Prof.Dr.H.Ziegler Paderborn, Aug. 4,1997

# FCB and FCV-Bits and addressing

## A) General description

### I) Standard Reference

The FCB and FCV Bits are defined in IEC870-5-2 as part of the link layer in the master to slave telegrams. It is used also by the EN1434 part 3 (Formerly TC176 part 4). The FCB (Frame Count-Bit) in the REQ_UD2 can be considered as the LSB of a telegram counter of transmitted telegrams in the slave to master direction. On the other hand, the FCB in the SND_UD can be considered as the LSB of a (separate) telegram counter for the transmitted telegrams in the master to slave direction. A set FCV (Frame Count Valid)-Bit signals whether this frame count mechanism is active.

## B) Applications of the FCB-mechanism

### I) Slave to master

#### 1.) Multi-telegram answers (RSP_UD) from slave to master

If a total answer sequence from a slave will not fit into a single RSP_UD (respond user data) telegram from the slave to the master, the master signals by a toggled FCB-Bit together with a set FCV-Bit in the next REQ_UD (Request user data) telegram that its link layer has properly received the last RSP_UD-telegram from the slave. The slave answers to a REQ_UD-request with toggled FCB-Bit with a set FCV-bit from the master with a RSP_UD containing the next link layer telegram section of a multi-telegram answer, otherwise it will repeat the last telegram. Note that a slave with a single RSP_UD-telegram may simply ignore the FCB in the REQ_UD2-telegram and send always the same (single) telegram. Note also that a slave with exactly two (sequential) RSP_UD-answer telegrams may simply use the FCB of the REQ_UD2 to decide which of both telegrams should be transmitted. Thus a slave with one or two (sequential) RSP_UD answer-telegrams does not require an internal "Last- REQ_UD2-FCB"-image bit. A slave with three or more (sequential) RSP_UD answer telegrams requires such an internal memory bit. Note that such an internal

memory bit for the RSP_UD-direction must be independent of an possible additional internal memory bit for the SND_UD direction (see master to slave section).

2.) Frozen answer telegrams from slave to master

In same instances a slave will freeze the data of its last RSP_UD answer telegram into an additional temporary storage and will repeat these previously frozen RSP_UD answer, if the FCB has not been toggled. After the reception of a toggled FCB-Bit with a set FCV-Bit or after the reception of a REQ_UD2 with the FCV-Bit cleared, the slave will generate a next answer telegram reflecting the current state of all its data instead of repeating the data values frozen at the first REQ_UD2 attempt with toggled FCB. In meter applications this frozen-telegram aproach will result in possibly very old meter data if the last REQ_UD2 with toggled FCB-bit occured a long time ago. Thus for meter readout this frozen telegram technique is not recommended.

## II) Master to slave

1.) Multi-telegram data (SND_UD) from master to slave

If the master sends a large (sequentual) data block to a slave (e.g. RAM/EEPROM-initialize, code upload) which must be divided into multiple telegrams a similar situation might occur. If the slave receives a telegram correctly and answers with a positive acknowledge (usually by a $E5 single byte answer) but the master does not receive this positive answer correctly, the master will repeat the last telegram with the identical FCB-Bit as in the first attempt. From this the slave can recognize that this next telegram does not contain the next data block but repeats the last data block which has been received correctly. So the slave may either ignore this telegram repetition completely or may accept it thus overwriting the last telegrams data with the second identical data. In both cases an internal telegram sequence counter is not incremented. Note that a slave which will accept only single telegram master to slave communication may simply ignore the FCB in the SND_UD. Note also that a master which can accept exactly two (sequential) SND_UD-telegrams may simply use the FCB of the SND_UD to decide which of both telegrams has been sent. Thus a slave which can accept one or two (sequential) SND_UD answer-telegrams does not require an internal "Last-SND_UD-FCB"-image bit. A slave which can accept three or more (sequential) SND_UD telegrams requires such an

internal memory bit. Note that such an internal memory bit for the SND_UD-direction must be independent of an possible additional internal memory bit for the RSP_UD direction

2.) Incremental actions in slave initiated by master

If single telegram SND_UD will initiate some incremental action in a slave (like toggling a relais or counting something) in contrast to sending some "absolute" data or parameters the FCB-mechanism allows as in the multi-telegram SND_UD situation a distinction between a repetition of the last telegram due to missed acklowledge reception and the next action. In this case the action is only taken if the FCB of the current SND_UD-telegram is different from the FCB in the previous SND_UD-telegram.

# C) Implementation aspects for primary addressing

## I) Implementation for primary addresses 0-250

1.) Master

The master must always contain a "Next REQ_UD2-FCB-image bit" and also a "Next SND_UD-FCB image bit" for each primary slave address used by its application layer. After sending a SND_NKE-request to a slave adress both these "Next FCB-image bit" associated with this address contained in the request must be set. Thus for each primary address the first REQ_UD2 or SND_UD telegram after a SND_NKE contains a set FCB-Bit. Note that after a memory loss (usually due to a power failure) of these "Next FCB-image bits" the master is required to send a SND_NKE to all affected addresses. All subsequent RSP_UD2-telegrams must contain the "Next REQ_UD2- FCB-image bit" of the appropriate primary address as a FCB. This "Next REQ_UD2 FCB-image bit" is toggeled after an error free link layer RSP_UD telegram has been received.
All subsequent SND_UD-telegrams must contain the "Next SND_UD- FCB-image bit of the appropriate primary address as a FCB. If a SND_UD has been successfully transmitted to a slave (reception of a valid acknowledge byte $E5 or a valid RSP_ACK telegram) this "Next SND_UD-FCB-image bit" associated with this address is toggled.

