We consider the clock synchronization problem in the (discrete) beeping model: Given a network of n nodes with each node having a clock value $\delta(v) \in \{0,\ldots T-1\}$, the goal is to synchronize the clock values of all nodes such that they contain the same value in any round.

We further assume \emph{asynchronous activations} for all nodes, i.e., nodes start their protocol in different rounds, as the problem would be trivial otherwise.
We give an asymptotically optimal algorithm that runs in $4D + \Bigl\lfloor \frac{D}{\lfloor T/4 \rfloor} \Bigr \rfloor \cdot (T \mod 4) = O(D)$ rounds, where D is the diameter of the network.

Once all nodes are in sync, they beep at the same round every T rounds.
The algorithm drastically improves on the $O(n D)$-bound of \cite{firefly_sync}.
Our algorithm is very simple as nodes only have to maintain 2 bits in addition to the $O(\log T)$ bits needed to maintain the clock.

Furthermore we investigate the complexity of \emph{self-stabilizing} solutions for the clock synchronization problem: We first show a lower bound of $\Omega(\max\{T,n\})$ rounds on the runtime for any such protocol.

Afterwards we present a protocol that runs in $O(\max\{T,n\})$ rounds using at most $O(\log(\max\{T,n\}))$ bits at each node, which is asymptotically optimal with regards to both, runtime and memory requirements.