PiCLOG
(Version 6.31)

Electric logging system without PC

(Winner of Japanese minister prize in 2010)

7K1PYG
Hiroyuki Nagasawa

Table of contents

1 Outline of PiCLOG

In case of low power operation (QRP), the electric power for communications can be supplied by batteries. However, electric logging using PC consumes much larger electric power than that for transceiver. This system, called 'PiCLOG', has been designed using micro controllers in order to reduce the power consumption for electric logging. This recodes QSO date/time, callsign, name, RS(T) reports, QTH, band, mode, messages to be written on QSL card, and so on. An extended keyboard makes on-time-logging easy. Peripheral RTC (Real Time Clock) always keeps UTC information, and it can be recorded every QSO.

The features of PiCLOG are;

- It launches up within one second after turning main switch on. It keeps operation for more than eight hours using four rechargeable batteries.
- Owing to optimized configuration of component, weight of the system is reduced below 0,5kg.
- QSO information can be entered using a commercially available keyboard via PS/2 port, and it recodes QSO information, like date/time(UTC), callsign, name, RS(T) reports, QTH, band, mode, frequency, and so on. The recording capacity exceeds 2000 QSOs, and stored data can be transferred to PC via RS-232C communication port.
- PiCLOG acts as a 'touch-coder', so that Morse cord can be sent by key input. This function never excludes keyer function. Morse code can be generated by external paddle as popular keyer is doing. The dot/dash ratio, key-pitch, and dot/space ratio can be adjusted by keyboard operation.
- Internal LCD shows key-setting, date/time, remaining memory, and characters to be sent.
- This includes enough function memory of 47 characters x 48 channels x 4 domains. Those can be selected/changed by keyboard. As popular keyer does, PiCLOG can send typical sentence for CQ-calling automatically/repeatedly.
- On recording callsign of other station, PiCLOG send it as Morse code automatically. Add to this, RST report and contest number can be sent.
- The input callsign can be searched, and this function is used for duplicate-check.
- The setting of PiCLOG can be switched between '1' and '2' immediately. This means PiCLOG includes two types of setting and memory.
- This includes searching function of callsign. This function shows you previous QSO with input callsign.
- Receiving trainer of Morse codes is implemented.
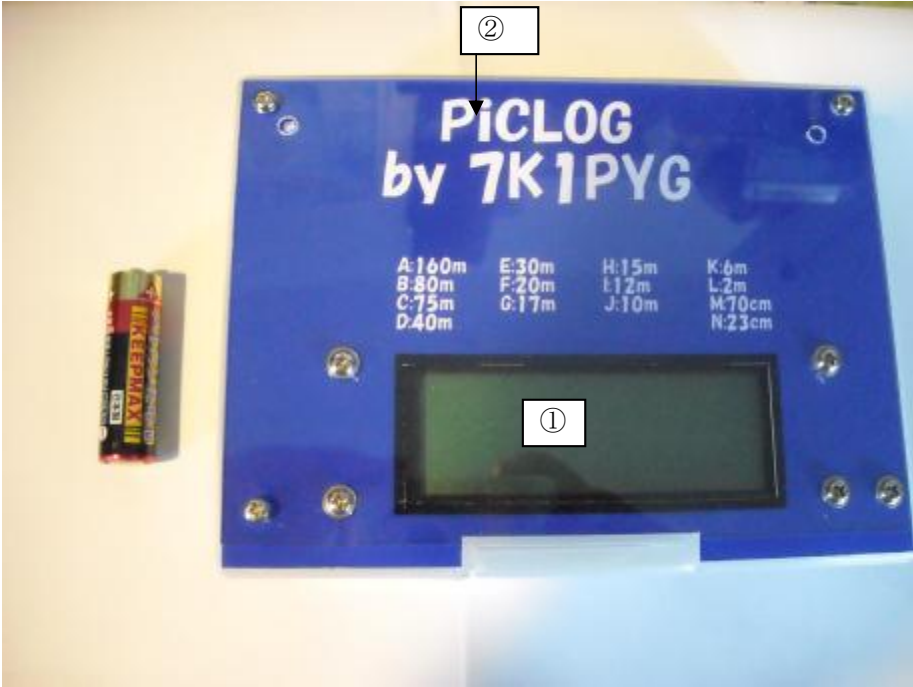
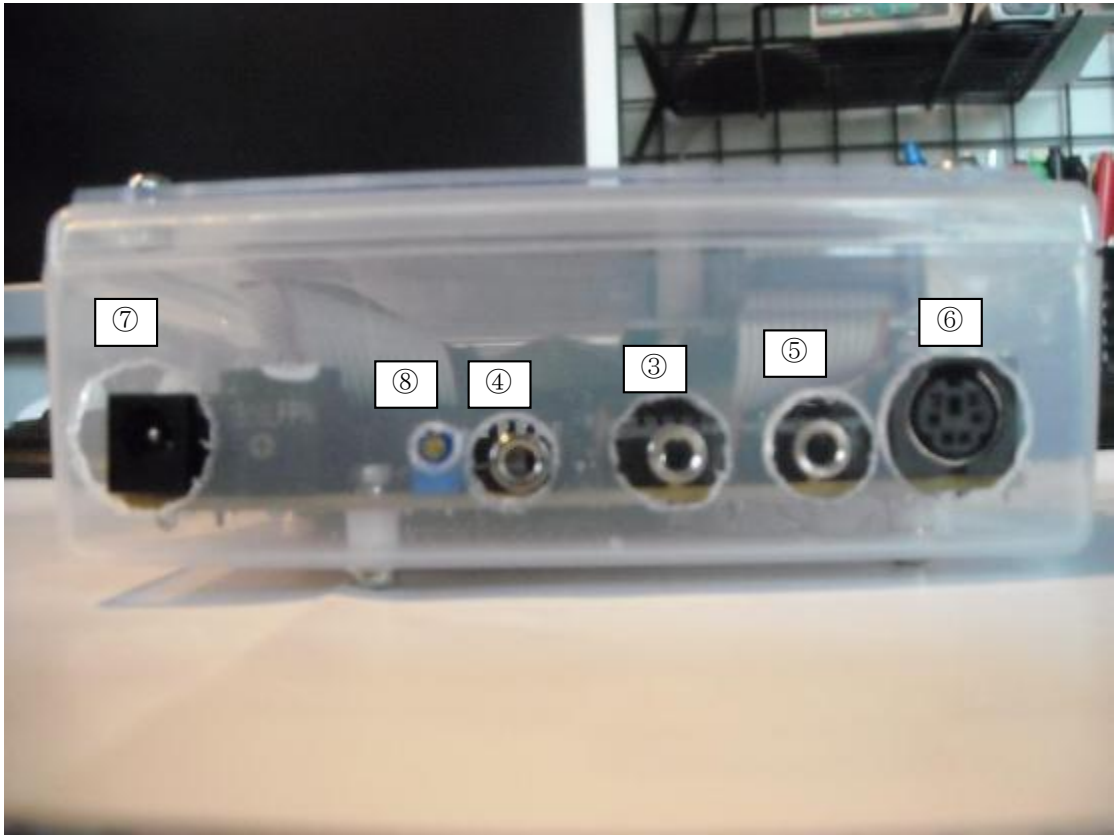## 2 Consistency of PiCLOG

