

CC-USB



Programming Guide

INTERNAL LOGICAL DEVICES of CC-USB

The CC-USB presents to the user five internal devices or addresses shown in Table 1:

Table 1. Internal devices of CC-USB and their addresses

Address	Device
1	Register Block (RB)
2	Camac Data Readout Stack (CDS)
3	Camac Scaler Readout Stack (CSS)
4	Camac NAF Generator (CNAF)
5	Common Output Buffer

The Register Block is composed of a number of registers identified by subaddresses as shown in Table 2:

Table 2. Register subaddresses and their functionality

Sub-address	Register	Note
0	Firmware ID	Read-only
1	Global Mode	Read/Write
2	Delays	Read/Write
5	Scaler Readout Frequency	Read/Write
6	User LED Source Selector	Read/Write
7	User NIM Output Source Selector	Read/Write
8	LAM Mask	Read/Write – 24-bits in two words
10	Action	Read/Write
12	Camac LAM	24-bits Read-Only
13	Serial Number	11 bits, Read-Only

Firmware ID Register

This Firmware ID register identifies the acting FPGA firmware in four hexadecimal digits MYFR, where M and Y represent the month and year of creation, and F and R represent the firmware and revision numbers, respectively.

Global Mode Register

The global mode register has the following 16-bit structure:

13-15	12	9-11	8	6	4,5	0 - 3
Unused	Arbitr.	WdgFreq	HeaderOpt	EvtSepOpt	Unused	BuffOpt

The BuffOpt bits (0-2) define the output buffer length. Bit 3 controls the mode of buffer filling, such that 0 closes buffers at event boundaries and 1 allows spreading events across the adjacent buffers:

BuffOpt Value	Buffer Length (words)
0	4096
1	2048
2	1024
3	512
4	256
5	128
6	64
7	Single Event

The EvtSepOpt set the number of event terminator word (hexadecimal FFFF), such that EvtSepOpt=0/1 cause one/two terminator word/s written at the end of each event.

The HeaderOpt bit controls the structure of the buffer header, such that HeaderOpt=0 writes out one header word identifying the buffer type (bit 15=1 – watchdog buffer, bit 14=0 – data buffer, bit 14=1 – scaler buffer) and the number of events in buffer. When HeaderOpt = 1, the second header word is written out listing the number of words in the buffer.

The WdgFreq bits define the frequency at which the watchdog is forcing writing of output buffer during data acquisition. The three bit number represents the time interval in seconds, counting from the end of an event, after which the watchdog triggers when no new event has been observed.

The Arbitr Bit, when set to 1 activates Camac bus arbitration.

Delays Register

The delays register stores the desired trigger delay (from the start signal applied to the NIM input to the actual start of the Camac readout) – least significant 8 bits and the LAM timeout period – most significant 8 bits. Both delays are in units of us.

Scaler Readout Frequency Register

The Scaler Readout Frequency Register stores the number defining the frequency at which scalers are to be read out (scaler stack is executed) during the data acquisition. The stored value is equal to the number of data events separating the scaler readout events. When the value is zero, scaler readout is suppressed.

USER LED and NIM Output Selectors

Numbers stored in these registers identify sources of User LEDs and NIM Outputs.

LAM Mask Register

The LAM Mask Register stores the LAM Mask defining what combination of LAMs triggers event readout during the data acquisition. When zero, the readout is triggered by a signal applied to the NIM input.

Action Register

Bit 0 of the Action Register activates data acquisition in list mode, when event readout is triggered either by a start signal applied to the User NIM input I1 or a combination of LAMs coinciding with the LAM mask.

Serial Number Register

The Serial Number Register is a Read-Only register containing the serial number of the CC-USB.

COMMUNICATING WITH CC-USB

Communication with the CC-USB consists in writing and reading of buffers of data to/from the USB2 port of the CC-USB using bulk-transfer mode. Borrowing from the USB language, the buffers to be written to the CC-USB will be called Out Packets, and they are sent to pipe 0 of the USB port. The buffers to be read will be called In Packets, and they are read from pipe 2 of the USB port. The USB controller IC, when connected to a USB2 port configures packet lengths to 512 bytes. For USB1 (full speed), the packet length is set to 64 bytes. The Out Packets must be properly formatted to be understood by the internal devices of CC-USB and, by the same token, the format of the In Packets retrieved from the CC-USB must be understood by the user in order to be useful.

