

# ZeitControl Balance Reader

(Document Version: 1.02)

## 1. Hardware

Currently available is a balance reader with 10 digits display. First 4 digits can display alphanumeric characters. Last 6 digits can display numeric (including hexadecimal) characters. This device uses DeviceID 1 and DeviceVersion 0. It supports programmable processor cards using ISO7816 compatible T=1 protocol, as for example ZeitControl BasicCard.

## 2. Developer Guide and Reference

### 2.1 Overview

The balance reader uses cards by sending an command (APDU) to the card as soon as the card is inserted into the reader. This command will include information about capabilities of the balance reader. In return the card must respond to this command with information about data to display. If requested in previous respond the balance reader will repeat the command, so the card can respond with further data to display. To give the card the choice in showing different data on each command, each command includes are record number which starts with 0 and is increment for each additional command call by the balance reader.

### 2.2 Usage with ZeitControl BasicCard

With BasicCard each supported APDU call is coded as a *Command* inside card side Basic program. Current version of BasicCard development software contains an include file (\basiccrd\inc\preader.def) which contains prototypes for this command. For some reason the prototypes are slightly different for Compact BasicCard and Enhanced BasicCard.

#### 2.2.1 PRDisplay command

The BasicCard command looks as follows:

```
Declare Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, DigitCount as Byte, _
    DecimalPoint as Byte, Delay as Byte, MoreData as Byte, _
    Data as String)
```

(for Enhanced BasicCard) or

```
Declare Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, DigitCount as Byte, _
    DecimalPoint as Byte, Delay as Byte, MoreData as Byte, _
    Data as String<=11)
```

(for Compact BasicCard).

This declaration is included inside PReader.def include file.

## 2.2.2 Input Data for BasicCard PRDisplay Command

When the PRDisplay command is called by the balance reader, several information are passed to the card. This information can be used to get knowledge about balance reader capabilities. You may ignore most of them if you expect ZeitControl balance reader to be used. At least you must check for RecordNumber.

**Note:** For technical reasons the same parameters names must be used for input and output data. So the parameters are named to match the purpose of the output data. In some cases the use for the input data is completely different.

Command Parameter	Used As	Value	Description
RecordNumber	RecordNumber	0 to 254	Number of requested record to display. Starting with 0.
DataFormat	DataFormat	see Description	General supported data format. This describes the format supported by all display characters. So this formats could be expected to be available without limitations. More enhanced formats could be available as well, but with limitations, which means it may be supported by some of the display characters only (e.g. only first 4 characters of ZeitControl Balance Reader supports alpha numeric characters). You can check FormatDetails for locating enhanced features of each display character. DataFormat is described as binary or of all supported format types. Please see below: PRAAlpha = 1 (alphanumeric characters) PRHex = 2 (numeric hexadecimal characters) PRNum = 4 (numeric decimal characters) PRNumSign = 8 (numeric decimal characters including sign) PRCur = 16 (major currency symbols)
DigitCount	DigitCount	1 to 254	General number of digits or characters supported (excluding possible sign character and decimal point). So when not using sign one additional character may be available.
DecimalPoint	Reserved	0	Reserved for future use
Delay	DeviceID	See device table (Appendix)	Identifier to identify the reader device.
MoreData	DeviceVersion	0 to 254	Internal version and revision number of device firmware and hardware.
Data	FormatDetails	ASCII character string	This optional string include details about display capabilities for each character in display. Each display character is marked by one of the following characters: S = Sign ('+' or '-') s = Sign (only '-') A = Alphanumeric (including '+' and '-') a = Alphanumeric (including '-' but not '+') C = Alphanumeric (including '+' and '-') + major Currency symbols c = Alphanumeric (including '-' but not '+') + major Currency symbols H = Numeric hexadecimal (including '+' and '-') h = Numeric hexadecimal (including '-' but not '+') N = Numeric (including '-' and '+') n = Numeric (including '-' but not '+')

### 2.2.3 Output Data for BasicCard PRDisplay Command

To display something inside balance reader display, the PRDisplay command must return the display data inside parameters of PRDisplay command. Please see below:

Command Parameter	Value	Description
RecordNumber	0 to 254	Record number returned. Must be same as send to card. So do not change this parameter.
DataFormat	Either: PRAppha = 1 PRHex = 2 PRNum = 4 PRNumSign = 8	See description of data for more information.
DigitCount	0 or 1 to 254	Number of digits or characters to show. If 0 show all given characters for alphanumeric format or all digits suppressing trailing zeros for numeric format.
DecimalPoint	0 or 1 to 254	For use with numeric data formats. 0=No decimal point, 1-254=Decimal point at given position. E.g. 3 for 2 numbers following the comma.
Delay	0 or 1 to 254	Time to show data in steps of 0.2s or 0 to show till card is removed.
MoreData	0 or 1	0 if this is last record to show. 1 if more records are available.
Data	See description.	If DataFormat is PRAppha you can assign the string to be displayed. For numeric formats you have to assign the numerical data. See example for more information.

