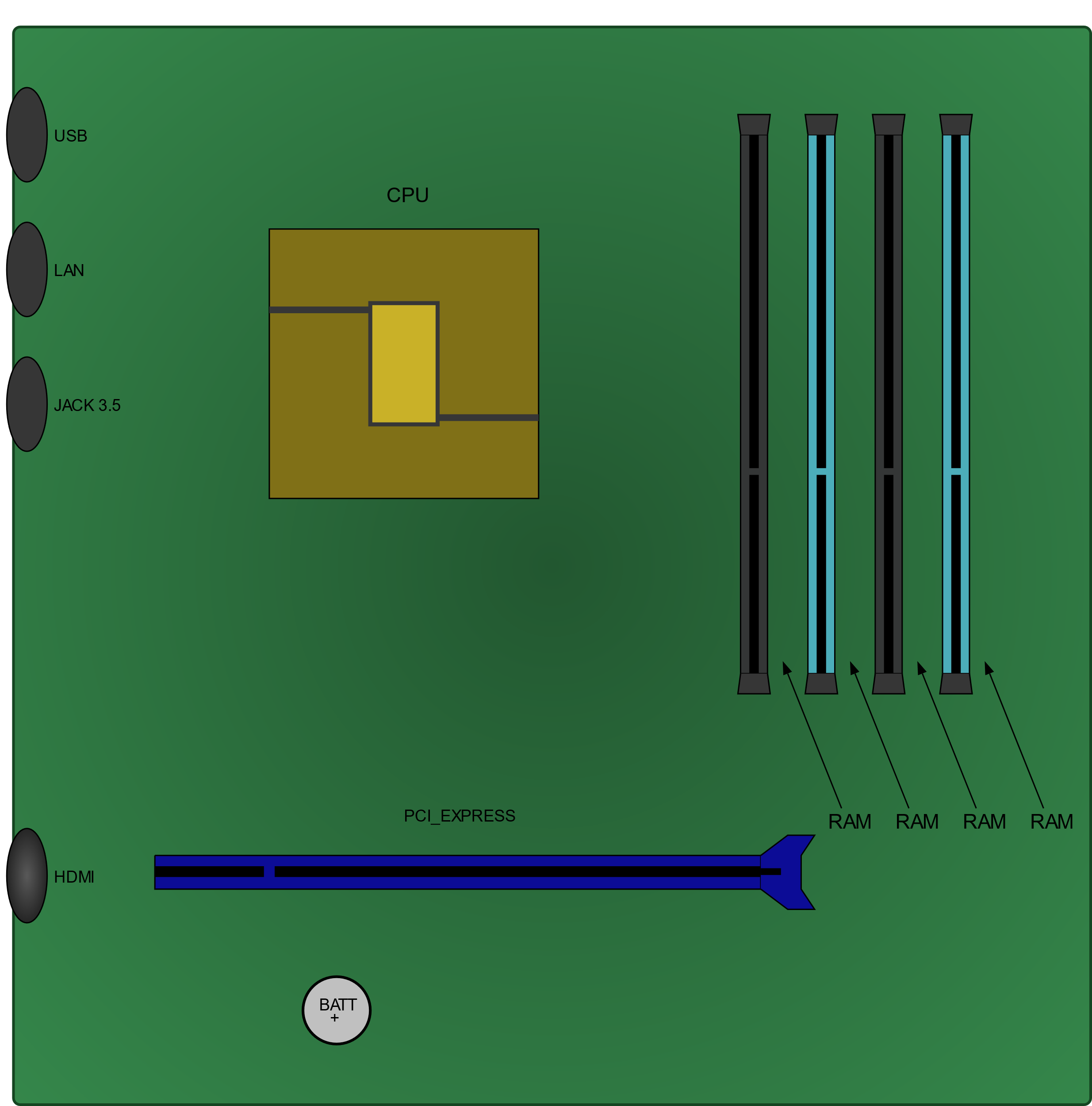
Tudorache Alexandru-Theodor

Grupa 342

Tema 2 TPAG

1.

