

# Tastatur- und Mausschnittstellen (AT, PS/2)

## Kurzbeschreibung

*Inhalt:*

1. Das Interface
2. Kommandos vom PC an die Tastatur
3. Kommandos von der Tastatur an den PC
4. Tastaturcodes
5. Die PS72-Maus
6. Anhang: Die serielle Maus

## 1. Das Interface

Tastatur- und Mausanschluß sind gleichartig aufgebaut. Deshalb können wir uns auf die Beschreibung der Tastaturschnittstelle beschränken.

### Signale

Das Tastaturinterface umfaßt zwei Signalleitungen, eine Stromversorgungsleitung und eine Masseleitung:

- KEYBOARD DATA: Datenleitung..
- KEYBOARD CLOCK: Taktleitung.
- + 5 V (Stromversorgung),
- Masse.

*Übertragungsrichtung:*

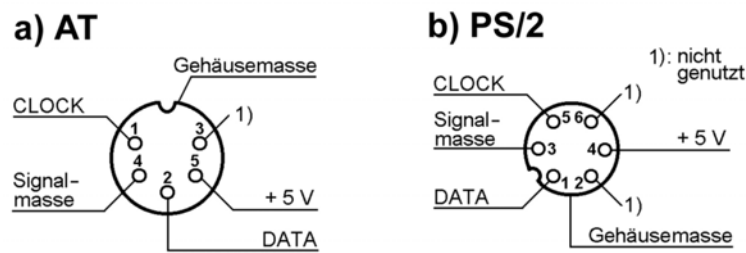
- KEYBOARD DATA: bidirektional. Erregt von der jeweils sendenden Einrichtung.
- KEYBOARD CLOCK: bidirektional. Ansteuerung:
  1. als Strobesignal: von der Tastatur (für beide Übertragungsrichtungen).
  2. zwecks Besetztanzeige und Selbsttestauslösung: vom PC.

*AT- und PS/2-Interface*

Beide Interfaces können als gleichartig angesehen werden; sie unterscheiden sich im Grunde nur im Steckverbinder.

### Steckverbinder

Die Geräte werden über 5- oder 6polige DIN-Rundsteckverbinder (Diodenbuchsen/Diodenstecker) angeschlossen (Abb. 1.1). Das AT-Interface nutzt einen 5poligen Steckverbinder in normaler, das PS/2-Interface einen 6poligen in Miniaturbauform. Anordnung: Buchse im PC, Stecker am Kabel.



**Abb. 1.1** Anschlußbelegungen des Tastaturinterfaces. Blick von vorn auf den Stecker am Tastaturkabel

#### Hinweise:

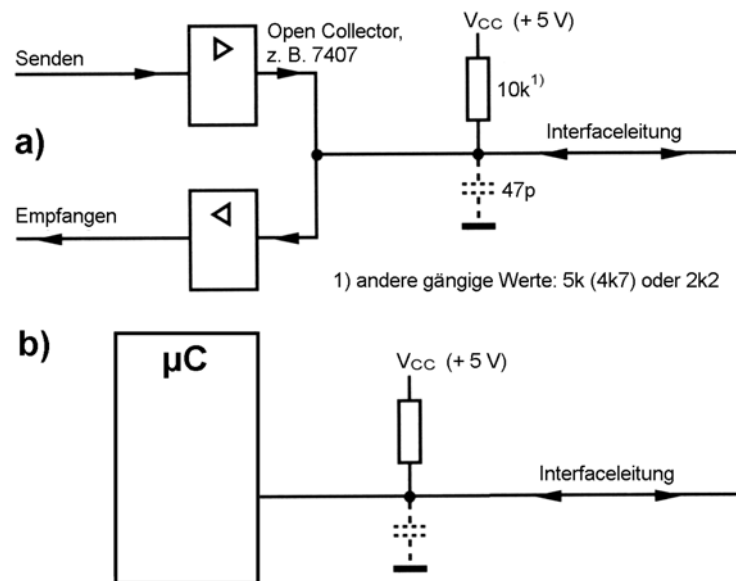
1. Das Metallgehäuse der PC-seitigen Diodenbuchse ist mit Masse verbunden, die metallische Umhüllung des Kabelsteckers mit dem Schirm des Kabels.
2. Achtung: 5polige Steckverbinder gibt es in zwei Bauformen, nämlich mit 45° oder mit 60° Teilung. Wir brauchen die 45°-Ausführung (Typ Nr. 4; daran erkennbar, daß die Anschlüsse 1...5 einen Halbkreis belegen).
3. Zwischenadapter und Zwischenkabel für beide Anschlußkombinationen (AT-Tastatur an PS/2-PC und umgekehrt) sind marktgängig.

#### Signalpegel und Flanken

Die Signale sind in Pegeln und Flankensteilheiten TTL-kompatibel. Zu den funktionellen Eigenheiten gehört, daß die Signale gelegentlich vom PC oder von der Tastatur auf Low gezogen werden. Deshalb werden die Leitungen typischerweise mit Open-Collector-Stufen angesteuert (z. B. Treiber 7407).

#### Leitungsabschluß

Die Signalleitungen sind beidseitig mit Pull-up-Widerständen beschaltet (Abbildung 1.2). Gelegentlich vorgesehene Kondensatoren dienen zum Unterdrücken von Störungen.

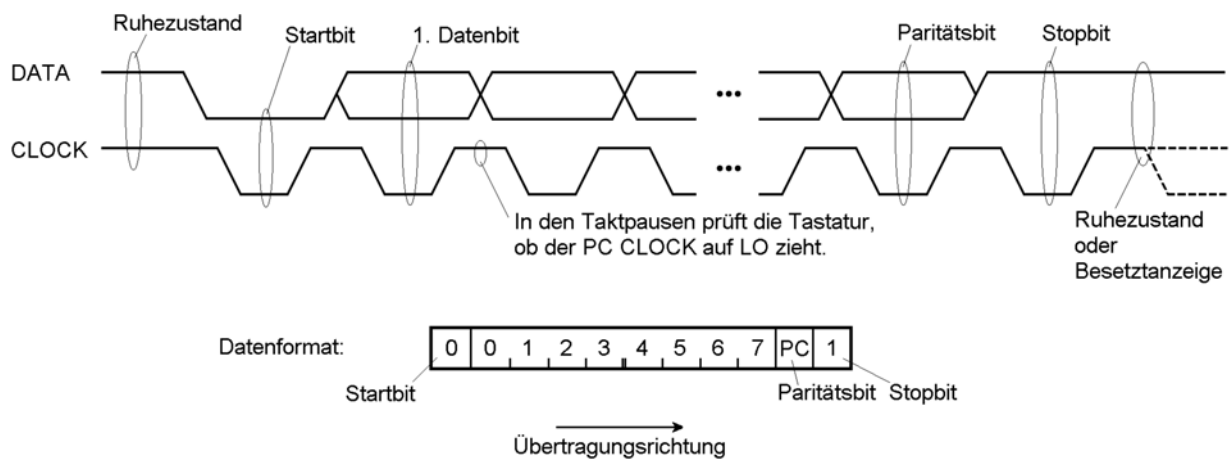


**Abb. 1.2** Kopplung und Leitungsabschluß eines Tastatursignals (es ist eine der beiden Seiten dargestellt)

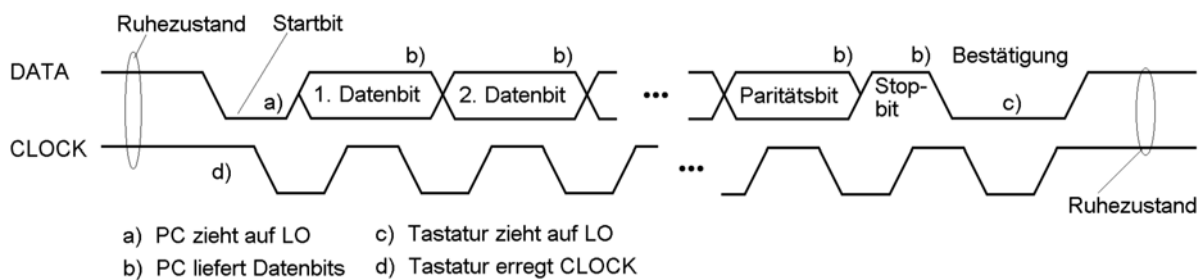
- a) Ankopplung über Open-Collector-Stufe. Senden: Aufschalten von Nullen oder Einsen. Empfangen: Sendesignal so einstellen, daß die Open-Collector-Stufe deaktiviert wird. Interfaceleitung wird freigegeben und durch Pull-up-Widerstand auf High gezogen. Die andere Einrichtung kann jetzt senden.
- b) Ankopplung an Mikrocontroller mit universellen E-A-Ports. Das Open-Collector-Verhalten wird durch Schalten des Richtungssteuerbits nachgebildet. Das Datenbit wird anfänglich fest mit Null belegt. Senden einer Eins oder Freigabe der Leitung (zum Empfangen): Richtungssteuerbit auf Eingabe (Anschluß wird hochohmig). Senden einer Null: Richtungssteuerbit auf Ausgang (Anschluß wird mit dem Datenbit – also mit einem Nullpegel – belegt).

