Deel 1 — Titel, samenvatting en Hoofdstuk 1 & 2

WA8DED HOST MODE GEBRUIKERSHANDLEIDING

==========
Oorspronkelijk door Paul T. Williamson, KB5MU, 23 mei 1986
Bijgewerkt door Peter Guelzow, DB2OS, 8 aug 1993

Samenvatting

Dit document beschrijft het gebruik van de **WA8DED Host mode** met de **WA8DED AX25 versie 2 multi-channel TNC firmware**, versie 1.1 voor de TAPR TNC-1 en **TheFirmware TF versie 2.x** voor de TNC-2 van **NORD><LINK e.V.**
TheFirmware TF is ook beschikbaar voor de **TNC-3 (68302) hardware**.

Het document is gebaseerd op:

- het WA8DED-document (versie 1.0),
- de W6IXU mailboxsoftware (revisie 13 mei 1986),
- en de WA8DED Split Screen Terminal Emulator (SS, versie 28 april 1986).

Het is bedoeld voor **applicatieprogrammeurs** die met de TNC willen communiceren in **Host mode**, en gaat ervan uit dat de lezer al bekend is met het gebruik van de WA8DED-firmware en TheFirmware in **user mode**.

Hoofdstuk 1

Wat is Host mode?

De WA8DED-TNC-firmware biedt twee volledig onafhankelijke gebruikersinterfaces.
De standaardinterface (user mode) is ontworpen voor gebruik door een menselijke operator via een terminal. **Host mode** daarentegen is bedoeld voor gebruik door een speciaal geschreven computerprogramma.

Host mode stelt de computer in staat om **strikte controle** te houden over de computer-TNC-verbinding en om **status- en controlegegevens** te verkrijgen die anders moeilijk uit de normale datastroom te halen zijn.

Een van de grootste problemen bij computer-TNC-koppelingen is altijd **flow control** geweest.
De TNC kan data sneller verzenden dan de computer kan verwerken, en omgekeerd kan de computer sneller data sturen dan de TNC kan uitzenden.

Theoretisch kan dit worden opgelost met:

- **hardware flow control** (RTS/CTS-lijnen op de RS-232-verbinding), of
- **software flow control** (XON/XOFF-tekens in de datastroom).

Beide methoden zijn echter foutgevoelig:

- hardware flow control is niet altijd beschikbaar;

- software flow control veroorzaakt problemen met **datatransparantie** (wat als de data zelf XON of XOFF bevat?).

Als alles niet perfect is ingesteld, gaat er data verloren.

Oplossing door Host mode

WA8DED Host mode lost dit probleem op door:

1. **Alleen te spreken wanneer daarom wordt gevraagd**, en
2. **Databursts te beperken tot maximaal 256 bytes**.

Hierdoor kan het computerprogramma andere taken uitvoeren (schrijven naar schijf, console afhandelen, koffie zetten ☕) zonder bang te zijn dat binnenkomende data verloren gaat.

De beperking tot 256 bytes maakt het ook mogelijk dat trage computers (zoals een Commodore 64 zonder UART) eerst een buffer vullen en daarna de data verwerken zonder overlopen.

Snellere computers kunnen zelfs eenvoudige byte-voor-byte polling gebruiken (bijvoorbeeld via BIOS-seriële routines op een IBM-PC).

⚠ **Belangrijk:**
In Host mode gebruikt de TNC **geen hardware- of software-flow-control**. Zorg ervoor dat je seriële drivers **geen XON/XOFF-tekens** naar de TNC sturen.

Omdat de TNC alleen antwoordt wanneer hij wordt bevraagd, moet de computer de TNC **continu pollen** om te zien wat er gebeurt.

In user mode wordt ontvangen AX.25-data direct weergegeven.
In Host mode mag de TNC **geen ongevraagde data sturen**, dus moet het programma periodiek om data vragen.

Dit betekent:

- de computer-TNC-verbinding is vrijwel continu actief;
- **frequent pollen** en een **hogere baudrate** (bijv. 4800 of 9600 baud) zijn wenselijk voor snelle respons;
- bij te lange poll-intervallen zijn merkbare vertragingen zichtbaar voor de gebruiker.

