Timing accuracy of mouse response registration on the IBM microcomputer family

JÖRG BERINGER

Institut für Psychologie, Technische Hochschule Darmstadt, Darmstadt, Germany

Data and theoretical estimates reported by Segalowitz and Graves (1990) and Crosbie (1990) regarding delays of serial Microsoft mouse response registration are corrected and extended. Empirically measured mouse responses clearly confirm a theoretical minimum delay time of 22.5 msec for Microsoft-compatible serial mouse devices. Timing accuracy of key actions for the serial, bus, and PS/2 mouse devices are presented.

SERIAL MOUSE

Segalowitz and Graves (1990) present timing details for the serial mouse and recommendations concerning time corrections. Although the reported delay times of the serial Microsoft mouse seem to be valid, the theoretical estimates of the transfer rate of the Microsoft mouse were based on an incorrect assumption. The Microsoft mouse transfers a 3-byte information block at 1,200 baud, with 7-bit word length, and 1 start and 1 stop bit. Thus, 27 bits ($9 \text{ bits} \times 3$) are to be transferred for each information block, and this takes at least 22.4999 msec.

Segalowitz and Graves (1990) assumed a 5-byte block similar to the Mouse Systems mouse. On the basis of that assumption, and ignoring the data format of serial communication, they derived the number of bits to be transferred on each block ($5 \times 8 = 40$) that would produce a minimum delay of 33 msec. In addition to the wrong generalization from the Mouse Systems mouse to the Microsoft mouse protocol, the published measurements also contain inconsistencies. For example, although the minimum transfer delay is estimated at 33 msec, the authors reported 25 msec as the shortest recorded interkey interval, and a mean delay of 31 msec.

To determine the minimum delay, both mouse keys of a Microsoft-compatible NCE Hypermouse 860/PC were electronically connected via an open collector circuit with the parallel interface of an 8-MHz AT clone. Mechanical influences of the mouse buttons were therefore not included in the tests. Standard features of the Experimental RunTime System (ERTS)¹ were used for validating mouse response registration. ERTS includes output via the parallel interface as well as interrupt-driven mouse registra-

tion via a user-defined event handler (Microsoft, 1986; Rinck, 1988). Trials were defined in such a way that ERTS itself triggered the mouse responses 100 msec after the beginning of response time registration.

A moving mouse was also found to distort the delay times by as much as one information block (Segalowitz & Graves, 1990). All measurements were therefore made with the mouse ball removed. The mouse cursor was turned off in all test situations to avoid extra execution times of the mouse driver. Cursor updating may cost considerable time and is even sometimes synchronized with the vertical retrace.

Table 1 lists the recorded mouse registration times for the NCE mouse and Version 6.13 of the Microsoft mouse driver. Keypress times were varied in 5-msec steps, from 5 to 35 msec. The true response-delay time was 100 msec. On the average, observed values for the make action (key pressed) are delayed by 24.4 msec. This result is clearly in line with the theoretical assumptions of a minimum transfer delay of 22.5 msec. The shortest observed delay is 23.9 msec, which is slightly beyond the theoretical minimum. The 1.4-msec difference between the expected delay and the observed delay seems to be due to the execution times of the mouse driver, the trigger/record delay of the ERTS, and the mouse's processor logic. The minimum registration times for break actions (key released) are listed in the lower part of Table 1. The high values (24.6 msec) show that the interval between consecutive make/break actions cannot be smaller than the minimum transfer rate.