### Signalfolgen

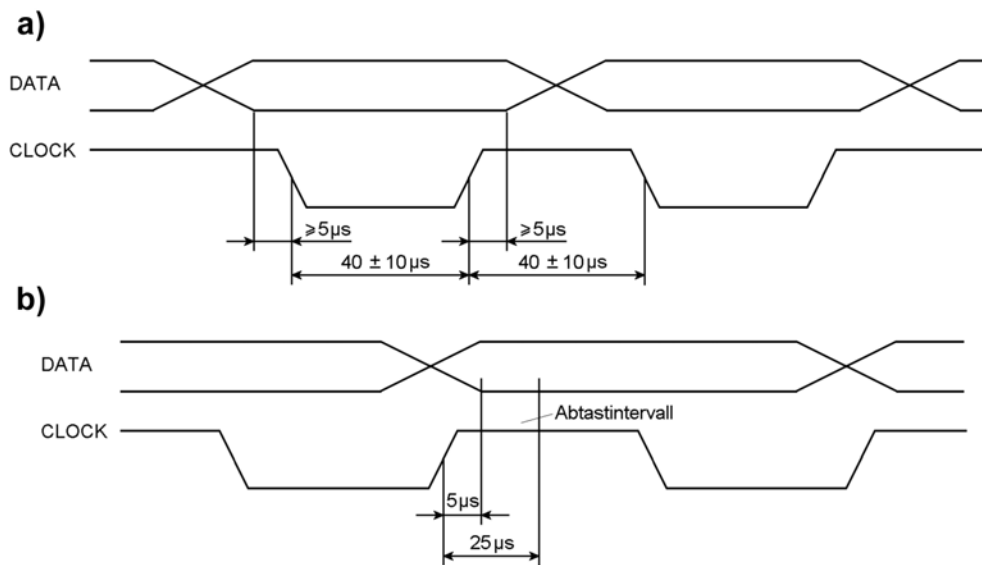
Die Datenübertragung ist in beiden Richtungen möglich. Die Abbildungen 1.3 bis 1.5 veranschaulichen die Signalverläufe und Zeitverhältnisse.



**Abb. 1.3** Oben: Datenübertragung von der Tastatur zum PC. Darunter das Datenformat



**Abb. 1.4** Datenübertragung vom PC zur Tastatur



**Abb. 1.5** Zeitverhältnisse. a) Richtung Tastatur => PC (der Taktimpuls begleitet das Datenbit und wirkt als Strobe), b) Richtung PC => Tastatur (der Taktimpuls ist eine Aufforderung an den PC, das Datenbit zu liefern)

### Übertragungsformat

Es handelt sich um ein festes Format: 1 Startbit (Low) + 8 Datenbits (beginnend mit dem niedrigstwertigen) + 1 Paritätsbit (das die 8 Datenbits auf ungerade Parität ergänzt) + 1 Stopbit (High).

### Übertragungszustände

Die verschiedenen Übertragungszustände werden durch „Ziehen“ der Takt- und Datenleitungen repräsentiert (Tabelle 1.1).

DATA	CLOCK	Zustand	Bemerkung
High	High	Ruhezustand; Tastatur darf senden	
High	Low	PC besetzt; Tastatur darf nicht senden	PC zieht CLOCK auf Low
Low	High	PC will senden	PC zieht DATA auf Low
Low	Low	undefiniert	Zwischenzustand, der durchlaufen werden kann, wenn der PC ein Kommando zur Tastatur schicken will

**Tabelle 1.1** Übertragungszustände des AT/PS/2-Tastaturinterfaces

### Übertragung von der Tastatur zum PC

Die Tastatur leitet eine Übertragung nur dann ein, wenn sie das Interface im Ruhezustand vorfindet (CLOCK und DATA beide High). Zunächst wird das Startbit (Low) übertragen, dann folgen die 8 Datenbits, das Paritätsbit sowie das Stopbit (High). Vgl. Abb. 1.3.

### Abbruch der Datenübertragung

Die Tastatur bricht die Datenübertragung ab, sobald CLOCK auf Low gezogen wird oder auf Low verharrt (Besetztanzeige des PC).

**Besetztanzeige**

Nach der Übertragung des Stopbits kann der PC CLOCK zeitweilig auf Low ziehen, wenn er zunächst nicht in der Lage ist, eine weiteres Datenbyte anzunehmen. Ein solcher Besetztzustand dauert typischerweise bis zu 50  $\mu$ s.

**Fehlerbehandlung**

Stellt der PC einen Paritätsfehler fest, so sendet er ein Kommando RESEND an die Tastatur. Diese antwortet darauf durch erneutes Senden des zuletzt übertragenen Bytes.

**Übertragung vom PC zur Tastatur**

Die Übertragung wird vom PC angefordert, indem er DATA auf Low zieht. CLOCK muß dabei High sein. Die Tastatur beginnt daraufhin, CLOCK-Impulse zu senden. Mit dem ersten CLOCK-Impuls wird die Low-Belegung von DATA übernommen (Startbit). Es folgen 8 Datenbits, ein Paritätsbit (ungerade Parität) sowie ein Stopbit (High). Vgl. Abb. 1.4.

**Bestätigen der Übertragung**

Nach dem Abholen des Stopbits (High) zieht die Tastatur DATA auf Low und sendet einen weiteren Taktimpuls. Dies dient dem PC als Bestätigung (Acknowledge) dafür, daß die Tastatur das Byte empfangen hat.

**Fehlerbehandlung**

Stellt die Tastatur einen Paritätsfehler fest, so wartet sie, bis DATA wieder High wird (Ruhezustand) und überträgt dann ein RESEND-Kommando zum PC.

**Zeitverhältnisse beim Übertragen von der Tastatur zum PC**

Breite eines Taktimpulses:  $40 \pm 10$  ns, Pause zwischen zwei Taktimpulsen ebenfalls  $40 \pm 10$  ns. Die Datenbelegung wird in den Pausen umgeschaltet (CLOCK = High), und zwar frühestens 5  $\mu$ s nach der Low-High-Taktflanke und spätestens 5  $\mu$ s vor der High-Low-Taktflanke. Siehe Abbildung 10.5a.

**Zeitverhältnisse beim Übertragen vom PC zur Tastatur**

Breite eines Taktimpulses:  $40 \pm 10$  ns, Pause zwischen zwei Taktimpulsen ebenfalls  $40 \pm 10$  ns. Ein Taktimpuls (CLOCK = Low) zeigt dem PC an, daß die Tastatur ein Datenbit erwartet. Der PC sollte also bei aktivem Taktimpuls das geforderte Datenbit auf die DATA-Leitung legen. Es wird von der Tastatur in einem bestimmten Zeitfenster (5...25  $\mu$ s) nach der Low-High-Flanke des Taktes ausgewertet. Vgl. Abb. 1.5b.

**Rücksetzen und Selbsttest (Basic Assurance Test BAT)**

Der PC löst in der Tastatur ein Rücksetzen aus, indem er die CLOCK-Leitung längere Zeit (Richtwert: 150 ms...2 s) auf Low hält. Nach dem Rücksetzen läuft in der Tastatur der Selbsttest (BAT) an. Testdauer: typischerweise zwischen 300 und 500 ms. Meldung des Testergebnisses durch Senden eines Bytes AAH (o.k.-Meldung) oder FDH (Fehlermeldung). Nach dem Selbsttest ist die Tastatur abgeschaltet (disabled). Das heißt, sie sendet keine Scan-Codes, kann aber Kommandos vom PC empfangen. Der jeweilige Code (AAH oder FDH) wird typischerweise 450 ms...2,5 s nach dem Rücksetzen zum PC übertragen.

## 2. Kommandos vom PC an die Tastatur

### Grundsätzliches zum Kommandoablauf

#### *Sendezeitpunkt*

Der PC prüft, wie weit eine Datenübertragung seitens der Tastatur (in Richtung PC) fortgeschritten ist. Läuft eine solche Übertragung und ist der 10. Taktimpuls noch nicht wirksam geworden, so bricht der PC die Übertragung ab, indem er CLOCK nach Low zieht. CLOCK verharrt für ungefähr 60 µs auf Low, damit die Tastatur Gelegenheit hat, die Anforderung zu erkennen. Dann zieht der PC DATA auf Low (Startbit) und gibt CLOCK wieder frei.

#### *Bytezahl*

Manche Kommandos bestehen nur aus einem Byte. Bei anderen folgen dem eigentlichen Kommandocode noch weitere Bytes nach, die auf jeweils spezifische Weise interpretiert werden. Jedes einzelne Byte muß durch Senden eines ACK-Codes (Acknowlegde) bestätigt werden.