### 2.2.4 Remarks and Examples

A complete example for using balance reader with BasicCard is included in current version of BasicCard development software (c:\basiccrd\examples\pocket). Some piece of this is following here:

#### 2.2.4.1 General Notes

You always should include PReader.def as included with this BasicCard development software.

E.g.:

```
#include PReader.def
```

#### 2.2.4.2 Transforming numeric data to be returned as String

PRDisplay command uses String type to return display data to the reader. This is used because string can have any size and we can pass different types of data using string type. To display numeric data you must convert your data into string type before assigning it to Data parameter of command. This could be done by use of *at* statement and two variables of different types.

First you need a variable of type Long. E.g.:

```
Dim BalanceData as Long
```

Data of type Long requires 4 byte of storage. So when converting to string we use String\*4 which also requires 4 byte storage. E.g.:

```
Dim BalanceDataStr as String*4 at BalanceData
```

The statement "at BalanceData" causes no extra memory to be allocated for BalanceDataStr. Instead it uses the same memory as BalanceData variable. So when using BalanceDataStr you use the binary representation of BalanceData as content of your String BalanceDataStr. This does not cause any conversion of the data.

So by assigning your numeric value to the Long variable, E.g.:

```
BalanceData = 1000
```

You access the same data as String\*4 from BalanceDataStr. So just assign the String\*4 type data to the Data parameter of PRDisplay command. E.g.:

```
Data = BalanceDataStr
```

### 2.2.4.3 Pocket Reader Extension for Debit Example

```

Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, _
                               DigitCount as Byte, _
                               DecimalPoint as Byte, Delay as Byte, _
                               MoreData as Byte, _
                               Data as String)

Dim BalanceData as Long
REM convert long type numeric data to string data type
Dim BalanceDataStr as String*4 at BalanceData
select case RecordNumber
  case 0
    DataFormat=PRNumSign ' Number with sign
    DigitCount=0         ' show all digits
    DecimalPoint=3       ' decimal point at 3 character
                        ' (2 digits follow point)
    Delay=0              ' show till card is removed
    MoreData=PRNoMoreData ' no more data to show
    BalanceData=Balance  ' convert long to string
    Data=BalanceDataStr  ' and send to balance reader
  case else
    DataFormat=PRAAlpha
    DigitCount=0
    DecimalPoint=0
    Delay=1000 / PRDelayUnits ' 1 second to show
    MoreData=PRNoMoreData
    Data="ERR" ' Error message
end select
End Command

```

### 2.2.4.4 Displaying String Data

```

Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, _
                               DigitCount as Byte, _
                               DecimalPoint as Byte, Delay as Byte, _
                               MoreData as Byte, _
                               Data as String)

select case RecordNumber
  case 0
    DataFormat=PRAAlpha ' alphanumeric data
    DigitCount=0        ' show all characters
    DecimalPoint=0      ' decimal point is ignored
    Delay=1000 / PRDelayUnits ' show for 2 seconds
    MoreData=PRNoMoreData ' no more data to show
    Data="DM 5.00"       ' and send to balance reader
  case else
    DataFormat=PRAAlpha
    DigitCount=0
    DecimalPoint=0
    Delay=1000 / PRDelayUnits ' 1 second to show
    MoreData=PRNoMoreData
    Data="ERR" ' Error message
end select
End Command

```

**2.2.4.5 Display Numeric Data**

EEPROM Value as Long

.....

```

Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, _
                             DigitCount as Byte, _
                             DecimalPoint as Byte, Delay as Byte, _
                             MoreData as Byte, _
                             Data as String)

Dim ValueData as Long
REM convert long type numeric data to string data type
Dim ValueDataStr as String*4 at ValueData
select case RecordNumber
  case 0
    DataFormat=PRNum      ' numeric data
    DigitCount=0         ' show all characters
    DecimalPoint=0       ' no decimal point
    Delay=2600 / PRDelayUnits ' show for 2.6 seconds
    MoreData=PRNoMoreData ' no more data to show
    ValueData=Value      ' now we can access as string
    Data=ValueDataStr    ' in ValueDataString
  case else
    DataFormat=PRAlpha
    DigitCount=0
    DecimalPoint=0
    Delay=1000 / PRDelayUnits ' 1 second to show
    MoreData=PRNoMoreData
    Data="ERR" ' Error message
end select
End Command

