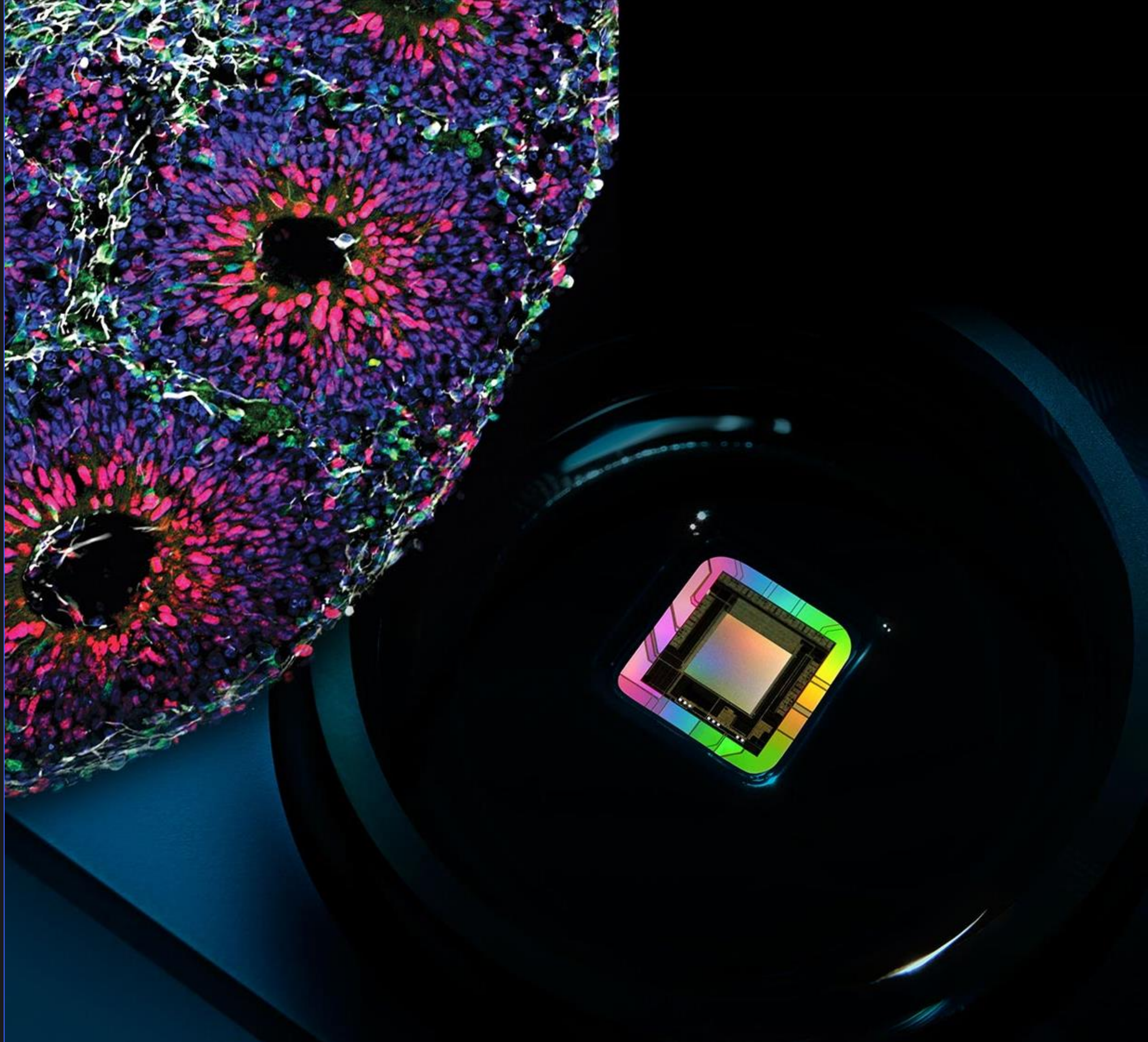
Microchip-based technologies for life science