User may send Out Packets to four devices – the Register Block (RB), Camac Readout Stacks (CDS and CSS), and the NAF Generator (RB, CDS, CCS, CNAF). User may read In Packets only from the Common Output Buffer. Reading back data from the RB, CDS, and CSS is achieved by, first sending a data request Out Packet to these devices and then by reading the In Packet containing the requested data from the Common Output Buffer. Writing to the Camac NAF Generator constitutes implicitly a request for data, such that in response to such a writing, CC-USB performs the requested Camac operation and returns the Camac data in the Common Output Buffer. Both, In and Out Packets are of a variable length, depending on which internal address is involved and what the content of the message is.

Important Note:

Read operations from the USB port are blocking operations such that the host program will stop executing until the data are available at the port. Therefore, the host program must make sure (by first requesting data) that CC-USB has placed data in the Common Output Buffer (physically this is the FIFO of the USB controller IC), before the read command is issued. CC-USB provides a mechanism for supplying data, even when the host program is “frozen” in a state of waiting for data. The mechanism consists in starting a second copy of the program and issuing a bare request for data command from this second copy, not followed by the read IN Packet command.

Also, it is important to specify a sufficiently long In Packet size to be at least of the size of the actual data buffer available at the Common Output Buffer. This is especially important in the case of reading Camac data buffers which differ in size substantially depending on the structure of the Camac Readout Stack.

General structure of Out Packets

Since internally, the USB controller of the CC-USB is set up as a 16-bit wide FIFO (First- In-First-Out Memory), the In and Out Packets are organized as collections of 16-bit words. For the purpose of the software, and more specifically, of the Windows Application Programming Interface (API) routines, the data are packed in byte-wide buffers, a process that may remain transparent to the user when proper set of routines (DLLs) is used. Also, much of the technical information on writing and reading back data from the internal devices of the CC-USB may be considered redundant, when a set of routines is available to perform the task. This information is, however, necessary for writing such routines.

First (16-bit) word in an Out Packet identifies the internal device/address for which the packet is intended and whether the packet represents a request for data or represents the data to be stored/interpreted to/by the target device. The latter information is coded in bit 3 (value=4) of the header word, with bit 3 set for requests for data. The meaning of the second word in the Out Packet depends on the address and represents the sub-address in the case of the Register Block and the number of words to follow, in the case of the Camac Stacks (CDS and CSS) and the Camac NAF Generator (CNAF). The subsequent words in the buffer, if any, represent the data to be stored in the target device or the data to be interpreted and acted upon by the target device (in the case of the CNAF). A detailed description of Out Packets for the four target devices is given below.

Writing to the Register Block

The Out Packet for writing data to various registers of the Register block is composed of the following words:

1. Target Address = 1 the target address identifying the register block

The Camac stack consists of a sequence of properly encoded simple (one line for Camac “Read” commands and 3 lines for Camac “Write” commands) or complex (multi-line) Camac commands.

Simple commands specify only the desired N, A, and F to be issued by CC-USB and, additionally, whether the data is 24-bits (Long Mode) or 16-bits long. For the Camac “Write” commands, additional two lines specify the data to be written.

The data word for a simple command has the following structure:

15	14	9-13	5-8	0-4
0	Long Mode	N	A	F

This data word can thus be calculated as

$$\text{SimpleCommand} = F + 32 * A + 512 * N + 16384 * \text{LongMode}$$

Note, that the Camac “Read” commands have both bits 3 and 4 (the most significant bits) of F set, while the “Write” commands have only bit 4 set. Control commands have bit 3 set.

Complex commands are possible only for “Read” operations. The first word of a complex command is similar to a simple command, except that it has bit 15 (continuation bit) set, i.e.

15	14	9-13	5-8	0-4
1	Long Mode	N	A	F

The second word is a modifier word, detailing the mode of readout to be performed or the nature of the data to be read. Depending on which bits in the second word are set, a number of additional words, if any, will follow. The continuation bit (bit 15) of the second word is set whenever additional data are to follow. The structure of the modifier word is as follows:

14-15	12-13	11	9	8	7	6	5	4	3	2	1	0
U	NT	U	AP	FC	LM	RM	AS	QS	HM	ND	S2	HD

Where the individual bits have the following meaning:

HD Hit Data - identifies the data as a 16-bit hit register data (coincidence register data), to be used for the conditional readout of subsequent Camac modules. The hit register must be the first module to be read out, i.e., the HD bit may be set only in the first word of the Camac readout stack (CDS).

