**WY-30 CONTROL SEQUENCES**

Control Sequences for the native personality:

|  |  |
| --- | --- |
| **Communication Modes** | |
| Turn full-duplex mode on | ESC C ESC D F |
| Turn block mode on | ESC B |
| Turn block mode off | ESC C |
| Turn half-duplex block mode on | ESC D H ESC B |
| Turn half-duplex mode on | ESC C ESC D H |
| Monitor mode on | ESC U |
| Monitor mode off | ESC u or ESC X |
| **Compatible and Enhance Modes** | |
| Set comaptible and Enhance modes | ESC ~ *mode* |
| *mode* Compatible Mode  " WY-30  # Televideo 910+  $ Televideo 925  % ADDS VP A2  Space Enhance mode off  ! Enhance mode on | |
| **Editing Modes** | |
| Turn remote edit keys mode on | ESC l |
| Turn local edit keys mode on | ESC k |
| **Sending Data in Block Mode** | |
| Send one character | ESC M |
| Send entire line | ESC 6 |
| Send unprotected line | ESC 4 |
| Send page | ESC 7 |
| Send unprotected page | ESC 5 |
| Place STX character | ESC 8 |
| Place ETX character | ESC 9 |
| Send a block of data | ESC s |
| Send a block of unprotected characters | ESC S |
| **Identifying the Terminal** | |
| Send terminal identifier | ESC SPACE |
| When the terminal receives  this sequence, it identifies  itself by transmitting this  three byte string to the  computer:  3 0 CR  33H 30H 0DH | |
| **Page Print Functions** | |
| Print formatted page | ESC P |
| Print formatted unprotected page | ESC @ |
| Print unformatted page | ESC p or ESC L |
| **Function Keys** | |
| Program a programmable key | ESC z *key sequence* DEL |
| Key? *key* shifted key  ESC  TAB !  BACKSPACE "  BREAK/DEL #  RETURN $  LINEFEED %  UP +  DOWN ,  LEFT -  RIGHT .  HOME /  ENTER s  KPD , p  KPD - q  KPD . r  KPD 0 t  KPD 1 u  KPD 2 v  KPD 3 w  KPD 4 x  KPD 5 y  KPD 6 z  KPD 7 {  KPD 8 |  KPD 9 }  F1 @ `  F2 A a  F3 B b  F4 C c  F5 D d  F6 E e  F7 F f  F8 G g  ----------------------------------  *sequence* Up to 64 bytes of data  to be loaded into key.  Limit total of non-  volatile memory is  121 characters, but  more can be entered.  DEL The | |
| Clear a programmable key | ESC z *key* DEL |
| Key? *key* shifted key  ESC  TAB !  BACKSPACE "  BREAK/DEL #  RETURN $  LINEFEED %  UP +  DOWN ,  LEFT -  RIGHT .  HOME /  ENTER s  KPD , p  KPD - q  KPD . r  KPD 0 t  KPD 1 u  KPD 2 v  KPD 3 w  KPD 4 x  KPD 5 y  KPD 6 z  KPD 7 {  KPD 8 |  KPD 9 }  F1 @ `  F2 A a  F3 B b  F4 C c  F5 D d  F6 E e  F7 F f  F8 G g  ---------------------------------- | |
| **Print Modes** | |
| Copy print mode on | CTRL R |
| Copy print mode Off | CTRL T |
| Transparent print mode on | CTRL X |
| Transparent print mode off | CTRL T |
| **Character Sets** | |
| Turn graphics mode on | ESC H CTRL B |
| Turn graphics mode off | ESC H CTRL C |
| Display a graphics character | ESC H *char* |
| Graphics *char* Graphics *char*  Character Character  0 8  1 9  2 :  3 ;  4 < 5=6>  7 ? | |
| **Local Commands** | |
| Lock keyboard | CTRL O or ESC # |
| Unlock keyboard | CTRL N or ESC " |
| Turn monitor mode on | ESC U |
| Turn monitor mode off | ESC u or ESC X |
| **Display Message Fields** | |
| Program/Display message on status line | ESC F *message* CR |
| Displays message on status line.  Can be up to 46 characters | |
| Program/Display message on unshifted label line | ESC z ( *message* CR |
| Displays message on unshifted  label line. Contains up to  78 characters | |
| Program message on shifted label line | ESC z ) *message* CR |
| Programs message on shifted label  line. Contains up to 78 characters | |
| Turn off shifted label line | ESC z DEL |
| Program/Display one label | ESC z *field label* CR |
| Function *Field Code*  *Key Unshifted Shifted*  *F1 0 P*  *F2 1 Q*  *F3 2 R*  *F4 3 S*  *F5 4 T*  *F6 5 U*  *F7 6 V*  *F8 7 W*  *label is a character string of up*  *to 8 characters* | |