For analyzing multiple key actions in more detail, another test situation was designed in which the interkey interval (IKI) between two consecutive keypresses was varied. The results are listed in Table 2. The observed IKI for simultaneous keypresses ($\text{IKI} = 0 \text{ msec}$) is zero. When both keys are pressed at the same time, the information is sent to the mouse driver within one information block. Longer IKIs are transferred sequentially as two separate information blocks. Thus, for expected IKIs between -20 msec and $+20 \text{ msec}$, constant IKIs of 24.6 msec are observed. IKIs longer than the minimum transfer delay

The author thanks W.-D. Mueller at Technische Hochschule Darmstadt, and Nordphon Computer Electronic NCE, 2399 Tarp, Germany, for technical support. Requests for reprints should be addressed to J. Beringer, Institut für Psychologie, Technische Hochschule Darmstadt, Hochschulstrasse 1, 6100 Darmstadt, Germany (e-mail: xbr1d80v@ddatht21.bitnet).

Table 1
Mean Mouse Registration Times (in Milliseconds) for a Single Key of a NCE Hypermouse
for Make (Key Pressed) and Break (Key Released) Action

Key Action	Keypress Time	No. Keypresses	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
Make	5	100	124.412	0.254	123.9	124.9	1.0
Make	10	100	124.371	0.237	123.9	124.8	0.9
Make	15	100	124.390	0.250	124.0	124.8	0.8
Make	20	100	124.375	0.265	123.9	124.9	1.0
Make	25	100	124.396	0.271	123.9	124.9	1.0
Make	30	100	124.414	0.272	124.0	124.9	0.9
Make	35	100	124.377	0.262	123.9	124.9	1.0
Break	5	100	24.670	0.046	24.6	24.7	0.1
Break	10	100	24.665	0.048	24.6	24.7	0.1
Break	15	100	24.673	0.047	24.6	24.8	0.2
Break	20	100	24.668	0.049	24.6	24.8	0.2
Break	25	100	25.254	0.430	24.6	25.6	1.0
Break	30	100	30.192	0.345	30.0	31.0	1.0
Break	35	100	35.255	0.335	34.4	35.5	1.1

Note—True response delay was 100 msec.

are recorded in correspondence with their true interval time.

To determine the reasons for the deviations between the observed average response delay of 31 msec reported by Segalowitz and Graves (1990) and the delay of 24.4 msec presented in Tables 1 and 2, the time behavior of an original Microsoft mouse was analyzed. The duration of a keypress was varied in steps of 1 msec between 5 and 11 msec (see Table 3).

The data in Table 3 show that keypresses shorter than 7 msec are not registered at all. The Microsoft mouse's processor logic obviously includes a time threshold of about 7 msec to overcome bouncing key contacts. This threshold adds 7 msec to the theoretical minimum transfer delay of 22.5 msec. The observed minimum transfer delay is $132.9 - 100 = 32.9$ msec (upper part of Table 3). Thus, the delay times of the serial Microsoft mouse reported by Segalowitz and Graves (1990) and Crosbie (1990) in the range of 31–33 msec are confirmed. Subtracting the threshold from the delay time $32.9 - 7 = 25.9$ msec leaves 1.3 msec with reference to the observed

minimum IKI of 24.6 msec (lower part of Table 3). This residual cannot be explained.

The mouse-dependent results show that the differences between mouse hardware must not be neglected. Although measurements of the serial Microsoft mouse performance agree with results from other papers (Crosby, 1990; Segalowitz & Graves, 1990, there is no guarantee that there will be no variations between serial Microsoft mice with respect to the internal logic. Individual testing of installed equipment is therefore recommended. Regardless of the brand, one can be sure that the minimum transfer delay of 22.5 msec is always valid for Microsoft-compatible mice. To complete the tests of mouse response registrations, the time behavior of a Microsoft bus mouse and a PS/2 mouse were also investigated.

BUS MOUSE

According to Crosbie (1990), the bus mouse is expected to be rather precise with respect to variations. Table 4 lists the measurements of a Microsoft bus mouse using a Microsoft mouse driver V6.25Z on a Toshiba 3200. The mouse driver was loaded with default options. A delay of 33 msec was therefore expected, which corresponds to the default interrupt rate of 30 Hz (Microsoft, 1986).

