

# PV DATA SENDER/DATA RECORDER/BKP CONVERTER MANUAL

Copyright © 2000 CASIO Computer Co.,LTD

## 1.INTRODUCTION

This manual describes the following tools, which support add-in dictionary development.

### Windows Applications

PV DATA SENDER (**PVS\_dev.exe**)

BKP CONVERTER (**BKPC.exe**)

### PV Application for Add-in

DATA RECORDER (**datarec.bin**)

### 1.1 PV DATA SENDER

PV DATA SENDER is an add-in dictionary development support tool for Windows and sends the dictionary data from a personal computer (to be referred to as the PC hereinafter) to a pocket viewer (to be referred to as the PV hereinafter). (**See 4. OPERATION (PV DATA SENDER)**)

### 1.2 DATA RECORDER

DATA RECORDER is an add-in dictionary development tool only for the PV and receives the dictionary data sent from PV DATA SENDER and registers them as the add-in data in the PV. (**See 5. OPERATION (DATA RECORDER)**)

### 1.3 BKP CONVERTER

BKP CONVERTER is an add-in dictionary development support tool for Windows and converts the dictionary data registered in the PV into a memory file (MEM/DMEM file), which can be reproduced by the SIM3022 simulator for Windows (**Sim3022.exe**)(to be referred to as the simulator hereinafter). (**See 6. OPERATION (BKP CONVERTER)**)

## 2.EXECUTION ENVIRONMENT

It is recommended to run PV DATA SENDER/BKP CONVERTER in the following environment.

<b>OS</b>	Windows 95/98, and Windows NT 4.0SP3 or later
<b>CPU</b>	Pentium 200MHz or more
<b>Memory</b>	64 MB or more
<b>HDD</b>	Free capacity 10 MB or more

DATA RECORDER works with the PV where add-in registration is enabled.

### 3.DEVELOPMENT OF ADD-IN DICTIONARY DATA

This section describes how to develop the add-in dictionary data and the details of each process.

#### 3.1 Add-in Dictionary Data Development Flow

<1> Using SDK, prepare the add-in dictionary software.

<2> Create the dictionary data corresponding to the add-in dictionary software.

<3> Using PV APPLICATION MANAGER (**PVM.exe**), register the exclusive add-in software, DATA RECORDER (**datarec.bin**), in the PV.

**PC->PV DATA RECORDER**

<4> Using PVDATA SENDER (**PVS\_dev.exe**) (PC tool) and DATA RECORDER (PV tool), send the dictionary data from the PC to the PV for registration.

**PC->PV Dictionary data**

<5> Using PC SYNC FOR WINDOWS (**SFIWIN.exe**), prepare a backup file for the PV.

**PV->PC BKP file (Back up file)**

<6> Using BKP CONVERTER (**BKPC.exe**), convert the backup file into the memory file.

**BKP file->MEM file**

<7> Using the memory file with the simulator, debug it on the simulator (**Sim3022.exe**).

<8> Evaluate on the PV.

### 3.2 Precautions for Development

The following lists the precautions for developing the add-in dictionary data.

- (1) When creating the dictionary data, be sure to include discriminative processing recognizable by the PV add-in dictionary in the dictionary data.
- (2) The maximum size of the dictionary data is **1 MB**.  
(However, this value is effective when the user data is not registered. Create the dictionary data, considering the amount of user data.)
- (3) The dictionary data should be created in the format, which takes into account division, by 1 to 30 KB (Optionally selectable in increments of 1 KB) in order to register into the PV.
- (4) When sending the dictionary data, it is necessary to set and confirm the following items:
  - (a) Registered file name (Up to 15 characters; alpha numerals only)
  - (b) Maximum size of the 1 record (1 to 30 KB in increments of 1 KB)
  - (c) Port number (COM1 to COM5)
- (5) The dictionary data cannot be sent in the following cases:
  - (a) When the free memory capacity is less than the dictionary data.
  - (b) When the total number of add-in registered data is **15**.
- (6) Prior to sending the dictionary data, execute the PV hard icons-[MenuBar]-[System]-[Memory management] in that order. (In order to secure the free memory capacity)
- (7) If you register the same dictionary data name as the one registered in the PV, it will be overwritten.
- (8) If communication is interrupted halfway, the dictionary data being registered will be deleted.
- (9) Prior to communication, close all other applications being started.
- (10) When using BKP CONVERTER, the simulator's CPJ file may be rewritten. Back up the file for safety.
- (11) When porting to the simulator, be sure to confirm the model.  
Simulation is not allowed between different models.
- (12) When you do the communication end, shutdown the power supply of PV once.  
When dictionary data is transplanted to the simulator, it is a necessary procedure.