#### *Mehrere Bytes*

Umfaßt das Kommando mehrere Bytes, so wird nach dem Empfang des ersten Bytes eine laufende Tastaturabfrage solange ausgesetzt, bis das Kommando vollständig ausgeführt wurde. Wird eines der Bytes als inkorrekt erkannt, so wird ein RESEND-Code zum PC übertragen, und die Kommandoausführung wird abgebrochen. Ausbleibende Bytes bewirken, daß die Tastatur hängenbleibt.

#### *Inkorrekte Bytes*

Auf inkorrekte Bytes reagiert die Tastatur mit einem RESEND-Code anstelle eines ACK-Codes.

#### *Weiterarbeit in der Tastatur*

Es gibt Kommandos, die den Arbeitszustand der Tastatur beeinflussen. Beeinflußt ein Kommando den Arbeitszustand nicht, so wird nach Abschluß der Kommandoausführung die Arbeit gemäß dem jeweiligen Zustand fortgesetzt. Eine eingeschaltete (enabled) Tastenabfrage wird weitergeführt.

#### *Arbeitszustände*

Die Tastatur hat zwei Arbeitszustände:

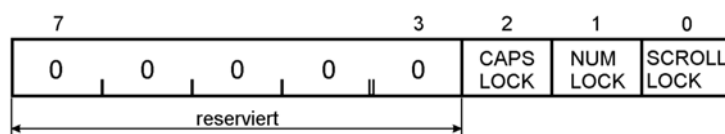
- Enabled. Die Tastenabfrage ist eingeschaltet, Tastenbetätigungen werden an den PC gemeldet.
- Disabled. Die Tastenabfrage ist ausgeschaltet. Tastenbetätigungen werden nicht erkannt.

### 00H...ECH – Ungültige Kommandos

Alle Kommandos, die mit 00H...ECH codiert sind, werden nicht akzeptiert und mit einem Kommando RESEND (FEH) beantwortet.

### EDH – STATUS

Mit diesem Kommando können die LED-Anzeigen der Tastatur umgeschaltet werden. Das Kommando hat zwei Bytes. Nach dem Empfang des zweiten Bytes (Abb. 2.1) werden die LEDs entsprechend angesteuert.



**Abb. 2.1** Byte zur LED-Steuerung

*Hinweis:* CAPS LOCK („Oberlage“ (Upper Case) ständig wirksam) und SCROLL LOCK (Rollmodus ständig wirksam; kaum genutzt) sind reine Anzeigen (die eigentlichen Funktionen werden vom BIOS erbracht). NUM LOCK wirkt hingegen auch in der Tastatur und beeinflusst die Scan-Codes der Cursor- sowie verschiedener Steuertasten (Einzelheiten in Abschnitt 4). Ist NUM LOCK aktiv, wirkt das numerische Tastenfeld als solches (im Sinne der Zifferneingabe), ist NUM LOCK inaktiv, wirken die entsprechenden Zifferntasten dieses Feldes als Cursor- bzw. Steuertasten.

### **EEH – ECHO**

Nach Empfang des Kommandos sendet die Tastatur ein Echo-Byte EEH zum PC zurück.

### **EFH – NOP (No Operation)**

Nach Empfang des Kommandos sendet die Tastatur ein RESEND-Kommando (FEH) zurück.

### **F0H – SETSCAN (Set Alternate Scan Codes)**

Mit diesem Kommando wird in der Tastatur einer von 3 verschiedenen Scan-Codes gewählt (Einzelheiten in Abschnitt 4). Das Kommando hat zwei Bytes. Das zweite Byte kennzeichnet den Scan-Code:

- 00H - keine Änderung. Nach Bestätigung des Kommandos schickt die Tastatur ein Byte zum Controller, das den aktuellen Scan-Code kennzeichnet.
- 01H - Scan-Code 1,
- 02H - Scan-Code 2,
- 03H - Scan-Code 3.

### **F1H – NOP (No Operation)**

Nach Empfang des Kommandos sendet die Tastatur ein RESEND-Kommando (FEH) zurück.

### **F2H – ID (Read Identification)**

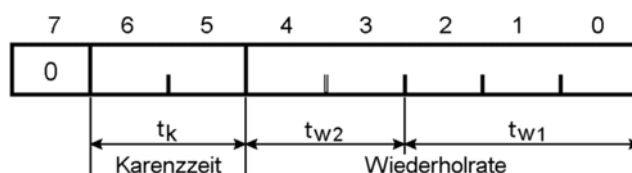
Nach Bestätigung des Kommandos sendet die Tastatur 2 Identifikationsbytes. Die gängigen Werte: 1. Byte: ABH, 2. Byte: 83H<sup>\*</sup>. Das 2. Byte sollte spätestens 500 µs nach dem ersten gesendet werden. In manchen PCs bleibt der Anfangstest (POST) hängen, wenn die Identifikationsbytes nicht eintreffen.

*Hinweise:*

1. Das zuerst übertragene Byte ist das niederwertige. Wird der genannte Identifikationscode als Festwert hingeschrieben (z. B. in einem Testprogramm), lautet er 83ABH.
2. Nach Microsoft (Windows 95) sollte das Kommando keinen Pauschalwert, sondern einen Code zurückliefern, der die Hardware näher kennzeichnet.

### **F3H – TYPEMATIC (Set Typematic Rate)**

Mit diesem Kommando wird sowohl die Betätigungsfrequenz als auch die Karenzzeit für die Typematic-Funktion gewählt (automatische Tastenauslösung bei länger dauerndem Drücken). Das Kommando hat zwei Bytes. Im zweiten Byte sind die genannten Kennwerte codiert (Abb. 2.2, Tabelle 2.1).



$$\text{Karenzzeit} = 250 \text{ ms} \cdot (1 + t_k); \text{ Toleranz } \pm 20\%$$

$$\text{Wiederholperiode } T = (8 + t_{w1}) \cdot 2^{t_{w2}} \cdot 0,00417 \text{ s}$$

$$\text{Wiederholrate} = \frac{1}{T}$$

Festwerte (Default Values): - Karenzzeit: 500 ms  
- Wiederholrate: 10 Betätigungen /s

**Abb. 2.2** Byte zur Typematic-Einstellung

Karenzzeit		Wiederholrate							
Zeit	Wert	W/s	Wert	W/s	Wert	W/s	Wert	W/s	Wert
0,25 s	0	30,0	0	15,0	8	7,5	16	3,7	24
		26,7	1	13,3	9	6,7	17	3,3	25
0,5 s	32	24,0	2	12,0	10	6,0	18	3,0	26
		21,8	3	10,9	11	5,5	19	2,7	27
0,75 s	64	20,0	4	10,0	12	5,0	20	2,5	28
		18,5	5	9,2	13	4,6	21	2,3	29
1 s	96	17,1	6	8,6	14	4,3	22	2,1	30
		16,0	7	8,0	15	4,0	23	2,0	31

W/s = Wiederholungen je Sekunde. Um ein Byte gemäß Abbildung 10.7 aufzusetzen, sind beide Wertangaben zu addieren (Karenzzeit + Wiederholrate)

**Tabelle 2.1** Typematic-Einstellungen

#### **F4H – ENABLE**

Einschalten der Tastenabfrage. Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht und mit der Tastenabfrage begonnen. Trifft ein Kommando ENABLE ein und ist die Tastenabfrage bereits eingeschaltet (enabled), so werden alle bisher gespeicherten Tastenbetätigungen (einschließlich der Typematic-Behandlung gedrückter Tasten) gelöscht.

#### **F5H – DISABLE**

Ausschalten der Tastenabfrage und Rücksetzen in den Grundzustand. Die Wirkung ist dem Rücksetzen vergleichbar. Die Tastatur gelangt in den Grundzustand, wobei für Scan-Code und Typematic-Einstellungen Festwerte (Default Values) gesetzt werden. Die Tastenabfrage bleibt ausgeschaltet (disabled), die Tastatur wartet aber auf weitere Kommandos.

#### **F6H – SET DEFAULT**

Rücksetzen in den Grundzustand. Die Wirkung ist dem Rücksetzen vergleichbar. Die Tastatur gelangt in den Grundzustand, wobei für Scan-Code und Typematic-Einstellungen Festwerte (Default Values)



gesetzt werden (s. weiter unten unter „Grundeinstellungen“). Der Zustand der Tastenabfrage bleibt aber - im Gegensatz zum Kommando F5H - erhalten. War die Tastenabfrage zuvor eingeschaltet (enabled), wird sie fortgesetzt.

