The SAM format, CIGAR strings, and indels

When it comes to phasing two variants with sequencing data, we need to establish which alleles are present on the same read. Are the two variants phased AB/BA or AA/BB? To retrieve this information from a BAM file, we need to establish the two nucleotides present at the variant site for each read (pair) overlapping both SNPs.
There are various ways to accomplish this, but the most intuitive approach in Pysam – the pileupcolumn – does not offer access to read names, which are the identifying feature to tie nucleotides to the same read or fragment. An alternative way is therefore the fetch() operation, which retrieves reads overlapping a specified interval from the BAM file (see Workshop 6).
Once a read has been retrieved, its starting position relative to the SNP site can be used to determine the corresponding nucleotide at the SNP. However, the start of the alignment is not necessarily the first base of the read sequence! Indels can offset the start of the alignment relative to the SNP site, so this has to be accounted for in your program. Another edit operation is the so-called soft-clip, which postpones or abort aligning a read entirely for several bases. All of this information is available in the SAM record of the read (the read object in Pysam).
The Samtools specification has detailed explanations of the SAM format (https://samtools.github.io/hts-specs/SAMv1.pdf), but this document explains some of the relevant points for your convenience.
Consider the following toy example:
Ref: AACCTTGTTCCAA
Read: CCTGT
Aligning the read to the reference might deem it to match position 3. Position 7 in the reference is a known SNP (marked in red), which we would like to genotype or phase. Since the read maps to positon 3, its 5th character (1-based) should contain the variant. Obviously, the 5th base is a T and not the correct character to read from the sequence. Since, the read aligned with an insertion prior to the reference base of interest (an inserted T), we have to retrieve the 4th character (the G) as the correct nucleotide for the SNP.
A soft-clip has a similar effect: None of the clipped bases are aligned to the reference.
Example Read: AAAGTT
This read might get aligned only from the 4th base onward and its position set to 7 in the reference. The three leading As get soft-clipped. The correct nucleotide, however, is not the first, but the fourth.
A deletion has the opposite effect.
There are several ways to determine whether indels are affecting the alignment positon or not in Pysam. One of them is to parse the CIGAR string. The CIGAR string is an abbreviated form of an edit transcript. It consists of numbers followed by letters. The number indicating the length of the edit, the letter the type. For example, 3D indicates three deleted bases in a row, 4S, four soft-clipped bases (can only occur at start or end of read). Most of the bases will be included in the M type, which stands for match. Note that match here stands for a diagonal operation in the Smith-Waterman algorithm, so includes mismatching as well as matching bases, but, importantly, not shift in the alignment.

This information can be conveniently accessed as pairs (length, type) in Pysam via the *cigartuples* field for an aligned segment.
If you use the fetch operation and then parse the *cigartuples* to establish which base to retrieve from the reads sequence, you need to account for all indels and soft-clips *before* the relevant reference base. Conveniently, all reads are stored as if they aligned to the forward strand in a BAM file. That means, the read sequence has been reverse complemented if the read aligned to the reverse strand, and the CIGAR string is presented accordingly.