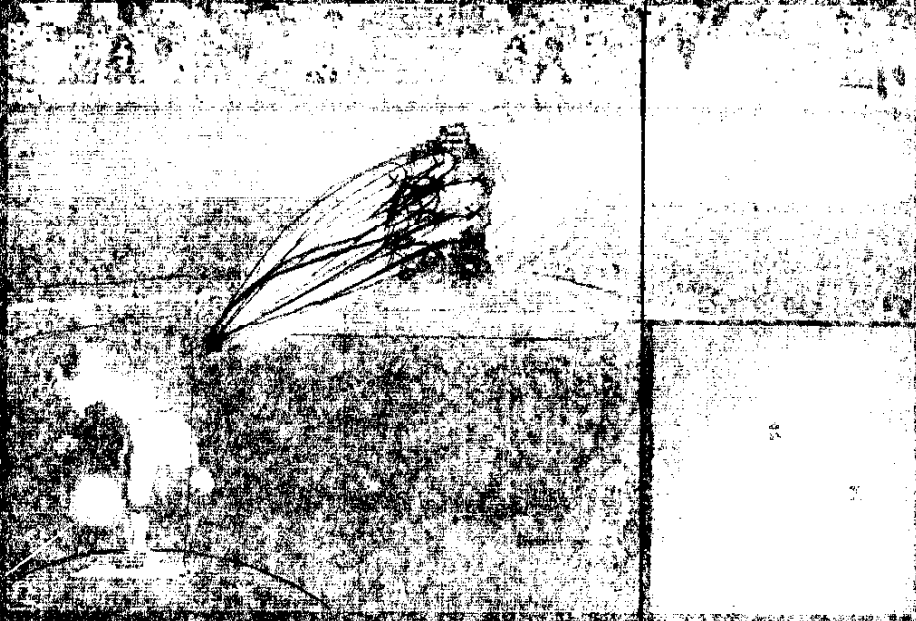


HardBall!



ACCOLADE

CPU-RAM CHART

00 MYWS	20 PRSTAT	40 GEAR 0-45	60 DOD	80 (sub stack)	A0 (data stack)
01	21 ↓	41 SPEED 0-150	61 0	81	A1
02	22	42 ALT 0-10,000	62 DOT 1	82	A2
03	23	43 ↓	63	83	A3
04	24 Y1	44 PITCH -99 to +99	64 DOT 2	84	A4
05	25 ↓	45 FLAP 0-29	65	85	A5
06	26 X1	46 FUEL 0-29	66 DOT 3	86	A6
07	27 ↓	47 POWER 0-99	67	87	A7
08	28 Y2	48 L PITCH	68 DOT 4	88	A8
09	29 ↓	49 L FLAP	69	89	A9
0A	2A X2	4A L POWER	6A DOT 5	8A	AA
0B	2B ↓	4B L GEAR	6B	8B	AB
0C	2C TURN	4C L SPEED	6C DOT 6	8C	AC
0D	2D ↓	4D L FUEL	6D DOT 7	8D	AD
0E	2E TURN	4E HALT	6E	8E	AE
0F	2F ↓	4F ↓	6F	8F	AF
10	30 SAURIA	50	70 RAM SIZE	90	B0
11	31 ↓	51	71	91	B1
12	32 HORIZ Y	52 L HCOB	72 DATSTK (>A0)	92	B2
13	33 HORIZ Y	53 ↓	73 SUBSTK (>80)	93	B3
14	34 HORIZ X	54 ROT -45 to +45	74 KEYBRD	94	B4
15	35 HORIZ X	55 ↓ ROT -45 to +45	75 KEY	95	B5
16	36 HORIZ X	56 LAMP INTF	76 JOY Y	96	B6
17	37 HORIZ X	57 BRNG 0-360	77 JOY X	97	B7
18	38 SECONDS (FOR FUEL)	58	78 RANDOM	98	B8
19	39 HRS (COMBAT MODE)	59	79 TIMER	99	B9
1A	3A	5A	7A MOTION	9A	BA
1B	3B	5B	7B VDPSTT	9B	BB
1C	3C	5C	7C STATUS	9C	BC
1D	3D	5D	7D CB	9D	BD
1E	3E	5E	7E YPT Temp	9E	BE
1F	3F	5F	7F XPT	9F	BF