### 3.3 Details of Procedures

**<1> Using SDK, create the add-in dictionary software.**

Create the add-in dictionary, referring to the accessory sample dictionary software.  
(For the details of SDK, see its manual)

**<2> Create the dictionary data corresponding to the add-in dictionary software.**

Create the dictionary data in the format, which can be referred to by the add-in dictionary software.

**<3> Using PV APPLICATION MANAGER, register the exclusive add-in software, DATA RECORDER, in the PV.**

Using the following steps, register in the PV the exclusive add-in tool, DATA RECORDER, designed to register the dictionary data.

**Step 1** Set the PV in the cradle and connect the cradle's cable to the PC's communication port.

**Step 2** Execute PV APPLICATION MANAGER (**PVM.exe**).

(For the details, see the manual for PV APPLICATION MANAGER.)

**Step 3** In the PV menu screen, touch the **[MenuBar]** hard icon.

A menu bar appears at the top of the screen.

Touch **[Option]**-**[Download]**.

Now, the PV will wait for communication.

**Step 4** Select **[File]**-**[Add file]** in the PV APPLICATION MANAGER menu,  
and then, select **datarec.bin**.

**Step 5** Select **[SET]** in the PV APPLICATION dialog.

**Step 6** DATA RECORDER is registered into the PV.

**<4> Using PVDATA SENDER (PC tool) and DATA RECORDER (PV tool), send the dictionary data from the PC to PV for registration.**

Follow the procedure below to send the dictionary data from the PC to PV.

**Step 1** Set the PV in the cradle and connect the cradle's cable to the PC's communication port.

**Step 2** Execute PVDATA SENDER (**PVS\_dev.exe**) on the PC side, select **[FILE]-[OPEN]**, and then select the dictionary data.

**Step 3** Touch and execute DATA RECORDER on the PV side.

The PV will wait for reception.

**Step 4** Select **[MAKE]** under PVDATA SENDER.

The dictionary data is sent from the PC and registered in the PV.

**Step 5** when you do the communication end, shutdown the power supply once.

(When dictionary data is transplanted to the simulator, it is a necessary procedure.)

**<5> Using PC SYNC FOR WINDOWS, prepare a PV backup file.**

Prepare the PV backup file in the following procedure:

**Step 1** Set the PV in the cradle and connect the cradle's cable to the PC's communication port.

**Step 2** Execute PC SYNC FOR WINDOWS (**SFIWIN.exe**) and execute **[Casio]-[Backup]-[Receive]** in that order.

Backup operation starts.

**Step 3** After backup operation; name the backup file (BKP file).

**<6> Using BKP CONVERTER, convert the backup file into the memory file.**

Following the procedure below, use BKP CONVERTER to convert into the memory file which corresponding to the simulator.

**Step 1** Execute BKP CONVERTER (**BKPC.exe**).

**Step 2** Select **[FILE]-[OPEN]**, and then, select a BKP file.

**Step 3** At the BKP file, which cannot be recognized

Select a corresponding model file from **[CONFIG]**.

**Step 4** Select **[MAKE]**.

The BKP file is converted to generate the memory file.

**<7> Use the memory file with the simulator and debug it on the simulator.**

Using the memory file, the simulator for Windows (**Sim3022.exe**). can debug the dictionary data registered in the PV and the add-in dictionary.

**Step 1** With the editor; open the CPJ file specified in **7. HOW TO PORT TO SIMULATOR**.

**Step 2** Correct the spots specified in **7. HOW TO PORT TO SIMULATOR**.  
to save the CPJ file

**<8> Evaluate on the PV.**

After debugging of the memory file on the simulator is completed, debug it on the PV.

## 4. OPERATION (PVDATA SENDER)

This section describes the functions of menu commands and the roles of display items for PVDATA SENDER (PVS\_dev.exe).

