

PACTOR

Inhaltsverzeichnis

1 Digitale Betriebsarten im Detail: Pactor	2
1.1 Einleitung	2
2 Pactor-Einführung und Theorie	2
2.1 I. Überblick	2
2.1.1 Die Grundlagen für den Entwurf sind:	2
2.2 II. System-Details	3
2.2.1 1. Taktierung	3
2.2.2 2. Pakete	3
2.2.3 3. Steuersignale (CS)	4
2.2.4 4. Master <> Slave	4
2.2.5 5. Ändern der Senderichtung	4
2.2.6 6. Ändern der Geschwindigkeit	4
2.2.7 7. Ende eines PACTOR-Kontakts	5
2.2.8 8. Datenkompression	5
2.2.9 9. Memory-ARQ	5
2.2.10 10. Monitor (listen mode)	5
2.2.11 11. FEC-Aussendung	5
2.2.12 12. Praktische Aspekte	6
2.3 Pactor-Betrieb	6
3 Pactor im Vergleich	6

Digitale Betriebsarten im Detail: Pactor

Einleitung

Pactor (aus dem Lateinischen: der Vermittler) wurde von in Deutschland von Ulrich Strate (DK4KV) und Hans-Peter Helfert (DL6MAA) entwickelt. Ziel war es die für Kurzwelle ineffizienten Betriebsarten wie AMTOR und das störungsanfällige Packet Radio (mit 300Bd) zu verbessern. Pactor wurde von der Version I über II zu der momentan aktuellen Version III entwickelt.

- Pactor I erreicht die eine maximale Übertragungsgeschwindigkeit von 200 Bit/s
- Pactor II erreicht die eine maximale Übertragungsgeschwindigkeit von 700 Bit/s
- Pactor III erreicht die eine maximale Übertragungsgeschwindigkeit von 3600 Bit/s (bei 2.4 kHz @ -40 dB)

aus der deutschen Hilfe von Eike, DM3 ML

Datei: [pic pactor.gif](#) Beispiel eines Pactor-Signals im Wasserfall-Display von MixW

Pactor-Einführung und Theorie

von der Webseite von RICHARD B. GRIFFIN, NB6Z

(Anmerkung DM3ML : Die folgenden Ausführungen beziehen sich auf PACTOR-1)

I. Überblick

PACTOR (PT) wurde speziell für die Arbeit in gestörten und fluktuierenden Kanälen als halbduplexes ARQ-System entworfen, das die Zuverlässigkeit von PACKET mit der festen Paketlänge von AMTOR kombiniert.

Die Grundlagen für den Entwurf sind:

PACTOR kombiniert alle wichtigen AMTOR und Packet-2-Weg-Charakteristika :

- Fester Zeitrahmen und volle Synchronität zur Sicherung der maximalen Geschwindigkeit

- Schnelle und zuverlässige Tastenübergabe und Break-in
- Weniger als 600 Hz Bandbreite
- 100% ASCII-kompatibel (wahre Binärdatenübertragung)
- extrem niedrige Wahrscheinlichkeit unentdeckter Fehler mit 16 bit CRC
- unabhängig von der Shift-Polarität
- kein Mehrnutzerüberhang in Schmalbandkanälen
- billige Hardware (einfache Z80-Leiterplatte)
- hoher Betriebskomfort (Mailboxsystem eingebaut usw.)
- Monitor-Modus (listen-mode)
- FEC-Modus beim CQ-Ruf

Als Neuigkeiten im Amateurfernschreiben wurden weitere hochwertige Eigenschaften eingeführt :

- Optimaler kohärenter Modus, wenn beide Systeme auf ein Frequenznormal (DCF77, TV-Zeilenfrequenz oder hochstabiler Rundfunksender) eingerastet sind
- Online-Datenkompression (Huffmann-Codierung)
- automatischer Geschwindigkeitswechsel (100/200 Baud) ohne Verlust der Synchronisation
- quittiertes Verbindungsende
- Memory-ARQ (gestörte Pakete können durch Wiederholung restauriert werden)

II. System-Details

1. Taktierung

Die Pactor-Aussendung ähnelt vom Prinzip her AMTOR : Blöcke mit Information werden gesendet und durch ein kurzes Steuersignal (CS) von der empfangenden Station bestätigt. Die Shiftlage wird in jedem Zyklus gewechselt, um die ARQ (siehe unten) zu unterstützen. Eine Mark-Space-Definition wird dadurch überflüssig.

- Zykluslänge : 1.25 sec
- Paketlänge : 0.96 sec = 192 (96) bits bei 200 (100) baud
- Quittungssignal (CS) : 0.12 sec = 12 bits zu je 10 msec
- CS-receive gap : 0.29 sec

Eine Änderung der Übertragungsgeschwindigkeit ändert nur die interne Paketstruktur aber nicht die anderen Zeitparameter.