LANDING

TITLE BASEBALL

SET 0	SET 1	SET 2	SET 3	SET 4	SET 5	SET 6	SET 7	SET 8	SET 9	SET 10
RAM > 800 COLOR: 44	RAM > 840 COLOR: 45	RAM > 880 COLOR: 46	RAM > 8C0 COLOR: 47	RAM > 900 COLOR: 48	RAM > 940 COLOR: 49	RAM > 980 COLOR: 50	RAM > 9C0 COLOR: 51	RAM > A00 COLOR: 52	RAM > A40 COLOR: 53	RAM > A80 COLOR: 54
>00 WP >01 >02 >03 >04 >05 >06 >07	>08 WP >09 >0A >0B >0C >0D >0E >0F	>10 WP >11 >12 >13 >14 >15 >16 >17	>18 TEST BOX >19 >1A >1B >1C >1D >1E >1F	>20 SP >21 >22 >23 >24 >25 >26 >27	>28 (>29 >2A >2B >2C >2D >2E >2F	>30 0 >31 1 >32 2 >33 3 >34 4 >35 5 >36 6 >37	>38 8 >39 9 >3A : >3B : >3C V >3D = >3E > >3F ?	>40 @ >41 A >42 B >43 C >44 D >45 E >46 F >47 G	>48 H >49 I >4A J >4B K >4C L >4D M >4E N >4F O	>50 P >51 Q >52 R >53 S >54 T >55 U >56 V >57 W
SET 11	SET 12	SET 13	SET 14	SET 15	SET 16	SET 17	SET 18	SET 19	SET 20	SET 21
RAM > A00 COLOR: F1	RAM > B00 COLOR: 00	RAM > B40 COLOR: 00	RAM > B80 COLOR: 00	RAM > BC0 COLOR: 00	RAM > C00 COLOR: 00	RAM > C40 COLOR: 00	RAM > C80 COLOR: A3	RAM > CC0 COLOR: A3	RAM > D00 COLOR: AF	RAM > D40 COLOR: 00
>58 >59 >5A >5B >5C >5D >5E >5F	>60 >61 >62 >63 >64 >65 >66 >67	>68 >69 >6A >6B >6C >6D >6E >6F	>70 >71 >72 >73 >74 >75 >76 >77	>78 >79 >7A >7B >7C >7D >7E >7F	>80 >81 >82 >83 >84 >85 >86 >87	>88 >89 >8A >8B >8C >8D >8E >8F	>90 >91 >92 >93 >94 >95 >96 >97	>98 >99 >9A >9B >9C >9D >9E >9F	>A0 >A1 >A2 >A3 >A4 >A5 >A6 >A7	>A8 >A9 >AA >AB >AC >AD >AE >AF
SET 22	SET 23	SET 24	SET 25	SET 26	SET 27	SET 28	SET 29	SET 30	SET 31	SET 32
RAM > D80 COLOR: A3	RAM > DC0 COLOR: AF	RAM > E00 COLOR: FC	RAM > E40 COLOR: FC	RAM > E80 COLOR: FC	RAM > EC0 COLOR: FC	RAM > F00 COLOR: FC	RAM > F40 COLOR: 0F	RAM > F80 COLOR: 4E	RAM > FC0 COLOR: 0E	RAM > 000 COLOR: 00
>B0 >B1 >B2 >B3 >B4 >B5 >B6 >B7	>B8 >B9 >BA >BB >BC >BD >BE >BF	>C0 >C1 >C2 >C3 >C4 >C5 >C6 >C7	>C8 >C9 >CA >CB >CC >CD >CE >CF	>D0 >D1 >D2 >D3 >D4 >D5 >D6 >D7	>D8 >D9 >DA >DB >DC >DD >DE >DF	>E0 >E1 >E2 >E3 >E4 >E5 >E6 >E7	>E8 >E9 >EA >EB >EC >ED >EE >EF	>F0 >F1 >F2 >F3 >F4 >F5 >F6 >F7	>F8 >F9 >FA >FB >FC >FD >FE >FF	>00 >01 >02 >03 >04 >05 >06 >07

RBHI

$8 * 8 + 0, 14 * 8 + 1$
 $7 * 8 + 6, 14 * 8 + 2$
 $7 * 8 + 4, 14 * 8 + 2$
 $7 * 8 + 2, 14 * 8 + 3$
 $7 * 8 + 0, 14 * 8 + 4$
 $6 * 8 + 6, 14 * 8 + 5$
 $6 * 8 + 4, 14 * 8 + 6$
 $6 * 8 + 2, 15 * 8 + 0$
 $5 * 8 + 7, 15 * 8 + 2$
 $5 * 8 + 5, 15 * 8 + 4$
 $5 * 8 + 4, 15 * 8 + 6$
 $5 * 8 + 2, 15 * 8 + 0$
 $5 * 8 + 1, 16 * 8 + 2$
 $5 * 8 + 0, 16 * 8 + 4$
 $4 * 8 + 6, 16 * 8 + 0$
 $4 * 8 + 5, 16 * 8 + 2$
 $4 * 8 + 4, 16 * 8 + 4$
 $4 * 8 + 4, 16 * 8 + 6$
 $4 * 8 + 3, 17 * 8 + 0$
 $4 * 8 + 3, 17 * 8 + 2$
 $4 * 8 + 3, 17 * 8 + 4$
 $4 * 8 + 4, 17 * 8 + 6$
 $>FFFF$