Fig. 1 Front Panel

Fig. 2 Side view

①LCD:

Line 1	Operation mode, Key setting（Pitch, Dash/Dot, Space/Dot, Direction of paddle）, Buzzer, Band, Preset conditions
Line 2	Date/Time in UTC (YYMMDD-HHMMSS), EEPROM memory chip. Current-memory address on EEPROM
Line 3-4	Characters on buffer-memory

②PTT indicator：

③Paddle port：   3.5mm stereo jack

④CW port：   3.5mm stereo jack

⑤Data I/O port：   3.5mm stereo jack communicates with PiCLOG on RS-232C level (Baud rate=9600, 8bits, Non-parity, StopBit=1). Log data can be transferred to PC

⑥Keyboard port：   Keyboard with PS/2(mini DIN6P) inputs numeric/alphabetical characters

⑦External DC line：   DC 8-12V   G-(+)

⑧LCD contrast：

Fig. 3 Inside of PiCLOG

Fig. 3 shows inside of PiCLOG. This consists with PCB, battery box, and LCD. Two microcontrollers (PIC16F88 and PIC16F887(Typical or PIC16F877A, PIC16F1839) operate the entire system. PIC16F88 transfers the key-codes to ASCII-code. Then PIC16F887 operates LCD, EEPROM, keyer, and I/O functions. Peripheral modules like Real Time Clock (DS1305 for type B, RTC-8564NB for type C), RS-232C level converter (MAX232), and 20colum x 4line LCD(SC2004CSWB), are connected to the PICs.

Fig. 4 shows a typical configuration of external connections for PiCLOG.

Fig. 4 External connections for PiCLOG

3   Circuit diagram and list of components

Fig. 5 shows a circuit diagram of PiCLOG(Type C). The components listed in **Table. 1** are selected to make the system simplify as small as possible. Although PiCLOG can be operated using internal batteries, external power line (DC 8-12V) can be connected to keep the operation for longer time.

PIC16F88 converts the key input into 8bit ASCII code with TTL level. MAX232 converts the TTL level into RS232C level, and it is exported to data port. If the data port is unplugged, the RS232C level ASCII data turns back to MAX232, and it is converted to TTL level again to be conducted to USART terminal of PIC16F887 (PIC16F877A).

Two 512k EEPROMs and RTC are connected to PIC16F887 (PIC16F877A) via I^2C line. This makes PIC16F887 enable to read/write. Paddle input and CW output through the B port of PIC16F887. A photo-coupler, PC813, electrically isolates PiCLOG from RIG to avoid undesirable feedback circuit.

Fig. 5 Circuit diagram (Type C)

Table. 1 List of components

Parts name	Type	Specification
B1	Battery	CR2032H
BZ1	OSC Buzzer	BZ 5V
C2	Chemical capacitors	100uF 25V 85C
C9,C15		10uF 25V 85C
C1,C3,C6,C7,C8,C10,C11,C12,C13,C14	Ceramic capacitors	0.1uF
C4,C5	Ceramic capacitors	0.01uF
D1,D2	Small signal diodes	1S2076 150mA
D3,D4		1S133 100mA
F1	Resetable fuse	170mA
IC2	PICs	PIC16F887 (PIC16F877A, PIC16F1839)
IC3		PIC16F88
IC4	RS232C level converter	ADM3202AN
IC5	Real Time Clock	RTC-8564NB
IC6,IC7	EEPROMs	24LC512
IC1	Voltage regulator	7805
OK1	Photo-coupler	PC813
Q1	Ceramic resonator	4MHz
Q2	Ceramic resonator	10MHz
R4,R5,R6,R7,R10	Resistance (1/6W)	10k
R8,R9		150
R3,R11,R12		2k
R1,R2		1k
T1	npn Transistor	2SC1518Y
VR1	10k Trimmer	LCD contrast
J2,J3,J4	Jack	3.5mm Stereo
X4	mini DIN6P	PS/2
X5	PIN header 2x7	LCD data line
J1	DC jack	DC_IN 8-12V

LED	5mm	
LCD	20×4 LCD character display	SC2004CSLB
Battery	Battery BOX	AAA x 4

4 How to use PiCLOG

4.1 Preparation

① Turn the main switch off
② Set four rechargeable AA batteries (1.2V) into the battery box. Otherwise, supply DC 8 − 12 V via external power line.
③ Connect keyboard to PS/2 port
④ Connect paddle to PDL port
⑤ Turn the main switch on. Then PiCLOG is launched. (If not, check connections of battery, LCD cable, and so on)
⑥ Check LCD display:
  - Line1 shows key setting like "**aA8d80w80sx3kcmbhnrp**"
  - Line2 shows date/time with YYMMDD(UTC) format. This value is incremented every second.
⑦ Input some alphabetical/numerical characters through keyboard. Then, those are displayed on line3-4 simultaneously. (Non alphabetical characters, like Å, Ä, and Ö, are not supported)

4.1.1 Setting of internal real time clock (RTC)

First of all, present date and time (UTC) have to be stored in PiCLOG by following procedure.
- Put **[ESC]** to erase characters stored in buffer-memory. Then, line3 and 4 are cleared.
- Input date and time with YYMMDDHHMM format in UTC as like '1103061308' (This means 8minutes pass 1pm on March 6 in 2011).
- Put **[Ctrl]**+**[T]**. If the input characters correspond to validate date/time, current date and time are indicated as '**110306-1308**' on line 2. Since the ticking of RTC is maintained by backup battery (CR2032), the current date/time information is incremented automatically even after turning the switch off.