API v. 2.4 for BioCAM DupleX

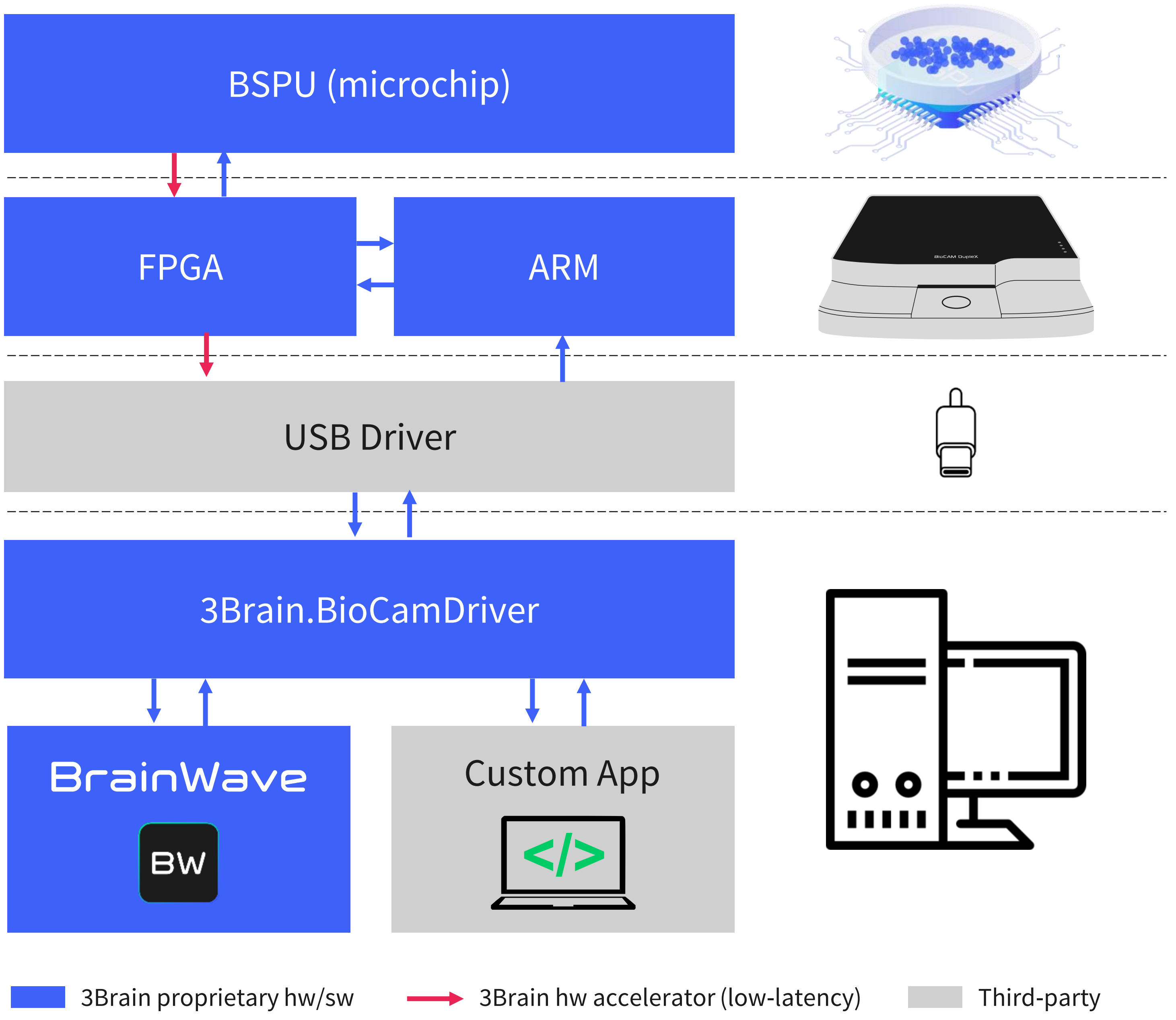


Last updated on 23.06.2023

Architecture and Specs

Specs

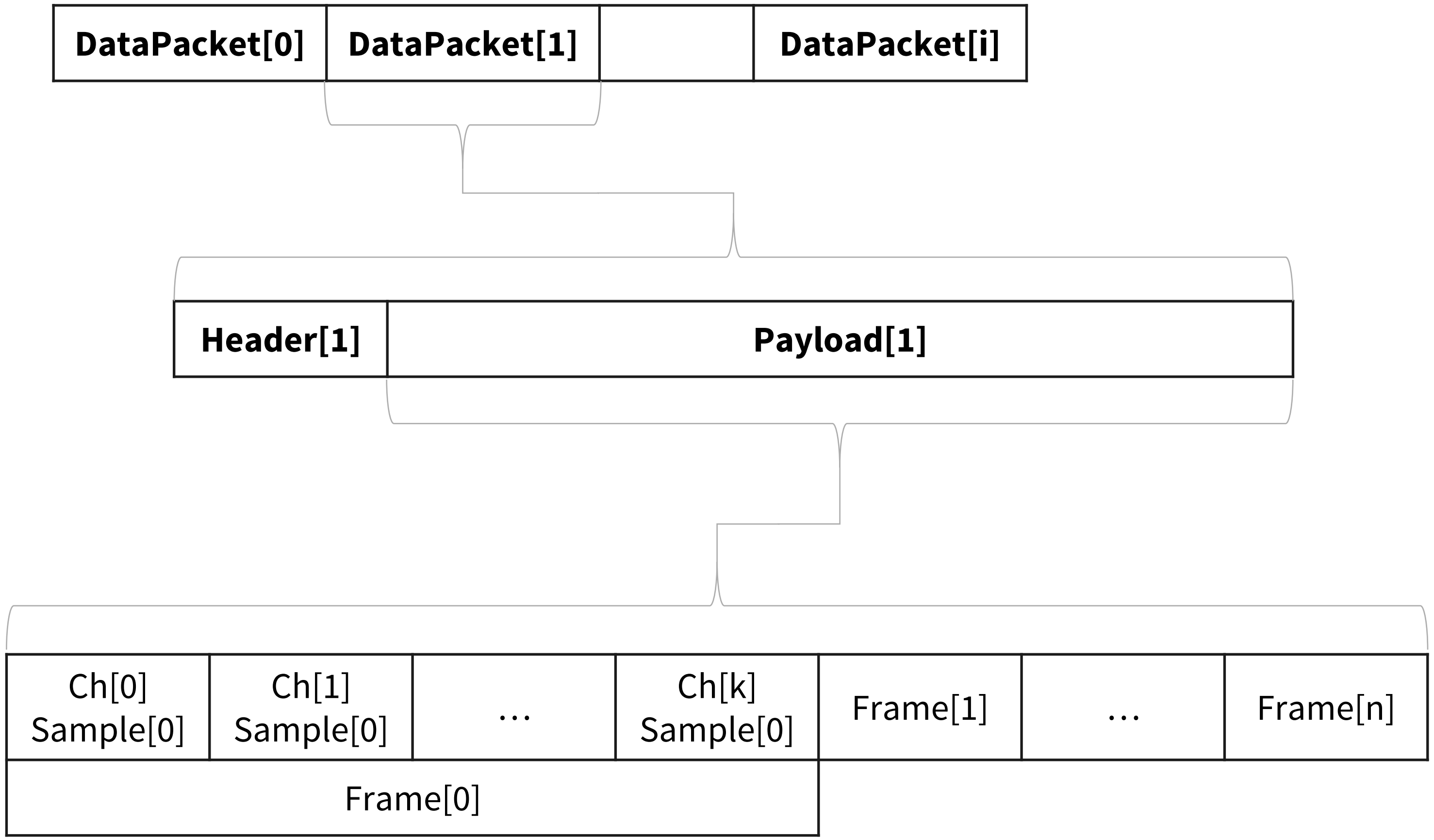
- Built for .NET Framework 4.7.
- Compatible with Windows 10 or 11.
- Can be used in C++, Java, Python, etc. by using a proper wrapper (not provided).
- Same toolset used in BrainWave (starting from BrainWave v. 5.0).