- S2 When set, S2 strobe is suppressed, the Camac cycle ending at the end of S1
- ND Numbers Data - identifies the data as representing the number of times the next command in stack has to be performed.
- HM Hit Mode - instructs the NAF Generator to condition the readout with the content of the hit pattern read in the first command of the stack (first command in an event). The Number of Product Terms used to condition the readout must be specified as well.
- QS Q-stop mode – the command is to be repeated as long as Q=1 (Q response from the addressed Camac module), but not more than the number specified in the following stack line..
- AS Address Scan – the command is to be repeated a number of times specified in the following word of the stack, with A incremented by 1 each time.
- RM Repeat Mode – repeat command a number of times specified in the following stack line.
- LM LAM Mode – wait for LAM, subject to LAM Timeout and perform the readout only when LAM is set.
- FC Fast Camac Mode – perform the readout in Fast Camac mode a number of times specified in the following stack line.
- AP Address Pattern Data – identifies the data as an address pattern to be used in conjunction with the command that follows. The subsequent command will be repeated for every address for which the bit is set in the address pattern data word.
- NT Number of Product Terms – specifies the number of words in the stack that follow and that constitute bit masks for constructing a logical equation used in deciding whether the given operation is to be performed for the particular hit register data.

The following rules apply:

(i) Whenever the Repeat Mode (RM), Address Scan (AS), Q-Stop, or FC bit is set, the stack line must be followed by another line defining the maximum number (11-bit number) of times the command is to be repeated.

(ii) When the Hit Mode (HM) bit is set, the Number of Terms bits must be declared. The stack line must be followed by the specified number of data lines representing bit masks BMask(1 to NT), to be used in constructing the logical condition for performing the command. The logical equation is:

$(BMask(1) \text{ AND } HD = BMask(1)) \text{ OR } (BMask(2) \text{ AND } HD = BMask(2)) \text{ OR } (BMask(3) \text{ AND } HD = BMask(3)) \text{ OR } (BMask(4) \text{ AND } HD = BMask(4)),$

i.e., the command will be performed whenever all bits in any of the specified Bit Maks are set in the hit register data.

Since the stack can be quite complex, it is advisable to write a proper routine to set up the stack. As an option, one may utilize the CCUSBWin Windows application by JTEC Instrument to format the stack and save it to disk.

THE STRUCTURE OF THE IN PACKETS

The General Output Buffer is associated with Endpoint 6 of the USB2 controller IC, which is configured as a 512 byte deep FIFO. This endpoint is configured for bulk transfer and one can specify lengths of buffers to be read of any length (up to 8192 bytes) compatible with the CC-USB functionality. All data supplied by the CC-USB is to be read from the Endpoint 6. While reading, it is important to specify the length of the buffer not shorter than the length of the actual data buffer written by the CC-USB into this endpoint.

The structure of data retrieved in conjunction with direct requests for data addressed to the Register Block and to the Camac Stacks is simple, such that the buffer consists only of the requested data.

The data buffers read during the data acquisition process have a structure depending on the mode of buffering, i.e., whether event data are allowed to span two buffers (bit 3 of BoffOpt set). Additionally, there are special rules for treating long events. These are discussed at the end of this section.

For the Integer Event Mode, the data buffer has the following structure:

- | | |
|---|--|
| 1. Header word | Bit 15 set indicates a watchdog buffer, bit 14 set indicates a scaler buffer. Bits 0 – 9 represent the number of events in the buffer. |
| 2. Optional 2 nd Header Word | Bits 0-11 represent the number of words in the buffer. |
| 3. Event Length | Event length including terminator words. |
| 4-N1. Event Data | |
| N2. Event Terminator | hexadecimal FFFF |
| N3. Optional 2 nd Terminator | hexadecimal FFFF |
| . | |
| . Subsequent Events | |
| . | |
| N5. Buffer Terminator | hex FFFF |

The unpacking of the events must be done in accordance with the Camac Stack that is involved in generating the buffer.

In the Split-Event mode, when events span two or more buffers, no buffer terminator is written.

For the direct access of the Camac NAFGEN (Interactive Camac operations), no header words are written and the In Packet contains only one event.

CC-USB has dedicated 2kWords-long event FIFO to assemble events. To handle longer events, CC-USB splits the long event into parts, each of which appears as a separate event in the output buffer. The partial events are distinguishable by bit 12 of the Event Length word set, except for the last part. Also, only the last “installment” is terminated by the Event Terminator word (s).

CC-USB has a provision to automatically change the output buffer packing mode to Split-Event mode, whenever the Event Length exceeds the length of the Integer-Event buffer. The fact of such a change is indicated by setting of bit 13 in the buffer header word.