4.1.2 How to switch the puddles of manipulator

The puddles of manipulator can be switched by **[Ctrl] + [K]**. This function switches combination with Dot/Dash and Left/Right puddles. On each operation, character on LCD (line 1, col 13) is changed between 'K' and 'k' alternately.

4.1.3 Buzzer on/off

The buzzer mode can be switched by **[Ctrl] + [S]**. On each operation, character on LCD (line 1, col 10) is changed between 'S' and 's' alternately. This function determines how buzzer operates during QSO. If the status character shows 'S' buzzer operates synchronizing Morse code. Otherwise, buzzer keeps quiet during QSO.

4.2 Operation mode of PiCLOG

PiCLOG includes four operation modes. Manual, Automatic, Chat, and Training. Manual mode makes QSO step by step on each key operation. On the other hand, Automatic mode manages QSO automatically like CQ calling, recording his callsign and RST reports, calling back his callsign. Chat mode act as touch-coder

so as to send Morse code on each key input. Training mode generates random callsign, and it is output via internal buzzer separating from RIG. This function brushes the hearing ability up.

The operation mode of PiCLOG is summarized in Table. 2

Table. 2 Operation mode of PiCLOG

Operation	Function	Remark
[Ctrl]+[A]	Selecting Automatic and Manual modes	The operation mode is switched into Manual or Automatic mode on each operation. The operating mode of PiCLOG is indicated in line1 col 1 with alphabetical. It toggles 'a' (Manual mode) and 'A' (Automatic mode) every [Ctrl]+[A] operation. Default='a'
[Ctrl]+[C]	Switch into Chat mode	This operation changes the PiCLOG into Chat mode. During Chat mode, character 'C' is indicated in line1 col 1. On key input, the correspond Morse code is send immediately. [ESC] or [Ctrl]+[A] change the mode into Automatic or Manual modes.
[Ctrl]+[M]	Switch into Training mode	This operation changes the PiCLOG into Training mode. During Training mode, character 'T' is indicated in line1 col 1. PiCLOG generates random callsign and outputs Morse codes via internal buzzer. However, the CW port is disconnected from RIG. Then received callsign has to be input via keyboard. If the input callsign is incorrect, PiCLOG output correct callsign via internal buzzer. Then, PiCLOG indicates the correct callsign and 'Hit any Key' on line 4. [Ctrl]+[M] operation after indicating 'Hit any Key' changes the PiCLOG into Automatic or Manual mode.
[Ctrl]+[X]	Sending interval of [F1] channel	The interval is indicated in line1 col 12 with numerical. Increment 0 to 9 stepping 1. This value means interval in seconds between sending sentences stored in [F1] channel. This function is available when PiCLOG is operated in Auto mode.
[Ctrl]+[K]	Dash/Dot configuration on paddle	The paddle configuration is indicated in line1 col 13 with alphabetical. Default='k'. It toggles 'k' (Left paddle corresponds to Dash) and 'K' (Right paddle corresponds to Dash) every [Ctrl]+[K] operation
[Ctrl]+[S]	Internal buzzer	The buzzer status is indicated in line1 col 10 with alphabetical. Default='s'. It toggles 's' (non beep) and 'S' (beep) every [Ctrl]+[S] operation.

4.3 Keyer function

PiCLOG acts as keyer by connecting the external paddle. This function is activated immediatly on turning the main switch on. The keyer setting is summarized in **Table. 3**.

Table. 3 Keyer setting

Operation	Function	Remark

[Shift]+[Y] or [Shift]+[Q]	Increasing the key-speed	The key-speed is indicated in line1 col2-3 with hexadecimal. Default=A8(20WPM). Increment step = 4. From 0 to FC
[Shift]+[U] or [Shift]+[Q]	Decreasing of key-speed	The key-speed is indicated in line1 col2-3 with hexadecimal. Default=A8(20WPM). Decrement step = 4. From FC to 0
[Shift]+[D]	Dash/Dot ratio	The Dash/Dot ratio is indicated in line1 col5-6 with hexadecimal. Default=80(3:1). Increment step = 4. From 0(1:1,5) to FC(1:4,5)
[Shift]+[W]	Space/Dot ratio	The Space/Dot ratio in line1 col8-9 with hexadecimal. Default=80(1:1). Increment step = 4. From 0(1:0,5) to FC(1:1,5)
[Shift]+[I]	Reset	This operation sets the keyer function parameter default.

4.4   Presetting in memory of PiCLOG

Some of static information like operator's callsign, QSO mode, band, frequency, operating location, power, and gridsquare of operating location can be recorded in PiCLOG as preset values. Those can be recorded into Log-memory on every QSO, and be easily transferred to PC. The data presetting is not mandatory for PiCLOG operation, but it makes logging easy. Table. 4 describes how to use the preset memory function.

Table. 4 Preset memory

Operation	Function	Remark
Setting [Alt]+[Q]		This operation is carried out to declare the following preset value is applied after the next QSO. Otherwise, input preset value is applied to latest QSO too.
Setting, Erasing [Ctrl]+[Alt]+[C] **Confirmation [?] – [Alt]+[C]** **Transmit [Alt]+[C]** or **[Home]**	Presetting operator's callsign	This operation sets the default value of operator's callsign into PiCLOG. Prior to input the callsign, buffer-memory has to be erased by [ESC]. The callsign can be input as is (for example '7K1PYG'). The input callsign is recorded in preset memory by operating [Ctrl]+[Alt]+[C]. After recording the callsign, indication of LCD line1 col 14 is turned from 'c' to 'C'. The callsign in preset memory is re-writable by same manner described in above. The preset memory for the callsign can be erased by operating [Ctrl]+[Alt]+[C] with empty buffer-memory. In this case, LCD indication of line1 col 14 is turned from 'C' to 'c'. The stored callsign can be sent as Morse code by operating [Alt]+[C] or [Home]. This operation never changes the buffer-memory. For confirmation of preset callsign, operate [Alt]+[C] after input [?]. Then preset callsign will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[M] **Confirmation [?] – [Alt]+[M]** **Transmit**	Presetting QSO Mode (x2) information	This operation sets the default value of mode(x2) information into PiCLOG. Prior to input the mode information, buffer-memory has to be erased by [ESC]. The mode information can be input as is or symbolic alphabet (for example 'CW' or 'C' for CW following the manner shown in **Table. 6**). The input mode information is recorded in preset memory by operating [Ctrl]+[Alt]+[M]. After recording the mode information, indication of LCD line1 col 15 is turned from 'm' to 'M'. The mode