| Clear function key label | ESC z *field* CR |
| Function *Field Code*  Key Unshifted Shifted  F1 0 P  F2 1 Q  F3 2 R  F4 3 S  F5 4 T  F6 5 U  F7 6 V  F8 7 W | |
| Clear unshifted label line | ESC z ( CR |
| Clear shifted label line | ESC z ) CR |
| **Screen and Cursor Display** | |
| Turn screen display off | ESC ` 8 |
| Turn screen display on | ESC ` 9 |
| Set cursor display feature | ESC ` *feature* |
| *feature* Cursor Display  0 Off  1 On (default)  2 Steady block  3 Blinking line  4 Steady line  5 Blinking block | |
| **Split Screen** | |
| Split screen horizontally | ESC x 1 *line* |
| Line *line* Line *line*  Num Num  1 space 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7 | |
| Activate the other data segment | ESC J or ESC K |
| Activate the upper data segment | ESC ] or ESC c |
| Activate the lower data segment | ESC } |
| Read active segment and cursor address | ESC / |
| In response to this, the terminal  sends the following string:  seg line col CR  seg=0 is upper; seg=1 is lower  line is line code  col is column code | |
| Redefine screen as one segment | ESC x 0 |
| **Scrolling** | |
| Set scrolling speed and type | ESC ` *type* |
| *type* Scrolling Type Speed lps  @ Jump scroll at Baud  < Smooth scroll 1=Smooth scroll 2 G Smooth scroll 3> Smooth scroll 4  ? Smooth scroll 8  H Smooth scroll 15  I Smooth scroll 20 | |
| Turn autoscroll mode off | ESC N |
| Turn autoscroll mode on | ESC O |
| **Display Attributes** | |
| Set field display attribute | ESC A *n attr* |
| *n* Screen Area  0 Data area  1 Function key label line  2 Terminal message field on  status line  3 Computer message field on  status line  *attr* Display Attribute  SPACE Space character  0 Normal  1 Blank  2 Blink  3 Blank  4 Reverse  5 Reverse/Blank  6 Reverse/Blink  7 Reverse/Blink/Blank  8 Underline  9 Underline/Blank  : Underline/Blink  ; Underline/Blink/Blank  < Underline Reverse=Underline/Reverse/Blank> Underline/Reverse/Blink  ? Underline/Reverse/Blink/  Blank  p Dim  q Dim/Blank  r Dim/Blink  s Dim/Blink/Blank  t Dim/Reverse  u Dim/Reverse/Blank  v Dim/Reverse/Blink  w Dim/Reverse/Blink/Blank  x Dim/Underline  y Dim/Underline/Blank  z Dim/Underline/Blank  { Dim/Underline/Blink/  Blank  | Dim/Underline/Reverse  } Dim/Underline/Reverse/  Blank  ~ Dim/Underline/Reverse/  Blink  DEL Dim/Underline/Reverse/  Blink/Blank | |
| Set display attributes | ESC ^ *n* |
| n Attribute  0 Turn reverse screen off  1 Turn reverse screen on  2 Attribute active to end of  page  3 Attribute active to end of  line | |
| **Addressing/Reading the Cursor** | |
| Address cursor | ESC a *line* R *column* C |
| *line* one or two digit  decimal value of the  line relative to home.  *column* one or two digit  decimal value of the  column relative to  home. | |
| Address cursor to line and column | ESC = *line column* |
| Line *line* Line *line*  Num Num  1 space 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  Column *col* Column *col*  Number Number  1 SPACE 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  25 8 26 9  27 : 28 ;  29 < 30=31> 32 ?  33 @ 34 A  35 B 36 C  37 D 38 E  39 F 40 G  41 H 42 I  43 J 44 K  45 L 46 M  47 N 48 O  49 P 50 Q  51 R 52 S  53 T 54 U  55 V 56 W  57 X 58 Y  59 Z 60 [  61 \ 62 ]  63 ^ 64 \_  65 ` 66 a  67 b 68 c  69 d 70 e  71 f 72 g  73 h 74 i  75 j 76 k  77 l 78 m  79 n 80 o | |
| Home cursor | ESC { or CTRL ^ |
| Move cursor to data segment and address | ESC - *seg line col* |
| *seg* Data Segment  0 Upper  1 Lower  Line *line* Line *line*  Num Num  1 space 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  Column *col* Column *col*  Number Number  1 SPACE 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  25 8 26 9  27 : 28 ;  29 < 30=31> 32 ?  33 @ 34 A  35 B 36 C  37 D 38 E  39 F 40 G  41 H 42 I  43 J 44 K  45 L 46 M  47 N 48 O  49 P 50 Q  51 R 52 S  53 T 54 U  55 V 56 W  57 X 58 Y  59 Z 60 [  61 \ 62 ]  63 ^ 64 \_  65 ` 66 a  67 b 68 c  69 d 70 e  71 f 72 g  73 h 74 i  75 j 76 k  77 l 78 m  79 n 80 o | |