The values in Table 4 show the minimum observed delay time (upper half) as well as the minimum observed make/break interval (lower half). The observed minimum make/break interval is exactly 30.8 msec. This is slightly faster than 30 Hz. The observed minimum delay time is only delayed by 0.8 msec, compared with the expected 100-msec response latency. From this it follows that the bus mouse theoretically transfers keypresses immediately, but its state is sampled only at the frequency of the interrupt rate. Keypresses shorter than one interinterrupt interval may not be registered if the keypress falls between two interrupts. This is shown in Table 4, which lists 100% of registrations only for keypresses longer than 30 msec. Observed mean response times for keypresses are listed

Table 2
Mean Mouse Registration Times (in Milliseconds) for the Interkey
Interval (IKI) Between Two Keypresses of an NCE Hypermouse

IKI	No. Keypresses	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
-30	50	-30.356	0.444	-31.0	-30.0	1.0
-25	50	-25.432	0.305	-25.6	-24.6	1.0
-20	50	-24.660	0.049	-24.7	-24.6	0.1
-15	50	-24.644	0.050	-24.7	-24.6	0.1
-10	50	-24.644	0.050	-24.7	-24.6	0.1
-5	50	-24.654	0.050	-24.7	-24.6	0.1
0	50	0.000	0.000	0.0	0.0	0.0
5	50	24.648	0.050	24.6	24.7	0.1
10	50	24.654	0.050	24.6	24.7	0.1
15	50	24.648	0.050	24.6	24.7	0.1
20	50	24.632	0.047	24.6	24.7	0.1
25	50	25.290	0.388	24.6	25.6	1.0
30	50	30.172	0.348	30.0	31.0	1.0

Table 3
Mean Mouse Registration Times (in Milliseconds) for a Single Key of a Serial Microsoft Mouse
for Make (Key Pressed) and Break (Key Released) Action

Key Action	Keypress Time	No. Keypresses	No. Registered	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
Make	5	50	0					
Make	6	50	0					
Make	7	50	2	132.900	0.000	132.9	132.9	0.0
Make	8	50	50	133.322	0.196	132.9	133.7	0.8
Make	9	50	50	133.282	0.227	132.9	133.7	0.8
Make	10	50	50	133.288	0.244	132.9	133.7	0.8
Break	5	50	0					
Break	6	50	0					
Break	7	50	2	24.700	0.000	24.7	24.7	0.0
Break	8	50	50	24.676	0.047	24.6	24.8	0.2
Break	9	50	50	24.674	0.044	24.6	24.7	0.1
Break	10	50	50	24.678	0.041	24.6	24.7	0.1

Note—True response delay was 100 msec.

Table 4
Mean Mouse Registration Times (in Milliseconds) for a Single Key of a Microsoft Bus Mouse
for Make (Key Pressed) and Break (Key Released) Action

Key Action	Keypress Time	No. Keypresses	No. Registered	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
Make	5	100	15	104.007	1.774	101.0	106.5	5.5
Make	10	100	33	106.248	2.761	100.8	110.7	9.9
Make	15	100	46	110.263	4.208	101.8	116.2	14.4
Make	20	100	63	111.803	5.716	101.1	120.9	19.8
Make	25	100	80	112.065	7.395	101.1	126.6	25.5
Make	30	100	99	115.304	9.203	101.3	131.3	30.0
Make	35	100	100	117.622	8.743	100.8	131.6	30.8
Break	5	100	15	30.900	0.000	30.9	30.9	0.0
Break	10	100	33	30.891	0.029	30.8	30.9	0.1
Break	15	100	46	30.891	0.028	30.8	30.9	0.1
Break	20	100	63	30.894	0.024	30.8	30.9	0.1
Break	25	100	80	30.899	0.019	30.8	31.0	0.2
Break	30	100	99	30.896	0.024	30.8	31.0	0.2
Break	35	100	100	35.222	10.724	30.8	61.8	31.0

Note—True response delay was 100 msec.

in the upper part of Table 4. Keypresses longer than one interrupt interval (keypress time greater than 30.8 msec) are expected to be delayed by half the interrupt interval, which is 15.4 msec. Expected mean delays of keypresses shorter than one interrupt interval depend on keypress time, because the size of the potential range (see Table 4, Range) is limited by the time a key is pressed.

