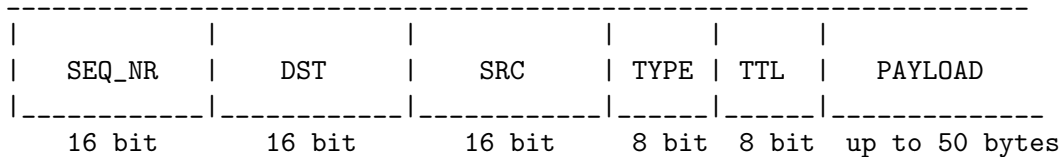

PWSN-Protokol Version 1.6

1. Paketformat



- *SEQ_NR* - Beginnt bei 0 und wird bei jedem versandtem Paket hochgezählt.
- *DST* - Die Routing-Zieladresse
- *SRC* - Die Routing-Absenderadresse
- *TYPE* - Der Pakettyp
- *TTL* - Time to live (beträgt 5 sofern nicht anders angegeben.)

2. Pakettypen

- 1 - PWSN_HELLO
Macht den Knoten in seiner Nachbarschaft bekannt. (Per Broadcast)
Enthält einen Wert vom Type `double energy`, der die verbleibende Energie angibt, einen 8 Bit-Wert `nb.size` und einen Puffer `neighbours`, der aus 16 Bit-Werten besteht. `neighbours` enthält dabei die Liste alle Routing-Adressen, die der Knoten direkt (in einem Hop) erreichen kann, `nb.size` gibt die Anzahl dieser Nachbarn an (max. 16).
Die Routing-Zieladresse und das TTL-Feld sind bei diesem Paket immer 0.
- 2 - PWSN_CHAT
Verschickt Textnachrichten an andere Knoten.
Enthält einen 8 Bit-Wert `len` und einen String `mesg`. `len` gibt die Länge von `mesg` inkl. dem Nullbyte an.
- 3 - PWSN_DATA_REQ
Fordert Daten von einem anderen Knoten an.
Enthält einen 16 Bit-Wert `id` und einen 8 Bit-Wert `type`. `id` wird bei jeder Anfrage hochgezählt, `type` spezifiziert die Anfrage wie folgt:
 - 1 - Temperatur
 - 2 - Relative Luftfeuchte
 - 3 - Verbleibende Energie
- 4 - PWSN_DATA_RESP
Beantwortet Datenanfragen anderer Knoten.
Enthält einen 16 Bit-Wert `id`, einen 8 Bit-Wert `data.len` und ein generisches Array `data.buf`. `id` gibt an auf welche Anfrage geantwortet wird, `data.len` ist die Länge der Daten und `data.buf` enthält die Daten selbst.
- 5 - PWSN_SERVICE_REQ
Fragt verfügbare Dienste von einem anderen Knoten ab.
Der Payload bleibt leer.