Communicatieformaat

Computer → TNC:

{kanaal}{info/cmd}{count}{data...}

- **Kanaal**: op welk kanaal de transmissie betrekking heeft
- **Info/cmd**: data of commando
- **Count**: aantal bytes dat volgt (−1 gecodeerd)
- **Data**: daadwerkelijke bytes

TNC → Computer (drie mogelijke vormen):

{kanaal}{code=0}

{kanaal}{code=1–5}{data...}{0}

{kanaal}{code=6–7}{count}{data...}

Aan de **code** is altijd te zien welk formaat volgt.
Details worden later uitgelegd.

Hoofdstuk 2

Host mode in- en uitschakelen

Host mode wordt geactiveerd met het commando:

JHOST 1

Bijvoorbeeld:

<ESC>JHOST1<CR>

Maar: als er al rommel in de invoerbuffer zit (bijvoorbeeld bij het opstarten), kan <ESC> als data worden geïnterpreteerd en gaat het commando verloren.

Betrouwbare methode

Stuur eerst:

- ^U of ^X om de buffer te legen
- eventueel een XON (^Q) als er een XOFF actief is

Daarom sturen we:

^Q ^X ^[JHOST1 ^M

Hex:

11 18 1B 4A 48 4F 53 54 31 0D

Dit werkt **altijd**, zolang de TNC in user mode is.

Als de TNC al in Host mode staat, moet dit worden herkend en hersteld.
Host-mode-herstel wordt later besproken.

Host mode verlaten

Verlaat Host mode met:

JHOST 0

De TNC bevestigt dit nog volgens Host-mode-protocol en keert daarna terug naar user mode.

⚠ Let op:

- QRES in Host mode zet je **direct terug naar user mode**, zonder Host-mode-bevestiging
- Host mode wordt **niet opgeslagen in NOVRAM**

- andere parameters kunnen wel permanent worden opgeslagen vanuit Host mode

---

👉 **Volgend bericht:**
Hoofdstuk 3 & 4 — Commando's sturen + pollen met het **G-commando**

Deel 2 — Hoofdstuk 3 & 4

Hoofdstuk 3

De TNC aansturen (Commanding the TNC)

Nu we in **Host mode** zitten, bekijken we een eenvoudige commandotransactie. Voor deze eerste uitleg slaan we enkele details over.

Stel dat we **Unattended mode** willen uitschakelen.
In normale **user mode** zouden we typen:

<ESC>U0<CR>

In Host mode moeten we hetzelfde commando verzenden, maar dan verpakt in het **computer-to-TNC format**.

Herinnering: het computer-to-TNC format is:

{channel}{info/cmd}{count}{data...}

- **Channel**: kanaalnummer
- **Info/cmd**: 1 voor een command
- **Count**: (lengte − 1)
- **Data**: het commando zelf

Het **U0**-commando bestaat uit twee ASCII-bytes (U en 0), dus:

- count = 2 − 1 = **1**
- info/cmd = **1**
- channel maakt niet uit → we kiezen **0**

Voorbeeldtransmissie

00 01 01 55 30

! ! ! U 0

! ! !

! ! +-- Count = 1 (twee bytes)

! +----- Info/cmd = 1 (dit is een command)

+-------- Channel = 0

⚠ **Belangrijk:**
Er mag **geen carriage return** (CR) worden meegestuurd. Zodra het opgegeven aantal bytes is ontvangen, accepteert de TNC de transmissie als compleet.

Antwoord van de TNC

Een **U0 command** kan niet mislukken en levert geen data op. De TNC antwoordt daarom met een **short format**:

00 00

! !