**F7H – SAK-T (Set All Keys to Typematic)**

Das Kommando schaltet die Typematic-Funktion für alle Tasten ein. Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**F8H – SAK-MB (Set All Keys to Make/Break)**

Das Kommando bewirkt, daß für alle Tasten sowohl beim Drücken (Make) als auch beim Loslassen (Break) Scan-Codes übertragen werden. Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**F9H – SAK-M (Set All Keys to Make Only)**

Das Kommando bewirkt, daß für alle Tasten Scan-Codes nur beim Drücken (Make) übertragen werden. Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**FAH – SAK-TMB (Set All Keys to Typematic and Make/Break)**

Das Kommando bewirkt, daß für alle Tasten sowohl beim Drücken (Make) als auch beim Loslassen (Break) Scan-Codes übertragen werden. Zudem wird die Typematic-Funktion für alle Tasten eingeschaltet. Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**FBH – SKT-T (Set Key Type to Typematic)**

Das Kommando schaltet die Typematic-Funktion für eine bestimmte Taste ein. Das Kommando hat zwei Bytes. Das zweite Byte enthält den Scan-Code der ausgewählten Taste (es gilt ausschließlich Scan-Code 3.) Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**FCH – SKT-MB (Set Key Type to Make/Break)**

Das Kommando bewirkt, daß für eine bestimmte Taste sowohl beim Drücken (Make) als auch beim Loslassen (Break) Scan-Codes übertragen werden. Das Kommando hat zwei Bytes. Das zweite Byte enthält den Scan-Code der ausgewählten Taste (es gilt ausschließlich Scan-Code 3.) Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**FDH – SKT-M (Set Key Type to Make Only)**

Das Kommando bewirkt, daß für eine bestimmte Taste Scan-Codes nur beim Drücken (Make) übertragen werden. Das Kommando hat zwei Bytes. Das zweite Byte enthält den Scan-Code der ausgewählten Taste (es gilt ausschließlich Scan-Code 3.) Nach Bestätigung des Kommandos wird der Ausgabepuffer (im Controller der Tastatur) gelöscht. Das Kommando betrifft nur Scan-Code 3.

**FEH – RESEND**

Das Kommando fordert die Tastatur auf, das zuvor gesendete Byte erneut zu übertragen (Wiederanlauf im Fehlerfall). War das zuvor gesendete (letzte) Byte ein RESEND-Code (FEH), so wird das vorletzte Byte gesendet. Der PC veranlaßt ein RESEND nur als Antwort auf eine vorausgegangene Übertragung in Richtung Tastatur => PC, und er überträgt dieses Kommando, bevor der Tastatur erlaubt, weitere Übertragungen zu starten (d. h. praktisch, unmittelbar im Anschluß an einen Besetztzustand).

**FFH – RESET**

Rücksetzen der Tastatur und Starten des Selbsttests im Mikrocontroller der Tastatur (Basic Assurance Test BAT). Die Tastatur beantwortet das Kommando mit einem ACK-Code. Der PC muß dies wiederum bestätigen, indem er DATA und CLOCK für wenigstens 500 µs auf High hält (also nicht auf Low zieht). Sofort nach Empfang des RESET-Kommandos schaltet die Tastatur die Tastenabfrage aus (Zustand „disabled“). Trifft vor dem Bestätigen des ACK-Codes (durch das beschriebene High-Halten von DATA und CLOCK) ein weiteres Kommando ein, so wird dieses neue Kommando bearbeitet. Wird hingegen der ACK-Code vom PC bestätigt, führt die Tastatur den BAT aus. Nach Absetzen der Endmeldung (o.k.: AAH, Fehler: FDH) gelangt die Tastatur in den Grundzustand, wobei für Scan-Code und Typematic-Einstellungen Festwerte (Default Values) gesetzt werden. Die Tastenabfrage bleibt ausgeschaltet (disabled), die Tastatur wartet aber auf weitere Kommandos. *Hinweis:* Während der BAT läuft, werden die Belegungen von DATA und CLOCK ignoriert.

**Grundeinstellungen der MF-II-Tastatur (Default Values)**

Scan-Code 1, Typematic-Karenzzeit: 500 ms, Typematic-Wiederholrate: 10 Betätigungen/s (Toleranz der Zeitangaben: ± 20%). Zusätzlich wird noch Num Lock aktiviert.

### 3. Kommandos von der (AT-) Tastatur an den PC

**00H – OVERRUN**

Überlaufanzeige. Dieser Code wird in die letzte Position des Ausgabepuffers eingetragen, wenn dieser voll ist, oder wenn zuviele Tasten gleichzeitig gedrückt sind (Stichwort: Key Rollover). Der Überlaufcode 00H gilt für die Scan-Codes 2 und 3.

**AAH – BAT o.k.**

Dieser Code wird gesendet, wenn der Selbsttests im Mikrocontroller der Tastatur (Basic Assurance Test BAT) erfolgreich durchgelaufen ist. Jede andere Antwort auf das Auslösen des BAT ist ein Fehleranzeichen.

**EEH – ECHO**

Dies ist die Rückmeldung auf ein ECHO-Kommando des PC. Hinweis: Echo-Kommandos werden sofort mit einem Echo-Code EEH bestätigt, ohne daß diesem ein Bestätigungscode ACK (FAH) vorangeht.

**FAH – ACKNOWLEDGE**

Dies ist die Bestätigung jedes gültigen Kommandos mit Ausnahme von ECHO und RESEND. Bricht der PC das Übertragen eines ACK-Codes ab, so wird die Bestätigungsmeldung beendet und das neue, vom PC gesendete Kommando bearbeitet.

**FDH – FAILURE**

Dieser Code wird gesendet, wenn der Mikrocontroller der Tastatur einen Hardwarefehler erkannt hat. Das betrifft sowohl den Selbsttests (Basic Assurance Test BAT) als auch den normalen Betrieb. Üblicherweise beendet die Tastatur nach einem Fehler die Tastenabfrage und wartet auf ein neues Kommando.

**FFH – OVERRUN**

Überlaufanzeige. Dieser Code wird in die letzte Position des Ausgabepuffers eingetragen, wenn dieser voll ist, oder wenn zuviele Tasten gleichzeitig gedrückt sind (Stichwort: Key Rollover). Der Überlaufcode FFH gilt für Scan-Code 1.

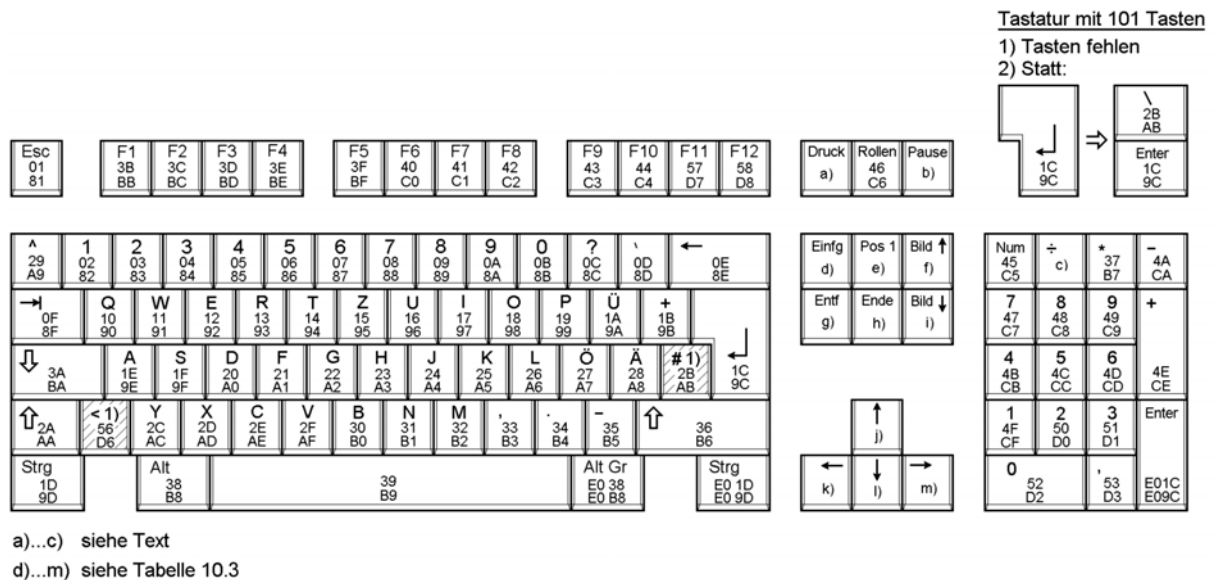
## 4. Tastaturcodes

### Scan-Codes

Sowohl das Drücken als auch das Loslassen einer Taste äußert sich in einem Scan-Code. Der beim Drücken erzeugte Scan-Code heißt Make-Code, der beim Loslassen erzeugte Break-Code. Die von der Tastatur gelieferten Scan-Codes sind von Tastaturtyp zu Tastaturtyp verschieden. Die MF-II-Tastatur ermöglicht es, zwischen drei verschiedenen Code-Zuordnungen zu wählen (Scan Code 1, 2, 3). Die Abbildungen 4.1 bis 4.4 veranschaulichen die drei verschiedenen Scan-Codes.