Reading Capabilities

Specs and Format

- Low-latency thanks to the 3Brain hw accelerator (see next slides).
- Data is received as DataPackets, with each DataPacket beign made up by a small Header, which contains useful information as the TimeStamp of the data, and the Payload, i.e. the data itself
- The size of the Payload is fixed during a certain acquisition and depends on the Acquisition Time Period (see next slides).
- Data inside the Payload is in raw format, as a sequence of frames where each frame contains the samples from all the electrodes.



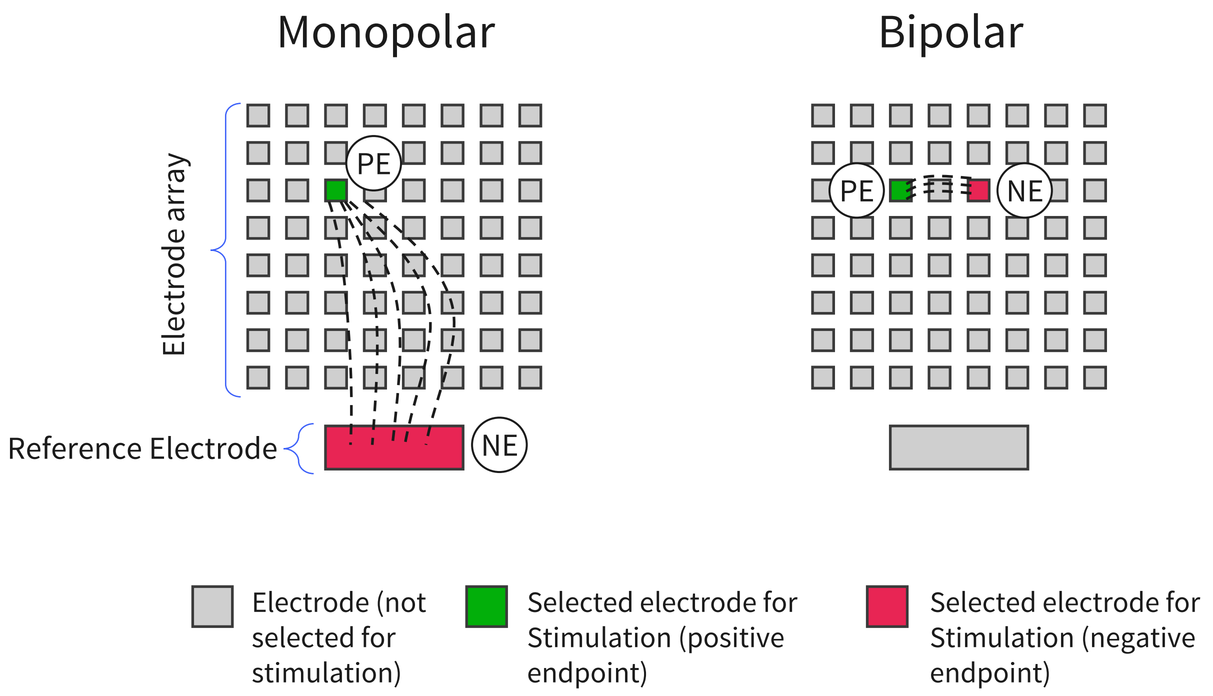
i = number of transmitted data packets; k = number of channels (electrodes); n = number of acquired frames

Stimulation Capabilities - 1

Types of Stimulation

Electrical stimuli are released between positive endpoints (PEs) and negative endpoints (NEs). At a given time, PEs and NEs can be assigned to either (and consequently stimulus released through):

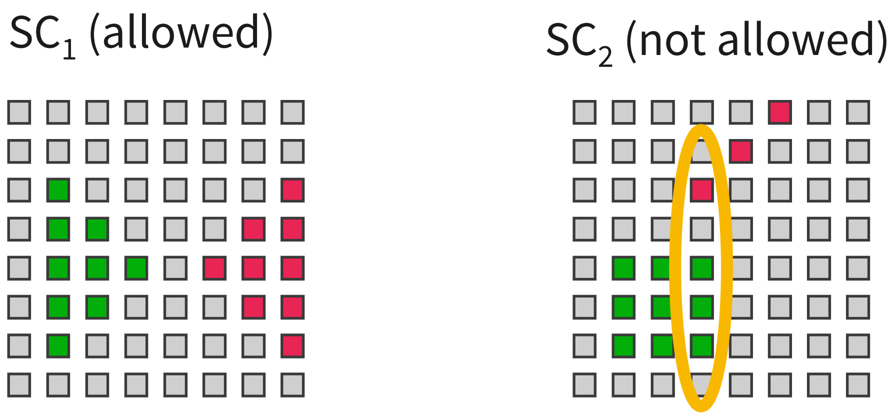
1. one electrode (or multiple electrodes) on the array and the reference electrode (Monopolar stimulation).
2. two electrodes or groups of electrodes on the array (Bipolar stimulation).