ABLD

$8 * 8 + 0, 14 * 8 + 3$
 $7 * 8 + 6, 14 * 8 + 4$
 $7 * 8 + 4, 14 * 8 + 6$
 $7 * 8 + 3, 15 * 8 + 0$
 $7 * 8 + 1, 15 * 8 + 2$
 $6 * 8 + 7, 15 * 8 + 4$
 $6 * 8 + 6, 15 * 8 + 6$
 $6 * 8 + 4, 16 * 8 + 0$
 $6 * 8 + 3, 16 * 8 + 2$
 $6 * 8 + 2, 16 * 8 + 4$
 $6 * 8 + 1, 16 * 8 + 6$
 $5 * 8 + 7, 16 * 8 + 0$
 $5 * 8 + 6, 17 * 8 + 2$
 $5 * 8 + 5, 17 * 8 + 4$
 $5 * 8 + 4, 17 * 8 + 6$
 $5 * 8 + 4, 18 * 8 + 0$
 $5 * 8 + 4, 18 * 8 + 2$
 $5 * 8 + 6, 18 * 8 + 4$
 $6 * 8 + 0, 18 * 8 + 6$
 $>FFFF$

RBMEB

$8 * 8 + 0, 14 * 8 + 1$
 $7 * 8 + 6, 14 * 8 + 2$
 $7 * 8 + 4, 14 * 8 + 4$
 $7 * 8 + 2, 14 * 8 + 6$
 $7 * 8 + 0, 15 * 8 + 0$
 $6 * 8 + 6, 15 * 8 + 1$
 $6 * 8 + 4, 15 * 8 + 3$
 $6 * 8 + 2, 15 * 8 + 5$
 $6 * 8 + 0, 15 * 8 + 7$
 $5 * 8 + 7, 16 * 8 + 1$
 $5 * 8 + 5, 16 * 8 + 3$
 $5 * 8 + 4, 16 * 8 + 5$
 $5 * 8 + 3, 16 * 8 + 7$
 $5 * 8 + 2, 17 * 8 + 1$
 $5 * 8 + 1, 17 * 8 + 3$
 $5 * 8 + 1, 17 * 8 + 5$
 $5 * 8 + 1, 17 * 8 + 7$
 $5 * 8 + 1, 18 * 8 + 1$
 $5 * 8 + 1, 18 * 8 + 3$
 $5 * 8 + 2, 18 * 8 + 5$
 $5 * 8 + 2, 18 * 8 + 6$
 $>FFFF$

PL157
ZBH1

BYTE 8*8+8, 14*8+2
 BYTE 8*8-1, 14*8+4
 BYTE 7*8+5, 14*8+6
 BYTE 7*8+3, 14*8+7
 BYTE 7*8+2, 15*8+1
 BYTE 7*8+0, 15*8+2
 BYTE 6*8+6, 15*8+4
 BYTE 6*8+4, 15*8+6
 BYTE 6*8+2, 15*8+7
 BYTE 6*8+0, 16*8+1
 BYTE 5*8+6, 16*8+3
 BYTE 5*8+4, 16*8+5
 BYTE 5*8+2, 16*8+7
 BYTE 5*8+0, 17*8+0
 BYTE 4*8+6, 17*8+2
 BYTE 4*8+4, 17*8+4
 BYTE 4*8+1, 17*8+7
 BYTE >FF, >FF

17 positions

2 per for fast
3 per for slow

ZBMEs

BYTE 8*8+1, 14*8+2
 8*8+1, 14*8+4
 7*8+6, 14*8+6
 7*8+4, 15*8+0
 7*8+3, 15*8+2
 7*8+1, 15*8+4
 7*8+0, 15*8+6
 6*8+6, 16*8+0
 6*8+4, 16*8+2
 6*8+2, 16*8+4
 6*8+1, 16*8+6
 6*8-1, 17*8+0
 5*8+6, 17*8+2
 5*8+4, 17*8+4
 5*8+2, 17*8+7
 >FFFF

ZBA0

8*8+1, 14*8+2
 8*8+0, 14*8+5
 7*8+7, 14*8+7
 7*8+6, 15*8+1
 7*8+5, 15*8+3
 7*8+3, 15*8+5
 7*8+2, 15*8+7
 7*8+1, 16*8+1
 7*8+0, 16*8+3
 6*8+7, 16*8+5
 6*8+6, 16*8+7
 6*8+5, 17*8+1
 6*8+4, 17*8+3
 6*8+3, 17*8+5
 6*8+2, 17*8+7
 >FFFF