! +--- Code = 0 (Success, no data follows)

+------ Channel = 0

Algemene Host-mode transactie

Elke Host-mode transactie heeft deze vorm:

1. De computer stuurt een command of data
2. De TNC antwoordt met een passend bericht

Het programma moet:

- uitgaande frames opbouwen
- binnenkomende frames interpreteren

Bij het ontvangen van een antwoord moet het programma kijken naar de **code byte** om te weten:

- hoeveel bytes er nog gelezen moeten worden
- welk format wordt gebruikt (short, null-terminated, of byte-count)

Een groot voordeel hiervan is dat de ontvangst-routine **geen kennis nodig heeft van de TNC-status**. Alles wat nodig is, zit in het ontvangen frame zelf.

Hoofdstuk 4

De TNC pollen: het G command

Het **G command** is de meest essentiële en fundamentele transactie in Host mode.
Een Host-mode applicatie zal voortdurend **G commands** naar de TNC sturen en de antwoorden interpreteren.

Omdat een G command altijd gericht is aan een **specifiek channel**, moet je alle actieve channels pollen:

- data op channel 1 → G naar channel 1
- data op channel 2 → G naar channel 2

- monitored data → **G naar channel 0**

Varianten van het G command

Er zijn drie vormen:

- G → alle mogelijke responses
- G0 → alleen information
- G1 → alleen link status

Dit is handig wanneer je:

- wacht op een statusverandering zonder data te willen verwerken
- of juist data wilt verwerken zonder statusmeldingen

Formaat van het G command

Voor een onbeperkt G command:

0x 01 00 47

! ! ! G

! ! !

! ! +--- Count = 0 (één byte data)

! +------ Info/cmd = 1 (command)

+--------- Channel = x (0 t/m 4)

Mogelijke antwoorden

De meeste G polls leveren dit antwoord op:

Code = 0 → Nothing available

Dit betekent dat er sinds de vorige poll **geen relevante gebeurtenis** is geweest.

Andere mogelijke responses:

- **Link status changes** (bijv. CONNECTED, DISCONNECTED) → via G of G1
- **Ontvangen dataframes** → via G of G0

Monitored data (speciale afhandeling)

Monitored information wordt **altijd op channel 0** afgehandeld.

De monitor header wordt **apart** verstuurd van de monitor information:

1. Eerst een monitor header (code 4 of 5)
2. Indien code 5 → de **volgende poll** levert de bijbehorende data (code 6)

Voordelen:

- de applicatie hoeft headers en data niet zelf te scheiden

- maximale transmissielengte blijft beperkt

⚠ Dit is de enige situatie waarin een TNC-bericht **afhankelijk is van een eerder bericht**.
Je wilt monitor headers met bijbehorende information daarom **direct achter elkaar** verwerken.

👉 **Volgend deel:**
Hoofdstuk 5 & 6 —
Details van **computer-to-TNC format** en **TNC-to-computer format** (inclusief alle codes en voorbeelden)

Deel 3 — Hoofdstuk 5 & 6

Hoofdstuk 5

Details van het computer-to-TNC format

Zoals eerder vermeld moeten **alle computer-to-TNC transmissies** altijd dit format volgen:

{channel}{info/cmd}{count}{data...}

De drie header-bytes (**channel**, **info/cmd** en **count**) worden in **binaire vorm** verzonden.
Om bijvoorbeeld de waarde 0 te verzenden, stuur je een **NULL-byte** (CHR$(0), ^^U, of ^Y0 afhankelijk van de taal).
⚠ **Niet** het ASCII-teken '0' verzenden.

Het **data field** bevat één van de volgende drie soorten informatie:

1. Data die via de **AX.25 link** moet worden verzonden
2. Een **command string**, zoals die in user mode tussen <ESC> en <CR> zou staan
3. "G", het Host-mode poll command

Informatie verzenden (info/cmd = 0)

Wanneer data via de AX.25 link moet worden verzonden:

- **info/cmd** = 0 (binaire nul)
- het data field mag **elke mogelijke bytewaarde** bevatten
- maximum: **256 bytes**

De **count byte** bevat:

aantal_bytes − 1

Voorbeeld: je bent verbonden op **channel 2** en wilt sturen:

Hello

Dan stuur je:

02 00 05 48 65 6C 6C 6F 0D

! ! ! H e l l o CR

! ! ! !