```

**2.2.4.6 Display Several Values**

EEPROM Value as Long

Const MaxValue=400\*100

.....

```
Command &HC8 &H00 PRDisplay(RecordNumber as Byte, DataFormat as Byte, _
                             DigitCount as Byte, _
                             DecimalPoint as Byte, Delay as Byte, _
                             MoreData as Byte, _
                             Data as String)
```

Dim ValueData as Long

REM convert long type numeric data to string data type

Dim ValueDataStr as String\*4 at ValueData

select case RecordNumber

case 0

```
DataFormat=PRAlpha      ' numeric data
DigitCount=0             ' show all characters
DecimalPoint=0          ' ignored
Delay=1600 / PRDelayUnits ' show for 1.6 seconds
MoreData=PRMoreData     ' more data available
Data="MAX 400.00"
```

case 1

```
DataFormat=PRNum        ' numeric data
DigitCount=0            ' show all characters
DecimalPoint=3          ' 2 character follow comma
Delay=5000 / PRDelayUnits ' show for 5 seconds
MoreData=PRNoMoreData  ' no more data available
' Convert data
ValueData = Value
Data=ValueDataStr
```

case else

```
DataFormat=PRAlpha
DigitCount=0
DecimalPoint=0
Delay=1000 / PRDelayUnits ' 1 second to show
MoreData=PRNoMoreData
Data="ERR" ' Error message
```

end select

End Command

## 2.3 Usage with other processor cards

### 2.3.1 APDU send by balance reader

Field	Size in bytes	Value	Description
CLA	1	0xC8	ISO CLA byte
INS	1	0x00	ISO INS byte
P1	1	0	ISO P1 byte – not used
P2	1	0	ISO P2 byte – not used
LC	1	6+m	ISO LC byte – length of following data (excluding LE)
RecordNumber	1	0 to 254	Number of requested record to display. Starting with 0.
DataFormat	1	binary or of 1, 2, 4, 8, 16	Supported DataFormats: binary or of 1=Alphanumeric, 2=Numeric hexadecimal, 4=Numeric decimal, 8=Numeric decimal with sign, 16=Major Currency Symbols
DigitCount	1	1 to 254	Number of digits or characters supported (excluding possible sign character and decimal point).
Reserved	1	0	Reserved for future use
Device ID	1	See device table (Appendix)	Identifier to identify the reader device.
DeviceVersion	1	0 to 254	Internal version and revision number of device firmware and hardware.
FormatDetails	m=0 or DigiCount (but max 230)	ASCII character string	This optional string include details about display capabilities for each character in display. Each display character is marked by one of the following characters: S = Sign ('+' or '-') s = Sign (only '-') A = Alphanumeric (including '+' and '-') a = Alphanumeric (including '-' but not '+') C = Alphanumeric (including '+' and '-') + major Currency symbols c = Alphanumeric (including '-' but not '+') + major Currency symbols H = Numeric hexadecimal (including '+' and '-') h = Numeric hexadecimal (including '-' but not '+') N = Numeric (including '-' and '+') n = Numeric (including '-' but not '+')
LE	1	0	ISO LE byte

## 2.3.2 Respond to be returned by card

Field	Size in bytes	Value	Description
RecordNumber	1	0 to 254	Record number returned. Must be same as send to card. If this value is not the same as send from balance reader balance reader will fail with error message in display.
DataFormat:	1	1, 2, 4, 8	Data format as included in data area of respond. 1=Alphanumeric, 2=Numeric hexadecimal, 4=Numeric decimal, 8=Numeric decimal with sign
DigitCount	1	0 or 1 to 254	Number of digits or characters to show. If 0 show all given characters for alphanumeric format or all digits suppressing trailing zeros for numeric format.
DecimalPoint	1	0 or 1 to 254	0=No decimal point, 1-254=Decimal point at given position. E.g. if 3 there are 2 digits right from decimal point. This value is ignored for Alphanumeric data.
Delay	1	0 or 1 to 254	Time to show following data in steps of 0.2s or 0 to show till card is removed.
MoreData	1	0 or 1	0 if this is last record to show. 1 if more records are available.
Data	n if DataFormat is alphanumeric. Multiple of 4 if DataFormat is numeric. For devices with 8 characters or less this must be 4 if numeric DataFormat. For devices with more than 8 up to 16 characters this could be either 4 or 8 bytes. ....	As required.	Data to display. Either ASCII string if DataFormat is Alphanumeric (1) or binary/numeric data to be displayed as number. Binary data must be packed MSB first. Signed values must be coded in two's-complement form.
SW1SW2	2	9000h or 61xxh	Status word. Both values are allowed.