Scena 2D pe care am ales sa o reprezint este o placa de baza. Rezultatul final arata astfel:



Pentru reprezentare au fost folosite 11 elemente distincte:

1. radialGradient (pentru fundal si pentru portul HDMI)

<!-- Fundal -->

<defs>

                <radialGradient

                    id="grad-background"

                    cx="50%"

                    cy="50%"

                    r="90%"

                    fx="50%"

                    fy="50%"

                    >

                    <stop

                        offset="0%"

                        style="stop-color: #225730; stop-opacity: 1"

                    />

                    <stop

                        offset="100%"

                        style="stop-color: #3a9452; stop-opacity: 1"

                    />

                </radialGradient>

</defs>

1. rect (pentru contur)

            <!-- Contur -->

            <rect

                x="10"

                y="20"

                rx="5"

                width="800"

                height="800"

                style="

                    fill: url(#grad-background);

                    stroke: #154521;

                    stroke-width: 2;"

            />

1. circle (pentru baterie)

<!-- Baterie -->

<circle

                r="25"

                cx="250"

                cy="750"

                stroke="black"

                stroke-width="2"

                fill="silver"/>

1. text (pentru tot textul din imagine)
2. tspan (pentru “+”-ul de pe baterie)

            <!-- Baterie -->

            <text x="237" y="750" fill="black" style="font-size: 12px;">

                BATT

                <tspan x="245" y="760">+</tspan>

            </text>

1. marker (pentru sagetile care arata spre slot-urile memoriei RAM)

                <!-- Sageata slot RAM -->

                <marker

                  id="arrowhead"

                  markerWidth="10"

                  markerHeight="7"

                  refX="0"

                  refY="3.5"

                  orient="auto"

                >

                  <polygon points="0 0, 10 3.5, 0 7" />

                </marker>

1. line (pentru sagetile care arata spre slot-urile memoriei RAM, pentru slot-urile memoriei RAM si PCI-EXPRESS si soclul CPU-ului)

                    <!-- Sageata slot RAM -->

                    <line

                        x1="675"

                        y1="600"

                        x2="635"

                        y2="500"

                        stroke="#000"

                        stroke-width="1"

                        marker-end="url(#arrowhead)"

                    />

1. path (pentru clips-urile de la slot-urile memoriei RAM si PCI-EXPRESS)

<!-- Clip PCI-EXPRESS -->

<path

            stroke="black"

            fill="#0c0c96"

            d="M 565 635 l20 -15 l20 0 l-10 15 l0 25 l10 15 l-20 0 l-20 -15"/>

1. polyline (pentru slot-urile memoriei RAM)

<!-- Slot RAM -->

<polyline

                    points="600,100 600,500 620,500 620,100 600,100"

                    style="fill: #343636; stroke: black; stroke-width: 1"

                    />

1. polygon (pentru sagetile care arata spre slot-urile memoriei RAM si pentru soclul CPU-ului)

<!-- Soclu CPU -->

<polygon points="200 170, 400 170, 400 370 200 370" fill="#807017" stroke="black" stroke-width="1"/>

1. ellipse (pentru porturile din stanga placii de baza: USB, LAN, JACK 3.5, HDMI)

                    <!-- Slot -->

                    <ellipse

                    cx="20"

                    cy="100"

                    rx="15"

