computers are bad

You are receiving this facsimile because you signed up for fax delivery of this newsletter. To stop delivery, contact Computers Are Bad by email or fax.

https://computer.rip - me@computer.rip - fax: +1 (505) 926-5492

2023-01-02 ANT plus is PAN for ants

One of the most interesting areas of modern consumer technology, to me, are low-power, low-range wireless networks. This category of network protocols were historically referred to as Personal Area Networks, or PANs. I say "historically" here because the term has never really had any use among consumers---I wager very few people on the street would be able to name a single PAN protocol, even though most people use one of them regularly. Part of the reason for this state of affairs is, of course, that there is only one PAN protocol that has achieved any real degree of success: Bluetooth.

Bluetooth exists in three major variants that are basically completely separate protocols, although all operate in the same band and are commonly implemented together in the same radio hardware. Bluetooth Classic is the evolution of the original Bluetooth protocol we have all known and loved whated since the early '00s. Bluetooth High-Speed is a bit of a cheat, where Bluetooth Classic is used only for the initial negotiation, and then an 802.11 network substantially similar to WiFi Direct is used for the actual data transfer.

This might initially seem like a surprising degree of added complexity, but since Bluetooth and WiFi operate in the same 2.4GHz band and are usually needed in the same devices, it was already extremely common for the same chip to implement both protocols. When a need emerged for a higher-speed Bluetooth connection, it made sense to simply exploit that fact rather than reinventing the wheel. Bluetooth just takes advantage of all the work already done for high-speed 802.11, and the Bluetooth standards constrain this 802.11 pseudo-bluetooth to a degree that connection setup is usually pretty reliable. Much of the disappearance of WiFi Direct can be attributed to the fact that Bluetooth HS is basically WiFi Direct anyway, and had the backing of the large set of Bluetooth device vendors.

The third variant of Bluetooth, and the more interesting one for our purposes, is Bluetooth Low Energy or BLE. BLE was not originally Bluetooth at all, it was developed as a completely separate protocol called Wibree. Amusingly, Wibree itself was originally loosely based on Bluetooth and is nearly as old as Bluetooth itself. BLE could be viewed as a case of a spin-off R&D effort that was successful enough to be acquired back into the parent organization, but the process happened through industry associations and standards bodies and was thus a great deal more politically complicated.

The major difference between Bluetooth Classic and BLE is, as the name suggests, power consumption. BLE uses a simplified association model, lightweight network protocol, and an almost MIB-like (management information base, from the ISO protocols) approach to the application layer that minimizes the need for negotiation or setup processes. This all makes BLE amenable to applications where a device wakes up and participates in the radio

1

network only occasionally and for a short period of time, which is great for battery-powered devices.

If you look at the table of BLE device profiles, you will notice a heavy slant in the direction of fitness trackers. Fitness trackers were one of the original motivators of low-power, low-bandwidth wireless connectivity. One of the early vanguards of this type of application was Nike+, launched in 2010. Nike+ consisted of a small module installed in a shoe, and then a receiver that could either be a standalone device or---most famously---an iPod with an accessory receiver. Nike+ used a simple proprietary protocol that, while its own unique branch of protocol design, is substantially similar to later fitness tracking approaches. It's possible that there's a direct relationship between Nike+ and later work like BLE, but I think it's more likely this is a case of convergent evolution. The requirements for a fitness-tracking wireless protocol are fairly straightforward.

Most fitness trackers track a small number of values and then report those at an interval. For an example that I am more familiar with, consider a bicycle. The most basic solutions measure only the RPM of the wheel, usually based on counting pulses from a hall-effect sensor. Somewhat more advanced products like Bontrager DuoTrap measure two pulse rates, the wheel and the crank. A more advanced product might also measure crank torque, getting you to three values, but in general the number of independent values involved in most fitness trackers is small and often only one. Heart rate sensors, for example, are a common class of low-energy, low-cost devices that report one value at a regular interval.

Typically the reporting interval is fairly low, with once a second being very common. The packet sizes are small since there are perhaps a few two-byte counters to report on the high end. In the interests of power saving and implementation simplicity, there's rarely a need for a connection-oriented protocol. As long as the tracking device has a unique ID that allows a receiver to differentiate it from other devices that might be nearby, there's not really even a need for two-way communications. The tracker can just broadcast the measured value each second. More complex receivers might have the user choose from a list of detected devices, but in most cases it's sufficient to use simple signal strength metrics to select the nearest device.

This all sounds delightfully simple, doesn't it? And like it could be far easier to interact with than the modern world of Bluetooth. Well...

WiBree, which became BLE, had a competitor. And a competitor that came directly from the fitness tracking industry, rather than from the broader alliance of mobile device vendors behind Bluetooth (one that has a huge interest in high-bandwidth audio as a primary use case). In 2003, Dynastream Innovations launched ANT. A few years later, Dynastream was acquired by Garmin Canada, giving ANT the imprimatur of a top fitness tracking brand. By 2010 or so ANT, in its 2004 variant ANT+, had become the most ubiquitous wireless protocol in commercial fitness equipment, and today we have basically all forgotten that it exists.