#### Hinweise:

- Die gezeigten Layouts betreffen die MF-II-Tastatur mit 102 Tasten. Zusätzlich ist dargestellt, worin sich die Tastatur mit 101 Tasten davon unterscheidet: (1) es fehlen je eine Taste rechts neben der linken Shift-Taste und links neben der Enter-Taste, (2) die Enter-Taste belegt nur eine Tasten-Reihe, darüber ist eine weitere Taste vorgesehen.
- Die Beschriftung der Tastenköpfe ist jeweils oben angegeben. Enthält ein Tastenkopf mehrere Zeichen oder Symbole, so ist nur jenes dargestellt, das der „Unterlage“ (Lower Case) entspricht.
- Alle Codes werden hexadezimal dargestellt. Codes, die mehrere Bytes lang sind, unterliegen bestimmten Bildungsregeln (aus den folgenden Tabellen 4.1 und 4.2 unschwer zu erkennen). Im folgenden sind aber alle Codes angegeben.



**Abb. 4.1** MF-II-Tastatur: Scan-Code 1 (XT)

In den Tastensymbolen ist zunächst der Make- und darunter der Break-Code angegeben. Der Break-Code entspricht grundsätzlich dem Make-Code, wobei zusätzlich Bit 7 gesetzt ist.

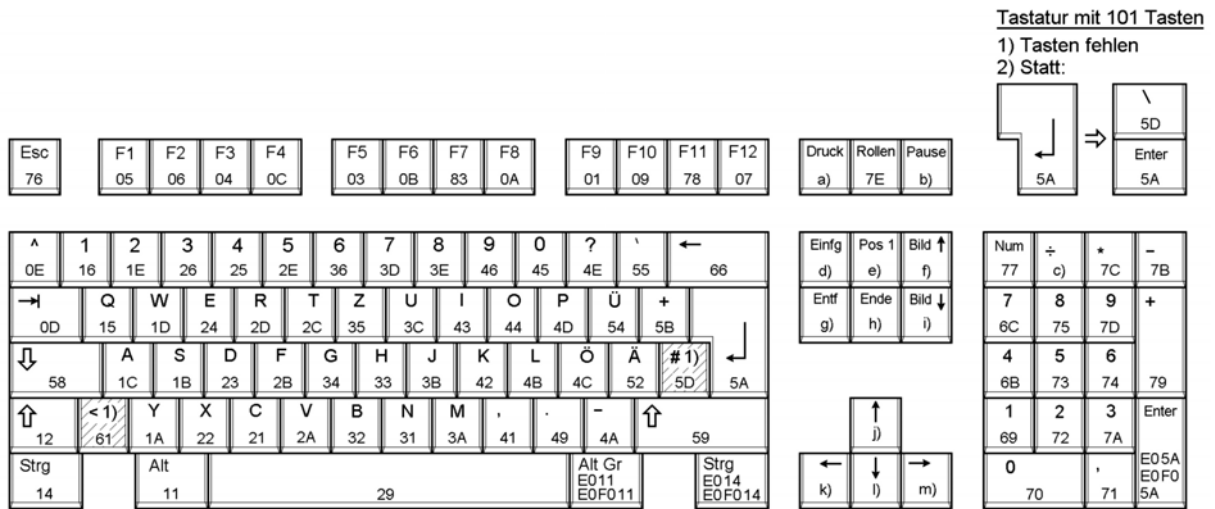
in Ab- bildung	Taste	Num Lock aktiv		Num Lock inaktiv	linke Shift-Taste <sup>1)</sup>		rechte Shift-Taste <sup>1)</sup>	
d)	Einfg	E0 2A E0 D2	E0 52 E0 AA	E0 52 E0 D2	E0 AA E0 D2	E0 52 E0 2A	E0 B6 E0 D2	E0 52 E0 36
e)	Pos 1	E0 2A E0 C7	E0 47 E0 AA	E0 47 E0 C7	E0 AA E0 C7	E0 47 E0 2A	E0 B6 E0 C7	E0 47 E0 36
f)	Bild 8	E0 2A E0 C9	E0 49 E0 AA	E0 49 E0 C9	E0 AA E0 C9	E0 49 E0 2A	E0 B6 E0 C9	E0 49 E0 36
g)	Entf	E0 2A E0 D3	E0 53 E0 AA	E0 53 E0 D3	E0 AA E0 D3	E0 53 E0 2A	E0 B6 E0 D3	E0 53 E0 36
h)	Ende	E0 2A E0 CF	E0 4F E0 AA	E0 4F E0 CF	E0 AA E0 CF	E0 4F E0 2A	E0 B6 E0 CF	E0 4F E0 36
i)	Bild 9	E0 2A E0 D1	E0 51 E0 AA	E0 51 E0 D1	E0 AA E0 D1	E0 51 E0 2A	E0 B6 E0 D1	E0 51 E0 36
j)	8	E0 2A E0 C8	E0 48 E0 AA	E0 48 E0 C8	E0 AA E0 C8	E0 48 E0 2A	E0 B6 E0 C8	E0 48 E0 36
k)	7	E0 2A E0 CB	E0 4B E0 AA	E0 4B E0 CB	E0 AA E0 CB	E0 4B E0 2A	E0 B6 E0 CB	E0 4B E0 36
l)	9	E0 2A E0 D0	E0 50 E0 AA	E0 50 E0 D0	E0 AA E0 D0	E0 50 E0 2A	E0 B6 E0 D0	E0 50 E0 36
m)	6	E0 2A E0 CD	E0 4D E0 AA	E0 4D E0 CD	E0 AA E0 CD	E0 4D E0 2A	E0 B6 E0 CD	E0 4D E0 36

1): ist Num Lock aktiv, bewirkt das Niederhalten einer Shift-Taste, daß die Codes gemäß „Num Lock inaktiv“ geliefert werden.

**Tabelle 4.1** Steuer- und Cursortasten bei Scan-Code 1. Jeweils oben Make-Code, darunter Break-Code

#### Besondere Make- und Break-Codes:

- Druck.** Make = E0 2A E0 37, Break = E0 B7 E0 AA. **Druck + Strg:** Make = E0 37, Break = E0 B7. **Druck + Alt (bzw. Alt Gr):** Make = 54, Break = D4.
- Pause.** Make = E1 1D 45 E1 9D C5. Kein Break-Code. **Pause + Strg:** Make = E0 46 E0 C6. Kein Break-Code. (Tastenfunktion: Programmunterbrechung/Break.)
- Taste ÷ .** Make = E0 35, Break = E0 B5. Wenn linke Shift-Taste niedergehalten: Make = E0 AA E0 35, Break = E0 B5 E0 2A. Wenn rechte Shift-Taste niedergehalten: Make = E0 B6 E0 35, Break = E0 B5 E0 36.



Die angegebenen Codes sind Make-Codes.

Break-Codes ist ein weiteres Byte F0 vorangestellt. Beispiel: Taste Q. Make 15, Break F0 15.

- a)...c) siehe Text
- d)...m) siehe Tabelle 10.4

**Abb. 4.2** MF-II-Tastatur: Scan-Code 2 (AT)

In den Tastensymbolen ist nur der Make-Code angegeben. Der Break-Code besteht aus zwei Bytes (Festwert F0 + Make-Code). Ausnahmen: die Tastensymbole für Alt Gr, Strg (rechts) und Enter (im numerischen Tastenfeld) enthalten sowohl die Make- als auch (darunter) die Break-Codes.

**Besondere Make- und Break-Codes:**

- a) **Druck.** Make = E0 12 E0 7C, Break = E0 F0 7C E0 F0 12. **Druck + Strg:** Make = E0 7C, Break = E0 F0 7C. **Druck + Alt (bzw. Alt Gr):** Make = 84, Break = F0 84.
- b) **Pause.** Make = E1 14 77 E1 F0 14 77. Kein Break-Code. **Pause + Strg:** Make = E0 7E E0 F0 7E. Kein Break-Code. (Tastenfunktion: Programmunterbrechung/Break.)
- c) **Taste ÷ .** Make = E0 4A, Break = E0 F0 4A. Wenn linke Shift-Taste niedergehalten: Make = E0 F0 12 E0 4A, Break = E0 F0 4A E0 12. Wenn rechte Shift-Taste niedergehalten: Make = E0 F0 59 E0 4A, Break = E0 F0 4A E0 59.
- d)...m) **Einfg, Pos1** usw. sowie Cursortasten. Siehe Tabelle 4.2.

