

FSERIAL.CPX V1.0 Release Notes

Roman Hodek

8. Januar 1993

Inhaltsverzeichnis

1	Zweck	2
2	Bedienung	2
3	Die FSERIAL-Infos	3
4	Portabilität	4
5	Bugreports	4
6	History	5

1 Zweck

FSERIAL.CPX ist Bestandteil von FAST_SER von Franz Sirl, das CPX wurde von Roman Hodek geschrieben. Es dient zur Konfiguration der seriellen Schnittstellen unter Berücksichtigung von FAST_SER.

FSERIAL.CPX tut im Prinzip dasselbe wie das bekannte MODEM.CPX von Atari, nur daß es eben merkt, wenn FAST_SER installiert ist und entsprechend handelt. Darüberhinaus kann FSERIAL.CPX ein paar Informationen über die Schnittstellen anzeigen und Breaks senden.

2 Bedienung

Im Hauptdialog verhält sich FSERIAL.CPX genauso, wie man es von einem CPX erwartet und wie man es von MODEM.CPX gewohnt ist. Es haben nur folgende kleine Anpassungen stattgefunden:

- Die Anordnung von Datenbits/Parität/Stopbits wurde umgestellt, so daß es der gewohnten Reihenfolge von z.B. 8N1 entspricht.
- Gibt es keine erweiterten Schnittstellen, so ist das Port-Popup abgeschaltet und fest auf Modem 1.
- Ist die aktuelle Schnittstelle Serial 1, so ist kein Rts/Cts-Handshake möglich, da Serial 1 keine solchen Handshake-Leitungen besitzt.
- Es gibt eine Infozeile, die die Version des CPX anzeigt und ob FAST_SER installiert ist und wenn ja, welche Version.
- Es gibt einen Button, um den Kompatibilitätsmodus von FAST_SER ein- bzw. auszuschalten. Ist dieser Modus an, so können zwei der erweiterten Speeds unter den Rsconf()-Indices 0 und 1 angesprochen werden,¹ die „alten“ Raten verschieben sich dementsprechend um 2 und bei Indices ≥ 18 findet man alle möglichen neuen Speeds. Um also z.B. bei Seriell 2 38400 bps einzustellen, wählt man in einem „nichtwissenden“ Programm 9600 bps (= Index 1) aus. Ist die Kompatibilität ausgeschaltet, so sind die neuen Geschwindigkeiten nur am Ende zu finden, d.h. sie können von Programmen, die nichts von FAST_SER wissen, nicht angesprochen werden. FSERIAL.CPX kommt mit beiden Konventionen zurecht, allerdings kann es beim Umschalten zwischen beiden Modi zu einem Verstellen der Baudrate kommen. Man sollte also immer zuerst den Kompatib.modus ändern, „OK“ drücken und danach noch einmal die bps-Rate kontrollieren.

Beim Booten werden alle Parameter inklusive der aktuellen seriellen Schnittstelle auf die in FSERIAL.CPX gespeicherten Werte gesetzt. Bei jedem Aufruf werden die Parameter erneut abgefragt, da sie ja von einem anderen Programm verstellt worden sein könnten. Man kann also FSERIAL.CPX auch zum Abfragen der aktuellen Einstellungen benutzen.

¹Seriell 2 und Modem 2 (Mega STE): 0↔57600, 1↔38400, Modem 2 (TT): 0↔38400, 1↔76800

Wird der Dialog mit „OK“ verlassen, so werden die Parameter für *alle* Ports gesetzt (nicht nur für den gerade unter „Port“ eingestellten). Ebenso werden bei „Sichern“ die Werte für alle Ports abgespeichert.

Leider ist aber der Handshakemode nicht per `Rscnf()` abfragbar. Daher wird hierfür direkt auf eine undokumentierte Variable der `IOREC`-Struktur zugegriffen.² Bei einem $TOS \leq 1.02$ kann leider die eingestellte bps-Rate nicht abgefragt werden. Dieser Fall wird auch nicht abgefragt, da man dann wahrscheinlich weder `FAST_SER` noch `FSERIAL.CPX` benötigt.

3 Die FSERIAL-Infos

Mit dem Button „Info“ kommt man zu den `FSERIAL`-Infos. Dort werden folgende Daten angezeigt:

Port: Der Port, zu dem die Daten gehören.

Chiptyp: Der Hardwarechip, der diese Schnittstelle steuert. Wenn `FAST_SER` installiert ist, bekommt man bei SCC-Ports eine genaue Angabe des Chips, ansonsten nur „SCC“.

Input-Puffer: Die Größe des Puffers für ankommende Daten.

Output-Puffer: dito für den Ausgabepuffer

Fehleranzeigen: `FAST_SER` verwaltet verschiedene Zähler für Fehler, die bei der Übertragung auftreten können. Ist `FAST_SER` nicht installiert, so bleiben die folgenden Angaben leer. SCC-bedingt sind manche der Angaben leider nicht immer 100%-ig verlässlich (speziell Frame Errors und Lost Chars).