2. Pakete

Eine Pactor-Paket besteht aus dem Kopf (Header), dem Datenfeld mit 20 bzw. 8 Byte bei 200 bzw. 100 Baud, einer Statuskennung und 2 Byte CRC (Cyclic Redundance Code = Prüfdaten)

- Header: Dieses Byte erlaubt eine schnelle Synchronisation und liefert zusätzliche Information (Memory-ARQ, Listen mode)
- Data: Binärdaten
- Status : Systemsteuerbyte mit einer 2-Bit-Paketnummer, TX-Mode, Breakanforderung oder QRT-Ankündigung
- CRC: 16-Bbit-CRC-Kode auf der Basis des CCITT-Polynoms $X^{16}+x^{12}+x^5+1$ berechnet über das ganze Paket ohne Header

3. Steuersignale (CS)

Es werden vier CS-Signale verwendet. Ein CS-Signal ist als Kompromiss zwischen schneller Erkennung und sicherer Übertragung 12 Bit lang. Die Codierung im Hex-Code ist :

CS1: 4D5 / CS2: AB2 / CS3: 34B / CS4: D2C

Die Hammingdistanz ist 8 bit, um die Chance des Empfangs eines falschen CS zu vermindern. CS1 /2 und CS3/4 haben ein symmetrisches bitinvertiertes Muster. CS1..3 haben die gleiche Funktion wie bei AMTOR, CS4 schaltet die Übertragungsgeschwindigkeit um, CS3 bewirkt die Umkehr der Senderichtung.

4. Master <> Slave

Die anrufende Station ist der Master und sendet spezielle Synchronisierungspakete bestehend aus :
/head (100 bd)/..address (8 bytes, 100 bd)../..address (8bytes, 100 bd) Normalerweise benutzt der Empfänger nur die 100-Baud-Sektion für eine schnelle Synchronisation. Die 200-Baud-Sektion liefert eine zusätzliche Information über die Kanalqualität. Falls Sie richtig empfangen wird, ist das erste CS eine CS4, sonst wird CS1 gesendet. Der Meister sendet dann nach der Synchronisation normale Pakete in 100 und 200 Baud. Das erste übertragene Zeichen ist die "system level number" (die Pactor-Software-Version), gefolgt von der Master-Adresse (seinem Rufzeichen).

5. Ändern der Senderichtung

Wie bei AMTOR kann die empfangende Station die Senderichtung wechseln, sobald sie ein gültiges Paket empfangen hat. Ein spezielles Umschaltpaket wird gesendet. Die sendende Station schaltet unmittelbar auf Empfang, wenn sie das Steuerpaket CS3 empfängt und wertet das Paket aus. Sie sendet CS1 und CS3 als Quittung oder ein CS2 als Wiederholungsanforderung.

6. Ändern der Geschwindigkeit

Eine Verringerung der Geschwindigkeit ist nur bei schlechten Bedingungen sinnvoll oder wenn nur wenige Daten z.B. bei manueller Eingabe zu übertragen sind. Beide Senderichtungen werden unsymmetrisch behandelt.

- langsamer

Der Empfänger kann nach jedem falsch empfangenen Paket mit CS4 eine Umschaltung von 200 Baud auf 100 Baud anfordern. Ein unbestätigtes 200-Baud-Paket wird dann mit 100 Baud wiederholt.

- schneller

jedes gültig empfangene Paket kann mit CS4 quittiert werden um beim Sender eine Erhöhung der Geschwindigkeit anzufordern. Wenn das anschließend mit 200 Baud gesendete Paket trotz mehrfacher Wiederholung nicht als richtig empfangen quittiert wird, schaltet der Sender automatisch auf 100 Baud zurück.

7. Ende eines PACTOR-Kontakts

Zum Ende eines ARQ-Link muss sicher gestellt werden, dass alle Daten übertragen worden. Pactor hat ein spezielles QRT-Paket eingeführt, das ein aktives QRT-Statusbit und die invertierte Empfängeradresse enthält. Wird diese Adresse gefunden, antwortet der Empfänger mit einem abschließenden CS und die Verbindung wird definitiv und im richtigen Zeitrahmen beendet.

8. Datenkompression

Eine Textanalyse eines typischen deutschen oder englischen Texts zeigt, dass die durchschnittliche Information je Zeichen kaum über 4 Bit liegt. ASCII-Texte mit ihren festen 8-Bit-Zeichen sind also zu 50% redundant. Dieser Überhang kann durch einen Code mit variabler Länge abhängig von der Zeichenverteilung abgebaut werden. Das populärste Beispiel für einen solchen Code ist das Morsealphabet. Bei Pactor wird der Huffman-Code mit einer ähnlichen Effizienz eingesetzt und ein Gewinn von bis zu 100% erzielt. Jedes Paket enthält einen komprimierten Zeichensatz, die Codelänge je Zeichen variiert von 2 bis 15 Bit.

