Project Thesis
Intelligent Hydroponic System

Arpit Sharma
P Number: P2613237,
Intelligent Systems and Robotics MSc.

Under the supervision of
Dr. Aboozar Taherkhani, De Montfort University, England.

Acknowledgements

I would like to thank Dr. Aboozar Taherkhani for being my supervisor for the project. It was because of his incessant guidance and valuable feedback throughout the period that such a complex system could be created for the project. Despite the COVID-19 outbreak, he was always available via MS Teams meetings and email. His deep knowledge about the hydroponic systems and machine learning made it possible to make the hydroponic system, intelligent by implementing and testing the state of art, TinyML, and IOT on it. To conclude, I am very grateful to my family and friends for their love and support during this tough academic year.

The final code used in the project

1 Android App Code

The code used for building the app using the Android Studio IDE is split in different files for readability as follows:

1.1 ControlPanel.java

```
package com.example.snoee.myapplistview.activities;

import android.annotation.SuppressLint;
import android.content.SharedPreferences;
import android.graphics.drawable.GradientDrawable;
import android.os.Build;
import android.support.annotation.RequiresApi;
import android.support.design.widget.TabLayout;
import android.support.v4.content.ContextCompat;
import android.support.v4.view.ViewPager;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;
import android.widget.Toast;

import com.example.snoee.myapplistview.AutoCutDialogFrag;
import com.example.snoee.myapplistview.BaseFragment;
import com.example.snoee.myapplistview.fragments.ControlPanelFrag;
import com.example.snoee.myapplistview.adapters.ViewPagerAdapter;
import com.example.snoee.myapplistview.fragments.StatisticsFrag;
import com.example.snoee.myapplistview.interfaces.Communicator;
import com.example.snoee.myapplistview.fragments.DevFrag;
import com.example.snoee.myapplistview.R;
import com.example.snoee.myapplistview.fragments.WeatherFrag;

import org.eclipse.paho.client.mqttv3.IMqttDeliveryToken;
import org.eclipse.paho.client.mqttv3.MqttCallbackExtended;
import org.eclipse.paho.client.mqttv3.MqttException;
import org.eclipse.paho.client.mqttv3.MqttMessage;

import java.util.ArrayList;
import java.util.List;
import java.util.Objects;
import java.util.Set;

public class ControlPanel extends AppCompatActivity implements Communicator {
    MQTThelper mqttHelper;
    Toolbar toolbar;
```

```
TabLayout tabLayout;
ViewPager viewPager;
int[][] colors = {{R.color.
ColorPrimaryLight_controlPanel,
R.color.ColorPrimary_controlPanel,
R.color.ColorPrimaryDark_controlPanel,
R.color.ColorAccent_controlPanel},
        {R.color.ColorPrimaryLight_statistics,
        R.color.ColorPrimary_statistics,
        R.color.ColorPrimaryDark_statistics, R.color.ColorAccent_statistics}};

String schedule;
BaseFragment page;
List<String> itemsSelected;
Set<String> set;
String message;

public static final String MY_PREFS_NAME = "MyPrefsFileControl";
SharedPreferences.Editor editor;

@SuppressLint("CommitPrefEdits")
@RequiresApi(api = Build.VERSION_CODES.O)
@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_control_panel);
    toolbar = findViewById(R.id.control_panel_toolbar);
    editor = getSharedPreferences(MY_PREFS_NAME, MODE_PRIVATE).edit();
    setSupportActionBar(toolbar);
    Objects.requireNonNull(getSupportActionBar()).setDisplayHomeAsUpEnabled(true);

    viewPager = findViewById(R.id.viewPager_id);

    ViewPagerAdapter adapter = new ViewPagerAdapter(getSupportFragmentManager());
    adapter.addFragment(new ControlPanelFrag(), "CONTROL PANEL");
    adapter.addFragment(new StatisticsFrag(), "STATISTICS");

    viewPager.setAdapter(adapter);

    tabLayoutInit();
    startMqtt();

    itemsSelected = new ArrayList<>();

    Bundle bundle = getIntent().getExtras();

    if (bundle != null) {
        int i = bundle.getInt("functionKey");
        TabLayout.Tab tab = tabLayout.getTabAt(i);
```

```
            Objects.requireNonNull(tab).select();
            toolbarAndTabGradient(colors[i][2]);
            changeTabTextColor(bundle.getInt("functionKey"));
        }
    }

    private void tabLayoutInit() {
        tabLayout = findViewById(R.id.tabLayout_id);
        tabLayout.setupWithViewPager(viewPager);
        Objects.requireNonNull(tabLayout.getTabAt(0)).setIcon(R.drawable.control_panel_icon);
        Objects.requireNonNull(tabLayout.getTabAt(1)).setIcon(R.drawable.statistics_icon);
        tabLayout.addOnTabSelectedListener(new TabLayout.OnTabSelectedListener() {
            @Override
            public void onTabSelected(TabLayout.Tab tab) {
                changeTabTextColor(tab.getPosition());
                toolbarAndTabGradient(colors[tab.getPosition()][2]);
            }

            @Override
            public void onTabUnselected(TabLayout.Tab tab) {
//                Toast.makeText(ControlPanel.this,
// tab.getText() + " onTabUnselected ", Toast.LENGTH_LONG).show();