2.) Slave

If a slave wants to utilize the FCB-Bit mechanism for the REQ_UD2-type (slave to master) transfers for more than two sequential telegrams it must provide a "Last REQ_UD2-FCB"-memory bit. If a valid REQ_UD2 telegram with a set FCV-Bit is received its FCB-Bit is compared to this "Last REQ_UD2-FCB-Bit". If they differ or the FCV-bit is clear, the next actual telegram data are used for the RSP_UD answer otherwise the last (stored) telegram is repeated.
If a slave wants to utilize the FCB-Bit mechanism for the SND_UD-type (master to slave) transfers for more than two sequential telegrams it must provide a "Last SND_UD-FCB"-memory bit. If a valid SND_UD telegram with a set FCV-Bit is received, its FCB-Bit is compared to this "Last SND_UD-FCB-memory Bit". If they differ or the FCV-bit is clear, the next actual telegram data are used for the RSP_UD answer otherwise the last (stored) telegram is repeated.
Note that after a valid reception of a SND_NKE to the primary address of the device or to the test adress 254 ($FE) or the broadcast adress 255 ($255) these internal "Last FCB memory bits must be cleared.

## II) Implementation for multiple address slaves

A slave might be configured to respond to more than one primary address. This could be useful for slaves which internally consist of more than one independent functional blocks. If this slave wants to utilize FCB-funcionalities they must implement the appropriate number of internal memory bits (0, 1 or 2) for each of these adresses.

## III) Implementation for the primary (broadcast) address 255

All transfers to the primary broadcast address 255 ($FF) are not answered and should hence be implemented by the master with the FCV-Bit cleared. Note that a SND_NKE to primary address 255 will clear the internal "Last received FCB"-Bits of all slaves with primary addresses 0-250 and with FCB-Bit implementation simultaneously.

## IV) Implementation for the primary (test) address 254 ($FE)

A slave should answer to all requests to the primary address 254 ($FE=test address) irrespective of its own primary address. The answer must contain its own primary address and not the address 254 ($FE). This test address is used by readout- or test equipment in point-to-point mode. Although this is a second primary address for each slave separate "Last received FCB"-Bit(s) are not required for this special case, since any test equipment or master is required to issue a SND_NKE after each

reconnect or power fail thus clearing the "Last received FCB"-Bit(s) and thus preparing for a virgin transaction irrespective of the previous communication history.

# D) Implementation for secondary addressing

## I) General aspects

Secondary addressing is implemented by "selecting" a certain slave by sending a "selection SND_UD-telegram" with CI=$52 or $56 (depending on byte sequence in multibyte data fields) to the pseudo primary address 253 ($FD) with data characterizing the slave by its 8-digit identification number, the manufacturing code, product identification, media code or other criteria. A slave with implemented secondary addressing will compare these data to its internal identification parameters and if they match completeley will go to the "selected" state and will acknowledge the selection SND_UD-telegram. All slaves with even minimally different identification parameters will be "deselected" and will not participate in any other communication via the pseudo primary address 253 ($FD) until selected.
All following communication via the pseudo primary address 253 ($FD) will be accepted, executed and acknowleged by a selected slave and ignored by a deselected slave. The RSP_UD-telegrams must contain the own primary address. A slave which implements only secondary addressing should not react to any primary address communication exept via the addresses 253 ($FD) if selected and to the addresses 254 ($FE) or 255 ($FF). Its RSP_UD-telegrams should contain 253 ($FD) to signal that it will not participate in primary addressing.
A select sequence with wild card symbols in the select data can be used to learn which meters with secondary address capability are connected to the logical segment associated with the master. Here the simultanous answers of multiple slaves which all meet the wildcarded and hence not unique selection criteria might produce collisions and hence uninterpretable answers. In these cases the master must sequentially test all legal combinations within the wildcard fields to find out the individual slaves which generated the collision. Note that the selection telegram themselves are always single telegram communications without incremental action. Therefore for these telegrams, the slave should simply acknowledge the selection telegram if its selection fits irrespective whether the FCB has been toggled or not.

## II) FCB-Implementation slave

A slave with implemented secondary addressing and with implemented FCB-administration must have an additional set of 0, 1 or 2 separate "Last Received FCB"-memery Bit(s) for all

communication via the pseudo primary address 253 ($FD). If it can communicate also alternatively over some other primary address (exept the special addresses 254 and 255) an additional set of 0, 1 or 2 "Last received FCB"-memry bit(s) for each of these primary addresses is required. A valid selection telegram will not only set the internal selection bit but will also clear all 0,1 or 2 internal "Last received FCB"-memory bit(s) associated with secondary addressing via the pseudo primary address 253 ($FD). The master will start the communication (REQ_UD2 or SND_UD) after any selection telegram (CI=$52 or $56) with the FCV-Bit set and the FCB-Bit set. If a slave has more than one alternative secondary identification, only a single set of 0,1 or 2 "Last received FCB"-memory bit(s) for all secondary addresses is required.

## III) FCB-Implementation master

The master must implement a separate pair of "Next FCB image"-Bits for pseudo primary address 253 ($FD) as for each other primary address. Although these "Next FCB image"-bits might be used for many slave, no confusion exists, since for accessing another slave a selection telegram is required which will define the future FCB sequence both for slave and master.