9. Memory-ARQ

In konventionellen ARQ-Systemen muss der Sender ein Paket so lange wiederholen, bis es der Empfänger fehlerfrei erkannt hat. Die Übertragungsgeschwindigkeit verringert sich bei einem schlechten S/N-Verhältnis dramatisch. Ein Ausweg ist, die Pakete deutlich zu verkürzen oder eine Fehlerkorrekturinformation einzubauen, die wieder bei guten Bedingungen die Übertragungsgeschwindigkeit deutlich verschlechtert. Pactor setzt dagegen einen Memory-ARQ ein. Jedes empfangene Bit eines Pakets wird mit einem 8-Bit-AD-Konverter bewertet. Die Amplitudenwerte zwischen 0 und 255 der Bits der wiederholt gesendeten Pakete werden im Speicher aufsummiert. Da die Shiftlage von Paket zu Paket gewechselt wird, werden konstante Störpegel eliminiert. Nach jeder Übertragung wird das Paket getestet, ob der CRC-Code erfüllt wird.

Anmerkung DM3ML : Pactor kann auf diese Weise bis zu einem S/N von -18 dB arbeiten. Voraussetzung ist, dass ein A/D-Wandler eingesetzt wird. Bei "aufgebohrten" (amerikanischen) Kontrollern wird in der Regel nur eine digitale 1 oder 0 abgespeichert, sodass der Vorteil der analogen Memory-ARQ bei weitem nicht erreicht wird.

10. Monitor (listen mode)

Pactor-Pakete können im Monitormodus mitgeschrieben werden. Es werden nur gültige Pakete mit erfolgreichem CRC-Test ausgegeben.

11. FEC-Aussendung

CQ- und Rundspruchsendungen werden in einem FEC-Modus gesendet. Die CS-Austastlücke entfällt. Die sendende Station kann Geschwindigkeit und Wiederholrate wählen.

Anmerkung DM3ML : Typisch ist der Unproto-Modus in 100 oder 200 Baud

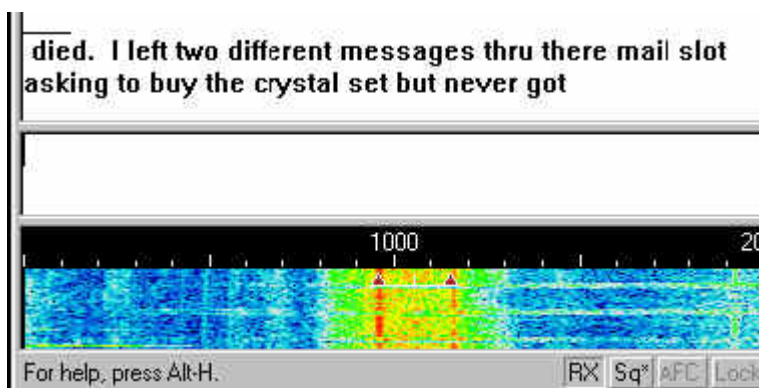
12. Praktische Aspekte

Die ersten PACTOR-Programme liefen auf Z80-Computern. Sie führten zu einem dedizierten 'PACTOR- Controller' mit eingebautem Modem und einer Abstimmmanzeige. Die konventionellen Sendearten BAUDOT und AMTOR wurden hinzugefügt um die Kompatibilität zu wahren und um Vergleiche zu ermöglichen. PACTOR lief bis zu viermal schneller als AMTOR.

Pactor-Betrieb

Mit der aktuellen Ausgabe von MixW Version 2 kann man Pactor (DM3ML: genauer Pactor-1) über das Soundkarteninterface mitschreiben. Es ist nicht möglich ohne einen zusätzlichen Hardware-TNC zu senden, der unter MixW problemlos konfiguriert werden kann.

Um Pactor ohne einen TNC zu empfangen, wählen Sie unter Mode |> Pactor oder klicken Sie auf das Modelfeld im Statusbalken. Ein Pactor-Signal erkennen Sie an den längeren chirpenden Pakete mit kurzen Quittungen dazwischen. Im folgenden Bild sehen Sie ein Pactor-QSO im Wasserfall :



Es sieht ähnlich aus wie RTTY und verwendet die gleichen mit einem Balken verbundenen Diamantcursoren.

Pactor im Vergleich

Pactor im Vergleich mit anderen Digi-Modes

Digital Mode Comparisons

4-Mar-2006 Firmware 3.6q