            }

            @Override
            public void onTabReselected(TabLayout.Tab tab) {
//                Toast.makeText(ControlPanel.this,
// tab.getText() + " onTabReselected", Toast.LENGTH_LONG).show();
            }
        });
    }

    private void startMqtt() {
        mqttHelper = new MQTThelper(getApplicationContext());
        mqttHelper.setCallback(new MqttCallbackExtended() {
            @Override
            public void connectComplete(boolean b, String s) {
            }

            @Override
            public void connectionLost(Throwable throwable) {
            }

            @Override
            public void messageArrived(String topic, MqttMessage mqttMessage) {
                page = (BaseFragment) getSupportFragmentManager()
                .findFragmentByTag("android:switcher:" + R.id.viewPager_id + ":"
                + viewPager.getCurrentItem());
                if (page != null) {
                    if (topic.equals(mqttHelper.batteryFeed)) {
```

```
                    page.setBattery(Integer.parseInt(mqttMessage.toString()));
                }
                if (topic.equals(mqttHelper.pumpSpeedFeed)) {
                    page.setSeekPin(Integer.parseInt(mqttMessage.toString()));
                }
                if (topic.equals(mqttHelper.pHFeed)) {
                    page.setphValue(Double.parseDouble(mqttMessage.toString()));
                }
                if (topic.equals(mqttHelper.statusFeed)) {
                    page.setphStatus(mqttMessage.toString());
                }
                if (topic.equals(mqttHelper.liquidLevelFeed)) {
                    page.setLiquidLevelTv(Double.parseDouble(mqttMessage.toString()));
                }
            }
        }

        @Override
        public void deliveryComplete(IMqttDeliveryToken iMqttDeliveryToken) {
        }
    });
}

public void pub(String topic, String message) {
    try {
        mqttHelper.mqttAndroidClient.publish(topic, message.getBytes(), 0, false);
    } catch (MqttException e) {
        e.printStackTrace();
    }
}

@Override
public void onDialogMessage(String tag, String msg) {
    message = msg;
    switch (tag) {

        case "PumpSliderFrag":
            pub(mqttHelper.pumpSpeedFeed, message);
            break;
        case "DevFrag":
            Toast.makeText(this, (message), Toast.LENGTH_SHORT).show();
            break;

        case "CityWeatherFrag":
            Toast.makeText(this, ("new city is " + message), Toast.LENGTH_SHORT).show();
            WeatherFrag.updateCity(message);
            break;

        case "SpeedPickerFrag":
            Toast.makeText(this, message + "", Toast.LENGTH_SHORT).show();
```

```
                schedule += "speed: " + message + " , ";
                break;

            case "SpeedPickerFromSetupFrag":
                Toast.makeText(getApplicationContext(), message + "", Toast.LENGTH_LONG).show();
                page = (BaseFragment) getSupportFragmentManager().findFragmentByTag
                ("android:switcher:" + R.id.viewPager_id + ":" + viewPager.getCurrentItem());
                if (page != null) {
                    page.setSeekPin(Integer.parseInt(message));
                }
                // pub(mqttHelper.pumpFeed, message);
                break;
            default:
                System.out.println("no match");
        }
    }

    @RequiresApi(api = Build.VERSION_CODES.JELLY_BEAN)
    void toolbarAndTabGradient(int colorA) {
        int[] colors = {ContextCompat.getColor(getApplicationContext(),
        colorA), ContextCompat.getColor(getApplicationContext(),
        R.color.Black)};
        //create a new gradient color
        GradientDrawable gd = new GradientDrawable(
                GradientDrawable.Orientation.LEFT_RIGHT, colors);
        gd.setCornerRadius(0f);
        //apply the button background to newly created drawable gradient
        toolbar.setBackground(gd);
        tabLayout.setBackground(gd);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.menu_settings_id) {
            Toast.makeText(getApplicationContext(), "Coming Soon!", Toast.LENGTH_SHORT).show();
//            AutoCutDialogFrag autoFrag = new AutoCutDialogFrag();
//            autoFrag.show(getFragmentManager(), "auto frag");
        } else if (id == R.id.menu_developer_id) {
            DevFrag devFrag = new DevFrag();
            devFrag.show(getFragmentManager(), "dev frag");
        } else if (id == R.id.item3_id) {
            Toast.makeText(getApplicationContext(), "item3 is selected", Toast.LENGTH_SHORT).show();
            finish();
```

```
        } else if (id == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);
    }

    public void changeTabTextColor(int customColor) {
        tabLayout.setTabTextColors(
                ContextCompat
                .getColor(getApplicationContext(), R.color.white),
                ContextCompat.getColor(getApplicationContext(), colors[customColor][3])
        );
    }

}
```

