Title: Reception Scheme changed for Ignoring IFG Traffic / Transformer Noise during Synchronized Operation

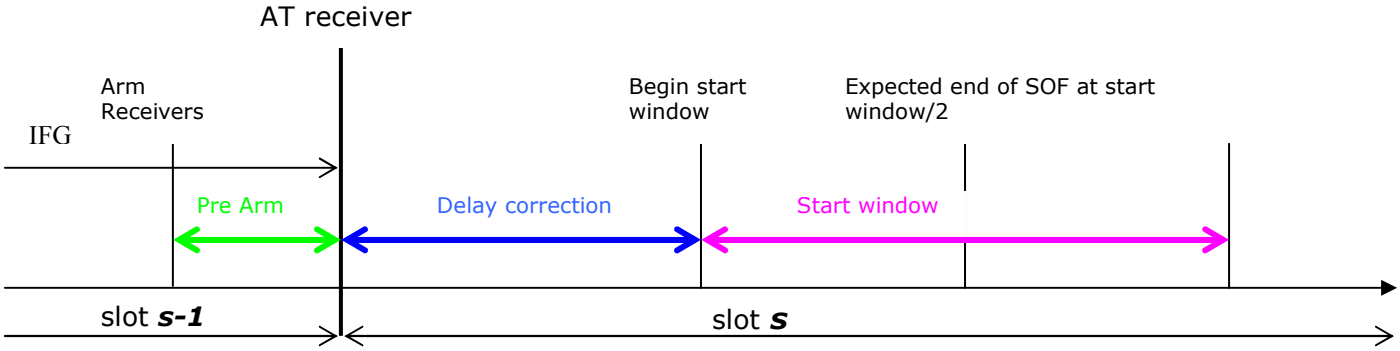
Subject: Protocol update to patch HW problem

Related Revision: Firmware V2.00 and later

Description:

In protocol versions prior to 2.00 the receivers are armed around 5 us before AT for reception. If during the interval between the point-in-time of arming and the begin of the startwindow (delayed action time) noise/traffic is detected on the bus the reception of a valid frame in the start window is prevented. The receivers should therefore be armed *after* the action time, after a stable preamble is on the channels (Manchester encoding) or after there is no possible noise on the channels (the sender drives the bus in MFM encoding). For MII encoding noise due to a floating bus can be precluded.

Receiver protocol version < 2.00



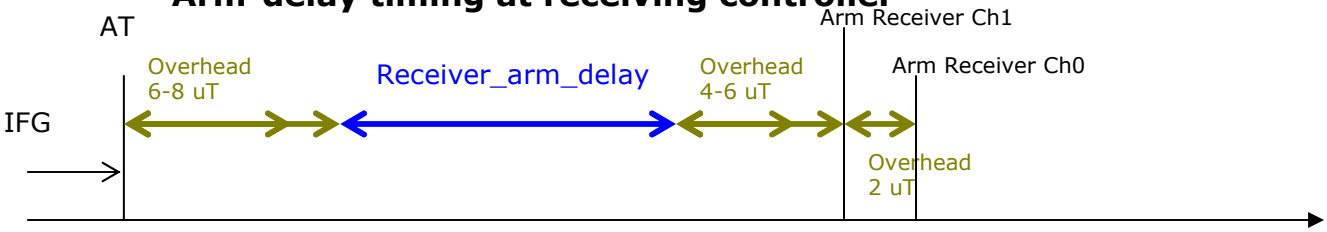
Fix:

The receivers are reset, but not armed before the AT. The controller waits for the AT then waits for additional *Receiver_arm_delay* micro ticks (uT) and then arms the receivers. The parameter *Receiver_arm_delay* is located in the MEDL and valid for both channels. The other timing parameters must be chosen according to the to formulas listed in this document.

