RF69 Native Format Driver

This driver is a patched version of the JeeLabs rf69.h driver. The changes are:

1. The default transmit power has been lowered to +7 dBm in order to avoid damage to the radio module in the event that the transmitter ip operated without an efficient antenna.
2. The point where the RSSI value is extracted has been moved because in the original position, the RSSI value of the previous message could occasionally be reported.

API

void init (uint8_t id, uint8_t group, int freq);
Initialises the driver with the NodeID, Group & Frequency. Node ID range is 1 – 60, 61 is send-only, 62 is reserved, 63 is receive-all. The OEM default group is 210. Frequency is in MHz and can have an arbitrary precision, e.g. 8686 = 868.6 MHz. There is no return value.

int receive (void* ptr, int len);
Returns the actual size of the received packet (0 – 62 bytes). The caller-supplied receive buffer pointed to by `ptr` (of size `len`) must be large enough to hold the entire packet.

void send (uint8_t header, const void* ptr, int len);
Transmit the data of length `len` in the buffer pointed to by `ptr`. The data must not exceed 62 bytes. Header is not normally required and should be zero. There is no return value.

void txPower (uint8_t level);
Sets the transmit power level. `Level` ranges from 0 – 31 representing -18 dBm to + 13 dBm in steps of 1 dBm. The default is 25 (+7 dBm) to avoid damage to the RFM69 if no antenna is fitted. There is no return value.

uint8_t rssi;
The received signal strength of the latest message, in steps of 0.5 dB below the reference level. i.e. to convert to the normal representation:

```
rfInfo.rssi = -rf.rssi/2;
```

void encrypt (const char* key);
Encrypts the data with a 16 character (max) / 128-bit key. If encryption is used, it must be used with the same key in all nodes in the same group. Encryption can be disabled with a null key:

```
rf.encrypt(0);
```

void sleep ();
Puts the radio module into sleep mode. There is no return value.

Some other variables are available – refer to the source code for details.