SIEMENS

Microcontrollers

ApNote AP1633

☐ additional file
APXXXX01.EXE available

Connecting a Fast External A/D Converter on the CAPCOM- Unit

Connecting an Fast External A/D Converter (800ns Conversion-Time) by using the PEC and the 50ns Resolution on the CAPCOM-Unit

Harald Lehmann / Siemens Cupertino

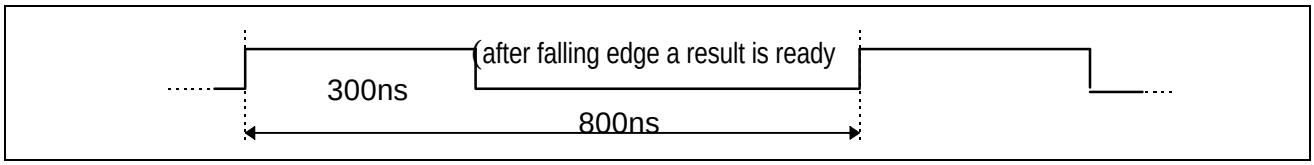
SIEMENS

AP1633 ApNote - Revision History		
Actual Revision : Rel.01	Previous Revison: none	
Page of actual Rel.	Page of prev. Rel.	Subjects changes since last release)

There are particular fast A/D converters (ADC) which are providing a result after three 800ns-cycles and then continuously every 800ns-cycle.

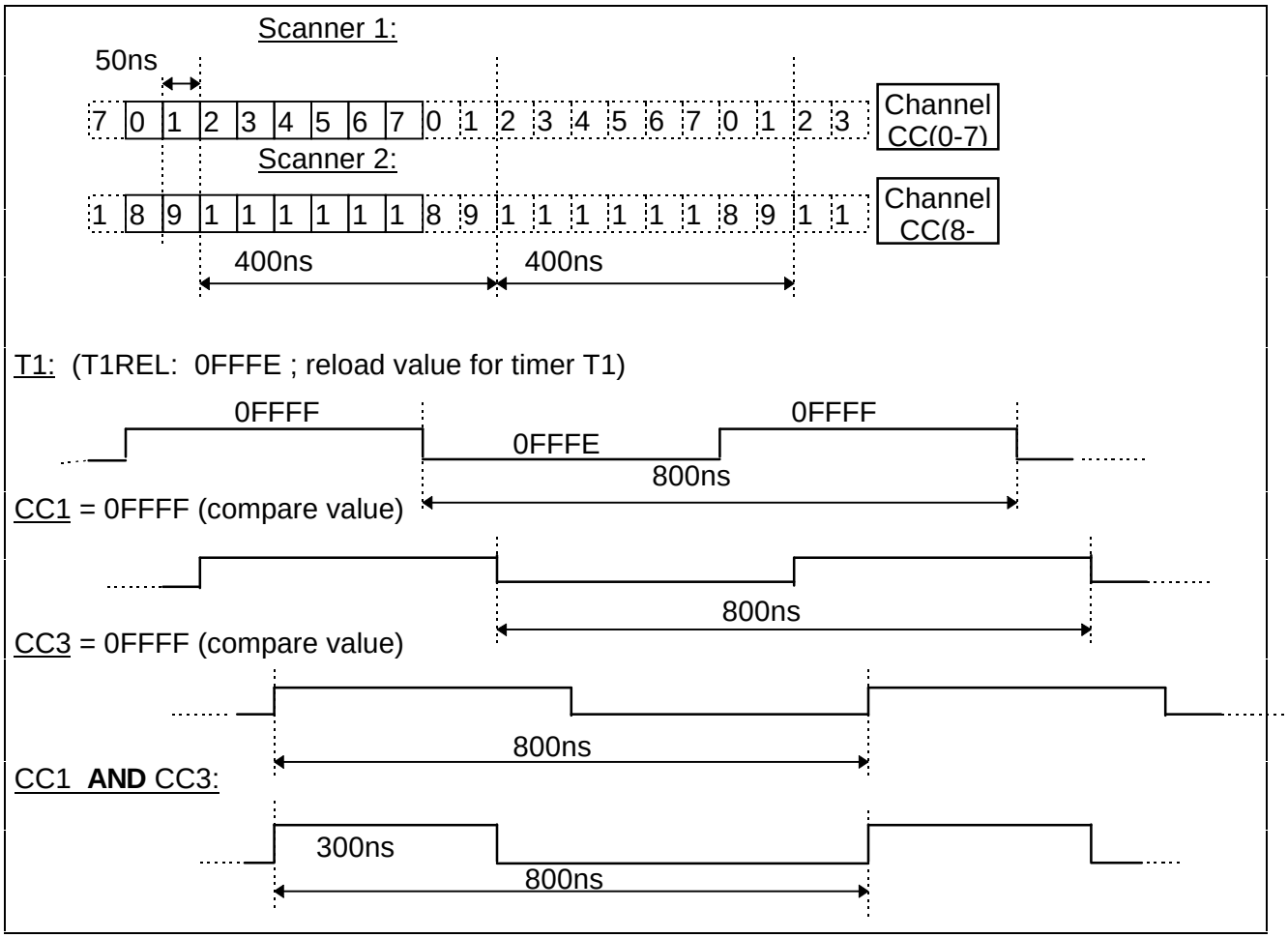
1 Triggering of the ADC

In order to use this ADC the signal as followed has to be generated:



This signal will be generated with the CAPCOM unit by using the 50ns shift of each channel as followed (see also ApNote “Generation of High Resolution PWM Signals”):

There are two internal scanners (parallel running) for the CAPCOM unit for the lower (0-7) and upper (8-15) channels:



2 Data transfer of the ADC result

2.1 The Hardware Solution:

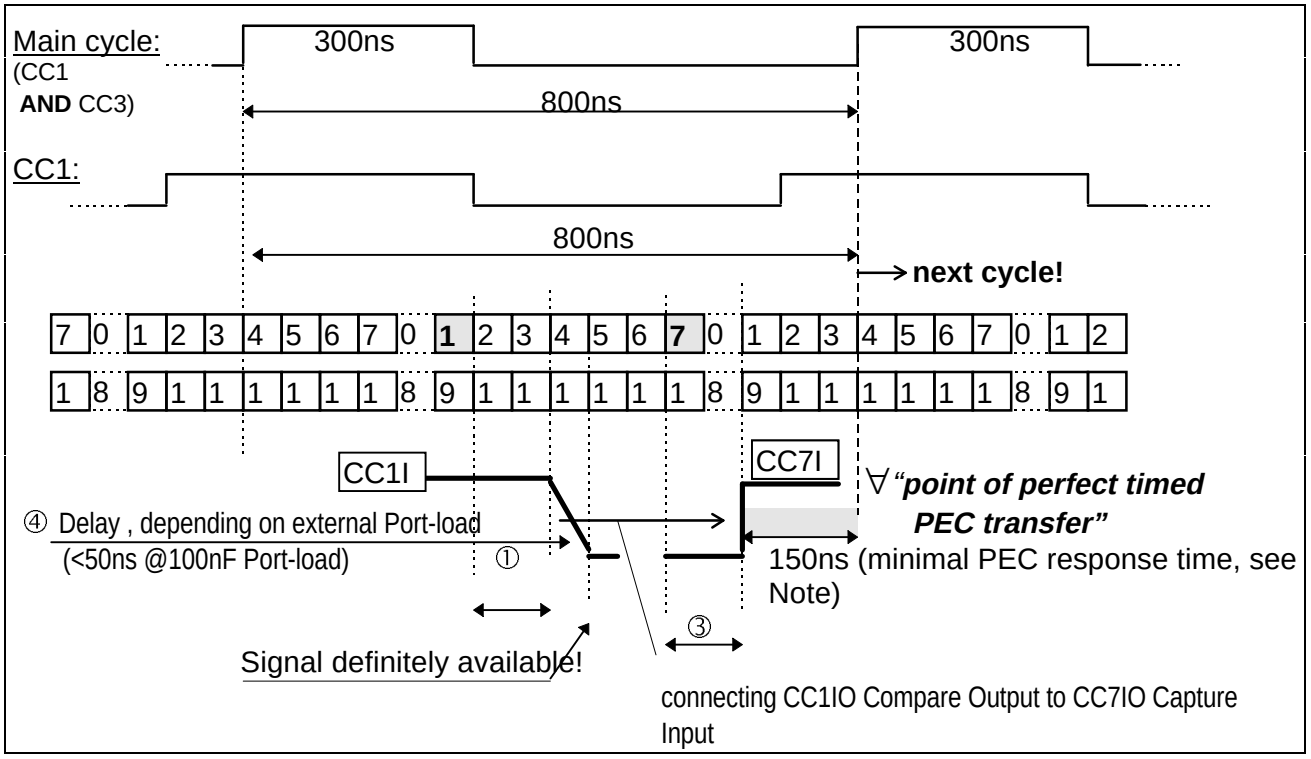
The external ADC will provide a result after three cycles and then continuously every cycle. After each 800ns cycle there have to be an (external) interrupt to trigger the PEC and transfer the ADC result (8 Bit from a Port) to a location RAM (intern or extern).

The external Interrupt will be generated as followed:

⇒ with the falling edge of CC3 channel.

Now the timing of PEC transfer can be fine tuned by selecting the Capture channel for generating the PEC.

Example: Have the interrupt request flag (IR) set at the beginning at each cycle.
As the previous page shows, the falling edge (CC1) will be generated when the scanner finished channel CC1/CC9. There are additional delays before the Port-pin will actually change its level:
Compare match to Output:100ns ①
plus the delay , depending on external Port-load (<50ns @100nF Port-load) ④
⟶ 100ns + ~50ns = ~150ns
For the input Capture of signals is only one delay to add on: Sample Pin to Capture IR Setting: 100ns ③



As shown in this example, the Interrupt flag CC7IR is set before the next cycle begins!
Now it can be *"fine tuned"* (by changing the Capture Input Channel) in respect of additional delays caused by the hardware and/or software.