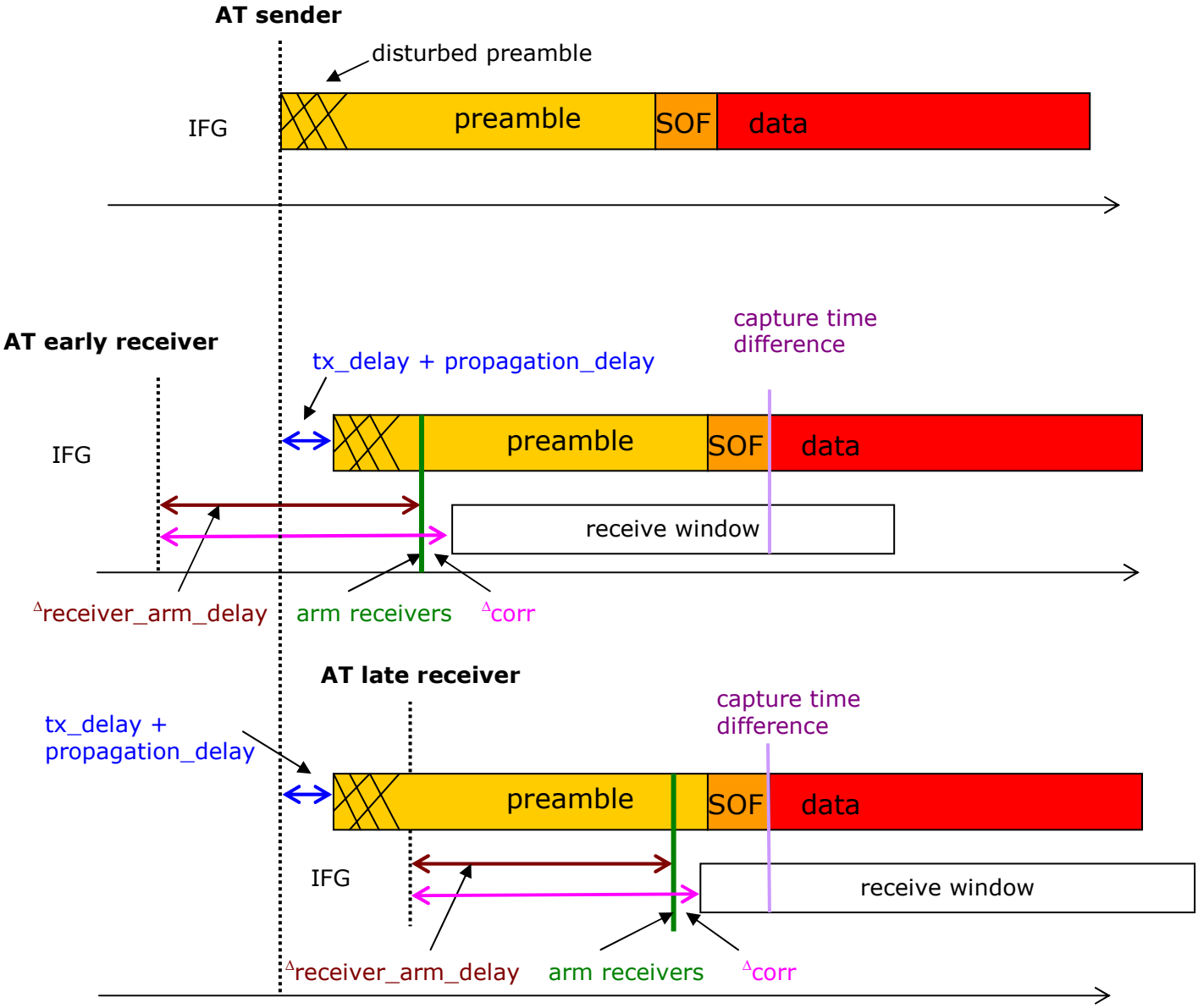
Important: the receiver arm delay is realized in the C2NF by the firmware code, so clock cycles are spent on the processing overhead (this additional delay and its jitter is caused by waiting for AT and the end of the arm delay):