Apart from the variations caused by the discrete interrupt rate, the registration is very precise. Setting the interrupt rate to 200 Hz, as Crosbie (1990) recommended, ensures registration of keypresses longer than 5 msec, which is sufficiently short for most experimental purposes. This high interrupt rate, however, produces a considerable overhead on the CPU, and a medium interrupt rate of 100 Hz or 50 Hz seems to be a better choice.

Whereas serial mouse measurements need to be corrected only by subtracting a constant minimum delay time, response times recorded by a bus mouse also require a correction of the variance. For keypresses longer than one

interrupt rate, the time behavior of the bus mouse can be compared to that of a clock running with low resolution. The statistical properties of low-resolution clocks are well defined (Ulrich & Giray, 1989). Ulrich and Giray presented two correction formulas for the mean and variance:

$$E(RT) = RT(PC) - (m/2) \quad (1)$$

and

$$var(RT) = var[RT(PC)] - (m^2/12), \quad (2)$$

where m is the clock resolution. According to these formulas, with an interrupt rate of 31 msec and an assumed true variance of zero, we expect the following values in our test situation: $RT(PC) = 100 + 31/2 = 115.5$, $var[RT(PC)] = (31 \times 31) / 12 = 80$, and $SD = 8.9$. The standard deviations shown in Table 4 fit very well to these theoretical estimates for keypresses of at least 30 msec.

Ulrich and Giray also emphasize that the mean difference between two continuous random variables is not de-

Table 5
Mean Mouse Registration Times (in Milliseconds)
for the Interkey Interval (IKI) Between
Two Keypresses of a Microsoft Bus Hypermouse

IKI	No. Keypresses	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
-40	50	-42.640	14.996	-61.8	-30.9	30.9
-30	50	-31.510	4.327	-61.8	-30.8	31.0
-20	50	-21.624	14.156	-30.9	0.0	30.9
-10	50	-8.034	13.554	-30.9	0.0	30.9
-5	50	-6.180	12.360	-30.9	0.0	30.9
0	50	0.000	0.000	0.0	0.0	0.0
5	50	4.326	10.722	0.0	30.9	30.9
10	50	11.124	14.832	0.0	30.9	30.9
20	50	22.860	13.550	0.0	30.9	30.9
30	50	30.900	0.000	30.9	30.9	0.0
40	50	40.168	14.157	30.9	61.8	30.9

pendent on the clock resolution. Mean differences less than clock resolution can be observed. This is confirmed by the means in Table 5, which show observed time differences for different IKIs. Although the bus mouse operated at intervals of 31 msec, smaller IKIs are observed. The standard deviation plus the minimum and maximum values, however, are obviously affected by the discrete sample rate.

PS/2 MOUSE

As Segalowitz and Graves (1990) noted, the time behavior of the PS/2 mouse is similar to that of keyboards. The PS/2 mouse is connected via the auxiliary device connector of the keyboard controller, which is a synchronously driven serial interface (IBM, 1987). The transfer is clocked by the keyboard controller. Measurements of single key actions are shown in Table 6.

The present results show the problems inherent in using the PS/2 mouse (Segalowitz & Graves, 1990). The pattern of observed response times is difficult to understand because appropriate technical information was not avail-

able. Response times were apparently delayed by an interaction of a discrete scan rate of 16.7 msec (see lower part of Table 6, minimum values), and other unknown factors (upper table minimums).