LBH1 $8 * 8 + 1, 14 * 8 + 1$
 $7 * 8 + 7, 14 * 8 + 2$
 $7 * 8 + 5, 14 * 8 + 3$
 $7 * 8 + 3, 14 * 8 + 4$
 $7 * 8 + 1, 14 * 8 + 6$
 $6 * 8 + 7, 15 * 8 + 0$
 $6 * 8 + 5, 15 * 8 + 2$
 $6 * 8 + 4, 15 * 8 + 4$
 $6 * 8 + 2, 15 * 8 + 6$
 $6 * 8 + 1, 16 * 8 + 0$
 $5 * 8 + 7, 16 * 8 + 2$
 $5 * 8 + 6, 16 * 8 + 4$
 $5 * 8 + 6, 16 * 8 + 6$
 $5 * 8 + 5, 17 * 8 + 0$
 $5 * 8 + 4, 17 * 8 + 2$
 $5 * 8 + 3, 17 * 8 + 4$
 $5 * 8 + 1, 17 * 8 + 5$
 $4 * 8 + 7, 17 * 8 + 5$
 $3 * 8 + 5, 17 * 8 + 4$
 $3 * 8 + 4, 17 * 8 + 2$
 $3 * 8 + 4, 17 * 8 + 0$

> FFFF

LBH0 $8 * 8 + 1, 14 * 8 + 3$
 $7 * 8 + 7, 14 * 8 + 5$
 $7 * 8 + 5, 14 * 8 + 7$
 $7 * 8 + 3, 15 * 8 + 1$
 $7 * 8 + 2, 15 * 8 + 3$
 $7 * 8 + 1, 15 * 8 + 5$
 $7 * 8 + 1, 15 * 8 + 7$
 $7 * 8 + 0, 16 * 8 + 1$
 $6 * 8 + 7, 16 * 8 + 3$
 $6 * 8 + 7, 16 * 8 + 5$
 $6 * 8 + 7, 16 * 8 + 7$
 $6 * 8 + 7, 17 * 8 + 1$
 $6 * 8 + 6, 17 * 8 + 3$
 $6 * 8 + 4, 17 * 8 + 4$
 $6 * 8 + 2, 17 * 8 + 3$
 $6 * 8 + 0, 17 * 8 + 2$
 $6 * 8 + 0, 17 * 8 + 0$
~~7 * 8 + 1~~

LBMEB

$8 * 8 + 1, 14 * 8 + 2$
 $7 * 8 + 7, 14 * 8 + 4$
 $7 * 8 + 5, 14 * 8 + 5$
 $7 * 8 + 3, 14 * 8 + 7$
 $7 * 8 + 1, 15 * 8 + 1$
 $7 * 8 + 0, 15 * 8 + 3$
 $6 * 8 + 6, 15 * 8 + 5$
 $6 * 8 + 5, 15 * 8 + 7$
 $6 * 8 + 4, 16 * 8 + 1$
 $6 * 8 + 2, 16 * 8 + 3$
 $6 * 8 + 1, 16 * 8 + 5$
 $6 * 8 + 1, 16 * 8 + 7$
 $6 * 8 + 0, 17 * 8 + 1$
 $6 * 8 + 0, 17 * 8 + 3$
 $5 * 8 + 7, 17 * 8 + 5$
 $5 * 8 + 5, 17 * 8 + 5$
 $5 * 8 + 3, 17 * 8 + 4$
 $5 * 8 + 2, 17 * 8 + 2$
 $5 * 8 + 2, 17 * 8 + 0$

> FFFF

SPRITE TABLE

VDP REG (1) =

SPRITE #	YPT	XPT	CHAR	COL.	VELO-CITY	Y	X	SPRITE #	YPT	XPT	CHAR	COL.	VELO-CITY	Y	X
0	>300	PITCHER ↑	>80	4	>780			16	>340				>7C0		
1	>304	PITCHER ↓	>84	4	>784			17	>344				>7C4		
2	>308	BATTER	>88	1	>788			18	>348				>7C8		
3	>30C		>8C	1	>78C			19	>34C				>7CC		
4	>310		>90	1	>790			20	>350				>7D0		
5	>314	✓	>94	1	>794			21	>354				>7D4		
6	>318	CORNER	>98	F	>798			22	>358				>7D8		
7	>31C	CATCHER	>9C	4	>79C			23	>35C				>7DC		
8	>320	LEG	>A0	4	>7A0			24	>360				>7E0		
9	>324	BALL OUTLINE	>98	1	>7A4			25	>364				>7E4		
10	>328				>7A8			26	>368				>7E8		
11	>32C				>7AC			27	>36C				>7EC		
12	>330				>7B0			28	>370				>7F0		
13	>334				>7B4			29	>374				>7F4		
14	>338				>7B8			30	>378				>7F8		
15	>33C				>7BC			31	>37C				>7FC		