Spatial Configurations (SCs)

A Spatial Configuration (SC) is the positive and negative endpoints that are selected for the stimulation *at a given time*. Multiple and different SCs can quickly follow one another (see next slide). SCs must meet these conditions:

1. The combined number of endpoints is ≤ 1’000.
2. Positive and negative endpoints do not share the same column(s) in the electrode array. In the aside figure, SC_1 is allowed, while SC_2 is not because one negative endpoint shares an electrode column that is already taken by 3 positive endpoints.



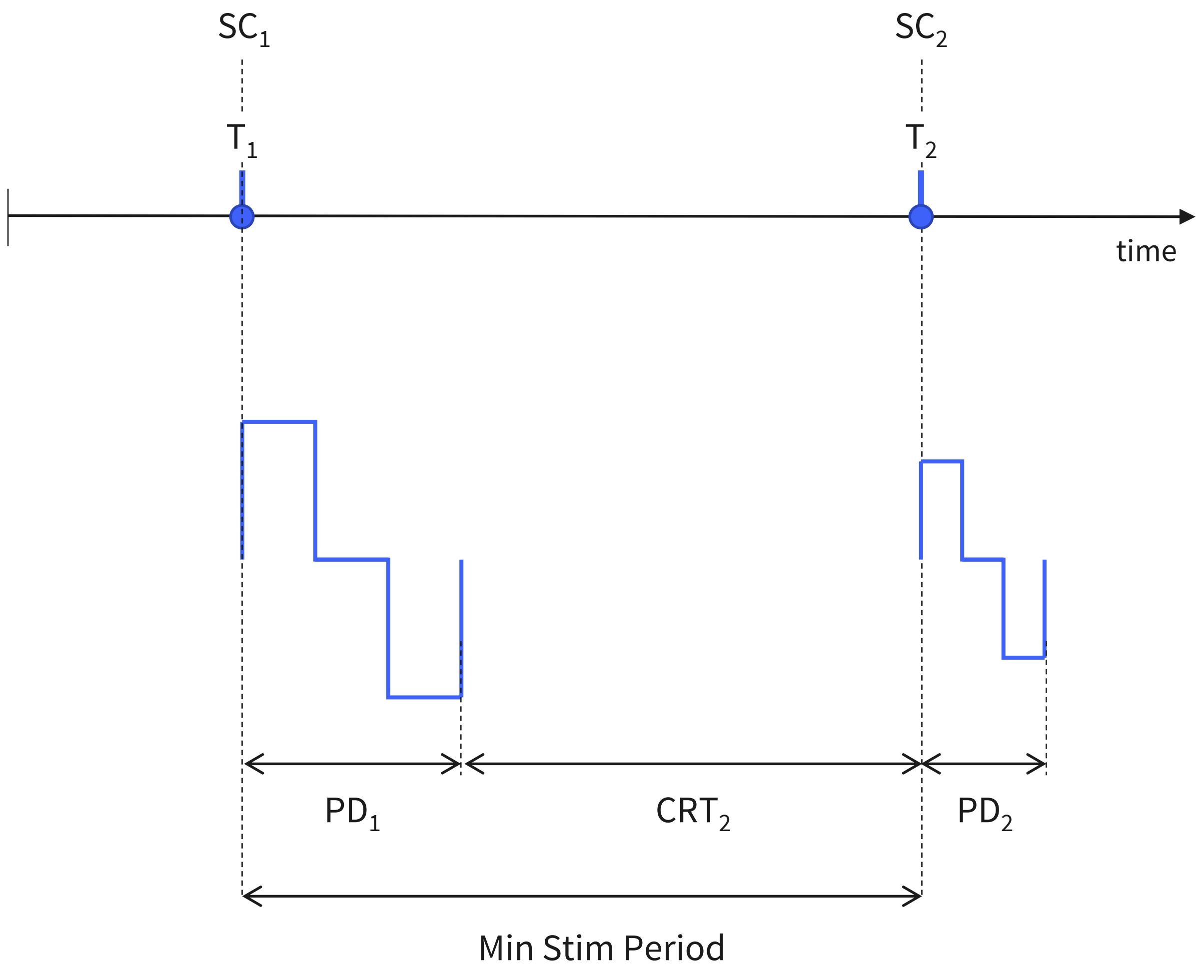
Stimulation Capabilities - 2

Frequency

- The minimum distance between two consecutive stimuli happening at T_1 and T_2 (Min Stim Period) is the sum of the following two values rounded up to the nearest stimulator time resolution increment (time resolution = 10μs):
    - The Pulse Duration of the first stimulus (PD_1)
    - The Chip Reconfig Time for the second stimulus (CRT_2), i.e. the time needed to reconfigure the chip based on the SC_2 (note that only SC_2 is relevant and not SC_1)