! ! +- Count = 5 (zes bytes data)

! +---- Info/cmd = 0 (information)

+------- Channel = 2

Let op:

- CR is **data**, geen deel van het Host-mode format
- data op **channel 0** wordt als **unproto** verzonden
- data op channel 1–4 wordt genegeerd als het channel niet connected is

---

Commands verzenden (info/cmd = 1)

Bij het verzenden van een command:

- **info/cmd** = 1
- het data field bevat de **ASCII command string**
- **geen <ESC> en geen <CR>**

Voorbeeld: TXDELAY 30 instellen:

00 01 02 54 33 30

! ! ! T 3 0

! ! !

! ! +- Count = 2 (drie bytes data)

! +---- Info/cmd = 1 (command)

+------- Channel = 0

Channel-afhankelijkheid van commands

De volgende commands zijn **channel-specifiek** in Host mode, net als in user mode:

C, D, F, G, L, N, O, V

- **L command** werkt anders in Host mode (later uitgelegd)
- **G command** is al behandeld
- overige commands mogen op elk channel worden verzonden (meestal channel 0)

Waarom byte-count format?

Het **byte-count format** zorgt voor **data transparency**:
elke bytewaarde is toegestaan, zonder escape-tekens.

Nadeel:
Als er een byte verloren gaat of extra wordt ingevoegd, raken computer en TNC **out of sync**.

Detectie van sync-verlies:

- channel byte: alleen waarden 0–4 geldig
- info/cmd byte: alleen 0 of 1 geldig
- timeouts kunnen helpen

Herstelmechanismen worden besproken in **Hoofdstuk 8**.

Hoofdstuk 6

Details van het TNC-to-computer format

Het **TNC-to-computer format** is complexer.
De sleutel tot correcte interpretatie is de **code byte**, die volgt op het channel byte.

Overzicht van codes

Code	Betekenis	Format
0	Success, nothing follows	short format
1	Success, message follows	null-terminated
2	Failure, message follows	null-terminated
3	Link status	null-terminated
4	Monitor header, no info	null-terminated
5	Monitor header, info follows	null-terminated
6	Monitor information	byte-count
7	Connected information	byte-count

Codes 0, 1 en 2 — Response op command of data

Code 0
Succes zonder extra informatie:

02 00

! !

! +---- Code = 0

+------- Channel = 2

Code 1
Succes mét informatie (null-terminated):

Voorbeeld: M command zonder argument

00 01 49 55 53 43 52 54 00

Code 2
Foutmelding, bijvoorbeeld:

INVALID COMMAND

TNC BUSY - LINE IGNORED

Codes 3 t/m 7 — Responses op G command

Code 0 (bij G):

Nothing available

Code 3 — Link status

Voorbeelden:

- CONNECTED
- DISCONNECTED
- LINK RESET
- FRAME REJECT

Deze berichten zijn identiek aan user mode.

Codes 4 en 5 — Monitor header

Bevat:

fm {call} to {call} via {digipeaters} ctl {name} pid {hex}

- Code 4 → geen info field
- Code 5 → info volgt (bij volgende poll)

Code 6 — Monitor information
Byte-count format, alleen op channel 0

Code 7 — Connected information
Byte-count format, alleen op connected channels (1–4)

👉 **Volgend deel:**
Hoofdstuk 7 & 8 —

Channel status (L command) en **Host mode error recovery**

Deel 4 — Hoofdstuk 7 & 8 + Begrippenlijst

Hoofdstuk 7

Channel status opvragen: het L command

Een van de krachtigste functies van **Host mode** is de mogelijkheid om de TNC om een **statusrapport per channel** te vragen.

Dit rapport lijkt op de uitvoer van het **L command** in user mode, maar is uitgebreider.

Channel Status opvragen

Stuur een **L command** naar het channel waarover je informatie wilt:

01 01 00 4C

! ! ! L

! ! !

! ! +---- Count = 0 (één byte data)

! +------- Info/cmd = 1 (command)

+---------- Channel = 1

Typische Channel Status response

01 01 30 20 30 20 30 20 30 20 30 20 30 00

! ! 0 0 0 0 0 0 !