1.2 MainActivity.java

```
 package com.example.snoee.myapplistview.activities;

import android.content.Intent;
import android.content.SharedPreferences;
import android.support.annotation.NonNull;
import android.support.design.widget.NavigationView;
import android.support.v4.view.GravityCompat;
import android.support.v4.widget.DrawerLayout;
import android.support.v7.app.ActionBarDrawerToggle;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ListView;
import android.widget.Toast;

import com.example.snoee.myapplistview.AutoCutDialogFrag;
import com.example.snoee.myapplistview.adapters.CustomAdapter;
import com.example.snoee.myapplistview.fragments.DevFrag;
import com.example.snoee.myapplistview.R;
import com.example.snoee.myapplistview.interfaces.Communicator;

public class MainActivity
extends AppCompatActivity implements NavigationView.OnNavigationItemSelectedListener,
Communicator {
    Toolbar toolbar;
    DrawerLayout drawerLayout;
```

```
NavigationView navigationView;
SharedPreferences.Editor editor;
public static final String MY_PREFS_NAME = "MyPrefsFileMain";
static String dir = "";

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.navigation_drawer);

    toolbar = (Toolbar) findViewById(R.id.toolbar);
    setSupportActionBar(toolbar);

    editor = getSharedPreferences(MY_PREFS_NAME, MODE_PRIVATE).edit();
    SharedPreferences prefs = getSharedPreferences(MY_PREFS_NAME, MODE_PRIVATE);
    dir = prefs.getString("directions", "none");


    drawerLayout = (DrawerLayout) findViewById(R.id.drawer_layout);
    navigationView = (NavigationView) findViewById(R.id.navigation_view);

    navigationView.setNavigationItemSelectedListener(this);

    ActionBarDrawerToggle toggle
    = new ActionBarDrawerToggle(this, drawerLayout, toolbar, R.string.open_drawer,
    R.string.close_drawer);
    drawerLayout.addDrawerListener(toggle);

    toggle.syncState();

    ListView listView = findViewById(R.id.category_list);
    listView.setDividerHeight(15);
    listView.setAdapter(new CustomAdapter(this));
    listView.setOnItemClickListener(new AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
            String[] titles = MainActivity.this.getResources().getStringArray(R.array.titles);
            functionLauncher(titles[position]);
        }
    });
}

public void functionLauncher(String title) {
    Intent intent = new Intent(this, ControlPanel.class);
    intent.putExtra("directions", "");

    switch (title) {
        case "Control Panel":
            intent.putExtra("functionKey", 0);
            break;
```

```
            case "Statistics":
                intent.putExtra("functionKey", 1);
                break;
        }
        startActivity(intent);
    }

    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {

        int id = item.getItemId();

        switch (id) {
            case R.id.settings_id:
                Toast.makeText(getApplicationContext(), "Coming Soon!", Toast.LENGTH_LONG).show();
//                AutoCutDialogFrag autoFrag = new AutoCutDialogFrag();
//                autoFrag.show(getFragmentManager(), "auto frag");
                break;
            case R.id.status_check_id:
                Toast.makeText(getApplicationContext(), "status check", Toast.LENGTH_LONG).show();
                break;
            case R.id.developer_id:
                DevFrag devFrag = new DevFrag();
                devFrag.show(getFragmentManager(), "dev frag");
                break;
        }

        drawerLayout.closeDrawer(GravityCompat.START);
        return true;
    }

    @Override
    public void onBackPressed() {
        if (drawerLayout.isDrawerOpen(GravityCompat.START)) {
            drawerLayout.closeDrawer(GravityCompat.START);
        } else {
            super.onBackPressed();
        }
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        getMenuInflater().inflate(R.menu.menu_main, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        int id = item.getItemId();
        if (id == R.id.menu_settings_id) {
```

```
            Toast.makeText(getApplicationContext(), "Coming Soon!", Toast.LENGTH_SHORT).show();
//            AutoCutDialogFrag autoFrag = new AutoCutDialogFrag();
//            autoFrag.show(getFragmentManager(), "auto frag");
        } else if (id == R.id.menu_developer_id) {
            DevFrag devFrag = new DevFrag();
            devFrag.show(getFragmentManager(), "dev frag");
        } else if (id == R.id.item3_id) {
            Toast.makeText(getApplicationContext(), "item3 is selected", Toast.LENGTH_SHORT).show();
            finish();
        } else if (id == android.R.id.home) {
            finish();
        }
        return super.onOptionsItemSelected(item);
    }

    @Override
    public void onDialogMessage(String tag, String message) {
        switch (tag) {
            case "AutoCutDialogFrag":
                Toast.makeText(this, (message), Toast.LENGTH_SHORT).show();
            case "DedicateDirectionsDialogFrag":
                Toast.makeText(getApplicationContext(), message + "", Toast.LENGTH_LONG).show();
                editor.putString("directions", message + "");
                editor.apply();
                dir = message + "";
                break;
            default:
                System.out.println("no match");
        }
    }
}
```