CHAR.	RAM	DATA	CHAR.	RAM	DATA	CHAR.	RAM	DATA	CHAR.	RAM	DATA
>80	>400	PITCHER UPPER	>98	>4C0	BALL	>B0	>580		>C8	>640	
>81	>408		>99	>4C8		>B1	>588		>C9	>648	
>82	>410		>9A	>4D0		>B2	>590		>CA	>650	
>83	>418		>9B	>4D8		>B3	>598		>CB	>658	
>84	>420	PITCHER LOWER	>9C	>4E0	CATCHER	>B4	>5A0		>CC	>660	
>85	>428		>9D	>4E8		>B5	>5A8		>CD	>668	
>86	>430		>9E	>4F0		>B6	>5B0		>CE	>670	
>87	>438		>9F	>4F8		>B7	>5B8		>CF	>678	
>88	>440	POWER UR	>A0	>500	PITCHER LEG	>B8	>5C0		>D0	>680	
>89	>448		>A1	>508	1/2 TRAP	>B9	>5C8		>D1	>688	
>8A	>450		>A2	>510		>BA	>5D0		>D2	>690	
>8B	>458		>A3	>518		>BB	>5D8		>D3	>698	
>8C	>460	BATTER LR	>A4	>520	BALL OUTLINE	>BC	>5E0		>D4	>6A0	
>8D	>468		>A5	>528		>BD	>5E8		>D5	>6A8	
>8E	>470		>A6	>530		>BE	>5F0		>D6	>6B0	
>8F	>478		>A7	>538		>BF	>5F8		>D7	>6B8	
>90	>480	BATTER UL	>A8	>540		>C0	>600		>D8	>6C0	
>91	>488		>A9	>548		>C1	>608		>D9	>6C8	
>92	>490		>AA	>550		>C2	>610		>DA	>6D0	
>93	>498		>AB	>558		>C3	>618		>DB	>6D8	
>94	>4A0	BATTER LL	>AC	>560		>C4	>620		>DC	>6E0	
>95	>4A8		>AD	>568		>C5	>628		>DD	>6E8	
>96	>4B0		>AE	>570		>C6	>630		>DE	>6F0	
>97	>4B8		>AF	>578		>C7	>638		>DF	>6F8	

CPU-RAM CHART

00 MYWLS	20 PITADS 0-4	40 PACTOP 1-1	60	80 (sub stack)	A0 (data stack)
01	21 PATADS 0-3	41 PACTEB 1-1	61	81	A1
02	22 PACTH 0-1, 2 LOW	42 PACTUL 0-1	62	82	A2
03	23 PACTK 0-1, 2 LOW	43 PACTW 0-1	63	83	A3
04	24 PACTS 0-1, 2 LOW	44 PACTX 0-1	64	84	A4
05	25 PACTT 0-1, 2 LOW	45	65	85	A5
06	26 PACTU 0-1, 2 LOW	46 PACTY 0-1	66	86	A6
07	27 PACTV 0-1, 2 LOW	47	67	87	A7
08	28 PACTW 0-1, 2 LOW	48	68	88	A8
09	29 PACTX 0-1, 2 LOW	49	69	89	A9
0A	2A PACTY 0-1, 2 LOW	4A	6A	8A	AA
0B	2B PACTZ 0-1, 2 LOW	4B	6B	8B	AB
0C	2C PACT0 0-1, 2 LOW	4C	6C	8C	AC
0D	2D PACT1 0-1, 2 LOW	4D	6D	8D	AD
0E	2E PACT2 0-1, 2 LOW	4E	6E	8E	AE
0F	2F PACT3 0-1, 2 LOW	4F	6F	8F	AF
10	30 STRIKE 0-2	50	70	90	B0
11	31 BALLS 0-3	51	71	91	B1
12	32 RATS 0-4	52	72	92	B2
13	33 BALLS 0-4	53	73	93	B3
14	34 BALLS 0-4	54	74	94	B4
15	35 BALLS 0-4	55	75	95	B5
16	36 BALLS 0-4	56	76	96	B6
17	37 BALLS 0-4	57	77	97	B7
18	38 PISCOR	58	78	98	B8
19	39 PISCOR	59	79	99	B9
1A	3A PISCOR	5A	7A	9A	BA
1B	3B PISCOR	5B	7B	9B	BB
1C	3C PISCOR	5C	7C	9C	BC
1D	3D PISCOR	5D	7D	9D	BD
1E	3E PISCOR	5E	7E	9E	BE
1F	3F PISCOR	5F	7F	9F	BF

