is to allow a device to join a piconet quickly by paging, then hand control of piconet back to the former Master of the piconet.

### 5.7.2 Uniting Scatternets with Role Switch

The devices linking a scatternet are present on more than one piconet and have to time share, spending a few slots on one piconet and a few slots on the other. Each device has its own independent clock, and when devices join a piconet, they track the timing of the piconet's Master by keeping track of the offset between their clock and the Master's clock. This means that when devices are present on more than one piconet, they will have to track two sets of timings. When switching between timings, there will be some slots which cannot be used (for a fuller description of scatternet timings and how to manage them, refer to Chapter 19).

Sometimes it is desirable to have a device join a piconet as a Master as shown in Figure 5–14. Consider a LAN Access Point (LAP). It does not know which devices in the area wish to connect, and it would be wasteful of its resources to constantly poll devices

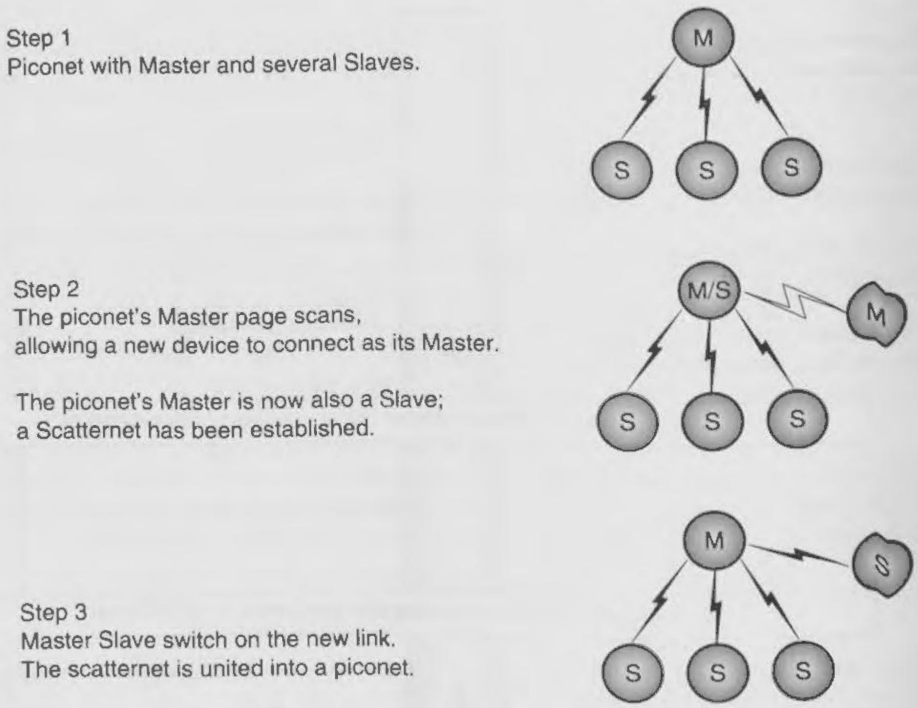


**Figure 5–14** Forming a scatternet and uniting it into a piconet.

to try to connect to them. Instead, it periodically page scans, and any devices wishing to connect page it. This means that connecting devices become Masters of a small piconet containing just themselves and the LAN access point.

If the situation were left like this, the LAN access point would lose control. It must be the Master of its links so that it can control the allocation of bandwidth to the devices connected to it. So, the LAN access point requests a Master / Slave switch as it accepts the connection. The new joiner accepts the switch, and the LAN access point is restored to working in a piconet rather than a scatternet, as shown in Figure 5–14.

## 5.8 LOW-POWER OPERATION

During Standby, Park, Hold, Sniff, or even between an active transmit or receive operation[2], a device may enter low-power operation, where any protocol or bit processing elements (hardware or software) may be turned off. All system clocks may be disabled, and

[2]When another device's packet header transmission has been received, indicating a multi-slot packet but with a different AM address, and thus is not directed at the present device.

the device may enter a very low power consumption mode of operation. Certain static data must be maintained, such as LC protocol state and buffer contents, and the only dynamic information which must be maintained is the native clock counter, CLKN. To conserve power, this may be clocked with a much lower power, and therefore less accurate 32-KHz oscillator (÷10 = 3.2 kHz). The tolerance specified in the standard on the Low-Power Oscillator (LPO) is ± 250 ppm.

An accuracy of 250 ppm gives rise to a worst case slippage of 1 slot every 2.5s. This is why regular resynchronisation is important and is explained further in the later section on low-power operation. By way of illustration, the maximum duration for which a device is allowed to remain inactive in Sniff or Hold mode, or between synchronising beacon instants in Park mode, is 40.9s. This equates to 65440 slots, which requires an uncertainty window of ± 17 slots.

It is worth noting that 32kHz crystals are not at present commonplace, unlike the 32.768kHz crystals commonly used in wristwatches and the like. The tolerance of a quartz crystal does not allow "pulling" over such a large distance, and so we must wait for the commercial success of Bluetooth to create a large demand for 32kHz parts to force the price of such components down.

## 5.9 BASEBAND / LINK CONTROLLER ARCHITECTURAL OVERVIEW

In this short section, we will tie together the material in both the previous chapter and this chapter to examine the overall architecture of a typical Link Controller / Baseband system.

Figure 5–15 shows a possible baseband/Link Controller system. The data path is either SCO (via a direct PCM interface, through HCI) processed by the audio CODEC subsystem, or ACL via HCI. The data is buffered, so it may be read out at system speed subsequently following reception, or stored awaiting transmission. Typically, double buffering is used to ease the scheduling of these operations. Indeed, double buffering on transmit is almost essential for a multi-link device where retransmissions must be anticipated.

The data path has already been discussed; it encodes or decodes data bursts during Tx or Rx, respectively. The Rx correlator effectively "sniffs" the received data, and when enabled, will search for the required access code. The sync word generator supplies a valid sync word derived from the appropriate LAP to the radio interface and the correlator as appropriate. The timebase produces a native clock: CLKN from the appropriate system reference clock, which must therefore be accurate to ± 20 ppm. An offset control function then maintains and applies the necessary offsets to produce CLKN, CLK, and CLKE as required.

The hop selection function combines the required CLK and BD address parts to produce the channel number and feeds these to the radio interface. Finally, the encryption key generator produces and stores keys which are then loaded up by the key stream generator and processed at symbol rate to produce a cipher stream for use by the bitstream data path.