1.3 MQTThelper.java

```
package com.example.snoee.myapplistview.activities;

import android.content.Context;
import android.util.Log;

import org.eclipse.paho.android.service.MqttAndroidClient;
import org.eclipse.paho.client.mqttv3.DisconnectedBufferOptions;
import org.eclipse.paho.client.mqttv3.IMqttActionListener;
import org.eclipse.paho.client.mqttv3.IMqttDeliveryToken;
import org.eclipse.paho.client.mqttv3.IMqttToken;
import org.eclipse.paho.client.mqttv3.MqttCallbackExtended;
import org.eclipse.paho.client.mqttv3.MqttClient;
import org.eclipse.paho.client.mqttv3.MqttConnectOptions;
import org.eclipse.paho.client.mqttv3.MqttException;
```

```
import org.eclipse.paho.client.mqttv3.MqttMessage;

public class MQTThelper {
    public MqttAndroidClient mqttAndroidClient;
    final String serverUri = "tcp://io.adafruit.com:1883";


    String clientId;
    final String pumpSpeedFeed = "arpitsmscproject/feeds/pumpSpeed";
    final String batteryFeed = "arpitsmscproject/feeds/battery";
    final String pHFeed = "arpitsmscproject/feeds/pH";
    final String statusFeed = "arpitsmscproject/feeds/pHStatus";
    public String liquidLevelFeed= "arpitsmscproject/feeds/liquidLevel";
    final String username = "arpitsmscproject";
    final String password = "aio_GPgg95Ao3NOndl8EpBI9k2E3HxNJ";

    public MQTThelper(Context context) {
        clientId = MqttClient.generateClientId();

        mqttAndroidClient = new MqttAndroidClient(context, serverUri, clientId);
        mqttAndroidClient.setCallback(new MqttCallbackExtended() {
            @Override
            public void connectComplete(boolean b, String s) {
                Log.w("mqtt", s);
            }

            @Override
            public void connectionLost(Throwable throwable) {

            }

            @Override
            public void messageArrived(String topic, MqttMessage mqttMessage) throws Exception {
                Log.w("Mqtt", mqttMessage.toString());
            }

            @Override
            public void deliveryComplete(IMqttDeliveryToken iMqttDeliveryToken) {

            }
        });
        connect();
    }

    public void setCallback(MqttCallbackExtended callback) {
        mqttAndroidClient.setCallback(callback);
    }

    private void connect() {
        MqttConnectOptions mqttConnectOptions = new MqttConnectOptions();
```

```
    mqttConnectOptions.setAutomaticReconnect(true);
    mqttConnectOptions.setCleanSession(false);
    mqttConnectOptions.setUserName(username);
    mqttConnectOptions.setPassword(password.toCharArray());
    try {
        mqttAndroidClient.connect(mqttConnectOptions, null, new IMqttActionListener() {
            @Override
            public void onSuccess(IMqttToken asyncActionToken) {

                DisconnectedBufferOptions disconnectedBufferOptions
                = new DisconnectedBufferOptions();
                disconnectedBufferOptions.setBufferEnabled(true);
                disconnectedBufferOptions.setBufferSize(100);
                disconnectedBufferOptions.setPersistBuffer(false);
                disconnectedBufferOptions.setDeleteOldestMessages(false);
                mqttAndroidClient.setBufferOpts(disconnectedBufferOptions);
                subscribeToTopic(pumpSpeedFeed);
                subscribeToTopic(batteryFeed);
                subscribeToTopic(pHFeed);
                subscribeToTopic(statusFeed);
                subscribeToTopic(liquidLevelFeed);
            }

            @Override
            public void onFailure(IMqttToken asyncActionToken, Throwable exception) {
                Log.w("Mqtt", "Failed to connect to: " + serverUri + exception.toString());
            }
        });

    } catch (MqttException ex) {
        ex.printStackTrace();
    }
}

private void subscribeToTopic(String topic) {
    try {
        mqttAndroidClient.subscribe(topic, 0, null, new IMqttActionListener() {
            @Override
            public void onSuccess(IMqttToken asyncActionToken) {
                Log.w("Mqtt", "Subscribed!");
            }

            @Override
            public void onFailure(IMqttToken asyncActionToken, Throwable exception) {
                Log.w("Mqtt", "Subscribed fail!");
            }
        });

    } catch (MqttException ex) {
        System.err.println("Exceptionst subscribing");
```

```
            ex.printStackTrace();
        }
    }
}
```