Let's talk a bit about the design of ANT+. It has general similarity and is partially a direct competitor to BLE, so it's probably easiest to discuss it in contrast to BLE. ANT+ is very slow, with perhaps 10 kbps as a typical achievable data rate (this depends somewhat on the network configuration in use). Unlike BLE (prior to 5.0 which appears to have gained a more sophisticated approach), ANT+ devices actively monitor channel usage in order to schedule an optimal time to transmit their payload. This allows a significantly larger number of ANT+ devices to operate in proximity than BLE.

2

ANT+ radios have extremely low power consumption, enabling ANT+ sensors to run off of a coin cell for an extended period, although BLE now achieves similar performance. ANT+ usually operates in a pure broadcast mode, but it does support reliable delivery when required.

The great thing about ANT+ though is the great simplicity of implementation compared to the Bluetooth stack. This isn't necessarily an inherent problem with BLE, but has more to do with the fact that BLE "sensor receivers" are usually devices with complete WiFi and Bluetooth stacks and so the degree of complexity that comes with managing multiple distinct Bluetooth modes. This isn't necessarily to say that BLE has significant usability problems on modern smartphones, it mostly seems to work very smoothly on modern devices besides Android's OS-level issues about whether or not BLE devices are associated as in a normal pairing relationship.

But the situation continues to irritate me. BLE is still saddled with a long history of Bluetooth standards with a high degree of diversity and user expectation of backwards compatibility. BLE is managed through complex operating system interfaces that are saddled with ideas from other Bluetooth variants and the history of Bluetooth on that platform. ANT+, as a protocol that is both new and very simple, manages to shed all of this complexity.

And that's why ANT+ remains popular today in high-turnover embedded applications, namely commercial fitness equipment. A lot of the cardio equipment you'll find in a modern gym has ANT+ built-in and will detect an ANT+ heart monitor in close proximity and use it instead of the integrated galvanometric heart monitor (the deal with the metal handgrips that tries to suss out electrical resistance changes resulting from your heartbeat from those resulting from the uneven contact of your sweaty hands and body thetans or whatever). This operation is completely association-free, rather the exercise equipment uses RSSI (signal strength) measurement to determine when such a tracker is reasonably close.

This surprisingly widespread support in fitness equipment is a victory for ANT+, but doesn't mean that much without larger consumer adoption. Anecdotally it seems like the number of people interested in purchasing an ANT+ heartrate monitor just for use at the gym isn't that large. So what of ANT+ in consumer devices? Well, it's kind of happened, almost...

Some Android smartphones have shipped with an integrated ANT+ radio, mostly from Samsung, which included ANT+ in pretty much all of their non-budget smartphones for over a decade. Unfortunately, it appears that the Galaxy S20 was the last in the series to feature ANT+. Across other vendors there's about 20 more models with ANT+ support, but all of them older. BLE has pretty much completely eliminated interest in ANT+ from the smartphone industry.

It's still pretty typical for cyclocomputers to have ANT+ support, and in general ANT+ has some ongoing adoption in the athletic industry, but it seems like the wind has gone out of its sails. Devices without ANT+ support can be made to work with a USB ANT+ adapter available fairly inexpensively from Garmin, but dongles have never really been popular with smartphones. As a result, a huge portion of ANT+ devices on the market simply also integrate BLE support... paving the way for BLE to replace ANT+ even for most people's existing equipment. The only thing that seems likely to remain a holdout for ANT+ is commercial fitness equipment in gyms, due to the more complex association process for BLE, but progress marches on and it seems likely that models with BLE support will appear nonetheless. Now that I've lamented the death of ANT+, let's dig into the details a bit.

Like many short-range digital radio protocols, ANT+ is based on GFSK or Gaussian frequency shift keying. GFSK is essentially a variant of FSK that uses a Gaussian transform to "soften" the edges of the FSK symbols---which has the result of reducing the effective bandwidth of the signal, making more efficient use of spectrum. ANT radios use a very low power of perhaps 1mw, since they are intended for use only at a range of a few meters in most cases.

The ANT+ logical protocol is proprietary, but the interface to ANT+ chips is openly documented. ANT+ networks consist of a master node and one or more slave nodes [1], but these roles are not necessarily what you would expect. Since the master device establishes channel parameters and many ANT+ devices are low-power sensors that operate in a simple broadcast mode, it is not unusual for the sensor to act as master and the receiver (such as a smartphone) as slave.

There are a wide variety of configuration packets used to set up data channels, but once a data channel is established, but to facilitate simple passive devices a sensor can establish a data channel in broadcast mode by using a set of default parameters (e.g. "network ID 0"). Alternately, devices wishing to communicate with reliable delivery can exchange configuration packets to establish a channel in acknowledgment mode, which will have unique identifying values so that the channel can be uniquely tracked. A typical packet has a payload of 8 bytes, but the special bulk transfer mode (used mostly for transferring firmware updates) can use larger packets to increase the effective bandwidth. Overall the protocol is reasonably simple, and the specification PDF is openly available and only 134 pages (more than half of which is appendices).

To implement collision avoidance and thus allow many devices to coexist in close proximity, ANT+ operates on a cycle defined by the "channel period," which can have various values depending on the type of device in use. ANT+ devices monitor the RF channel in use for other packets, and if they observe packets from unrelated devices they will "phase shift" to transmit later in the channel period instead. For devices of like type, a very large number of devices can thus operate in the same channel and physical space without any collisions (once the avoidance process has "settled").

And that's some useless knowledge about a dying protocol.

[1] I use here the terms found in the ANT+ specification document.