## P-WAVE MODEL INTRODUCTION

The P-wave model used in this project is based on a transfer function framework that synthesizes electrocardiogram (ECG) singles using three parallel linear subsystems, one for each of the principal ECG waves: P, QRSQ, and T[1]. In a physiological ECG, the P wave is generated by the depolarization of the atria, occurring before ventricular activation and thus before the QRS complex. Since it reflects atrial conduction and pacemaker behavior, accurate modeling of the P wave is essential for capturing sinus rhythm and many atrial conditions. Many detailed biophysical heart models can reproduce this behavior, but they are very complex and expensive to stimulate. The idea in [1] is to replace those large models with a much smaller linear system that still produces waveforms that look like clinical ECGs. In this setup, the P wave is treated as its own second order system, tuned directly from ECG data, so we can adjust atrial behavior in a clear and controlled way [1]

Within this framework, the P-wave is represented by the first term of a three branch transfer function written in the Laplace domain.The complete heartbeat model is expressed as a sum of three delayed second order transfer functions, but the P-wave can be written as

$$HB(s) = \sum_{i=1}^{3} k_i e^{-r_i s} \frac{a_i s - b_i}{s^2 + c_i s + d_i}, \quad (12)$$

Which is the $i = 1$ term of Equation (12) in [1]. Here, $k_1$ is the gain associated with the P-wave subsystem, $r_1$ is its time delay relative to the excitation impulse, and the coefficients $a_1, b_1, c_1$ and $d_1$ define the location of one zero and a pair of complex conjugate poles in the transfer function. Since this is a proper second order system with a first order numerator, its impulse response exhibits the damped, asymmetric shape that is typical of atrial depolarization. In contrast to a standard second order form, the non standard representation used here explicitly includes the numerator zero, allowing more flexible control of the initial slope and skewness of the waveform [1].

Each parameter in the P-wave transfer function alters a specific, visible feature of the waveform. The gain $k_1$ stretches the P-wave up or down without changing its width, so it controls the amplitude, which in real signals is usually on a few tenths of a millivolt. The delay $r_1$ shifts the entire P-wave in time and is used to match the observed PR interval (the time from the start of the P-wave to the start of the QRS complex) [1]. The coefficient $a_1$ sets the location of the zero. Moving this zero changes the initial slope and asymmetry between the upstroke and the downstroke of the P-wave, so it can model slower or faster atrial conduction. The pair $b_1$ and $d_1$ are linked to the natural frequency of the underlying second order oscillator. Changing them makes the P-wave narrower or wider in time. The coefficient $c_1$ controls the damping: higher damping makes the waveform quickly settle back to baseline with little ringing, while lower damping would allow small oscillations after the peak.

Rodríguez-Abreo et al. chose a non standard form for these second order transfer functions, meaning that the numerator and denominator coefficients are left free instead being written only in terms of natural frequency and damping ratio [1]. This choice is important for the P-wave model. Since the zero location is an independent variable parameter, the model can fine tune the skew and tilt of the P wave in addition to its basic width and height. That extra flexibility is useful because real P-waves are often slightly asymmetric and can vary in shape with different atrial conditions. In other words, the non-standard form makes it easier to match the subtle details of the P-wave without disturbing the QRS or T-wave branches.

To find acceptable parameter values for the P-wave model, the authors use a generic algorithm rather than manual tuning [1]. The algorithm starts with many random guesses for the parameters $a_1, b_1, c_1, d_1, k_1$ and $r_1$, simulates the heartbeat for each guess, and measures how close the result is to a target ECG beat. The best guesses are kept and combined to form a new generation of parameters, and P-wave branch is separated from the other branches, the algorithm can automatically adjust its parameters to better fit the small atrial deflection while the QRS and T-wave model that is consistent with the full ECG but still controlled by a small set of clear parameters.

The heartbeat formula in Equation (12) describes only one beat. To create a continuous ECG over many beats, the model uses Equation (13), which adds a periodic input term. For the P-wave branch, this becomes

$$ECG(s) = \sum_{i=1}^{3} \frac{k_i e^{-r_i s}}{1 - e^{\frac{s}{f}}} \frac{a_i s - b_i}{s^2 + c_i s + d_i}, \quad (13)$$

Where $f$ is the frequency of the impulse train (beats per second) [1]. The factor $1/(1 - e^{-s/f})$ is the Laplace transform of an infinite sequence of impulses separated by a time $T = 1/f$. This means the P-wave response of Equation (12) is simply repeated once every beat, always with the same shape and delay, while the value of $f$ sets the heart rate. A slow heart rate uses a small $f$, so the P-waves are farther apart in time. A fast heart rate uses a larger $f$, bringing the P-waves closer together. The important point is that changing $f$ does not change the shape of the P wave itself. Shape is controlled only by $a_1, b_1, c_1, d_1, k_1$ and $r_1$, while beat to beat timing is controlled by $f$.

Since each ECG component is modeled as its own branch, the P-wave subsystem can also be modified or removed independently, which is useful for representing atrial arrhythmias. It is a controlled, interpretable handle on atrial behaviors within a simple linear structure that still captures the essential features of human ECG signals.