1.4 CustomAdapter.java

```
package com.example.snoee.myapplistview.adapters;

import android.content.Context;
import android.content.res.Resources;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.ImageView;
import android.widget.TextView;

import com.example.snoee.myapplistview.R;

import java.util.ArrayList;

public class CustomAdapter extends BaseAdapter {
    ArrayList<SingleRow> list;
    Context c;

    public CustomAdapter(Context c) {
        this.c=c;
        list = new ArrayList<SingleRow>();
        Resources res = c.getResources();
        String[] titles = res.getStringArray(R.array.titles);
        String[] descriptions = res.getStringArray(R.array.description);
        int[] images = {R.drawable.control_panel_icon,R.drawable.statistics_icon};
        for (int i = 0; i < titles.length; i++) {
            list.add(new SingleRow(images[i], titles[i], descriptions[i]));
        }
    }

    @Override
    public int getCount() {
        return list.size();
    }

    @Override
    public Object getItem(int position) {
        return list.get(position);
    }
```

```
    @Override
    public long getItemId(int i) {
        return i;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        ViewHolder viewHolder;
        if(convertView == null){
            convertView = LayoutInflater.from(c).inflate(R.layout.categories_listview, null);

            viewHolder = new ViewHolder();

            viewHolder.imageViewCategory = convertView.findViewById(R.id.category_image);
            viewHolder.textViewTitle = convertView.findViewById(R.id.category_title);
            viewHolder.textViewDescription = convertView.findViewById(R.id.category_description);

            convertView.setTag(viewHolder);
        }else{
            viewHolder = (ViewHolder)convertView.getTag();
        }

        SingleRow row = list.get(position);

        viewHolder.textViewTitle.setText(row.title);
        viewHolder.textViewDescription.setText(row.description);
        viewHolder.imageViewCategory.setImageResource(row.image);

        return convertView;

    }
}

class ViewHolder{
    ImageView imageViewCategory;
    TextView textViewTitle;
    TextView textViewDescription;
}
```

1.5 SingleRow.java

```
 package com.example.snoee.myapplistview.adapters;

public class SingleRow {
     int image;
      String title;
     String description;
```

```
    public SingleRow(int image, String title, String description) {
        this.image = image;
        this.title = title;
        this.description = description;
    }
}
```

