

```

package net.minecraft.src;

import java.io.*;

// Created by MoareAI
// Contains data handled by the ModLoader

public class mod_DigitalFunctions extends BaseMod
{
    //ModLoader actions
    public mod_DigitalFunctions()
    {
        RenderGateID = ModLoader.getUniqueBlockModelID(this, false);
        ModLoader.getLoadedMods();
        ModLoader.RegisterTileEntity(net.minecraft.src.TileEntityDigitalMisc.class,
        "DigitalFunctions");

        //Item Icons
        if (!ReduceIcon)
        {
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDToggle.png",
            IconToggle);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDRS.png", IconRS);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDClock.png",
            IconClock);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDHalfAdder.png",
            IconHalfAdder);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDJK.png", IconJK);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDPulseGen.png",
            IconPulseGen);
            ModLoader.addOverride("/gui/items.png", "/MoareAI/Items/MDCounter.png",
            IconCounter);
        }

        ModLoader.AddName(ItemDigitalMisc, "Digital Functions");

        for(int damage = 0; damage < 8; damage++)
        {
            ModLoader.AddName(new ItemStack(ItemDigitalMisc, 1, damage),
            ItemDigitalMisc.getItemNameIS(new ItemStack(ItemDigitalMisc, 1, damage)));
        }

        if (EffectiveCrafting)
            CraftCost = 1;
        else
            CraftCost = 0;

        //Recipe for Toggle
        ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, CraftCost*5+1, 1), new
        Object[] {
            " Z ", "#XY", Character.valueOf('#'), Item.redstone,
            Character.valueOf('X'), Block.sand, Character.valueOf('Y'), Block.torchRedstoneActive,
            Character.valueOf('Z'), Block.lever
        });

        //Recipe for RS
        ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, CraftCost*2+1, 2), new
        Object[] {
            "XY#", "# #", "#YX", Character.valueOf('#'), Item.redstone,
            Character.valueOf('X'), Block.sand, Character.valueOf('Y'), Block.torchRedstoneActive
        });

        //Recipe for Pulse Clock
        ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, CraftCost*5+1, 3), new

```

```

Object[] {
    "#XY", Character.valueOf('#'), Item.pocketSundial, Character.valueOf('X'),
    Block.sand, Character.valueOf('Y'), Block.torchRedstoneActive
});

//Recipe for Half Adder
ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, 1, 4), new Object[] {
    "#", "X", Character.valueOf('#'), new
ItemStack(mod_LogicalGates.ItemLogicalGates, 1, 5), Character.valueOf('X'), new
ItemStack(mod_LogicalGates.ItemLogicalGates, 1, 4)
});

//Recipe for JK Flip Flop
ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, 1, 5), new Object[] {
    "#X", Character.valueOf('#'), new ItemStack(ItemDigitalMisc, 1, 2),
    Character.valueOf('X'), new ItemStack(ItemDigitalMisc, 1, 1)
});

//Recipe for Pulse Generator
ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, 1, 6), new Object[] {
    " #", "YX ", Character.valueOf('#'), new
ItemStack(mod_LogicalGates.ItemLogicalGates, 1, 4), Character.valueOf('X'), new
ItemStack(mod_LogicalGates.ItemLogicalGates, 1, 2), Character.valueOf('Y'),
Item.redstoneRepeater
});
ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, 1, 6), new Object[] {
    " #", "X ", Character.valueOf('#'), new ItemStack(ItemDigitalMisc, 1, 2),
    Character.valueOf('X'), Item.redstoneRepeater
});

//Recipe for Counter
ModLoader.AddRecipe(new ItemStack(ItemDigitalMisc, 1, 7), new Object[] {
    "#X", Character.valueOf('#'), new ItemStack(ItemDigitalMisc, 1, 4),
    Character.valueOf('X'), new ItemStack(ItemDigitalMisc, 1, 1)
});
}

public String Version()
{
    return "1.6.6";
}

//Block declaration
public static final Block BlockDigitalMisc;

//Render type declaration
public static int RenderGateID;

//Icon declaration
public static int IconToggle;
public static int IconRS;
public static int IconClock;
public static int IconHalfAdder;
public static int IconJK;
public static int IconPulseGen;
public static int IconCounter;

//Item declaration
public static Item ItemDigitalMisc;

//Properties
public static final File cfgfile;
public static int PulseGenLength = 2;
public static int ClockTime = 16;