-
- 6 - PWSN_SERVICE_RESP
Beantwortet Dienstanfrage anderer Knoten.
Enthält einen 16 Bit-Wert `id`. `id` ist ein Bitfeld mit allen auf dem Knoten verfügbaren Diensten. Dabei gibt es folgende Dienste:
 - Bit 1 - LED schalten
 - Bit 2 - Kaffee kochen
 - 7 - PWSN_SERVICE_SET
Benutzt einen Service.
Enthält einen 8 Bit-Wert `id` und einen 8 Bit-Wert `value`. `id` gibt den Service an, der benutzt werden soll. `value` ist der Wert der gesetzt werden soll. Für die LEDs geben die beiden niederwertigsten Bits den gewünschten Schaltwert für die grüne und rote LED an:
 - 00 - beide aus
 - 01 - grün aus, rot an
 - 10 - grün an, rot aus
 - 11 - beide an
 - 8 - PWSN_DHCP_DISCOVER
Sucht einen DHCP-Dienst-Anbieter.
Dieses Paket wird immer an die physische Broadcast-Adresse gesendet und enthält keine Nutzdaten.
Die Routing-Adressfelder müssen 0 sein. Das TTL-Feld ist 0.
 - 9 - PWSN_DHCP_OFFER
Bietet einem Knoten eine Routing-Adresse an.
Die Routing-Zieladresse sowie das TTL-Feld sind 0.
Das Paket enthält einen 16 Bit-Wert `xid`, einen 16 Bit-Wert `addr`, einen 16 Bit-Wert `subnet_mask`.
`xid` ist eine Zufallszahl, die die Transaktion eindeutig macht. `addr` enthält die neue Routing-Adresse, die dem Anfrager angeboten wird. Die Bitmaske `subnet_mask` bestimmt den Adressbereich, den der Anfrager nun selbst vergeben darf. Für diese Adressen gilt, dass alle gesetzten Bits der Maske unverändert bleiben müssen und von den übrigen Bits mindestens eins gesetzt sein muss.
 - 10 - PWSN_DHCP_REQ
Fragt eine zuvor angebotene Routing-Adresse an.
Dieses Paket wird als erstmalige Antwort auf das Adress-Angebot an die Broadcast-Adresse gesendet, zum Verlängern der Adresse wird es direkt an den DHCP-Server adressiert. Die Routing-Zieladresse ist 0, die Routing-Absender enthält die angefragte Adresse. Das TTL-Feld ist 0.
Das Paket enthält einen 16 Bit-Wert `xid`.
`xid` entspricht der Transaktions-ID, die im PWSN_DHCP_OFFER-Paket mitgeteilt wurde.
 - 11 - PWSN_DHCP_ACK
Bestätigt die Vergabe einer Routing-Adresse.
Das Paket enthält keine Nutzdaten. Das TTL-Feld ist 0.
 - 12 - PWSN_DHCP_NAK
Lehnt die Vergabe einer Routing-Adresse ab.
Das Paket enthält keine Nutzdaten. Das TTL-Feld ist 0.

-
- 13 - PWSN_DHCP_RELEASE
Gibt eine Adresse wieder frei. Dieses Paket wird immer an die Broadcast-Adresse (physikalische und Netzwerk) gesendet. Das TTL-Feld ist 0.
Es enthält den 16 Bit-Wert `net_addr` und den 8 Bit-Wert `phy_addr`.
`net_addr` benennt die Adresse, die freigegeben wird, `phy_addr` gibt die physikalische Adresse des vormaligen Besitzers dieser Adresse an.
 - 14 - PWSN_ROUTE_INFO
Enthält Routing-Status- und Kontrollinformationen.
Das Paket enthält einen 16 Bit-Wert `type`.
Dieser repräsentiert ein Bitfeld, wobei folgende Bits (beginnend bei dem niederwertigsten) vergeben sind:
 - 01 = DESTINATION UNREACHABLE

3. Ablauf

Ein Knoten sendet zunächst *DHCP_DISCOVER*-Pakete aus. Sobald er ein entsprechendes Angebot bekommen hat, fragt er die angebotene Adresse an. Wird diese Anfrage akzeptiert, wird der Anfrage selbst zum Server und beginnt zusätzlich er mit dem Verschicken von *HELLO*-Paketen. Eine Adresse muss innerhalb von fünf Minuten durch erneutes Aussenden eines *DHCP_REQ*-Pakets an den vergebenden Server verlängert werden. Bleibt eine entsprechende Anfrage aus, gibt der Server die Adresse wieder frei und gibt das via *DHCP_RELEASE* bekannt.

Jeder Knoten sendet dabei alle zehn Sekunden ein *HELLO*-Paket, solange er noch mehr als 5 mAh hat.¹ Wenn ein Knoten zehn Minuten nicht gesendet hat, hat er seine Energie wieder auf 8 mAh regeneriert und kann weiter senden.

Empfängt ein Knoten ein Paket mit einer Routing-Zieladresse, die seiner eigenen Routing-Adresse entspricht, wertet er das Paket aus. Andernfalls wird ermittelt, ob ein Pfad zu dem Zielknoten bekannt ist und das Paket ggf. dahin weitergeleitet. In allen anderen Fällen wird das Paket als Broadcast weitergeleitet. Beim Weiterleiten wird das TTL-Feld dekrementiert. Ist das Feld bereits 0, wird ein Paket nicht weitergeleitet. Pakete mit der Zieladresse 0 werden ebenfalls nicht weitergeleitet.

¹Wir gehen davon aus, dass ein Knoten eine Anfangsenergie von 10 mAh hat.