1.6 ViewPagerAdapter.java

```
package com.example.snoee.myapplistview.adapters;


import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentPagerAdapter;

import java.util.ArrayList;
import java.util.List;

public class ViewPagerAdapter extends FragmentPagerAdapter {

    private final List<Fragment> fragmentList = new ArrayList<>();
    private final List<String> fragmentTitleList = new ArrayList<>();


    public ViewPagerAdapter(FragmentManager fm) {
        super(fm);
    }


    @Override
    public Fragment getItem(int position) {
        return fragmentList.get(position);
    }


    @Override
    public int getCount() {
        return fragmentTitleList.size();
    }


    @Override
    public CharSequence getPageTitle(int position) {
        return fragmentTitleList.get(position);
    }

    public void addFragment(Fragment fragment, String title){
```

```
        fragmentList.add(fragment);
        fragmentTitleList.add(title);
    }

}
```

1.7 BatteryFrag.java

```
package com.example.snoee.myapplistview.fragments;

import android.content.Context;
import android.content.res.TypedArray;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.support.v4.app.Fragment;
import android.util.AttributeSet;
import android.util.Log;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import com.example.snoee.myapplistview.R;


public class BatteryFrag extends Fragment {
    public TextView backupStatusTv;
    public TextView estimatedRunningTimeTv;
    public TextView batteryTv;
    public ImageView battLevel;
    public ImageView battIcon;

    public int battery = -1;
    private int estimatedTimeRemaining = -1;
    public View v;
    static ViewGroup.LayoutParams layoutParams;
    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;
    int batteryLevelWidth;
    private static final double battToTimeConversionFactor = 0.02;
    private static final String estimatedTimeRemainingPrefix = "Estimated Time:";

    @Override
```

```
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {

    if (themeResId != NO_CUSTOM_THEME) {
        inflater = inflater.cloneInContext(
                new ContextThemeWrapper(getActivity(), themeResId)
        );
    }
    View view = inflater.inflate(R.layout.fragment_battery, container, false);
    backupStatusTv = view.findViewById(R.id.backup_status_id);
    estimatedRunningTimeTv = view.findViewById(R.id.estimated_running_time_id);
    battLevel = view.findViewById(R.id.batt_level_id);
    battLevel.post(new Runnable() {
        @Override
        public void run() {
            batteryLevelWidth = battLevel.getMeasuredWidth();
        }
    });
    battIcon = view.findViewById(R.id.battery_icon_id);
    battIcon.post(new Runnable() {
        @Override
        public void run() {
            battIcon.setAlpha(100);
        }
    });

    batteryTv = view.findViewById(R.id.batt);
    if(battery!=-1) {
        batteryTv.setText(String.valueOf(battery) + "%");
    }
    v = view;
    return view;
}

@Override
public void onInflate(
        @NonNull Context context,
        AttributeSet attrs,
        Bundle savedInstanceState
) {
    super.onInflate(context, attrs, savedInstanceState);
    TypedArray a = context.obtainStyledAttributes(
            attrs,
            R.styleable.ChildFragment
    );
    themeResId = a.getResourceId(
            R.styleable.ChildFragment_customTheme,
            NO_CUSTOM_THEME
    );
    a.recycle();
```

```
    }

    public void setBatteryTv(int batt) {
        Toast.makeText(getActivity(), "changed", Toast.LENGTH_LONG);
        battery = batt;
        if (batteryTv != null&&battery!=-1) {
            batteryTv.setText(String.valueOf(battery) + "%");
            layoutParams = battLevel.getLayoutParams();
            layoutParams.width = (int) (Double
            .valueOf(batteryLevelWidth / 100.0) * Double.valueOf(batt));
            battLevel.setLayoutParams(layoutParams);
            estimatedTimeRemaining = (int) (batt * battToTimeConversionFactor * 60);
            estimatedRunningTimeTv
            .setText(estimatedTimeRemainingPrefix + String.valueOf(estimatedTimeRemaining / 60)
            + " hours" + " ,
            + String.valueOf(estimatedTimeRemaining % 60)
            + " minutes");
        }
    }
}
```