$^{\Delta}$receiver_arm_delay (physical time) = Receiver_arm_delay (MEDL parameter)
+ 12 * $^{\Delta}$microtick (processing overhead)
+ [0,4] * $^{\Delta}$microtick (processing jitter)

Arm-delay timing at receiving controller



Arm-delay timing example (Manchester encoding) for an early and a late receiver with receiver protocol version >=2.00



As shown in the figure above, the $^{\Delta}$receiver_arm_delay must end *before the receive window* is opened (AT + $^{\Delta}$corr).
The arming of the receivers must be triggered *before SOF* and *after the disturbed preamble part*

These conditions must be fulfilled in any in temporal shift scenario between sender and receiver (+/- ∏).

Parameter selection for Manchester encoding ignoring noise / disturbed preambles (completing section 11.1.2 C2NFHW SDD V1.1.2):

Assumptions / Preconditions:

$\Pi > 16 * {}^{\Delta}microtick$

${}^{\Delta}rec\_window = 4 * \Pi$

Symmetric channels with same encoding and speed (same parameters for both channels).

Given:

Start of frame pattern size:	SOF_bits = 3.5 [bits]
Precision:	Π [s]
Transmission speed on bus:	speed [bit/s]
Disturbed preamble bits to be tolerated:	max_disturbed_preamble_bits [bits]
Maximum propagation delay[1]:	${}^{\Delta}$propagation_delay_max [s]
Minimum propagation delay[2]:	${}^{\Delta}$propagation_delay_min [s]
Duration of an microtick:	${}^{\Delta}$microtick [s]
Individual propagation delays	${}^{\Delta}$corr_ch0/1 [s]

Calculated:

${}^{\Delta}send\_delay = 0$

preamble_bits = ((${}^{\Delta}$propagation_delay_max - ${}^{\Delta}$propagation_delay_min) + 3*Π) * speed +max_disturbed_preamble_bits +1[3]

preamble_doublebits = ceil (preamble_bits / 2)

${}^{\Delta}send\_offset$ = (preamble_doublebits * 2 + SOF_bits) / speed

${}^{\Delta}corr\_base = {}^{\Delta}send\_offset - 2*\Pi + {}^{\Delta}microtick$[4]

${}^{\Delta}receiver\_arm\_delay = \Pi + {}^{\Delta}propagation\_delay\_max$ + max_disturbed_preamble_bits / speed

MEDL parameters:

Send_Delay_Ch0 = 0
Send_Delay_Ch1 = 0
Delay_Correction_Ch0 = ceil((${}^{\Delta}$corr_base + ${}^{\Delta}$corr_ch0) / ${}^{\Delta}$microtick)
Delay_Correction_Ch1 = ceil((${}^{\Delta}$corr_base + ${}^{\Delta}$corr_ch1) / ${}^{\Delta}$microtick)

Receiver_arm_delay = ceil(${}^{\Delta}$receiver_arm_delay / ${}^{\Delta}$microtick) - 12
Preamble_doublebits = preamble_doublebits

For only ignoring IFG noise set max_disturbed_preamble_bits = 0

---

[1] Between all nodes in the cluster, should include ${}^{\Delta}tx\_delay = 5 * {}^{\Delta}microtick$ (Manchester)
[2] Between all nodes in the cluster, should include ${}^{\Delta}tx\_delay = 5 * {}^{\Delta}microtick$ (Manchester)
[3] To tolerate receiver_arm_delay jitter
[4] For AS8202NF internal pipeline-delay of $recbytes register

Parameter selection for MFM encoding ignoring noise on floating channel (completing section 11.1.2 C2NFHW SDD V1.1.2):

Assumption:

Assumptions / Preconditions:

$\Pi > 16 * {}^{\Delta}microtick$

${}^{\Delta}rec\_window = 4 * \Pi$

Symmetric channels with same encoding and speed (same parameters for both channels).

