This is a **numerical model of a rabbit sinoatrial node pacemaker cell** that explicitly simulates a **network of individual ryanodine receptor (RyR) clusters** and all the major membrane currents. It produces **time series of action potentials and calcium signals**, plus images of the RyR network and local Ca.

---

1. What this model is

"A numerical model of a rabbit sinoatrial node cell featuring a functional network of individual RyRs. The model is written in Delphi (version 12.2) for Windows using Visual Component Library (VCL)." GitHub

Scientifically, it's the code used in the preprint

"A New Fight-or-Flight Pacemaker Mechanism via Ryanodine Receptor Abundance and Superclustering" (2025)

Authors: Valeria Ventura Subirachs, Tagirova SyevdaAlexander V Maltsev, Dongmei Yang, Edward G Lakatta, Michael D Stern, Victor A Maltsev

https://www.biorxiv.org/content/10.1101/2025.11.26.690788

The paper uses dSTORM super-resolution imaging of RyR clusters plus this model to study how **cluster size, number, and network topology control local Ca release and pacemaker rate**.

National Institute on Aging, NIH, Biomedical Research Center, 251 Bayview Blvd. Baltimore, MD 21224

Corresponding author: Victor A. Maltsev. Email: maltsevvi@mail.nih.gov

Thus, it is a **biophysically detailed SANC model with an explicit RyR cluster network + full membrane electrophysiology**.

---

2. How the model is structured (internals in plain language)

2.1 Geometry and RyR network (CRUs)

Inside uParams.pas and uSANCmodel.pas the code:

- Defines **cell geometry**:

    - Cell length, diameter, surface area S_cell_mkm2, volumes of:
        - **subspace** (narrow space under the membrane),
        - **ring cytosol** (peripheral shell),
        - **core cytosol**, and
        - **SR compartments**: free SR (fSR) and junctional SR (jSR).
- Discretizes the cell surface into a 2D grid (xGridLen, yGridLen) of "voxels".
- Builds a set of **Ca Release Units (CRUs)**, each being a **cluster of RyRs**:
    - Each CRU has:
        - Number of RyRs NRyR,
        - A set of subvoxels (voxels),
        - Local jSR Ca, subspace Ca, buffers, etc.
    - Set_CRUs places these CRUs on the grid using a size distribution that can be:
        - Loaded from experimental dSTORM / SIM RyR cluster data (through file input & ReadRySizeDistribution_dSTORM), and then scaled to match a desired total RyR density. MDPI+1

The end result is a **network of CRUs** whose sizes and spatial arrangement mimic experimentally measured RyR cluster networks in rabbit SANCs. PubMed+1

2.2 Calcium handling

The code sets up several calcium "fields" as 2D arrays:

- ArSubCyt – Ca in **subspace** (just under the membrane).
- ArRingCyt – Ca in **ring cytosol** (peripheral shell).
- ArRingFSR – Ca in **ring free SR**.
- Plus scalar core cytosol/core SR variables.

Then each timestep:

1. **RyR Ca release**
    - Each CRU has a stochastic release process:

        - Opening probability depends on **local subspace Ca** and parameters CRU_Casens, CRU_ProbConst, CRU_ProbPower.
        - Unit RyR current is set by Iryr_at_1mM_CaJSR and the Ca gradient between jSR and subspace.
    - The code keeps track of a CRU "spark" current Ispark and terminates it when current falls below a threshold related to Ispark_Termination_scale (representing termination by jSR depletion rather than arbitrary inactivation, in line with modern RyR physiology). Rockefeller University Press+1

2. **SERCA uptake & SR refilling**
    - SERCA flux is tabulated (Set_SERCA_tabulation) as a function of cytosolic and SR Ca, then applied to fSR and cytosol arrays.
    - This refills the SR and creates a **refractory period** for each CRU (time needed to reload jSR), which is crucial for the emergent clock-like periodicity of local Ca release. Rockefeller University Press+1
3. **Ca buffers**
    - Calsequestrin buffering in jSR.
    - Calmodulin buffering in subspace, ring, and core (Do_BufferCM, Do_Buffer_CQ_jSR).
4. **Diffusion**
    - 2D lateral diffusion of Ca within subspace, ring cytosol, and ring SR (Do_Diffusion2D).
    - Radial diffusion between subspace ↔ ring ↔ core, and fSR ↔ jSR.
    - This allows **Ca waves / local Ca releases (LCRs)** to propagate across the CRU network, not just as isolated sparks.