- Chip Reconfig Time = 0 if $SC_2 = SC_1$, i.e. when the spatial configuration does not change. In this scenario very high stimulation frequencies can be achieved (e.g. 4kHz for a Pulse Duration of 250μs).
- If $SC_2 \neq SC_1$, The Chip Reconfig Time depends from the number of independent rows (N_{ROWS}) required by SC_2, that is the union of the rows in the electrode array when considering all endpoints for SC_2, according to the formula:

  Chip Reconfig Time = 26μs + 8.4μs * (N_{ROWS} -1)

  For instance, if PD_1 = 150μs and SC_2 has 2 endpoints on 2 different rows (N_{ROWS} = 2), Chip Reconfig Time = 150 + 26 + 8.4μs = 184.4μs => rounded up to 190μs



Possibility of Programming Complex Stimulation Patterns

Examples and Constraints

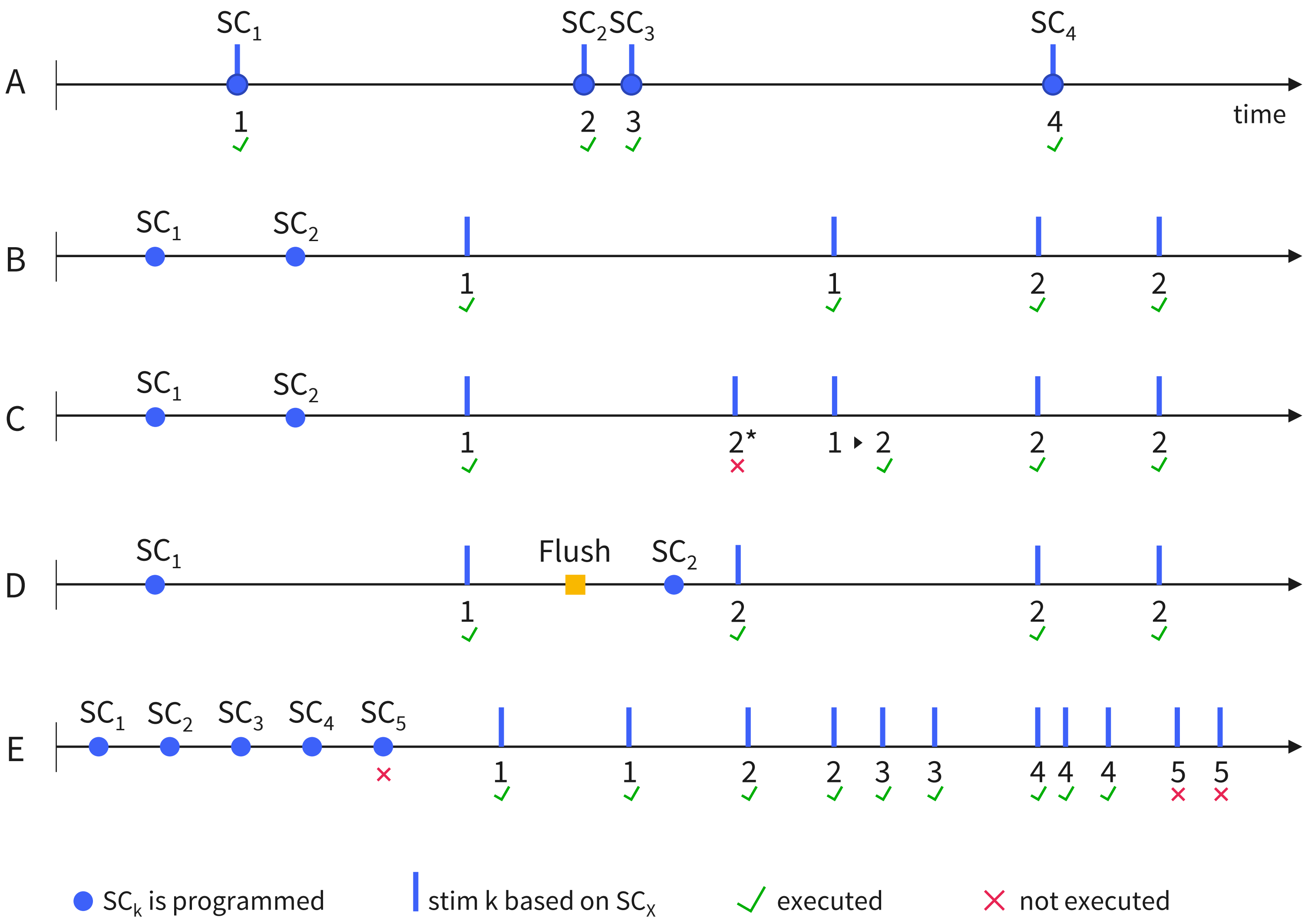
A. Stimuli based on different SPs are released immediately and in succession, even at short intervals (e.g. 2 and 3; for specs see prev. slide).