in Abbildung	Taste	Num Lock aktiv	Num Lock inaktiv	Linke Shift-Taste	rechte Shift-Taste
d)	Einfg	E0 12 E0 70 E0 F0 70 E0 F0 12	E0 70 E0 F0 70	E0 F0 12 E0 70 E0 F0 70 E0 12	E0 F0 59 E0 70 E0 F0 70 E0 F0 12
e)	Pos 1	E0 12 E0 6C E0 F0 70 E0 F0 12	E0 6C E0 F0 6C	E0 F0 12 E0 6C E0 F0 6C E0 12	E0 F0 59 E0 6C E0 F0 6C E0 F0 12
f)	Bild 8	E0 12 E0 7D E0 F0 7D E0 F0 12	E0 7D E0 F0 7D	E0 F0 12 E0 7D E0 F0 7D E0 12	E0 F0 59 E0 7D E0 F0 7D E0 F0 12
g)	Entf	E0 12 E0 71 E0 F0 71 E0 F0 12	E0 71 E0 F0 71	E0 F0 12 E0 71 E0 F0 71 E0 12	E0 F0 59 E0 71 E0 F0 71 E0 F0 12

in Abbildung	Taste	Num Lock aktiv	Num Lock inaktiv	Linke Shift-Taste	rechte Shift-Taste
h)	Ende	E0 12 E0 69 E0 F0 69 E0 F0 12	E0 69 E0 F0 69	E0 F0 12 E0 69 E0 F0 69 E0 12	E0 F0 59 E0 69 E0 F0 69 E0 F0 12
i)	Bild 9	E0 12 E0 7A E0 F0 7A E0 F0 12	E0 7A E0 F0 7A	E0 F0 12 E0 7A E0 F0 7A E0 12	E0 F0 59 E0 7A E0 F0 7A E0 F0 12
j)	8	E0 12 E0 75 E0 F0 75 E0 F0 12	E0 75 E0 F0 75	E0 F0 12 E0 75 E0 F0 75 E0 12	E0 F0 59 E0 75 E0 F0 75 E0 F0 12
k)	7	E0 12 E0 6B E0 F0 6B E0 F0 12	E0 6B E0 F0 6B	E0 F0 12 E0 6B E0 F0 6B E0 12	E0 F0 59 E0 6B E0 F0 6B E0 F0 12
l)	9	E0 12 E0 72 E0 F0 72 E0 F0 12	E0 72 E0 F0 72	E0 F0 12 E0 72 E0 F0 72 E0 12	E0 F0 59 E0 72 E0 F0 72 E0 F0 12
m)	6	E0 12 E0 74 E0 F0 74 E0 F0 12	E0 74 E0 F0 74	E0 F0 12 E0 74 E0 F0 74 E0 12	E0 F0 59 E0 74 E0 F0 74 E0 F0 12

**Tabelle 4.2** Steuer- und Cursortasten bei Scan-Code 2. Jeweils oben Make-Code, darunter Break-Code



Die angegebenen Codes sind Make-Codes.

Break-Codes ist ein weiteres Byte F0 vorangestellt. Beispiel: Taste Q. Make 15, Break F0 15.

- - Tasten sind "Make-Only". Beim Loslassen wird kein Break-Code gesendet. Typematic ist unwirksam.
- - Typematic ist unwirksam.

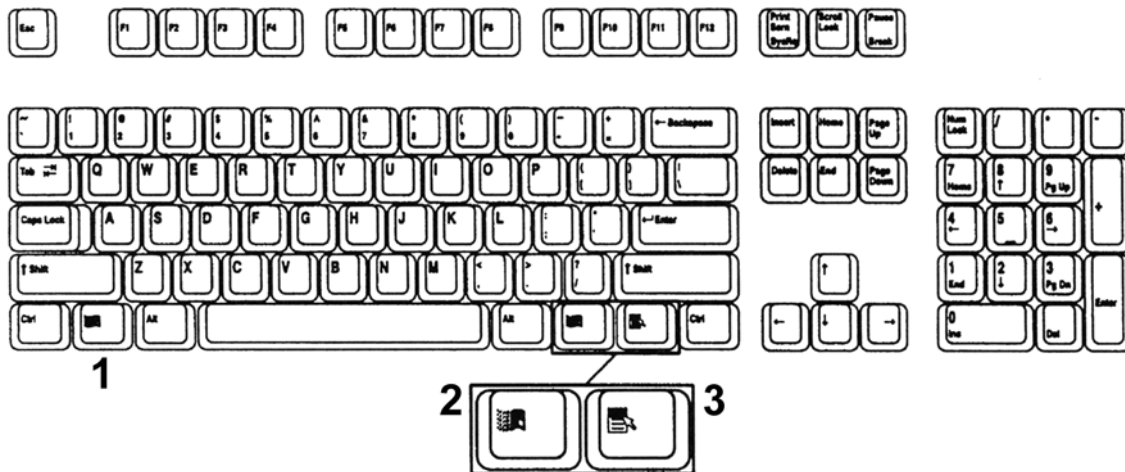
**Abb. 4.3** MF-II-Tastatur: Scan-Code 3 (PS/2)

**Hinweise:**

1. In den Tastensymbolen ist nur der Make-Code angegeben. Der Break-Code besteht aus 2 Bytes (Festwert F0 + Make-Code).
2. Die Abbildung zeigt die Grundeinstellung. Die Scan-Codes und die Typematic-Funktionen aller Tasten können durch Kommandos vom PC aus verändert werden (vgl. Abschnitt 3).

**Die Tastatur seit Windows 95**

Microsoft hatte seinerzeit ein Layout mit 3 zusätzlichen Tasten empfohlen (Abb. 4.4, Tabelle 4.3).



**Abb. 4.4** Tastaturlayout für Windows 95 (nach: Microsoft). Die zusätzlichen Tasten: 1 - linke Windows-Taste; 2 - rechte Windows-Taste; 3 - Anwendungstaste

Scan-Code	Linke Windows-Taste		Rechte Windows-Taste		Anwendungstaste	
	Make	Break	Make	Break	Make	Break
1; Unterlage <sup>1)</sup>	E0 5B	E0 DB	E0 5C	E0 DC	E0 5D	E0 DD
1; Oberlage	E0 AA E0 5B	E0 DB E0 2A	E0 AA E0 5C	E0 DC E0 2A	E0 AA E0 5D	E0 DD E0 2A
1; Num Lock aktiv	E0 2A E0 5B	E0 DB E0 AA	E0 2A E0 5C	E0 DC E0 AA	E0 2A E0 5D	E0 DD E0 AA
2; Unterlage <sup>1)</sup>	E0 1F	E0 F0 1F	E0 27	E0 F0 27	E0 2F	E0 F0 2F
2; Oberlage	E0 F0 12 1F	E0 F0 1F E0 12	E0 F0 12 27	E0 F0 27 E0 12	E0 F0 12 2F	E0 F0 2F E0 12
2; Num Lock aktiv	E0 12 E0 1F	E0 F0 1F E0 F0 12	E0 12 E0 27	E0 F0 27 E0 F0 12	E0 12 E0 2F	E0 F0 2F E0 F0 12
3	8B	F0 8B	8C	F0 8C	8D	F0 8D

1) auch: Num Lock + Shift

**Tabelle 4.3** Scan-Codes der zusätzlichen Windows-Tasten

## 5. Die PS/2-Maus

### Betriebsweisen

In der Maus sind 2 Betriebsweisen einstellbar:

- Streaming Mode. Alle Mausbewegungen werden ständig an den PC gemeldet, und zwar in einem festen Intervall (Report Rate). (Die normale Betriebsweise nach dem Zurücksetzen.)
- Remote Mode. Bewegungsmeldungen werden nur als Antwort auf ein Lesekommando (READ) zum PC übertragen.

**Auflösung**

Mäuse übertragen Koordinatenangaben für die X- und Y-Richtung der Bewegung. Die Auflösung kennzeichnet jene Wegstrecke, die die Maus gerade noch erkennen kann, die sich also in der niedrigstwertigen Bitposition einer Koordinatenangabe bemerkbar macht. Typische Auflösungen sind:

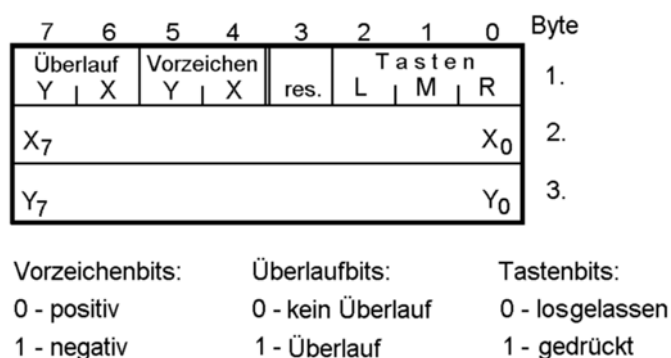
- 1/200 Zoll bzw. 200 dpi (PS/2, Apple),
- 1/400 Zoll bzw. 400 dpi (Microsoft).

*Hinweise:*

1. Die genauen Auflösungen der Original-Mäuse: IBM PS/2: 203,718 dpi; Microsoft: 417,07 dpi; Apple: 197,147 dpi.
2. 1/400" (0,0635 mm) wird sinnigerweise als „1 Mickey“ bezeichnet.
3. Die zum PC übertragenen Bewegungsangaben betreffen üblicherweise einen Bereich von - 128... + 127 Schritten (bzw. Mickeys) in jeder Koordinate (X und Y).

**Übertragung der Bewegungsmeldungen**

Bewegungsangaben sind in Form eines fest formatierten Paketes aus 3 Bytes untergebracht (Abb. 5.1).



**Abb. 5.1** Bewegungsangaben der PS/2-Maus

**Einschaltrücksetzen**

Das Einschalttrücksetzen wirkt wie ein Kommando RESET (FFH).