DISCUSSION

Because of the small variance and well-understood system-dependent delays, as response devices, the serial mouse and the bus mouse are superior to the standard keyboard and to the PS/2 mouse. If the influence of system-dependent response delays on mean and variance of observed data is known, statistics may be corrected. Even uncorrected data, however, do not greatly reduce the statistical power, provided that the variance of the response device is small in relation to the sample variance (Segalowitz & Graves, 1990; Ulrich & Giray, 1989). This is also true for many keyboards.

With the exception of the bus mouse, all investigated response devices are unsuitable for the registration of multiple key action with short IKIs. Even the detection of very short keypresses seems to be problematic. Fortunately, most reaction-time experiments do not require simultaneous keypresses or extraordinarily short keypress times. For most experiments, the serial mouse and the bus mouse are inexpensive, precise standard input devices. It is important, however, to use the mouse with the ball removed to avoid additional variance caused by the transmission of the mouse motion (see also Segalowitz & Graves, 1990).

Unless the timing behavior of input devices is included in the published technical specifications of the hardware, individual validation of the timing accuracy of the installed equipment is recommended before using correction algorithms. Even Microsoft-compatible mouse hardware may differ by a significant amount with respect to minimum delay time. Because in this study only one mouse model from each type was tested, variations within each type of mouse are not covered by the investigations. While

Table 6
Mean Mouse Registration Times (in Milliseconds) for a Single Key of an IBM PS/2 Mouse
for Make (Key Pressed) and Break (Key Released) Action

Key Action	Keypress Time	No. Keypresses	No. Registered	<i>M</i>	<i>SD</i>	Minimum	Maximum	Range
Make	10	300	3	114.733	4.135	110.2	120.2	10.0
Make	15	300	87	121.122	3.270	114.0	127.4	13.4
Make	20	300	180	123.768	4.046	113.8	132.5	18.7
Make	25	300	243	124.818	4.847	114.3	134.7	20.4
Make	30	300	300	127.320	5.513	115.0	139.4	24.4
Make	35	300	300	126.920	5.701	115.0	140.1	25.1
Make	40	300	300	127.077	5.655	114.3	139.7	25.4
Break	10	300	3	27.900	7.850	16.8	33.6	16.8
Break	15	300	87	22.922	8.031	16.6	33.7	17.1
Break	20	300	180	21.598	7.569	16.6	33.7	17.1
Break	25	300	243	26.477	8.243	16.6	33.8	17.2
Break	30	300	300	29.459	7.130	16.6	33.8	17.2
Break	35	300	300	34.648	8.713	16.7	50.4	33.7
Break	40	300	300	39.907	8.141	33.1	50.4	17.3

Note—True response delay was 100 msec.

the serial communication protocol can be expected to be constant for reasons of standardization, there is no certainty about the delay caused by the internal mouse logic that collects buttonpresses.

REFERENCES

- CROSBIE, J. (1990). The Microsoft mouse as a multipurpose response device for the IBM PC/XT/AT. *Behavior Research Methods, Instruments, & Computers*, **22**, 305-316.
- IBM (1987). *Personal System/2 and Personal Computer BIOS interface technical reference*. Portsmouth, U.K.: Author.
- MICROSOFT. (1986). *Microsoft mouse programmer's reference guide*. Author.
- RINCK, S. (1988). *Handbuch ERTS Experimentelles Run Time System* (J. Beringer, Ed.). Darmstadt: J. Beringer.
- SEGALOWITZ, S. J., & GRAVES, R. E. (1990). Suitability of the IBM XT, AT, and PS/2 keyboard, mouse, and game port as response devices in reaction time paradigms. *Behavior Research Methods, Instruments, & Computers*, **22**, 283-289.
- ULRICH, R., & GIRAY, M. (1989). Time resolution of clocks: Effects on reaction time measurement—Good news for bad clocks. *British Journal of Mathematical & Statistical Psychology*, **42**, 1-12.

NOTE

1. ERTS is a professional runtime and development system for psychological experiments based on reaction time. It was written by the author and was released on the European market in 1987.

(Manuscript received October 17, 1990;
revision accepted for publication January 13, 1992.)