B. Stimuli are programmed to be released later on (up to 1k future stimuli). When programming stimuli must be temporally ordered.

C. SC_2 tries to program three stimuli, however the first (*) is not executed (it's ahead the last from SC_1). Also, the last stimulus from SC_1 is converted to SC_2.(because SC_2 effectively takes place at (*) time).

D. To overwrite future stimuli, the Flush command is first executed.

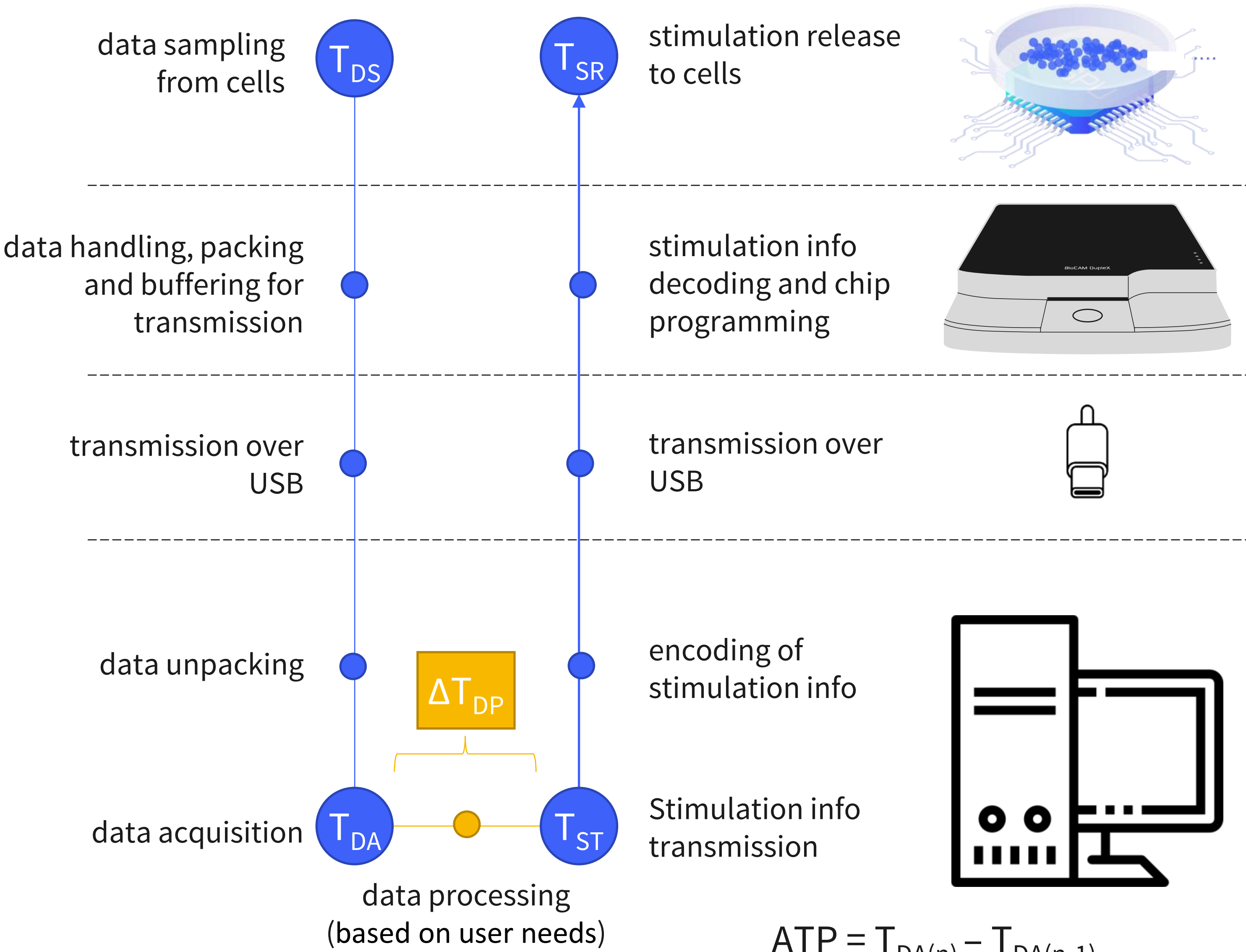
E. Multiple SCs can be appended one after the other for future stimuli. The maximum number of SCs that can be buffered at a given time instant (SC_{max}) depends on how many electrodes are involved in those SCs. If all SCs use two electrodes SC_{max} = 50; regardless the number of electrodes used by SCs, it is guaranteed to have always SC_{max} ≥4.



Closed-loop applications

Method

- The API can be set to operate with a custom Acquisition Time Period (ATP, i.e. a fixed amount of data that is acquired at each acquisition cycle) between 1ms and 250ms.