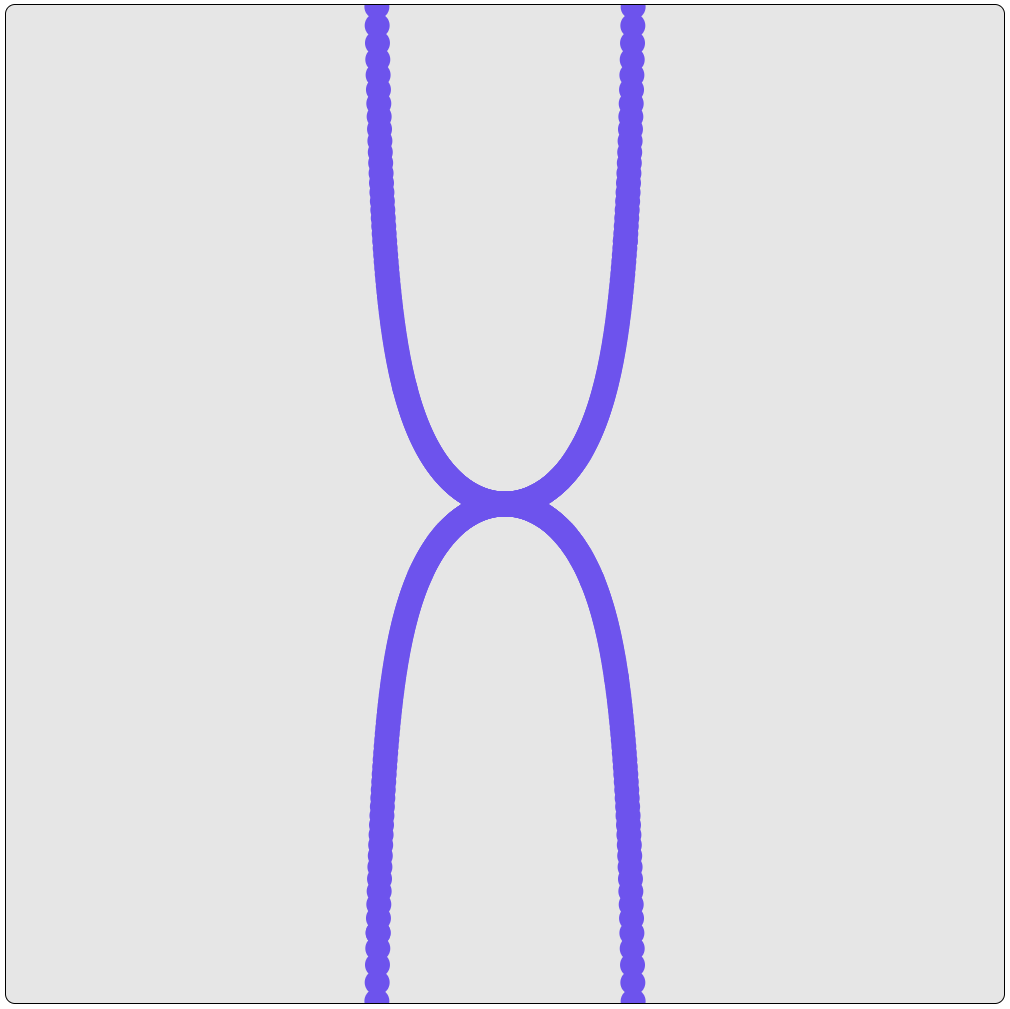
                    ry="35"

                    style="fill: #363636; stroke: black; stroke-width: 1"

                  />

2.

Curba pe care am ales sa o reprezint este [*Kappa Curve*](https://en.wikipedia.org/wiki/Kappa_curve). Rezultatul final arata astfel:



Metoda prin care am reprezentat curba este prin afisarea a 720 de puncte (am observant ca un numar mai mare de puncte nu aduce niciun beneficiu din punct de vedere vizual).

Aceste puncte au fost calculate folosind ecuatiile parametrice

,

unde constanta *a* a fost aleasa ca fiind .

Avand o fereastra de 800x800, pentru a avea coordonate in intervalul [-400, 400] in loc de [0, 800], a fost nevoie de centrarea viewbox-ului.

<svg width="1000" height="1000" viewBox="-400 -400 800 800" id="svg">

    <circle id="circle" r="10" fill="#6d53ed"/>

</svg>

Pentru afisarea punctelor a fost creat un element *circle* (fara coordonate) in interiorul ferestrei care apoi a fost clonat de 720 de ori. Apoi fiecarei clone i-au fost asignate coordonate astfel incat sa rezulte curba dorita.

<script>

    let a = 75 \* Math.sqrt(2);

    for (let t = 0; t < 720; t++) {

          let x = a \* Math.sin(t);

          let y = a \* Math.sin(t) \* Math.tan(t);

          const initial\_circle = document.getElementById("circle")

          const point = initial\_circle.cloneNode(true)

          point.setAttribute("cx", x);

          point.setAttribute("cy", y);

          document.getElementById("svg").appendChild(point);

    }

</script>