[Alt]+[M]		information in preset memory is re-writable by same manner described in above. The preset memory for the mode information can be erased by operating [Ctrl]+[Alt]+[M] with empty buffer-memory. In this case, LCD indication of line1 col 15 is turned from 'M' to 'm'. The stored mode can be sent as Morse code by operating [Alt]+[M]. This operation never changes the buffer-memory. For confirmation of preset mode, operate [Alt]+[M] after input [?]. Then preset mode will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[B] **Confirmation [?] – [Alt]+[B]** **Transmit [Alt]+[B]**	Presetting Band information	This operation sets the default value of band information into PiCLOG. Prior to input the band information, buffer-memory has to be erased by [ESC]. Band information can be input as wave length or symbolic alphabet (for example '15M' or 'H' for 21MHz band). The relationship between band and symbolic alphabet is shown in Table. 5. The input band information is recorded in preset memory by operating [Ctrl]+[Alt]+[B]. After recording the band information, indication of LCD line1 col 16 is turned from 'b' to 'B'. The band information in preset memory is re-writable by same manner described in above. The preset memory for the band information can be erased by operating [Ctrl]+[Alt]+[B] with empty buffer-memory. In this case, LCD indication of line1 col 16 is turned from 'B' to 'b'. The stored band can be sent as Morse code by operating [Alt]+[B]. This operation never changes the buffer-memory. For confirmation of preset band, operate [Alt]+[B] after input [?]. Then preset band will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[H] **Confirmation [?] – [Alt]+[H]** **Transmit [Alt]+[H]**	Presetting the operating location.	This operation sets the default value of operator's location into PiCLOG. Prior to input the location, buffer-memory has to be erased by [ESC]. The location can be input as is (for example 'STOCKHOLM') or by CNTY (like 'JCC1026'). The input location is recorded in preset memory by operating [Ctrl]+[Alt]+[H]. After recording the location, indication of LCD line1 col 17 is turned from 'h' to 'H'. The location in preset memory is re-writable by same manner described in above. The preset memory for the location can be erased by operating [Ctrl]+[Alt]+[H] with empty buffer-memory. In this case, LCD indication of line1 col 17 is turned from 'H' to 'h'. The stored location can be sent as Morse code by operating [Alt]+[H]. This operation never changes the buffer-memory. For confirmation of preset location, operate [Alt]+[H] after input [?]. Then preset location will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[N] **Confirmation [?] – [Alt]+[N]** **Transmit [Alt]+[N]**	Presetting the operator name.	This operation sets the default value of operator's name into PiCLOG. Prior to input the name, buffer-memory has to be erased by [ESC]. The name can be input as is (for example 'HIRO'). The input name is recorded in preset memory by operating [Ctrl]+[Alt]+[N]. After recording the name, indication of LCD line1 col 18 is turned from 'n' to 'N'. The name in preset memory is re-writable by same manner described in above. The preset memory for the name can be erased by operating [Ctrl]+[Alt]+[N] with empty buffer-memory. In this case, LCD indication of line1 col 18 is turned from 'N' to 'n'.

		The stored name can be sent as Morse code by operating [Alt]+[N]. This operation never changes the buffer-memory. For confirmation of preset name, operate [Alt]+[N] after input [?]. Then preset name will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[R] **Confirmation [?] – [Alt]+[R]** **Transmit [Alt]+[R]**	Presetting the operator's RIG.	This operation sets the default value of operator's RIG into PiCLOG. Prior to input the RIG name, buffer-memory has to be erased by [ESC]. The RIG name can be input as is (for example 'FT-817'). The input RIG name is recorded in preset memory by operating [Ctrl]+[Alt]+[R]. After recording the name, indication of LCD line1 col 19 is turned from 'r' to 'R'. The RIG name in preset memory is re-writable by same manner described in above. The preset memory for the RIG name can be erased by operating [Ctrl]+[Alt]+[R] with empty buffer-memory. In this case, LCD indication of line1 col 19 is turned from 'R' to 'r'. The stored RIG name can be sent as Morse code by operating [Alt]+[R]. This operation never changes the buffer-memory. For confirmation of preset RIG name, operate [Alt]+[R] after input [?]. Then preset RIG name will be displayed on LCD.
Setting, Erasing [Ctrl]+[Alt]+[P] **Confirmation [?] – [Alt]+[P]** **Transmit [Alt]+[P]**	Presetting the transmitting power.	This operation sets the default value of transmitting power into PiCLOG. Prior to input the power, buffer-memory has to be erased by [ESC]. The power can be input as is (for example '5WTS'). The input power is recorded in preset memory by operating [Ctrl]+[Alt]+[P]. After recording the name, indication of LCD line1 col 20 is turned from 'p' to 'P'. The power in preset memory is re-writable by same manner described in above. The preset memory for transmitting power can be erased by operating [Ctrl]+[Alt]+[P] with empty buffer-memory. In this case, LCD indication of line1 col 20 is turned from 'P' to 'p'. The stored power can be sent as Morse code by operating [Alt]+[P]. This operation never changes the buffer-memory. For confirmation of preset power, operate [Alt]+[P] after input [?]. Then preset power will be displayed on LCD.

Table. 5 Symbols represented the band information

Symbol	Band
A	160m
B	80m
C	75m
D	40m
E	30m
F	20m
G	17m
H	15m

I	12m
J	10m
K	6m
L	2m
M	73cm
N	23cm

Table. 6   Symbols represented the mode information

Symbol	Mode
C	CW
F	FM
S	SSB
A	AM

4.5   Touch-coder function

PiCLOG converts characters into Morse codes. This function makes us enable sending Morse codes without paddle. The characters input through the keyboard are indicated in LCD (line 3 − 4). Simultaneously, those are stored in buffer-memory. The buffer-memory has capacity for 47 characters.

Some of special Morse codes are represented by symbolic key shown in **Table. 7**.

Table. 7 Symbolic key related to special Morse cord

Key	Morse cord
}	$\overline{KN}$
]	$\overline{AR}$
[	$\overline{BK}$
=	$\overline{BT}$
>	$\overline{SK}$

4.5.1 Operation with Automatic or Manual mode

The stored characters in buffer memory are not immediately converted into Morse code. On hitting the [Enter] key, the stored characters are converted into Morse codes to be transmitted sequentially.