```

mod_DigitalFunctions.java

```
public static int DefaultCount = 5;
public static boolean EffectiveCrafting = true;
private int CraftCost;
public static boolean ReduceIcon = false;
public static int GUIStep = 5;
public static int DigitalFunctionsBlockID = 201;
public static int DigitalFunctionsItemID = 457;
public static String BlockTexture = "/MoareAI/Blocks/DigitalFunctions.png";

//Block and Item data
static
{
    //The configuration file
    cfgfile = new File(Minecraft.getMinecraftDir(), "/config/MoareAI's Digital
Functions.cfg");
    ConfigDigitalFunctions.Configure(cfgfile);

    if (!ReduceIcon)
    {
        IconToggle = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconRS = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconClock = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconHalfAdder = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconJK = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconPulseGen = ModLoader.getUniqueSpriteIndex("/gui/items.png");
        IconCounter = ModLoader.getUniqueSpriteIndex("/gui/items.png");
    }

    BlockDigitalMisc = new
BlockDigitalMisc(DigitalFunctionsBlockID).setBlockName("MiscDigital");

    ItemDigitalMisc = new ItemDigitalMisc(DigitalFunctionsItemID,
BlockDigitalMisc).setItemName("MiscDigital");

}

//The rendering for the gates
public boolean RenderWorldBlock(RenderBlocks renderblocks, IBlockAccess
iblockaccess, int x, int y, int z, Block block, int l)
{
    Tessellator tessellator = Tessellator.instance;
    if(block instanceof ICustomTextureBlock)
    {
        tessellator.draw();
        tessellator.startDrawingQuads();
        GL11.glBindTexture(3553 /*GL_TEXTURE_2D*/,
ModLoader.getMinecraftInstance().renderEngine.getTexture(((ICustomTextureBlock)block).g
etTextureFile()));
    }
    if(l == RenderGateID)
    {
        TileEntityDigitalMisc tileentity =
(TileEntityDigitalMisc)iblockaccess.getBlockTileEntity(x, y, z);
        int meta = iblockaccess.getBlockMetadata(x, y, z);
        int rotation = tileentity.Rotation;
        float light = block.getBlockBrightness(iblockaccess, x, y, z);
        float white = 1.0F;
        //Gate Model
        RenderGate.RenderGateModel(tessellator, block, iblockaccess, x, y, z,
white, light, rotation);

        if (meta > 0)
        {
            float red = 0.5F;
```

```

//Output A
red = 0.5F;
if (tileentity.OutputA)
    red = 1.0F;
RenderGate.RenderOutput(tessellator, block, x, y, z, red, light,
rotation, 0);
RenderGate.RenderConnectionPiece(tessellator, block, x, y, z, red,
light, rotation, 0);

//Output B
if (meta == 2 || meta == 4)
{
    red = 0.5F;
    if (tileentity.OutputB)
        red = 1.0F;
    RenderGate.RenderOutput(tessellator, block, x, y, z, red, light,
rotation, 1);
    RenderGate.RenderConnectionPiece(tessellator, block, x, y, z, red,
light, rotation, 1);
}

//Input Connection
if (tileentity.OutputA || tileentity.OutputB)
    red = 1.0F;
RenderGate.RenderInputConnection(tessellator, block, x, y, z, red,
light, rotation);

//InputA
float cyan = 0.0F;
if (meta == 5 || meta == 7)
{
    red = 0.5F;
    if (tileentity.InputA)
    {
        red = 1.0F;
    }
    if (!tileentity.ConnectionA)
    {
        cyan = 0.5F;
    }
    RenderGate.RenderInput(tessellator, block, x, y, z, red, cyan, light,
rotation, 3);
}

//InputB
cyan = 0.0F;
red = 0.5F;
if (tileentity.InputB)
{
    red = 1.0F;
}
if (!tileentity.ConnectionB)
{
    cyan = 0.5F;
}
RenderGate.RenderInput(tessellator, block, x, y, z, red, cyan, light,
rotation, 0);

//InputC
if (meta == 2 || meta == 4 || meta == 5 || meta == 7)
{
    cyan = 0.0F;

```

```

        mod_DigitalFunctions.java

        red = 0.5F;
        if (tileentity.InputC)
        {
            red = 1.0F;
        }
        if (!tileentity.ConnectionC)
        {
            cyan = 0.5F;
        }
        RenderGate.RenderInput(tessellator, block, x, y, z, red, cyan,
light, rotation, 1);
    }
}
}
if(block instanceof ICustomTextureBlock)
{
    Tessellator tessellator1 = Tessellator.instance;
    tessellator1.draw();
    tessellator1.startDrawingQuads();
    GL11.glBindTexture(3553 /*GL_TEXTURE_2D*/,
ModLoader.getMinecraftInstance().renderEngine.getTexture("/terrain.png"));
    return true;
}
return false;
}
}
}

```