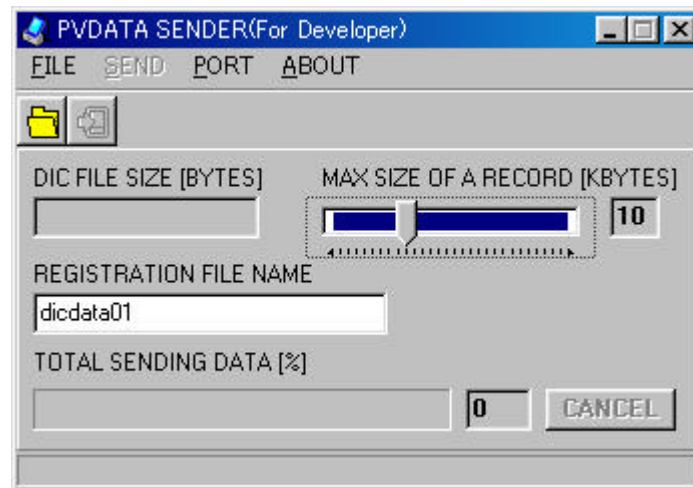


Fig 1. PVDATA SENDER Proper

### 4.1 Precautions for Operation

When operating PVDATA SENDER, observe the following:

#### (1) About FIXED SAVE

- (a) If **[FIXED SAVE (INITILE)]** is executed, you will not be able to alter the dictionary data name and the maximum size of one record.
- (b) If you want to alter the setting halfway development, delete the PVSCFG.INI file in the folder where the PVDATA SENDER executing file exists. This will enable alteration again.
- (c) Even if the PVSCFG.INI file is deleted, it will be started by default setting and new PVSCFG.INI will be generated by executing **SAVE (INIFIILE)**.

#### Default Setting

Dictionary Data Name	DICDATA01
Maximum Bytes	30 KB
Port Used	COM1

- (2) If transmission is cancelled during communication, all the data transmitted so far will be deleted. Redo from the beginning.

## 4.2 Menu Commands

### <1> [OPEN (DICFILE)][In FILE]

Opens the dictionary data (extension:..DIC).

### <2> [SAVE (INIFILE)][In FILE]

Saves the current settings (dictionary data name, maximum size of one record, port number).

### <3> [FIXED SAVE (INIFILE)][In FILE]

Fixes the settings of the dictionary data so that general users cannot alter the specifications of the add-in dictionary (**dictionary data name, maximum size of one record**) to be developed.

### <4> [EXIT][In FILE]

Exits PVDATA SENDER.

### <5> [SEND]

Sends the dictionary data to the PV.

### <6> [PORT]

Capable of setting communication ports. (**COM1-COM5**)

### <7> [ABOUT]

Displays "ABOUT PVDATA SENDER."

## 4.3 Description of Items

### <1> DIC FILE SIZE

Displays the dictionary data size (in bytes).

### <2> REGISTRATION FILE NAME

Input the file name to be registered in the PV.

### <3> TOTAL SENDING DATA

Displays the already sent data (in percentage).

### <4> CANCEL

Capable of canceling transmission.

### <5> MAX SIZE OF A RECORD

Select the maximum size of one record. (**1 to 30 KB in increments of 1 KB**)



## 5. OPERATION (DATA RECORDER)

DATA RECORDER is to receive and register the dictionary data sent from PVDATA SENDER (**datarec.bin**).

The settings required for registration, such as dictionary data name and maximum size of one record, are not to be directly set at DATA RECORDER, but received from PVDATA SENDER.

Therefore, it has no particular basic operation.

**Only operation allowed for DATA RECORDER is to cancel an application at the [ESC] hard icon.**

Processing is cancelled by pressing the **[ESC]** hard icon while executing DATA RECORDER.

**If cancelled (termination by error included), all the data in process will be deleted.**



FIG2 DATA RECORDER Icon (Screen for Simulator)

## 6. OPERATION (BKP CONVERTER)

This section describes the functions of menu commands and the roles of dialog items for BKP CONVERTER (BKPC.exe).