**Kommandosteuerung**

Die Maus wird wie die Tastatur über Kommandos gesteuert. Zum Signalspiel auf dem Interface siehe Abschnitt 1. Es folgen die einzelnen Kommandos.

**E6H – SET 1:1 SCALING**

Der Bewegungsmaßstab wird auf 1:1 gesetzt. Die Bewegungsdaten gemäß Abb. 5.1 entsprechen unmittelbar der Mausbewegung gemäß der jeweils eingestellten Auflösung (Tabelle 5.1).

**E7H – SET 2:1 SCALING**

Der Bewegungsmaßstab wird auf 2:1 gesetzt (vergrößerte Auflösung; 1 aufgenommener Bewegungsimpuls = Erhöhung bzw. Verminderung der Koordinatenangabe um 2). Nicht alle Mäuse führen das Kommando aus (sie geben aber trotzdem die Bestätigung (FAH) zurück).



**E8H – SET RESOLUTION**

Es folgt ein weiteres Byte, das die Auflösung der Mausbewegung kennzeichnet (Tabelle 5.1). Manche Mäuse akzeptieren das Kommando, verändern aber ihre Auflösung nicht.

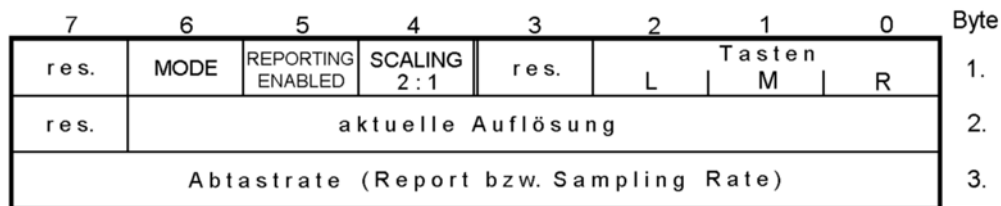
Auflösungsangabe	Teilverhältnis *)	Auflösung	
		Impulse/mm	dpi
00H	n : 8	1	25
01H	n : 4	2	50
02H	n : 2	4	100
03H	n : 1	8	200

\*) : n = Anzahl der tatsächlich abgetasteten Impulse

**Tabelle 5.1** Auflösung der Mausbewegung

**E9H – STATUS REQUEST**

Die Maus schickt als Antwort eine Zustandsmeldung zum PC, die aus 3 Bytes besteht (Abb. 5.2).



MODE:

0 - Streaming Mode

1 - Remote Mode

REPORTING ENABLED:

0 - zyklische Bewegungsmeldung ausgeschaltet

1 - zyklische Bewegungsmeldung eingeschaltet

Tastenbits:

0 - losgelassen

1 - gedrückt

**Abb. 5.2** Zustandsmeldung

**EAH – SET STREAMING MODE**

Die Betriebsweise „Streaming Mode“ (s. oben) wird eingeschaltet.

**EBH – READ DATA**

Auf dieses Kommando hin wird eine Bewegungsmeldung (gemäß Abb. 5.1) zum PC geschickt. Das Kommando wirkt in beiden Betriebsweisen.

**ECH – RESET ECHO (WRAP) MODE**

Der Echo- bzw. Wrap-Modus (s. das folgende Kommando) wird ausgeschaltet.

**EEH – SET ECHO (WRAP) MODE**

Der Echo- bzw. Wrap-Modus wird eingeschaltet. In dieser Betriebsart liefert die Maus jedes empfangene Kommando- oder Datenbyte an den PC zurück. Ausnahmen: die Kommandos RESET und RESET ECHO MODE.

**F0H - SET REMOTE MODE**

Die Betriebsweise „Remote Mode“ (s. oben) wird eingeschaltet.

**F2H – READ DEVICE TYPE**

Es wird ein Byte als Geräteerkennung zum PC geschickt (üblicher Wert: 00H).

**F3H – SET REPORT RATE**

Es folgt ein weiteres Byte, das angibt, in welchem Rhythmus Bewegungsmeldungen zum PC geschickt werden (im Streaming Mode). Siehe Tabelle 5.2. *Achtung:* Manche Mäuse akzeptieren das Kommando, stellen aber eine feste Abtastrate (Report Rate) ein (typischerweise 40/s).

Report Rate (je Sekunde)	Wertangabe	Report Rate (je Sekunde)	Wertangabe
10	0AH	80	50H
20	14H	100	64H
40	2BH	200	CBH
60	3CH		

**Tabelle 5.2** Einstellung der Report Rate (Abtast- bzw. Samplingrate; Bewegungsmeldungen je Sekunde)

**F4H – ENABLE REPORTING**

Mit diesem Kommando wird das zyklische Senden von Bewegungsmeldungen (im Streaming Mode, Intervalle gemäß Tabelle 5.2) angeschaltet. Im Remote Mode ist das Kommando wirkungslos.

**F5H – DISABLE REPORTING**

Dieses Kommando schaltet das zyklische Senden von Bewegungsmeldungen im Streaming Mode aus. Die Maus antwortet aber noch auf andere Kommandos. In diesem Zustand („Reporting Disabled“) gilt: soll ein Kommando zur Maus geschickt werden, das Rückgabewerte erwartet (z. B. STATUS REQUEST), so ist zuvor auch Streaming Mode auszuschalten (Kommando F0H).

**F6H – SET DEFAULT**

Dieses Kommando setzt die Maus in einen Grundzustand (Tabelle 5.3). Es wird kein Selbsttest ausgelöst.

Abtastrate (Report Rate):	100 Bewegungsmeldungen/s <sup>*)</sup>
Bewegungsmaßstab (Scaling):	1 : 1
Betriebsweise:	Streaming Mode
Auflösung:	4 Impulse/mm (100) dpi
Zyklische Bewegungsmeldung (Reporting):	ausgeschaltet (disabled)

\*) : Manche Mäuse haben eine konstante Abtastrate von 40/s.

**Tabelle 5.3** Grundzustand

**FEH – RESEND**

Das Kommando fordert die Maus auf, das zuvor gesendete Byte erneut zu übertragen. Zum Ablauf im einzelnen siehe die Kommandobeschreibung in Kapitel 10. Hinweis: Die Maus liefert (wie die Tastatur) einen RESEND-Code (FEH) an den PC, wenn ein empfangenes Byte als inkorrekt erkannt wurde.

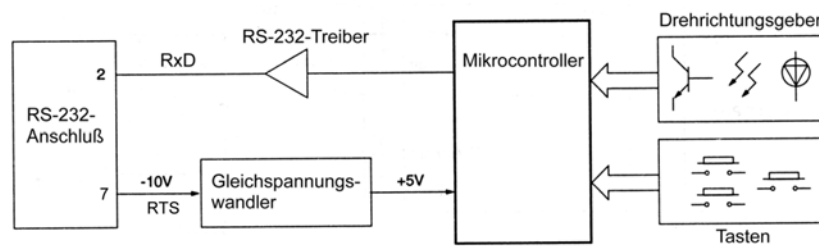
**FFH – RESET**

Rücksetzen der Maus-Hardware. Die Maus gelangt in den Grundzustand gemäß Tabelle 5.3. Sie bestätigt das Kommando zunächst mit FAH und meldet das Ende des Rücksetzablaufs durch zwei aufeinanderfolgende Bytes AAH und 00H.

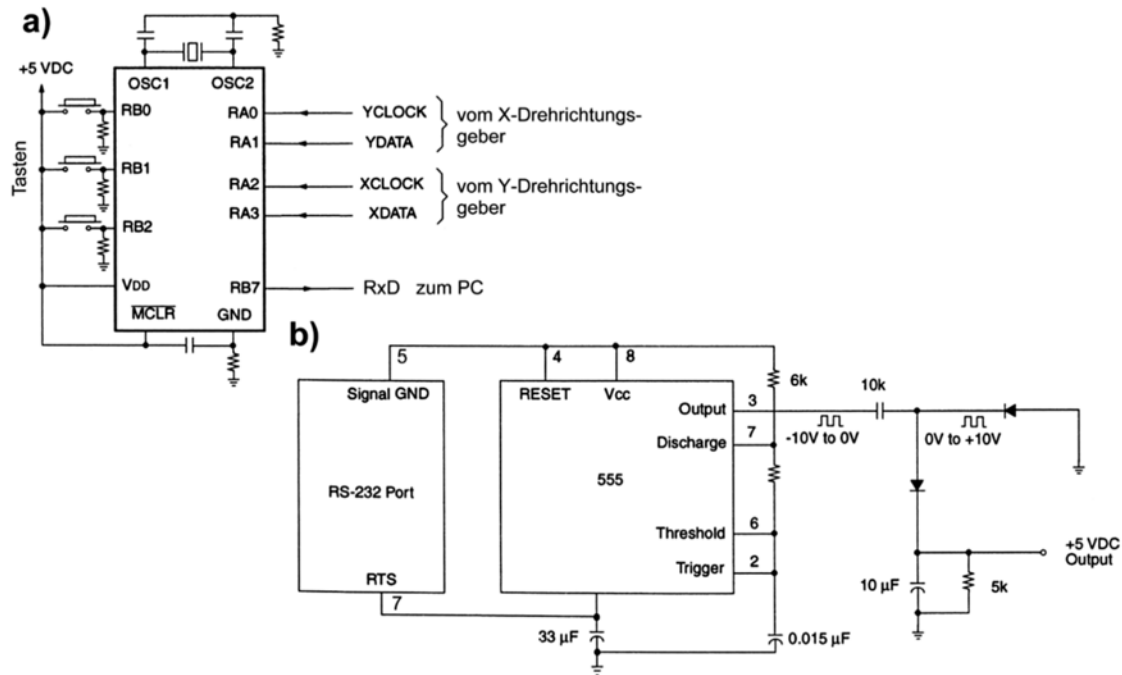
**6. Anhang: Die serielle Maus****Maus-Elektronik**