Given:

Precision:	Π [s]
Maximum propagation delay[5]:	${}^{\Delta}propagation\_delay\_max$ [s]
Minimum propagation delay[6]:	${}^{\Delta}propagation\_delay\_min$ [s]
Duration of an microtick:	${}^{\Delta}microtick$ [s]
Individual propagation delays	${}^{\Delta}corr\_ch0/1$ [s]

Calculated:

${}^{\Delta}send\_delay = 3 * \Pi + ({}^{\Delta}propagation\_delay\_max - {}^{\Delta}propagation\_delay\_min) + 6 * {}^{\Delta}microtick$[7]
${}^{\Delta}corr\_base = \Pi + ({}^{\Delta}propagation\_delay\_max - {}^{\Delta}propagation\_delay\_min) + 7 * {}^{\Delta}microtick$[8]
${}^{\Delta}receiver\_arm\_delay = \Pi + {}^{\Delta}propagation\_delay\_max$

MEDL parameters:

$Send\_Delay\_Ch0 = ceil({}^{\Delta}send\_delay / {}^{\Delta}microtick)$
$Send\_Delay\_Ch1 = ceil({}^{\Delta}send\_delay / {}^{\Delta}microtick)$
$Delay\_Correction\_Ch0 = ceil(({}^{\Delta}corr\_base + {}^{\Delta}corr\_ch0) / {}^{\Delta}microtick)$
$Delay\_Correction\_Ch1 = ceil(({}^{\Delta}corr\_base + {}^{\Delta}corr\_ch1) / {}^{\Delta}microtick)$

$Receiver\_arm\_delay = ceil({}^{\Delta}receiver\_arm\_delay / {}^{\Delta}microtick) - 16$

[5] Between all nodes in the cluster, should include ${}^{\Delta}tx\_delay = 5 * {}^{\Delta}microtick$ (MFM)
[6] Between all nodes in the cluster, should include ${}^{\Delta}tx\_delay = 5 * {}^{\Delta}microtick$ (MFM)
[7] For AS8202NF internal pipeline-delay of $recbytes register
[8] For AS8202NF internal pipeline-delay of $recbytes register

Parameter selection for MFM and MII encoding without noise (completes section 11.1.2 C2NFHW SDD V1.1.2):

Assumptions / Preconditions:

$\Pi > 16 * {}^{\Delta}microtick$

${}^{\Delta}rec\_window = 4 * \Pi$

Symmetric channels with same encoding and speed (same parameters for both channels).

Given:

Precision:	Π [s]
Individual propagation delays	${}^{\Delta}corr\_ch0/1$ [s]
Duration of an microtick:	${}^{\Delta}microtick$ [s]
Transmitter delay:	${}^{\Delta}tx\_delay = 5 * {}^{\Delta}microtick$ (MFM)
	${}^{\Delta}tx\_delay = 13 * {}^{\Delta}microtick$ (MII)

Calculated:

${}^{\Delta}send\_delay = 2 * \Pi - {}^{\Delta}tx\_delay + 17 * {}^{\Delta}microtick$[9]
${}^{\Delta}corr\_base = 17 * {}^{\Delta}microtick$[10]

MEDL parameters:

MEDL parameters (for receiver r in sending slot s):

$Send\_Delay\_Ch0 = ceil({}^{\Delta}send\_delay / {}^{\Delta}microtick)$
$Send\_Delay\_Ch1 = ceil({}^{\Delta}send\_delay / {}^{\Delta}microtick)$

$Delay\_Correction\_Ch0 = ceil(({}^{\Delta}corr\_base + {}^{\Delta}corr\_ch0) / {}^{\Delta}microtick)$
$Delay\_Correction\_Ch1 = ceil(({}^{\Delta}corr\_base + {}^{\Delta}corr\_ch1) / {}^{\Delta}microtick)$

$Receiver\_arm\_delay = 0$

[9] +1 microtick for AS8202NF internal pipeline-delay of $recbytes register
[10] +1 microtick for AS8202NF internal pipeline-delay of $recbytes register