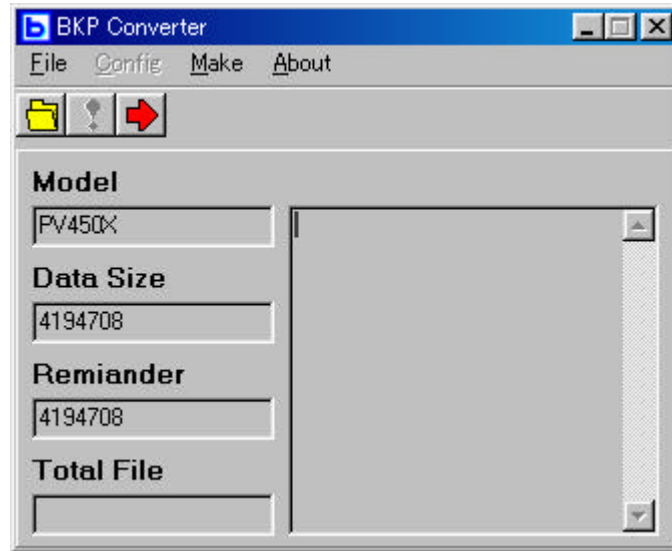


FIG3 BKP CONVERTER Proper

### 6.1 Precautions for Operation

When operating BKP CONVERTER, observe the following:

Do not delete the BKPCFG.INI file in the folder where the BKP CONVERTER executing file is included. (Because the model file has been saved)

### 6.2 Menu Commands

#### <1> [OPEN][In FILE]

Opens the backup file (extension:.BKP).

#### <2> [EXIT][In FILE]

Exits BKP CONVERTER.

#### <3> [CONFIG]

Opens the CONFIG dialog to allow you to check the details of each model file.  
(At the BKP file which cannot be recognized)

#### <4> [MAKE]

Converts the BKP file into the memory file (MEM/DMEM file)

#### <5> [ABOUT]

Displays "ABOUT BKP CONVERTER."

## 6.3 Description of Items

### <1> MODEL

Displays the model file.

### <2> DATASIZE

Displays the number of bytes of the BKP file (in bytes).

### <3> REMAINDER

Displays the number of bytes of the unconverted BKP file.

### <4> TOTAL FILE

Displays the total number of memory files.

### <5> DISPLAY

Displays the converted memory file name and number of bytes.

The converted memory file name automatically assumes **MEMX.BIN** (X is 0, 1, 2, 3, 4, and so on).

If compounded with the simulator's memory file, the memory file name will be **DMEMY.BIN** (Y is the number of the united memory file).

## 6.4 CONFIG Dialog

The CONFIG dialog displays a list of model files.

Of the model files, the one displayed in blue can be selected.

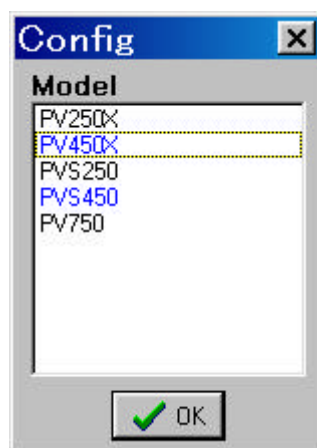


FIG4 CONFIG Dialog

If the model file is double-clicked on the dialog, the **MODEL SETTING dialog** will be opened, which allows you to confirm the settings of that model file.

## 6.5 MODEL SETTING Dialog

The MODEL SETTING dialog is to confirm the settings of each model.

The image shows a 'Parameter Edit' dialog box with a blue title bar and a close button (X) in the top right corner. The dialog is divided into two main sections: 'Basic Parameter' on the left and 'For Dual' on the right. In the 'Basic Parameter' section, there are four text input fields: 'Model' with 'PV450X', 'File Size [bytes]' with '4194708', 'Header Size [bytes]' with '404', and 'Number of division' with '2'. Below these is a 'Mem file information' section containing a text area with 'mem0=2097152' and 'mem1=2097152'. The 'For Dual' section contains three text input fields: 'Dual memory File path' with 'C:\CASIO\PV2EN01\SIM\DUFLA', 'Connected memory file number' with '0', and 'Cut dual memory address [byted]' with '2097152'. At the bottom right of the dialog is an 'OK' button with a green checkmark icon.

Fig.5 MODEL SETTING Dialog

### <1> MODEL

Displays the model file name.

### <2> BKP SIZE

Displays the BKP file size (in bytes) used for this model.

### <3> HEADER SIZE

Displays the header size in the BKP file.

### <4> NUMBER OF DIVISION

Displays the number of divisions of the MEM file.

### <5> MEM FILE INFORMATION

Displays the number of bytes of each memory file (MEM file)(in bytes).

### <6> DUAL MEMORY FILE PATH

Displays the full path of the simulator's memory file used to generate the DMEM file required for a specific model.

### <7> CONNECTED MEMORY FILE NUMBER

Displays the number of MEM file used to generate the DMEM file required for a specific model.