2.3 Membrane electrophysiology

The membrane side is in Set_electrophysiology and a set of find_I… procedures:

- Reversal potentials:
    - EK, ENa, EKs via Nernst equation using Ko, Ki, Nao, Nai.
- Total cell capacitance:

- Cm = C_pF_per_mkm2 * S_cell_mkm2.

Currents included:

- **ICaL** (L-type Ca current): split into Cav1.2 and Cav1.3 components (gCaL * fractions).
  - Each CRU has its local Cav1.2 / Cav1.3 conductance (CRUgCav12, CRUgCav13).
  - Local Ca-dependent inactivation via fCa.
- **ICaT** + background Ca current **IbCa**.
- **INCX** – electrogenic Na–Ca exchanger, using local subsarcolemmal Ca; this is the main coupling between the calcium clock (LCRs) and the membrane clock. Rockefeller University Press+1
- **If** ("funny" current) – mixed Na/K inward current, with gating variable y and half-activation V_Ih12.
- **IKr** and **IKs** – delayed rectifier K currents.
- **Ito** and **Isus** – transient outward and sustained K currents.
- **INaK** – Na/K pump.

At each timestep:

1. All ionic currents are computed: ICaL, ICaT, IKr, IKs, Ito, Isus, If, INaK, INCX, IbCa, plus any applied Iext.
2. Total membrane current:

$$I_m = ICaL + ICaT + IKr + IKs + Ito + Isus + I_f + INaK + INCX + IbCa + I_{ext}$$

3. Membrane potential is advanced via:

$$\frac{dV_m}{dt} = -\frac{I_m}{C_m}$$

implemented as:

dVdt := -Im / Cm;

Vm := Vm + dVdt * TimeTick;

2.4 β-adrenergic stimulation

There is a β-AR block (fight-or-flight mode):

- When SimTime exceeds bAR_onset and bAR_stimulation_apply is true, it scales:
    - gCaL (L-type Ca conductance),
    - Pup (SERCA pump rate),
    - gKr (rapid delayed rectifier),
    - and shifts V_Ih12 (If activation).

Those changes mimic PKA-mediated phosphorylation under β-adrenergic stimulation, and the associated papers show how they modulate pacemaker rate via the RyR network.

---

3. What it *does* in practice

When you run Model_Run (the main integration routine), the code:

1. Initializes parameters from the GUI (cell size, RyR density, β-AR settings, time step, simulation duration, etc.).
2. Builds the CRU network and sets diffusion / SERCA tables.
3. Sets initial Ca concentrations, gating variables, and Vm.
4. Enters a repeat … until time loop:
    - Advances SimTime by TimeTick.
    - Optionally applies β-AR parameter changes.
    - Computes all membrane currents and updates Vm.
    - Updates CRU states, Ca buffers, diffusion, and SERCA flux.
    - Accumulates Ca changes into the arrays.
    - Every few steps:
        - Plots Vm vs time on the fPlotVm form.
        - Optionally updates a 2D image of Ca or RyR network on fImage2.
        - Updates textual status (simulation time, compute time, etc.).

This gives you a **self-consistent simulation of SANC automaticity**, where:

- Local Ca release dynamics (sparks → LCRs) emerge from the RyR network.

- Those LCRs drive INCX.
- INCX contributes to diastolic depolarization.
- The membrane clock (ionic currents) and Ca clock (RyR/SR system) become coupled and determine beating rate.

---

4. What are the outputs?

From the code:

1. **Action potential trace (Vm vs time)**
    - Displayed on the TfPlotVm form (Chart1.Series1).
    - Can be exported as a text file with two columns: time and Vm (using the "Save as TXT" button in UplotVm.pas), for further analysis.
2. **2D images / movies of the CRU network and Ca distribution**
    - TfImage2 displays a bitmap of:
        - RyR cluster locations and sizes, or
        - jSR Ca, or
        - subspace Ca, depending on the chosen mode.
    - It uses cell geometry and ArCRU, ArSubCyt, etc., to color-code the image.
3. **Parameters and CRU statistics**
    - Procedures like WriteParams and file export blocks write out:
        - Total number of CRUs,
        - Total number of RyRs, mean cluster size, etc.,
        - Cell geometry and simulation parameters.
    - Useful for linking morphology → function, as in the 2024 Cells paper. MDPI+1

So the "output" is **time series + spatial maps + summary statistics** of how the RyR network and Ca/membrane dynamics behave under chosen conditions.