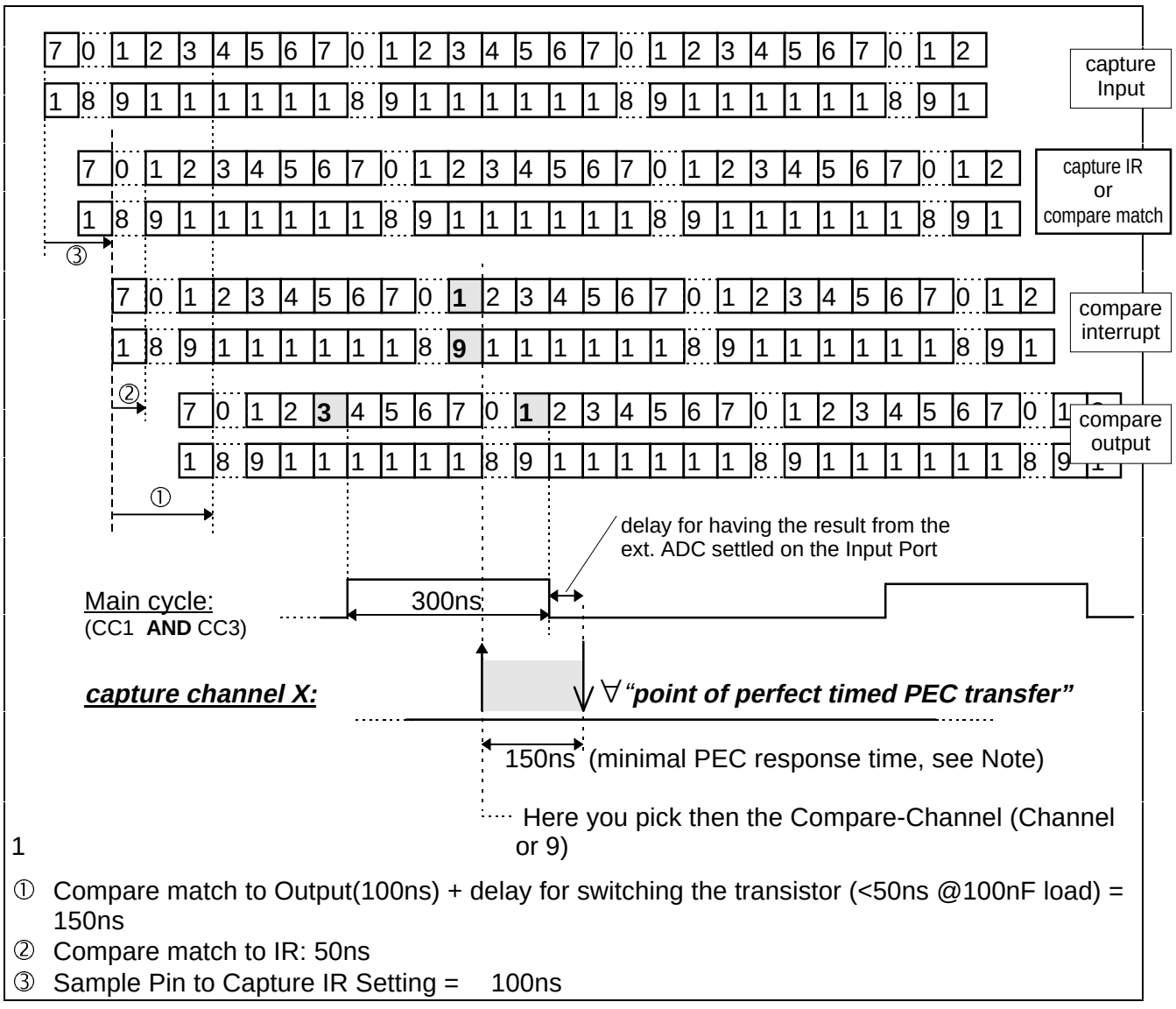
Note: Now you have to add up only the PEC response time as is described in the User's Manual to get the timing for the transfer of the results.

For example: The worst case PEC response time during internal ROM program execution adds to 9 state times (450ns @ 20 MHz CPU clock).

The minimal response is 150ns, this 150ns can be incorporated in the *"fine tuning"*, too.

2.2 The Software Solution:

With the following sketch is it an easy way to determine the appropriate Compare Channels for generating the interrupt to trigger the PEC-transfer. Please note that you have to start at the point where do you wish to have the PEC-transfer scheduled (*"point of perfect timed PEC transfer"*).



① Compare match to Output(100ns) + delay for switching the transistor (<50ns @100nF load) = 150ns
② Compare match to IR: 50ns
③ Sample Pin to Capture IR Setting = 100ns

CAP/COM Scan Mechanism Overview

Now it can be *"fine tuned"* (by changing the Compare-Channel) in respect of additional delays caused by the hardware and/or software.

Note: Now you have to add up only the PEC response time as is described in the User's Manual to get the timing for the transfer of the results.

For example: The worst case PEC response time during internal ROM program execution adds to 9 state times (450ns @ 20 MHz CPU clock).

The minimal response is 150ns, this 150ns can be incorporated in the “fine tuning”,