- We measure the Closed-Loop Latency (CLL) as the total time between the data sampling from cells (T_{DS}) and the stimulation release (T_{SR}), i.e. the time to close the loop, without the time needed to process the data (ΔT_{DP}), which is based on user needs. The CLL measure is very accurate and it is done by timestamping (accuracy ~10ns) both sampled data and stimulation information.



$$ATP = T_{DA(n)} - T_{DA(n-1)}$$

$$CLL = (T_{DA} - T_{DS}) + (T_{SR} - T_{ST})$$

Performances for closed-loop applications

Considerations

- Windows is not a real-time OS, therefore the measured CLL can experience fluctuations subject to several factors, such as processor clock speed and processor load.
- CLL is also affected by the number of EndPoints selected for a given stimulation, which affects the time to send instructions over USB and programming the chip.

Acq. Time Period [ms]	# of Total EndPoints	Closed-Loop Latency [ms]	
		Mean ± SD	Max
1	2	1.14 ± 0.03	**1.52**
2	2	2.21 ± 0.04	**2.60**
3	2	3.27 ± 0.10	**4.41**
5	2	5.44 ± 0.20	**6.12**
5	100	5.45 ± 0.24	**6.20**
10	2	10.65 ± 0.20	**12.10**
10	100	10.60 ± 0.22	**11.47**

Tests done on AMD Ryzen 9 7950X (4.50GHz) on 23 June 2023, fw v.1.7.0.12. For each row, the values were calculated from a sample of 11k CLL measurements.