| Read cursor address | ESC ? |
| Terminal returns a three byte  address:  *line col* CR  Line *line* Line *line*  Num Num  1 space 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  Column *col* Column *col*  Number Number  1 SPACE 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  25 8 26 9  27 : 28 ;  29 < 30=31> 32 ?  33 @ 34 A  35 B 36 C  37 D 38 E  39 F 40 G  41 H 42 I  43 J 44 K  45 L 46 M  47 N 48 O  49 P 50 Q  51 R 52 S  53 T 54 U  55 V 56 W  57 X 58 Y  59 Z 60 [  61 \ 62 ]  63 ^ 64 \_  65 ` 66 a  67 b 68 c  69 d 70 e  71 f 72 g  73 h 74 i  75 j 76 k  77 l 78 m  79 n 80 o | |
| Read cursor address | ESC b |
| Terminal responds with a seven  byte address:  *ll* R *ccc* C  *ll* is a is a two digit  decimal value of the line  relative to home.  *ccc* is a two digit decimal  value (with a leading  zero) of the column  relative to home. | |
| Read segment number and cursor address | ESC / |
| Terminal responds with a four  character string as:  *seg line col* CR  *seg* Data Segment  0 Upper  1 Lower  Line *line* Line *line*  Num Num  1 space 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  Column *col* Column *col*  Number Number  1 SPACE 2 !  3 " 4 #  5 $ 6 %  7 & 8 '  9 ( 10 )  11 \* 12 +  13 , 14 -  15 . 16 /  17 0 18 1  19 2 20 3  21 4 22 5  23 6 24 7  25 8 26 9  27 : 28 ;  29 < 30=31> 32 ?  33 @ 34 A  35 B 36 C  37 D 38 E  39 F 40 G  41 H 42 I  43 J 44 K  45 L 46 M  47 N 48 O  49 P 50 Q  51 R 52 S  53 T 54 U  55 V 56 W  57 X 58 Y  59 Z 60 [  61 \ 62 ]  63 ^ 64 \_  65 ` 66 a  67 b 68 c  69 d 70 e  71 f 72 g  73 h 74 i  75 j 76 k  77 l 78 m  79 n 80 o | |
| **Moving the Cursor** | |
| Move cursor to column one | CTRL M |
| Move column right one column | CTRL L |
| Move cursor left one column | CTRL H |
| Move cursor down one line in column (scroll) | CTRL J |
| Move cursor up one line in column (wrap) | CTRL K |
| Move cursor down one line in column (no wrap) | CTRL V |
| Move cursor to column one of next line (scroll) | CTRL \_ |
| Move cursor up in column (scroll) | ESC j |
| **Tab Stops** | |
| Set tab stop | ESC 1 |
| Clear tab stop at cursor | ESC 2 |
| Clear all tab stops | ESC 0 |
| Tabulate cursor | CTRL I |
| Backtab | ESC I |
| **Creating Protected Forms** | |
| Turn write protect mode on | ESC ) |
| Turn write protect mode off | ESC ( |
| Turn protect mode on | ESC & |
| Turn protect mode off | ESC ' |
| Protect column | ESC V |
| Set protect character attribute | ESC ` *attr* |
| *attr* Attribute  7 Dim  A Normal  C Double wide  E Double wide and dim | |
| **Editing** | |
| Turn insert mode on | ESC q |
| Turn insert mode off | ESC r |
| Insert one space character | ESC Q |
| Insert line of space characters | ESC E |
| Delete cursor character | ESC W |
| Delete line | ESC R |
| **Clearing Data** | |
| Clear screen to nulls | ESC \* |
| Turn insert mode off | ESC r |
| Insert one space character | ESC Q |
| Insert line of space characters | ESC E |
| Delete cursor character | ESC W |
| Delete line | ESC R |
| **Local Keyboard Commands** | |
| Toggle keyclick | CTRL ENTER |
| Toggle monitor mode | CTRL LEFT ARROW |
| Toggle status line | CTRL RIGHT ARROW |
| Toggle BLK/FDX | CTRL BACKSPACE |
| Increase scrolling rate | CTRL UP ARROW |
| Decrease scrolling rate | CTRL DN ARROW |
| Unlock keyboard | SETUP |
| Reset all terminal modes | CTRL SETUP |
| Clear screen to nulls, home cursor | CTRL HOME |
| Toggle copy print mode | CTRL KPD 2 |
| Toggle transparent print mode | CTRL SHIFT KPD 2 |
| Print screen unformatted | CTRL KPD 0 |
| Print Screen Formatted | CTRL SHIFT KPD 0 |