Die Steuerelektronik einer Maus besteht üblicherweise aus einem einzigen Schaltkreis (das ist typischerweise ein Mikrocontroller), der aus der seriellen Schnittstelle mit Betriebsspannung versorgt wird (Abb. 6.1 bis 6.3). Die typische Datenrate auf dem seriellen Interface beträgt 1200 Bits/s. Das Interface selbst hat oft nur 3 Leitungen:

- Signalmasse (GND).
- RxD. Die Maus sendet (als DCE), der PC empfängt (als DTE).
- RTS oder DTR als Spannungsversorgung. Diese Signale werden vom PC ständig auf Mark-Pegel (typischerweise -12 V) gehalten. Stromaufnahme der Maus: typischerweise unter 10 mA.

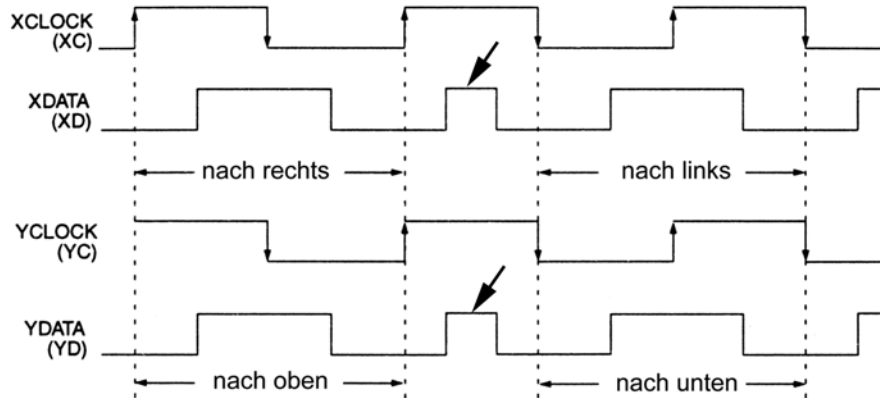


**Abb. 6.1** Blockschaltbild einer seriellen Maus (Microchip)



**Abb. 6.2** Weitere Einzelheiten (Microchip). a) - die Beschaltung des Mikrocontrollers; b) - Spannungsversorgung.

Der Mikrocontroller ist direkt mit den Tasten und den photoelektrischen Drehrichtungsgebern verbunden. Die Taktfrequenz liegt typischerweise bei 4 MHz. Die negative Spannung des Schnittstellensignals RTS wird mittels einer Ladungspumpe in + 5 V gewandelt. Grundlage der Ladungspumpe ist hier ein Zeitgeberschaltkreis 555. Die Stromaufnahme der Maus beträgt maximal 10 mA. Die Schaltung arbeitet noch bei RTS-Pegeln von - 10 V.

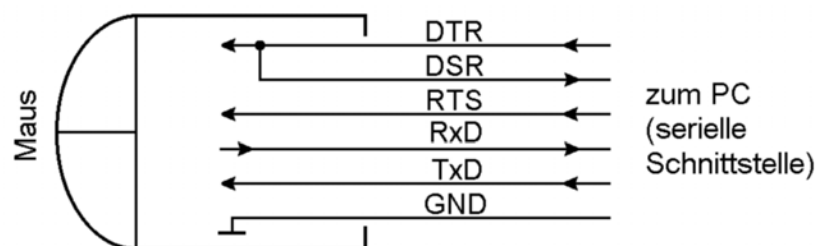


**Abb. 6.3** Die Signale der Drehrichtungsgeber ( Microchip)

Die Geber liefern jeweils zwei Signale, die gegeneinander um 90° phasenverschoben sind. Das Vorzeichen der Phasenverschiebung ergibt sich aus der Drehrichtung. Bei Drehrichtungswechsel tritt ein Phasensprung auf (die schmalen Impulse (Pfeil) auf den Leitungen XDATA, YDATA in der Mitte des Diagramms).

### Die serielle Maus seit Windows 95

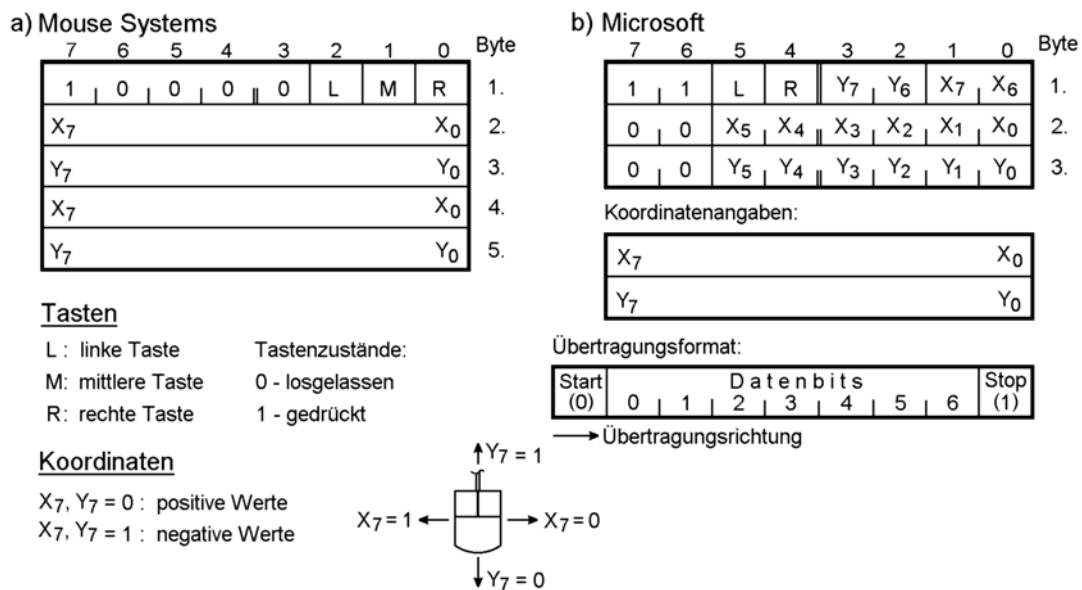
Abb. 6.4 zeigt, wie die Maus an das Interface anzuschließen ist. Wie jedes andere Gerät an einer seriellen Schnittstelle muß die Maus DSR aktivieren, um anzuzeigen, daß sie angeschlossen ist. Das wird auf einfachste Weise durch Verbinden von DSR und DTR erreicht. Die Maus muß RTS abfragen. Ist RTS inaktiv, verbleibt sie in einem Wartezustand. Wird RTS aktiv, sendet die Maus ihre Geräte-Identifikationsdaten. Anschließen wird der Normalbetrieb aufgenommen. Wird RTS dabei inaktiv, schaltet die Maus in den Wartezustand zurück.



**Abb. 6.4** Das serielle Maus-Interface gemäß den Windows-95-Anforderungen (Microsoft)

### Übertragungsformate

Es gibt verschiedene Übertragungsformate (Abb. 6.5). In mehreren fest formatierten Bytes wird sowohl die Tastenbetätigung als auch der zurückgelegte Weg in beiden Koordinaten übertragen. Der Weg wird in dpi bzw. Mickeys gemessen und als vorzeichenbehaftete Binärzahl dargestellt. Bereichsüberschreitungen (Wrap Around) müssen von der Maustreibersoftware behandelt werden.



**Abb. 6.5** Die übertragene Information (serielle Maus)

*Hinweise:*

1. Die Abbildung zeigt zwei verbreitete Übertragungsformate. Dabei sind die Bytes so dargestellt, wie sie ein auswertendes Programm im RAM vorfindet.
2. Mouse-Systems-Format: Die x- und y-Koordinaten werden zweimal übertragen.
3. Microsoft-Format: Die Microsoft-Mäuse übertragen nach dem Einschalttrücksetzen zunächst ein Zeichen „M“ (4DH) an den PC.
4. Serielle Übertragung: 1200 Bits/s, 1 Startbit, 1 Stopbit. Mouse Systems: 8 Datenbits, Microsoft: 7 Datenbits. Bit 0 wird zuerst übertragen.
5. Auf dem seriellen Interface hat die logische 1 negativen Pegel (Mark).