1.8 ControlPanelFrag.java

```
 package com.example.snoee.myapplistview.fragments;

import android.os.Bundle;
import android.support.annotation.Nullable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import com.example.snoee.myapplistview.BaseFragment;
import com.example.snoee.myapplistview.R;

public class ControlPanelFrag extends BaseFragment {
    View view;
    Bundle bundle;
    BatteryFrag battFrag;
    WeatherFrag wFrag;
    PumpSliderFrag psFrag;

    public String[] outputValues;
    String inputValue;

    public String getInputValue() {
        return inputValue;
    }
```

```
@Override
public View provideYourFragmentView(LayoutInflater inflater,
ViewGroup parent, Bundle savedInstanceState) {
    View view = inflater.inflate(R.layout.fragment_control_panel, parent, false);
    return view;
}

@Override
public String getFragmentName() {
    return "AgroFrag";
}

@Override
public String getWindDirection() {
    WeatherFrag weathtFrag = (WeatherFrag) getChildFragmentManager()
    .findFragmentById(R.id.weath_id);
    return weathtFrag.getWindDirection();
}

@Override
public void setBattery(int bat) {
    BatteryFrag battFrag = (BatteryFrag) getChildFragmentManager()
    .findFragmentById(R.id.batt_id);
    battFrag.setBatteryTv(bat);
}

@Override
public void onActivityCreated(@Nullable Bundle savedInstanceState) {
    super.onActivityCreated(savedInstanceState);
}

@Override
public void setSeekPin(int i) {
    PumpSliderFrag psFrag = (PumpSliderFrag) getChildFragmentManager()
    .findFragmentById(R.id.ps_id);
    psFrag.setSeekPinPs(i);
}

@Override
public void setphValue(double i) {
    phFrag pf = (phFrag) getChildFragmentManager().findFragmentById(R.id.ph_id);
    pf.setPhValueTv(i);
}

@Override
public void setphStatus(String i) {
    phFrag pf = (phFrag) getChildFragmentManager().findFragmentById(R.id.ph_id);
    pf.setPhStatusTv(i);
}
```

```
    @Override
    public void setLiquidLevelTv(double ll) {
        LiquidLevelFrag llf = (LiquidLevelFrag) getChildFragmentManager()
        .findFragmentById(R.id.liquid_level_id);
        llf.setLiquidLevelTv(ll);
    }

    @Override
    public void displayScheduleTv(String str) {
    }
}
```

1.9 DevFrag.java

```
 package com.example.snoee.myapplistview.fragments;

import android.app.DialogFragment;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.example.snoee.myapplistview.R;

public class DevFrag extends DialogFragment implements View.OnClickListener {
    Button sendFeedback;
    EditText feedbackEt;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_dev, container, false);

        feedbackEt = (EditText) v.findViewById(R.id.feedback);
        sendFeedback = (Button) v.findViewById(R.id.send_feedback);
        sendFeedback.setOnClickListener(this);

        return v;
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.send_feedback:
                Toast.makeText(getActivity(), feedbackEt.getText()+"
```

```
                feedback sent", Toast.LENGTH_SHORT).show();
                dismiss();
                break;
        }
    }
    @Override
    public void onResume() {
        super.onResume();
        getDialog().getWindow().setLayout(getResources()
        .getDimensionPixelSize(R.dimen.dialog_width), getResources()
        .getDimensionPixelSize(R.dimen.dialog_height));
    }
}
```

1.10 LiquidLevelFrag.java

```
 package com.example.snoee.myapplistview.fragments;

import android.content.Context;
import android.content.res.TypedArray;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.support.v4.app.Fragment;
import android.os.Bundle;
import android.util.AttributeSet;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.TextView;
import com.example.snoee.myapplistview.R;

public class LiquidLevelFrag extends Fragment {
    TextView liquidLevelTv;
    public double liquidLevel;

    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        //
        if (themeResId != NO_CUSTOM_THEME) {
            inflater = inflater.cloneInContext(
                    new ContextThemeWrapper(getActivity(), themeResId)
            );
```

```
        }
        //
        View view = inflater.inflate(R.layout.fragment_liquid_level, container, false);
        liquidLevelTv = view.findViewById(R.id.liquid_level);
        return view;
    }

    @Override
    public void onInflate(
            @NonNull Context context,
            AttributeSet attrs,
            Bundle savedInstanceState
    ) {
        super.onInflate(context, attrs, savedInstanceState);
        TypedArray a = context.obtainStyledAttributes(
                attrs,
                R.styleable.ChildFragment
        );
        themeResId = a.getResourceId(
                R.styleable.ChildFragment_customTheme,
                NO_CUSTOM_THEME
        );
        a.recycle();
    }

    public void setLiquidLevelTv(double ll) {
        liquidLevel = ll;
        if (liquidLevelTv != null) {
            liquidLevelTv.setText(String.valueOf(ll));
        }
    }
}
```