0 = 02
 1 = 00
 2 = 02
 3 = 11
 4 = 01
 5 = 11
 6 = 11
 7 = 01
 8 = 00
 9 = 00
 10 = 22
 11 = 02
 12 = 00
 13 = 11
 14 = 01
 15 = 11
 16 = 11
 17 = 00
 18 = 11
 19 = 04
 20 = 00
 21 = 00
 22 = 11
 23 = 00
 24 = 00
 25 = 01
 26 = 00
 27 = 11
 28 = 00
 29 = 11
 30 = 00
 31 = 00
 32 = 00
 33 = 02
 34 = 12
 35 = 00
 36 = 12
 37 = 00
 38 = 00
 39 = 00
 40 = 00
 41 = 01
 42 = 01
 43 = 34
 44 = 12
 45 = 00
 46 = 00
 47 = 01
 48 = 00
 49 = 00
 50 = 11
 51 = 00
 52 = 11
 53 = 11
 54 = 00
 55 = 22
 56 = 11
 57 = 01
 58 = 11
 59 = 00
 60 = 11
 61 = 00
 62 = 13
 63 = 00

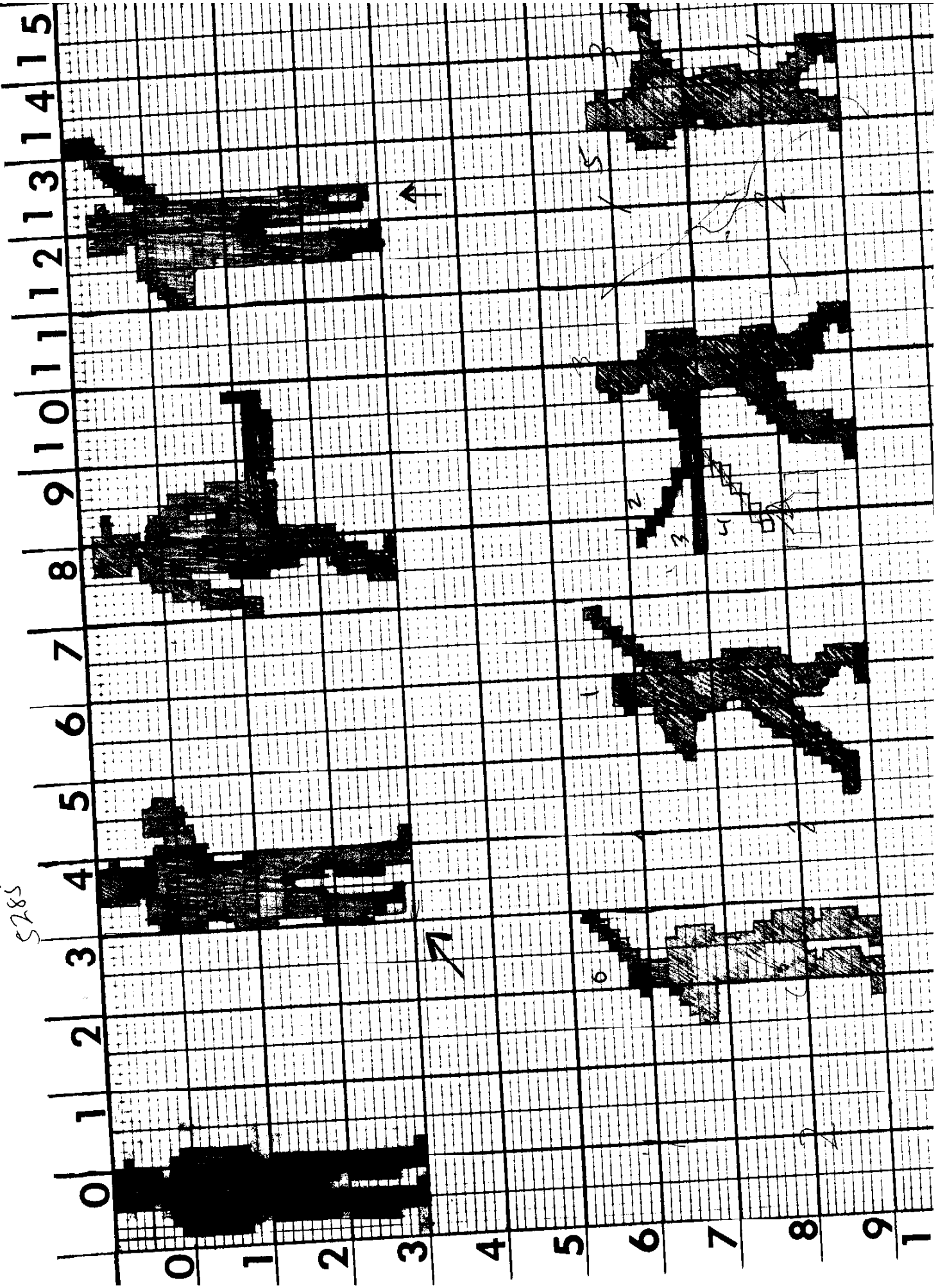
SINGLE	10 ⁰ , 14, 41, 45
DOUBLE	15 ² , 16, 55 ³ , 57
TRIPLE	5 ⁴ , 6, 52 ⁵ , 54
HR	25 ⁶ , 25, 29 ⁷ , 29
FLY OUT	0 ⁸ , 4, 46 ⁹ , 51
GROUND	17 ¹⁰ , 24, 58 ¹¹ , 63
FOUL	7 ¹² , 9, 33 ¹³ , 40
POP	26 ¹⁴ , 28, 30 ¹⁵ , 32