### 3. Appendix

#### 3.1 Using Balance Reader with Compact BasicCard

To avoid EEPROM heap is being used, which may corrupt the Compact BasicCard EEPROM heap if card is pulled at this time, you should verify that at least about 30 bytes RAM heap is available. You can check size of RAM heap by using “-OM” option of ZeitControl Basic Compiler and checking length of RAMHEAP in created map file. If not enough RAMHEAP is left you can try to reduce stack by using stack statement like this.

```
#ifdef CompactBasicCard
#stack 30
#endif
```

You should verify now that enough stack is left for you card program and that enough RAMHEAP is available.

#### 3.2 Device IDs

Currently assigned device IDs and supported formats:

Device Name	FormatDetails	Device ID
ZeitControl Balance Reader	CCCChhhhh (4 alphanumeric characters supporting major currency symbols + 6 numeric characters)	1

#### 3.3 Alphanumeric Characters

Alphanumeric characters based on ASCII character set. Following characters are supported:

Character	Hex Value
'0'	30
'1'	31
'2'	32
'3'	33
'4'	34
'5'	35
'6'	36
'7'	37
'8'	38
'9'	39
'A'	41
'B'	42
'C'	43
'D'	44
'E'	45
'F'	46
'G'	47
'H'	48
'I'	49
'J'	4A
'K'	4B
'L'	4C
'M'	4D
'N'	4E
'O'	4F
'P'	50
'Q'	51
'R'	52
'S'	53
'T'	54
'U'	55
'V'	56

ZeitControl Balance Reader

'W'	57
'X'	58
'Y'	59
'Z'	5A
'a'	61
'b'	62
'c'	63
'd'	64
'e'	65
'f'	66
'g'	67
'h'	68
'i'	69
'j'	6A
'k'	6B
'l'	6C
'm'	6D
'n'	6E
'o'	6F
'p'	70
'q'	71
'r'	72
's'	73
't'	74
'u'	75
'v'	76
'w'	77
'x'	78
'y'	79
'z'	7A
' ' (space)	20
'.'	2E
'+'	2B (optional see format details, replaced by ' ' if not supported)
'_'	2D
'/'	2F (optional, could be replaced by ':', '÷' or ' ' if not available)
'*'	2A (optional, could be replaced by multiplication point '·' or ' ' if not available)

If either capital (A-Z) or small (a-z) letters could not be displayed they are converted to related letter of other type (e.g. 'A' to 'a' or 'h' to 'H').

If '.' is following a numeric (0-9) or alphabetic (a-z, A-Z) character the decimal point ('.') is added to this preceding character instead.

Characters not listed here should not be used by developers.

### 3.4 Major Currency Symbols

Additionally to alphanumeric characters as described above balance reader with ability to show major currency symbols implement some (not necessarily all) of the currency symbols as listed below:

Character	Name	Hex Value
‘€’	Euro	A4 (ISO8859), 80 (CP1252) or D5 (CP858) all values are translated to Euro symbol
\$	Dollar	24
£	Pound	9C
¥	Yen	9D
₣	French Franc	B0 (no standard)
₵	Lira	B1 (no standard)

It is possible only some of this characters are supported. In this case not supported currency symbols are replaced by ‘ ’ (space). Developers should check if desired currency is supported by verify DeviceID and DeviceVersion.

### 3.5 DataFormat and DigitCount

DataFormat and DigitCount gives minimal display capability available by each character.

### 3.6 DecimalPoint

DecimalPoint is only valid for numeric formats. It is ignored for alphanumeric data to display.

### 3.7 FormatDetails

Format details give more details about pocket reader display capabilities. This string is optional and may be omitted, if display capabilities are described completely by DataFormat and DigitCount.