! !

! +------- Code = 1 (Success with info)

+---------- Channel = 1

De response bevat **zes ASCII-decimale waarden**, gescheiden door spaties.

Channel 0

Channel 0 heeft minder statusinformatie:

00 01 30 20 33 00

Betekenis van de waarden

a b c d e f

Veld	Betekenis
a	Aantal link status messages nog niet verwerkt
b	Aantal receive frames nog niet verwerkt
c	Aantal send frames nog niet verzonden
d	Aantal verzonden frames nog niet geacknowledged
e	Aantal retries op huidige operatie
f	Link state

Link states

Waarde	Betekenis
0	Disconnected
1	Link Setup
2	Frame Reject
3	Disconnect Request
4	Information Transfer
5	Reject Frame Sent
6	Waiting Acknowledgement
7	Device Busy
8	Remote Device Busy
9	Both Devices Busy
10	Waiting Ack + Device Busy
11	Waiting Ack + Remote Busy
12	Waiting Ack + Both Busy
13	Reject Frame Sent + Device Busy
14	Reject Frame Sent + Remote Busy
15	Reject Frame Sent + Both Busy

Opmerkingen:

- Channel 0 toont alleen velden **a** en **b**
- Bij firmware versie 1 zijn alleen states **0 t/m 4** mogelijk

Gebruik in applicaties

- Terminalprogramma's tonen de status
- Mailboxsoftware wacht op acknowledgements
- Forwarders controleren of alle data is bevestigd
- Applicaties kunnen throughput bewaken en beslissen om te stoppen

---

Hoofdstuk 8

Host mode error recovery

Fouten op de computer–TNC-link kunnen ontstaan door:

- losse kabels
- RF-instraling
- corrupte RS-232-data
- Murphy 😁

De recovery-routine moet aannemen dat de TNC zich in een **onbekende toestand** bevindt.

Doel van recovery

- de TNC terugbrengen naar een **bekende state**
- zo min mogelijk **ongewenste AX.25-transmissies**

Stappenplan

1. **Gooi alle binnenkomende data weg**
   Deze kan niet betrouwbaar worden geïnterpreteerd.
2. **Stuur herhaaldelijk ^A (0x01)**
   - mogelijk vult dit een resterende byte-count
   - maximaal 256 keer nodig
3. Zodra de TNC reageert, weet je dat hij weer **command-ready** is

Waarom ^A?

Een reeks ^A resulteert uiteindelijk in een ongeldig command:

01 01 01 01 01

Dit levert een foutmelding op:

INVALID COMMAND

De inhoud van de foutmelding is niet belangrijk — **de reactie zelf** is dat wel.

⚠ Recovery kan traag zijn, maar is zelden nodig.

Begrippenlijst (Engels → uitleg)

AX.25
Packet radio protocol gebruikt in amateurradio.

Byte-count format
Transmissieformaat waarbij eerst het aantal bytes wordt opgegeven (−1 gecodeerd).

Channel
Logisch communicatiekanaal binnen de TNC (0–4).

Command
ASCII-opdracht aan de TNC (zoals C, D, L, G).

Connected information
Data ontvangen via een actieve AX.25-verbinding.

Flow control
Mechanisme om datasnelheid tussen apparaten te regelen.

G command
Host-mode poll command om events en data op te vragen.

Host mode
TNC-bedrijfsmodus waarbij een computer volledige controle heeft.

Info/cmd byte
Header-byte die onderscheid maakt tussen data (0) en command (1).

Link status
Statusinformatie van een AX.25-verbinding.

Monitor header / Monitor information
Ontvangen AX.25-frames die niet voor een actieve connectie zijn.

Null-terminated format
Data eindigt met een null-byte (0x00).

Polling
Periodiek opvragen van status of data.

Short format
Response zonder extra data (alleen channel + code).

TNC (Terminal Node Controller)
Hardware die computer en radio verbindt voor packet radio.

User mode
TNC-bedrijfsmodus voor directe bediening door een gebruiker.