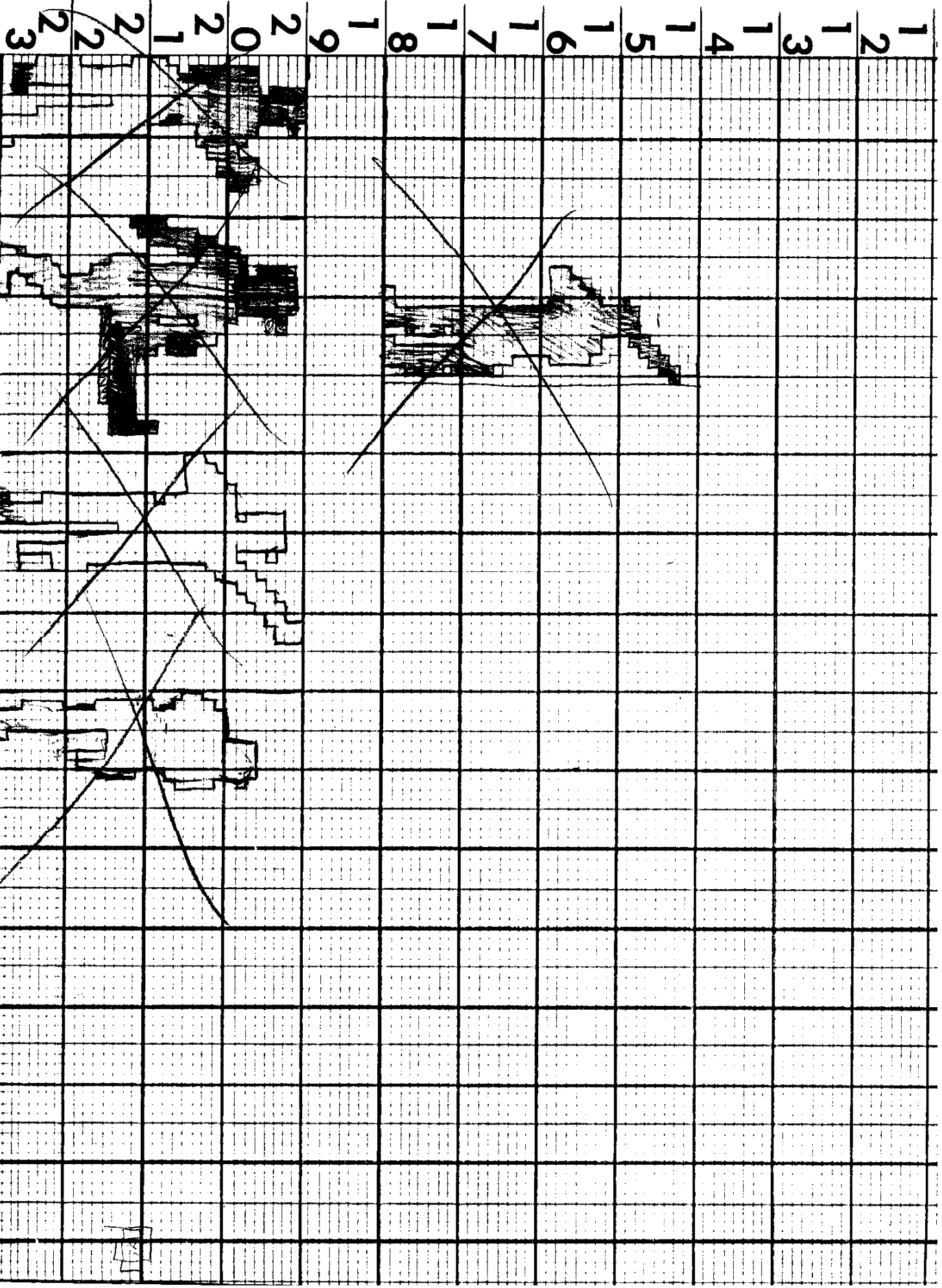
1 1 1 1 1 1 2 3 4
 change
 1 1 1 1 1 1 2 3 123 123
 (2) (2) (2)

Striked = not zero on screen
 on hit batter doesn't follow
 thru
 Remove shadow on hit