4.5.2 Operation with Chat mode

In order to switch PiCLOG into Chat mode, operation by [Ctrl]+[C] is needed. During Chat mode, characters input through keyboard are transmitted immediately as Morse code. Newly input characters during Morse code transmission are stored in buffer memory, and those are transmitted sequentially. However, number of unsent characters in buffer memory cannot exceed 47. Chat mode can be terminated

by [ESC] or [Ctrl]+[C].

4.6 CW memory function

PiCLOG consists of several memory, buffer-, stamp-, current-, preset-, and log-memories. As shown in Fig. 6, Buffer-memory is connected with LCD directory. LDC shows characters in buffer-memory any time. The characters in buffer-memory can be transferred to current memory by combining [Ctrl] and some of function key. If [Alt] and/or [Shift] are combined with above operation, the content in buffer-memory is transferred to Stamp-memory. Some combination of [Ctrl] and alphabet transfers the content of buffer-memory to preset-memory. On updating the current-memory, the content in buffer-memory is also recorded in log-memory by combining data in preset-memory.
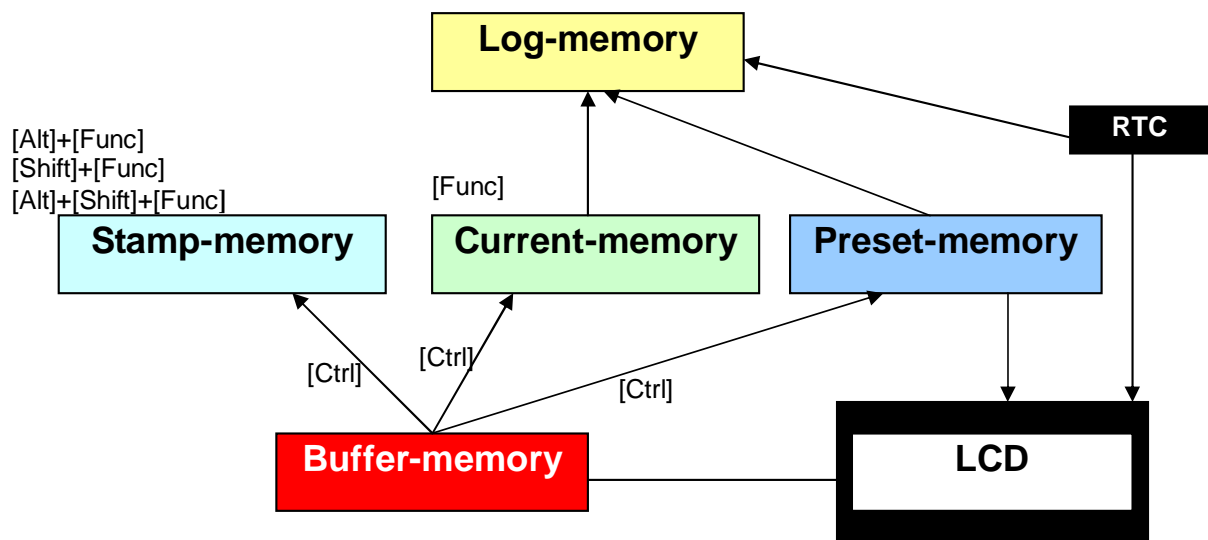

Fig. 6 Memory configuration

The memory function is summarized in Table. 8.

Table. 8 CW memory operations

Operation	Function	Remarks
[F*] **(*=1~12)**	This sends content of current-memory as Morse code.	
[Ctrl]+[F*] **(*=1~12)**	This records content in buffer-memory to current-memory	The content for buffer-memory is recorded in Log-memory simultaneously.
([Alt]) and/or ([Shift]) + [F*] **(*=1~12)**	This sends content of stamp-memory as Morse code.	Stamp-memory of 36 channels can be selected by combining the [ALT], [Shift],and [F*].
[Ctrl]+([Alt]) and/or ([Shift])+[F*] **(*=1~12)**	This records content of buffer-memory into stamp-memory.	
[Ctrl]+[Z]	This changes EEPROMs between chip1 and chip2	This operation toggles chip1 and chip2. By changing the EEPROMs, Keyer

		setting also changed synchronizing the chip number.
[TAB]	This records content in buffer-memory to stamp- and log-memories as his callsign. Content in stamp-memory [F2] is also recorded to log-memory. On finishing the recording, the stored his callsign and RST report are sent by Morse codes.	This function works independently with operation mode of PiCLOG, 'auto' or 'manual'.
[HOME]	This sends operator's callsign, which is recorded in preset memory, by Morse codes.	This operation never changes the content of buffer-memory.
[↑]	This records content in buffer-memory to stamp- and log-memories as his callsign. Contents in stamp-memory his callsign, [F2](RST) and [F9](STX) are also recorded to log-memory as callsign, RST and STX, respectively. On finishing the recording, the stored his callsign, RST and STX are sent by Morse codes sequentially.	The contents in stamp-memory are also updated, and this function is useful for CQ-calling. This function works independently with operation mode of PiCLOG, 'auto' or 'manual'.
[→]	This records contents in stamp-memory [F2] and [F9] to log-memory as RST and STX, respectively. On finishing the recording, the updated RST and STX are sent by Morse codes sequentially.	The contents in stamp-memory are also updated, and this function is useful for CQ-calling. The content in buffer-memory is NOT recorded as his callsign. This function works independently with operation mode of PiCLOG, 'auto' or 'manual'.
[↓]	This sends his callsigne, [F2](RST) and [F9](STX in stamp-memory by Morse codes sequentially.	The contents in stamp-memory are sent without updateding. This function is used on receiving a request for re-send.
[←]	This sends [F2](RST) and [F9](STX in stamp-memory by Morse codes sequentially.	The contents in stamp-memory are sent without updateding. This function is used on receiving a request for re-send.
[PageUp]	This records content in buffer-memory to stamp- and log-memories as his callsign. Then contest number of 001 style is stored in stamp-memory after incrimination. Contents in stamp-memory his callsign, [F2](RST) and [F9](STX) are also recorded to log-memory as callsign, RST and STX, respectively. On finishing the recording, the stored his callsign, RST and STX are sent by Morse codes sequentially.	The contents in stamp-memory are also updated, and this function is useful for CQ-calling on contest. This function works independently with operation mode of PiCLOG, 'auto' or 'manual'.
[PageDown]	This records contest number of 001 style in stamp-memory after incrimination. Then the contents in stamp-memory [F2](RST) and [F9](STX) are also recorded to log-memory as RST and STX, respectively. On finishing the recording, the stored RST and STX are	The contents in stamp-memory are also updated, and this function is useful for CQ searching on contest. This function works independently with operation mode of PiCLOG, 'auto' or 'manual'.

	sent by Morse codes sequentially.	
[End]	This initializes 001-style contest number as 000	The initial contest number is incremented on [PageUp] or [PageDown] operation, and sent as 001
[Alt]+[+]	This increments 001-style contest number manually. On incrimination, the latest value is indicated on LCD	Maximum value of the number is 9999
[Alt]+[-]	This decrements 001-style contest number manually. On discrimination, the latest value is indicated on LCD	Minimum value of the number is 000
[Ctrl]+[O]	This switches sending of RST and STX as alphabetical or number.	With this operation, indication on LCD Line-2, col-16 is switched between '_' (Under bar as alphabetical) and '-' (Hyphen as numeric). During the alphabetical mode, some Arabic number is converted into alphabet as shown following manner. ('0'→'T'、'1'→'A'、'9'→'N')
[Ctrl]+[Shift]+[E]	This erases totally log-memory, which is used currently.	On this operation, base address of log-memory (0900) is indicated on LCD line-2, col-17
[Ctrl]+[Shift]+[R]	This recovers the erased log-memory. This means cancellation of log-memory erasing	On this operation, latest address of log-memory is indicated on LCD line-2, col-17
[Ctrl]+[Ins]	This executes callsign searching as regarding that content in buffer-memory is target callsign	This operation never changes the content of buffer-memory, even if some words is added on LCD. During searching,'-' is added after the target callsign. On finding the target callsign, QSO date/time is indicated just after the '-'. Otherwise, 'X' is add. The hyphen and following characters are not added in buffer-memory.
[Ctrl]+[Del]	This deletes a latest QSO record from log-memory	Latest data is deleted on operating [Del]. On deleting the record, address on LCD line-2, col-17 is updated.

4.6.1 Memory mapping on EEPROM

PiCLOG recodes temporally QSO information like callsign, operator's name, RST report, QTH, and so on. Add to this, typical sentences for QSO can be stored in EEPROMs. PiCLOG involves buffer-memory of 48 channels x 47 characters. Among of them, memory area corresponds to 12 channels, called current-memory, recodes tentative information which be modified every QSO. However, Log-memory, that is described later, stores the content of current-memory simultaneously to remain every QSO information. The recodes stored in Log-memory can be downloaded as ADI format later.

Reserved memory area, which corresponds to 36 channels, recodes typical sentences those might not be changed each QSO. This memory area is called stamp-memory. Among of stamp-memory, channels related [F1] key can be used for 'CQ-calling'. When PiCLOG is set as Auto-mode, sentences related with [F1] key are sent repeatedly with desired interval determined by [Ctrl]+[X]. **Table. 9** shows memory mapping of PiCLOG.

Table. 9 Memory mapping of PiCLOG: Two EEPORMs are connected to

PiCLOG. Those include same memory configuration.

Memory address (Hexadecimal)	Name of memory
0000 - 023F	Stamp-memory [Alt]+[Function-keys]
0240 - 047F	Current-memory
0480 - 06BF	Stamp-memory [Shift]+[Function-keys]
06C0 - 08FF	Stamp-memory [Alt]+[Shift]+[Function-keys]
0900 - FFFF	Log-memory

4.6.2 Current-memory (synchronized recording into log-memory)

Typical QSO information can be stored in 'current-memory' of PiCLOG. The current-memory consists of 12 channels x 3 domains, and this recodes QSO information like callsign, RS(T) reports, band, mode, name, frequency, RIG, QTH, contest number, comments, and message to be written on QSL card. Each function key between [F1] to [F12] is related to specific QSO information. **Table. 10** shows correlation between QSO information and function key.

The current memory can be modified by combining [Ctrl] and [Function key]. For example, if his callsign is stored in buffer memory, put [Ctrl]+[F1]. Then callsign stored in buffer memory is transferred to current memory. Simultaneously, the alternative data in current memory is recorded in Log-memory synchronizingly. This means that QSO information can be recorded in Log memory on changing data in current memory. Specially, current memory recording in [F1] channel, this is rewriting of callsign, results in entering new QSO record as a manner described below. When [F1] current-channels (His callsign) are modified by [Ctrl]+[F1] operation, callsign (this corresponds to data in buffer memory), date/time(UTC) information and contents in preset-memory (operator's callsign, operator's name, band, mode, RIG, TX-power, QTH, and gridsquare) are recorded into log-memory simultaneously.

Stored data in current memory, like his callsign, his name, and so on, are thought to be employed during QSO sometime. The stored data can be transferred as Morse codes by operation of [function key]. For example, [F1] results in sending of callsign, [F2] for his RST, and [F4] for his name.

Table. 10 Correlation between current-memory and function-key

Function key	Content to be recorded in current-memory
F1	His callsign
F2	RS(T) to be sent
F3	Recieved RS(T)
F4	His name
F5	His QTH
F6	Frequency in MHz

F7	His RIG
F8	His power in Watts
F9	Contest number to be sent (STX)
F10	Received contest number (SRX)
F11	Remark
F12	Message to be written on QSL card (QSL_MSG)

4.6.3 Stamp-memory

Typical sentences, those are employed frequently during QSO in dependent on person in QSO, are stored in stamp-memory. Say, the stamp-memory is just employed for "RUBBER STAMP QSO".

Stamp-memory consists with three domains, those are selected by [Function key] combined with [Alt] and/or [Shift] keys. Each domain is divided into 12 channels. Each channel is related with specific function keys between [F1] to [F12]. Among of them, channels related with [F1] keep typical sentence for CQ-calling (like "CQ CQ DE 7K1PYG PSE K"), and the sentence for CQ-calling can be sent automatically/repeatedly when PiCLOG is set as Auto-mode.

Any data can be recorded by combining the [Ctrl] and [Function key]. [Alt] and/or [Shift] keys have to be put to select memory domain.   Otherwise, the data in buffer memory is stored in current memory.

The recording procedure targeting [F1] channel stamp memory of [Alt]-domain is written below.

①Hit the [ESC] key to erase the buffer-memory.

②Input the desired CQ-calling sentence like 'CQ CQ DE 7K1PYG 7K1PYG/1 PSE K' using keyboard.

③Push [Ctrl]+[Alt]+[F1] sequentially. Then sentence stored in buffer-memory is transferred to stamp-memory, and LCD line 3 and 4 are erased.

④Push [Alt]+[F1] sequentially to send the stored sentence. The sentence is sent repeatedly as CQ-calling when PiCLOG is operated with Auto-mode. The interval between each CQ-calling is determined by [Ctrl]+[X].

⑤The CQ-calling can be immediately terminated by hitting any key or paddle, so as to input received callsign as regarding CQ-calling.

4.6.4 Switchable EEPROM

PiCLOG includes two 512kbyte EEPROMs, those are defined as EEPROM-1 and EEPROM-2. One of them is activated depend on the EEPROM selection. Although both EEPROMs include equivalent memory mapping shown in **Table. 9**, contents in each memory are totally separated. The setting of keyer is also synchronized with EEPROM selection. This means that the PiCLOG involves two customized operating condition that includes own log-memory, stamp-memory, current-memory, and keyer-setting. By operating [Ctrl]+[Z], EEPROM-1 or EEPROM-2 is activated alternately. This function makes PiCLOG to adopt different situations. For example, EEPROM-1 is employed at home, while EEPROM-2 at outdoor. The activated EEPROM number is indicated at line 2, col 15. Each EEPROM recodes about 1000 QSOs.

4.6.5 'Record-and-Play' function

In order to make the logging operation easy, PiCLOG includes several automatic functions, those are called 'Record-and-Play' function, described below.

Automatic recording/sending of his callsign and RST:

1. If you receive callsign as a reply for your CQ-calling, input it using keyboard. Then the CQ-calling is automatically terminated and his callsign is stored in buffer-memory.
2. Put [TAB] key. Next, callsign stored in the buffer-memory is transferred to [F1] channel of current-memory and log-memory simultaneously. Of course, current date/time and preset memories are also stored in log-memory. Then, callsign and His_RST (this is pre-recorded in [F2] channel of current-memory) are sent sequentially.

This function is available independently on operation mode of PiCLOG, automatic- or manual-mode.

Automatic recording/sending on contest (in case a contest number that constant with number of QSO):

If the contest number is constant independently with QSO count, like CQWW DX contest, [↑]、[↓]、[→]、[←] keys make operation easy as manner described below.

1. Start CQ-calling using [F1] channel of stamp-memory by ([Alt] and/or [Shift])+[F1] key.
2. The callsign received is input via keyboard, and it terminates the CQ-calling.
3. Then put [↑] key. Callsign stored in buffer-memory is transferred to current- ([F1]) and log- memories. Simultaneously, the received callsign, RST-report([F2]), and contest number([F9]:STX) are sent sequentially.
4. As a reply to your contest number (STX), you might receive contest number (SRX). Input the SRX using keyboard, then put [Ctrl]+[F10] to record it to current- and log- memories.

If you put [→] key instead of [↑], RST-report([F2]), and contest number([F9]:STX) are sent/recorded sequentially, but not callsign. This operation might be helpful for CQ-searching.

If you are requested re-send the contest number by 'NR?', put [←] (or [↓]). This operation sends (callsign), RST-report([F2]), and contest number([F9]:STX) skipping recording, so as to avoid undesirable recording in log-memory.

Automatic recording/sending on contest (in case the incremental contest number '001'):

If the contest number is sequential and dependents on the QSO count, [PageUp] or [←] keys make the operation easy as described below.

1. Prior to the contest, put [End] key to initialize the contest number as '000'.
2. Start CQ-calling using [F1] channel of stamp-memory by ([Alt] and/or [Shift])+[F1] key.
3. The callsign received is input via keyboard, and it terminates the CQ-calling.
4. Then put [PageUp] key. Callsign stored in buffer-memory is transferred to current- ([F1]) and log- memories. Simultaneously, the received callsign, RST-report([F2]), and incremented contest number([F9]:STX) are sent sequentially.
5. As a reply to your contest number (STX), you might receive contest number (SRX). Input the SRX using keyboard, then put [Ctrl]+[F10] to record it into current- and log- memories.

If you put [PageDown] key instead of [PageUp], RST-report([F2]), and incremented contest number([F9]:STX) are sent/recorded sequentially, but not callsign. This operation might be helpful when you join the contest as CQ-searching.

If you are requested re-send the contest number, put [←] (or [↓]). This operation sends (callsign), RST-report([F2]), and contest number([F9]:STX) skipping recording and increment, so as to avoid duplicated recording in log-memory.

The contest number can be adjusted by [+]([-]) key for increment (decrement). On incrementing (decrementing), the new contest number is indicated on LCD.

4.6.6 Alphabetical RST and contest number

The RST and STX can be sent as alphabetical ('T' as 0, 'A' as 1, 'N' as 9) or number as is. [Ctrl]+[N] operation switches the sending manner of RST and STX. The setting is indicated in LCD line 2, col 16 as '-'(case of numeric) or '_' (case of alphabetical).

4.6.7 Record deletion in log-memory

The latest record in log-memory can be erased by [Ctrl]+[Del]. Each [Ctrl]+[Del] operation deletes newest record in log-memory sequentially. Duplicate [Ctrl]+[Del] operation results in undesirable deletion of record.

Entire log-memory can be erased by operating [Ctrl]+[Shift]+[E].

4.6.8 Log searching

Callsign in log-memory can be searched by [Ctrl]+[Ins]. In this case, the stored data in buffer memory is regarded as target callsign. The character '-' is attached tail of target callsign displayed on LCD. The searching is started from latest record to past. The searching is stopped at out of record or on finding a latest QSO with target callsign. The callsign in log-memory has to be exactly equal with target one. This means that the callsign "7K1PYG" differs from "7K1PYG/1".

On finding a record agreed with target callsign, QSO-DateTime is added just after the '-' by YYMMDDHHMM style. If the target callsign cannot be founded in the log-memory, just 'X' is added after '-'.

This function never changes the content in buffer-memory, even though some additional characters are added on LCD. Therefore, the target callsign in buffer-memory can be recorded in current- and log-memories by [Ctrl]+[F1].

4.6.9 Sending callsign in preset-memory

As described in 4.4, operator's callsign, mode, band, frequency, tx-power, QTH, and gridsquare can be stored in preset-memory. Among of the preset-memories, operator's callsign can be sent as Morse code skipping the buffer-memory by [Home] or [Alt]+[C] operation. This function never affects the content of buffer-memory, and it is useful on CQ-searching. Assuming that callsign 'JH1ZMK' is stored in buffer-memory during CQ-searching. Then operator's callsign '7K1PYG' in preset-memory can be sent by [Home] (or [Ctrl]+[C]). In this case, the callsign 'JH1ZMK' is remained in buffer-memory, and it can be stored in current- and log-memories by [Ctrl]+[F1] on receiving a call back from JH1ZMK.

4.7 How to transfer the log data

The log data stored in PiCLOG can be transferred to PC through RS-232C series line. The RS-232C setting is summarized in **Table. 11**.

Table. 11 RS-232C setting

Item	Value
Baud rate	9600
Character bit	8
Stop bit	1
Parity	No

Flow control	No

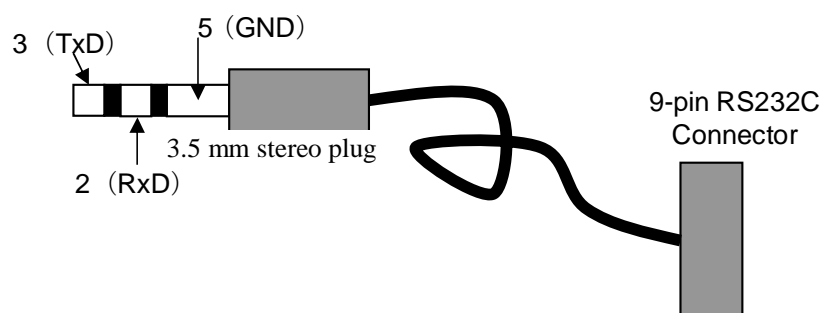**Fig. 7** shows data cable to be connected to PiCLOG.

Fig. 7 PiCLOG data cable

4.7.1 Manual transfer of log data
The data transfer is carried out manually by following procedure.
① Prior to conect the data cable, turn the main switch on. Then select the EEPROM chip number by [Ctrl]+[Z].
② Turn the main switch off. Then connect PC and PiCLOG using data cable.
③ Activate communication software on PC. Then turn the main switch on.
④ Send 'q' to PiCLOG from PC. Then PiCLOG starts data transfer. The data transfer is terminated at end of record.

The stored data in log-memory can be erased by [Ctrl]+[Shif]+[E] after detaching the data cable. Otherwise, send 'e' to PiCLOG from PC. On erasing the log-memory, EEPROM address displayed on LCD is updated as '0900'. If the log-memory is erased by mistake, [Ctrl]+[Shift]+[R] operation (or send 'R' to PiCLOG from PC) recovers the log-memory.

PiCLOG can be operated by PC using some control codes listed in Table. 12.

Table. 12 Controlling code of PiCLOG

Controling code	Function
z	This code switches the EEPROM chip 1 or 2
q	This code executes the transfer of log-data. The data format is indicated in **Fig. 8**
e	This code erases all of the data in log-memory
r	This code restores the erased log-data
n	This code forces PiCLOG to send RST and contest number as numerical.

l	This code executes the transfer of PiCLOG setting
i	This code initializes the keyer setting
a	This code changes the operation mode of PiCLOG auto- or manual-mode.
b	This code sets the content of buffer-memory as band in preset-memory
c	This code sets the content of buffer-memory as operator's callsign in preset-memory
d	This code increments the Dash/Dot ratio
f	This code sets the content of buffer-memory as frequency in preset-memory
h	This code sets the content of buffer-memory as QTH in preset-memory
g	This code sets the content of buffer-memory as gridsquare in preset-memory
m	This code sets the content of buffer-memory as mode in preset-memory
p	This code sets the content of buffer-memory as tx-power in preset-memory
s	This code turns the internal buzzer on or off
t	This code sets the content of buffer-memory as date and time
y	This code increments the keying speed
u	This code decrement the keying speed
w	This code increments the Dot/Space weight
x	This code increments the interval of CQ-calling
Alphabet with uppercase or number	The input characters are stored in buffer-memory sequentially
[Enter]	Characters stored in buffer-memory are sent as Morse code sequentially

Fig. 8 Log-data format

80 (Hex) Record head	ASCII character array	Delimiters by hex (Table. 13)	ASCII character array	· · · ·	Null(0) or FF (Hex)

Table. 13 Delimiter in log-data

Delimiter（Hex）	Definition
01	Operator's callsign in preset-memory
02	Mode in preset-memory
03	Frequency (MHz) in preset-memory
04	Band in preset-memory
05	My QTH in preset-memory
06	TX-power (W) in preset-memory
07	Gridsquare of my QTH in preset-memory
08	QSO data and time (YYMMDDHHMM)
80	His callsign (This act as record header)
81	His RST
82	My RST
83	His name
84	His QTH
85	RX-frequency (MHz)
86	TX-power (W)
87	RX-power (W)
88	Sent contest number (STX)
89	Received contest number (SRX)
8A	Comments
8B	Message to be written on QSL card (QSL_MSG)

4.7.2 Data transfer using software 'PiCUpp'

The data transfer software called 'PiCUpp' can operate on Windows XP. Performance on other version of windows is not guaranteed. PiCUpp can be installed using installer.

- On starting PiCUpp, available COM-ports appear in left listbox.
- Select a COM-port on listbox, which is connected with PiCLOG. Then PiCUpp communicates with PiCUpp and shows version information on status bar.
- Push 'Fetch' button or select fetch on menu. Then data transfer is started. The data transfer is terminated when latest record is transferred.
- On finishing the data transfer, dialog box appears to ask you to erase the log-memory. Push [Ok] for erase. Otherwise, push [Cancel].
  The erasing or recovering can be carried out by selecting on Data menu.
  EEPROM chip can be switched from PiCLOG by selecting [Data]-[Change EEPROM]
  In case that you are unable to download the log-data by [Fetch], the data structure in PiCLOG might be destroyed by some reason. In order to bail the available log data as much as possible, in that case, check [Data]-[Bulk Fetch]. This operation makes PiCLOG to ignore the tail-delimiter, and all of the data will be downloaded. You might get past data,

those are not meaningless. Finally, you can erase the meaningless data using editor.

those are not meaningless. Finally, you can erase the meaningless data using editor.