Framing Errors: Zu soviel Zeichen wurden keine bzw. nicht die richtige Anzahl Stopbits empfangen.

Parity Errors: Anzahl der Zeichen, für die eine inkorrekte Parität festgestellt wurde.

Rx Overflows: Anzahl der Zeichen, die wegen vollem Empfangspuffers verloren gingen.

Lost Chars: Anzahl der Zeichen, die verloren gingen, weil der SCC nicht schnell genug abgefragt wurde.

Die Fehlerzähler laufen normalerweise immer weiter, wenn neue Fehler passieren. Mit dem Button „Fehler löschen“ kann man alle Zähler wieder auf Null setzen, falls dies nötig sein sollte.

Zusätzlich zu den Infos kann man aus diesem Dialog auch noch einen Break (d.h. eine Dauerlage Nullen ohne Stopbits) senden. Der Break dauert solange, wie man sich mit gedrückter Maustaste auf dem Button befindet.

²Dabei wird auch ein TOS-Fehler in diesem Zusammenhang umschifft.

4 Portabilität

Bei der Vielfalt der seriellen Schnittstellen und der TOS-Fehler ist es nicht einfach, wirklich portabel zu programmieren. Daher gibt es ein paar Einschränkungen:

- `FSERIAL.CPX` kann nur mit einer TOS-Version ≥ 1.04 verwendet werden, da man vorher die eingestellte bps-Rate nicht auslesen kann.
- `FSERIAL.CPX` verläßt sich öfters darauf, daß es entweder 1, 3 (Mega STE) oder 4 (TT) serielle Ports gibt, und dies genau bei den genannten Maschinen. Falls es einmal Schwierigkeiten hiermit geben sollte, bitte Bugreport an mich. Ebenfalls wird angenommen, daß bei 3 Ports diese „Modem 1“, „Modem 2“ und „Serial 2“ heißen. Bei 4 Ports wird „Serial 1“ vor „Serial 2“ eingeschoben. „Serial 1“ hat dabei die besondere Eigenschaft, daß es keine Handshakeleitungen besitzt.
- `FSERIAL.CPX` erlaubt aktuell bis zu maximal 29 verschiedene bps-Raten an einer Schnittstelle. Da es `FAST_SER` prinzipiell erlaubt, eigene Geschwindigkeiten zu definieren, könnte dies dazu führen, daß manche Speeds nicht mehr einstellbar sind. Die Grenze wäre jedoch durch einfaches Neucompilieren hochzusetzen.
- Man kann mit `Rscnf()` nicht den Handshakemodus erfahren. Dieser ist aber in einer (undokumentierten) `IOREC`-Variablen gespeichert. Es gibt auch schon mehrere andere Programme, die sich darauf verlassen, daß diese Angabe dort steht. Leider funkt auch hier wieder ein TOS-Fehler dazwischen: In manchen Versionen war sich Atari nicht einig, ob der Modus nun im High- oder im Lowbyte des Words zu speichern ist. Daher wird folgendermaßen abgefragt: Wenn im Lowbyte ein Wert ungleich Null steht, so wird dieser benutzt. Nur wenn dort eine Null ist, wird der Wert des Highbytes benutzt. Auch hier gilt: Wenn mein Vorgehen Schwierigkeiten verursacht: Bugreport an mich.
- Auch beim Break-Senden ist Atari mal wieder ein Tippfehler unterlaufen: Bei den Original `SCC-Rscnf`'s wird das Breakbit im falschen Parameter erwartet. Um dies zu umschiffen, habe ich folgende Methode gewählt: Zuerst wird das Bit an der richtigen Stelle gesetzt und dann im `TSR`-Rückgabewert nachgesehen, ob dort das Bit nun wirklich 1 ist. Wenn ja ist alles ok, sonst wird das Bit im Argument davor gesetzt.

5 Bugreports

`FSERIAL.CPX` wurde in einem mehrmonatigen Betatest geprüft und scheint inzwischen einigermaßen fehlerfrei zu sein. Falls Sie doch einen Bug finden oder einen Verbesserungsvorschlag habe, so schreiben Sie bitte an:

Roman Hodek
Baiersdorfer Str. 7
8521 Möhrendorf
Roman Hodek @ N (Mausnetz)
cdb0@fau60.informatik.uni-erlangen.de
rnhodek@cip.informatik.uni-erlangen.de

am Besten per E-Mail. Für eine Diagnose ist eine genaue Beschreibung des Fehlers und seiner Symptome sowie eine genaue Angabe der benutzten Systemkonfiguration (Hard- und Software) äußerst wichtig!

6 History

V0.1 β : Erste veröffentlichte Version

V0.2 β : Watermark-Anzeige im Infodialog ersetzt (da wenig sinnvoll) durch die Fehlerzähler; Button für `baud_table_flag` eingebaut; Fehler beim Window-Verschieben beseitigt

V0.3 β = V1.0: Kleiner Vertipper bei Stopbits/Parität-Einstellung beseitigt; ansonsten scheint das CPX inzwischen weitgehend fehlerfrei zu sein.