### <8> CUT DUAL MEMORY ADDRESS

Displays the address to be cut from the simulator's memory file to be used.

## 7.HOW TO PORT TO SIMULATOR

This section describes how to use the memory file on the simulator, which has been converted by BKP converter.

**If the CPJ file is altered by mistake, the simulator will be disabled.**

**When altering it, be sure to back it up.**

### 7.1For the PV-250X

<1> Open the C:/CASIO/PV2EN01/SIM/PV250X.CPJ file with the editor.

<2> Rewrite the following portion in the PV250X.CPJ file to save the CPJ file.

#### PV250X.CPJ File

```
[CSGROUP8]
CHIPNAME0=USERFLASH
CHIPFILE0=Sim/STDflash16.bin  <-Alter to Sim/MEM0.bin
CHIPPROGAREA0=0
CHIPOFFSET0=0
CHIPKIND0=2
CHIPCODE0=MBM29LV160B
CHIPBUS0=16
```

## 7.2 For the PV-450X

- <1> Open the C:/CASIO/PV2EN01/SIM/PV450X.CPJ file with the editor.
- <2> Rewrite the following two portions in the PV450X.CPJ file to save the CPJ file.

### PV450X.CPJ File

```
[CSGROUP5]
CHIPNAME0=SYSTEM00
CHIPFILE0=BIOS/BIOSBIND.00
CHIPPROGAREA0=0
CHIPOFFSET0=0
CHIPKIND0=1
CHIPCODE0=
CHIPBUS0=8
CHIPNAME1=SYSTEM01
CHIPFILE1=C/Bin/APLALL.bin
CHIPPROGAREA1=0
CHIPOFFSET1=20000
CHIPKIND1=1
CHIPCODE1=
CHIPBUS1=8
CHIPNAME2=DUALFLASH
CHIPFILE2=Sim/DUFLASH32.BIN <-Alter to Sim/DMEM0.BIN
CHIPPROGAREA2=200000
```

```
[CSGROUP8]
CHIPNAME0=USERFLASH
CHIPFILE0=Sim/Uflash16.bin <-Alter to Sim/MEM1.BIN
CHIPPROGAREA0=0
```

### 7.3 For the PV-S250

- <1> Open the C:/CASIO/PV2EN02/SIM/PVS250.CPJ file with the editor.
- <2> Rewrite the following portion in the PVS250.CPJ file to save the CPJ file.

#### PVS250.CPJ File

```
CHIPNAME2=DUALFLASH  
CHIPFILE2=Sim/DUFLASH32.BIN <-Alter to Sim/DMEM0.BIN  
CHIPPROGAREA2=200000  
CHIPOFFSET2=0  
CHIPKIND2=2  
CHIPCODE2=MBM29DL324B  
CHIPBUS2=16
```

### 7.4 For the PV-S450

- <1> Open the C:/CASIO/PV2EN02/SIM/PVS450.CPJ file with the editor.
- <2> Rewrite the following two portions in the PVS450.CPJ file to save the CPJ file.

#### PVS450.CPJ File

```
CHIPNAME2=DUALFLASH  
CHIPFILE2=Sim/DUFLASH32.BIN <-Alter to Sim/DMEM0.BIN  
CHIPPROGAREA2=200000  
CHIPOFFSET2=0  
CHIPKIND2=2  
CHIPCODE2=MBM29DL324B  
CHIPBUS2=16
```

```
[CSGROUP8]  
CHIPNAME0=USERFLASH  
CHIPFILE0=Sim/STDflash16.bin <-Alter to Sim/MEM1.BIN  
CHIPPROGAREA0=0  
CHIPOFFSET0=0  
CHIPKIND0=2  
CHIPCODE0=MBM29LV160B  
CHIPBUS0=16
```

## 7.5 For the PV-750 (PV-750Plus)

<1> Open the C:/CASIO/PV2EM01/SIM/PV750.CPJ file with the editor.

(For the PV-750Plus, open the C:/CASIO/PV2EM02/SIM/PV750.CPJ file.)

<2> Rewrite the following portion in the PV750.CPJ file to save the CPJ file.

### PV750.CPJ File

CHIPNAME2=DUALFLASH

CHIPFILE2=Sim/DUFLASH32.BIN <- Alter to Sim/DMEM0.BIN

CHIPPROGAREA2=200000

CHIPOFFSET2=0

CHIPKIND2=2

CHIPCODE2=MBM29DL324B

CHIPBUS2=16