WHERE DATA SINGLE, SINGLE
 DOUBLE, DOUBLE
 TRIPLE, TRIPLE
 HR, HR
 FLYOUT, FLYOUT
 GROUND, GROUND
 FOUL, FOUL
 POPUP, POPUP

5285





16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

The image shows a grid of graph paper with a vertical line drawn through it. The grid is composed of small squares. There are several dark, irregular smudges or marks on the grid. One large smudge is located in the lower-left quadrant, roughly between the vertical line and the right edge, and between the horizontal lines corresponding to the numbers 16 and 19. Another smaller smudge is located in the upper-left quadrant, roughly between the vertical line and the right edge, and between the horizontal lines corresponding to the numbers 22 and 25. A small arrow points to the right from the bottom edge of the grid, near the vertical line, at approximately the horizontal level of the number 17. The numbers 16 through 30 are printed along the top edge of the grid, aligned with the vertical lines.

CH#	0	1	2	3	4	5	6	7	
D8	A8	C8	C8	C8	80	80	80	80	DIAMONDS
D9	05	03	03	03	01	01	01	01	
DA	80	00	A0	90	88	84	82	81	
DB	01	7E	7E	7E	7E	7E	7E	01	
DC	3C	42	81	BD	BD	81	42	3C	
DD	80	7E	7E	7E	7E	7E	7E	80	
DE	01	03	05	09	11	21	41	81	
DF	80	80	80	40	40	20	20	20	
EO	01	01	01	62	02	04	04	04	
E1	10	08	04	02	01	00	00	00	
E2	00					FF	00	00	
E3	08	10	20	40	80	00	00	00	
E4	81	7E	7E	7E	7E	3C	18	00	
E5	00								
E6	18	5A	7E	18	18	3C	24	66	
E8	F7	F7	00	EF	EF	00	F7	00	BRICK
FD	00	18	3C	18	18	3C	7E	7E	FAN ROW 1 & 3
F8	00	81	C3	81	81	C3	E7	E7	FAN ROW 2

CH#	0	1	2	3	4	5	6	7
0	FF	FF	FF	FF	FF	FF	FF	FF
8	18	3C	7F	18	18	18	18	18
10	18	18	18	18	18	7F	3C	FF
18	FF	80	80	80	80	80	80	80
19	FF	00						
1A	FF	01						
1B	80							
1C	01							
1D	80							FF
1E	00							FF
1F	01							FF
90	00							
91	00				01	3F	07	FF
92	00	00	00	0F	FF	FF	FF	FF
93	00	03	FF	FF	FF	FF	FF	FF
94	1F	FF						
95	FF							
96	FF						FE	F8
97	FF				F8	E8	00	00
98	FF	FF	FF	3F	1F	0F	03	01
99	FF	FF	FF	FE	ED	00	00	00
9A	FF	F8	80					
9B	F8	FF						
9C	FF	FF	FF	FC	F8	F0	C0	80
9D	00							
A0	FF	FE	FE	FC	FD	FA	F3	E9
A1	FF	00	FF					
A2	FF	7F	7F	3F	9F	CF	CF	E7
A3	E7	CF	9F	3F	00	00	FF	FF
A4	FF	FF	FF	FF	00	00	FF	FF
A5	FF	FE	F8	F0	F0	FF	FF	FF
A6	FF	7F	1F	0F	0F	FF	FF	FF
A7	E7	F8	F9	FC	00	00	FF	FF
B0	00							3F
B1	00						FF	FF
B2	00					FF	FF	FF
B3	00					FF	FF	FF
B4	00					C0	FF	FF
B5	00							FC
B6	80	E0	80	F8	F8	FE	FE	FF
B7	FF							
B8	C0	80	00	FF				
B9	00	01	03	FF				
C8	00					01	06	18
C9	00				3F	C0	00	00
CA	00			FF	00			
CB	00			FC		03	00	00
CC	06					80	60	18
CD	00	00	03	04	0C	10	20	40
CE	60	80	00					
CF	00	7E						81
D0	06	01	00					
D1	00	00	C0	20	10	80	40	20
D2	80	81	82	84	84	88	90	90
D3	80	60						
D4	01	02	04	08	10	20	40	80
D5	80	40	20	10	08	04	02	01
D6	01	00						
D7	09	81	41	21	21	11	09	09

FILL
↑
↓

UL
HORIZ TOP
WR
TEXT BOX VERT LEFT
VERT RIGHT
ALSO >C0 - C7
LL
HORIZ BOT
LR

BASEPATH

BATTER'S BOX

MOUND

DIAMOND

INSTRUCTIONS:

black always bottom of inning
blue always top of inning

AUDIO/TONES:

AUDIO/TONES:

HOME COMPUTER "GROM" DEVELOPMENT DATE:



