33.11 Service Request Generation

The EDSADC can activate service request output signals to issue an interrupt, to trigger a DMA channel, or to trigger other on-chip modules. Each channel provides 2 service request lines, SRxM and SRxA.

The product-specific appendix details the available service request connections.

Several events can be assigned to each service request output. Service requests can be generated by several types of events:

- **Result events:** indicate the availability of new valid results. Usually, this triggers a read action by CPU or DMA. Result events are generated at the output rate of the configured filter chain (indicated by bits RESEVx in register **EVFLAG**). The output rate of result service requests depends on the selected FIFO fill level (see bitfield SRLVL in register **RFCx (x=0-13)**).
  Can be issued via SRxM.
- **Alarm events:** indicate that a conversion result value is within a programmable value range. This offloads the CPU/DMA from background tasks, i.e. a service request is only activated if the specified conversion result range is met or exceeded.
  Can be issued via SRxA.
- **Special events:** indicate specific circumstances of previously configured functions.
  - Timestamp trigger event can generate a service request.
    Can be issued via SRxM (see read sequences) or SRxA (separate).
  - Capture event for sign delay measurement can generate a service request.
    Can be issued via SRxA.

Each event is indicated by a dedicated flag that can be cleared by software. If a service request is enabled for a certain event, the service request is generated for each event, independent of the status of the corresponding event indication flag. This ensures efficient DMA handling of EDSADC events (the event can generate a service request without the need to clear the indication flag).

Note: Event flags SDCVAL are cleared when the filter chain is initialized,
i.e. when the channel is started (CHxRUN = 1), when an integration window is started with FCR = 0, or when the calibration algorithm begins or ends.
Event flags ALEVx, TSVAL and RESEVx are cleared when the channel is started (CHxRUN = 1).

Note: The following registers provide a set of bits for each available channel.
The number of available channels depends on the chosen device type.

Event Flag Register

The register below shows the maximum configuration.

Other products of the family may have less channels and, consequently, less valid RESEVx/ALEVx flags.

EVFLAG
Event Flag Register **($00E0_H$)** **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0		**ALEV13**	**ALEV12**	**ALEV11**	**ALEV10**	**ALEV9**	**ALEV8**	**ALEV7**	**ALEV6**	**ALEV5**	**ALEV4**	**ALEV3**	**ALEV2**	**ALEV1**	**ALEV0**
r		rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0		**RESEV13**	**RESEV12**	**RESEV11**	**RESEV10**	**RESEV9**	**RESEV8**	**RESEV7**	**RESEV6**	**RESEV5**	**RESEV4**	**RESEV3**	**RESEV2**	**RESEV1**	**RESEV0**
r		rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh	rwh

Field	Bits	Type	Description
RESEVx (x=0-13)	x	rwh	**Result Event** *Note: Bit RESEVx is cleared when result register RESMx is read, or when bit RESECx is written with 1.* 0_B No result event 1_B New result value is generated by the filter chain
ALEVx (x=0-13)	x+16	rwh	**Alarm Event** 0_B No alarm event 1_B An alarm event has occurred
0	15:14, 31:30	r	**Reserved, write 0, read as 0**

Event Flag Clear Register

The register below shows the maximum configuration. Other products of the family may have less channels and, consequently, less valid RESECx/ALECx control bits.

EVFLAGCLR
Event Flag Clear Register **($00E4_H$)** **Application Reset Value: $0000\ 0000_H$**

31	30	29	28	27	26	25	24	23	22	21	20	19	18	17	16
0		**ALEC13**	**ALEC12**	**ALEC11**	**ALEC10**	**ALEC9**	**ALEC8**	**ALEC7**	**ALEC6**	**ALEC5**	**ALEC4**	**ALEC3**	**ALEC2**	**ALEC1**	**ALEC0**
r		w	w	w	w	w	w	w	w	w	w	w	w	w	w

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
0		**RESEC13**	**RESEC12**	**RESEC11**	**RESEC10**	**RESEC9**	**RESEC8**	**RESEC7**	**RESEC6**	**RESEC5**	**RESEC4**	**RESEC3**	**RESEC2**	**RESEC1**	**RESEC0**
r		w	w	w	w	w	w	w	w	w	w	w	w	w	w

Field	Bits	Type	Description
RESECx (x=0-13)	x	w	**Result Event Clear** 0_B No action 1_B Clear bit RESEVx
ALECx (x=0-13)	x+16	w	**Alarm Event Clear** 0_B No action 1_B Clear bit ALEVx
0	15:14, 31:30	r	**Reserved, write 0, read as 0**

Note: Software can set flags RESEVx and ALEVx and trigger the corresponding event by writing 1 to the respective bit. Writing 0 has no effect.
Software can clear these flags by writing 1 to bit RESECx and ALECx, respectively.