1.11 phFrag.java

```
package com.example.snoee.myapplistview.fragments;

import android.content.Context;
import android.content.res.TypedArray;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.support.v4.app.Fragment;
import android.os.Bundle;
import android.util.AttributeSet;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
```

```
import android.widget.TextView;

import com.example.snoee.myapplistview.R;

public class phFrag extends Fragment {
    TextView phValueTv, phStatusTv;
    public double phValue;
    public String phStatus;

    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        //
        if (themeResId != NO_CUSTOM_THEME) {
            inflater = inflater.cloneInContext(
                    new ContextThemeWrapper(getActivity(), themeResId)
            );
        }
        //
        View view = inflater.inflate(R.layout.fragment_ph, container, false);
        phValueTv = view.findViewById(R.id.ph_value);
        phStatusTv = view.findViewById(R.id.ph_status);
        return view;
    }

    @Override
    public void onInflate(
            @NonNull Context context,
            AttributeSet attrs,
            Bundle savedInstanceState
    ) {
        super.onInflate(context, attrs, savedInstanceState);
        TypedArray a = context.obtainStyledAttributes(
                attrs,
                R.styleable.ChildFragment
        );
        themeResId = a.getResourceId(
                R.styleable.ChildFragment_customTheme,
                NO_CUSTOM_THEME
        );
        a.recycle();
    }

    public void setPhValueTv(double ph) {
        phValue = ph;
        if (phValueTv != null) {
```

```
            phValueTv.setText(String.valueOf(ph));
        }
    }

    public void setPhStatusTv(String status) {
        phStatus = status;
        if (phStatusTv != null) {
            phStatusTv.setText(String.valueOf(status));
        }
    }
}
```

1.12 PumpSliderFrag.java

```
package com.example.snoee.myapplistview.fragments;

import android.app.Activity;
import android.content.Context;
import android.content.res.TypedArray;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.support.v4.app.Fragment;
import android.os.Bundle;
import android.util.AttributeSet;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.SeekBar;
import android.widget.TextView;

import com.example.snoee.myapplistview.interfaces.Communicator;
import com.example.snoee.myapplistview.R;

public class PumpSliderFrag extends Fragment {
    Communicator communicator;
    TextView pumpSpeed;
    SeekBar pumpSlider;
    int seekPin;
    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        //isto nao vai funcionar se a nossa actividade nao implementar interface
        communicator = (Communicator) activity;
```

```
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        //
        if (themeResId != NO_CUSTOM_THEME) {
            inflater = inflater.cloneInContext(
                    new ContextThemeWrapper(getActivity(), themeResId)
            );
        }
        //
        View view = inflater.inflate(R.layout.fragment_pump_slider, null);
        pumpSpeed = view.findViewById(R.id.pumpSpeed);
        pumpSlider = view.findViewById(R.id.pumpSlider);
        pumpSliderInit();
        return view;
    }

    public void pumpSliderInit() {

        pumpSlider.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
//              Toast.makeText(getActivity(), "seekbar progress: " +
// progress, Toast.LENGTH_SHORT).show();
                seekPin = progress;
                pumpSpeed.setText("Pump Speed : " + progress);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
//              Toast.makeText(getActivity(), "seekbar touch started!",
// Toast.LENGTH_SHORT).show();
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {
//              Toast.makeText(getActivity(), "seekbar touch stopped!",
// Toast.LENGTH_SHORT).show();
                communicator.onDialogMessage("PumpSliderFrag", "" + seekPin);
            }
        });
    }

    @Override
    public void onInflate(
            @NonNull Context context,
            AttributeSet attrs,
            Bundle savedInstanceState
```

```
    ) {
        super.onInflate(context, attrs, savedInstanceState);
        TypedArray a = context.obtainStyledAttributes(
                attrs,
                R.styleable.ChildFragment
        );
        themeResId = a.getResourceId(
                R.styleable.ChildFragment_customTheme,
                NO_CUSTOM_THEME
        );
        a.recycle();
    }

    public void setSeekPinPs(int i) {
        pumpSlider.setProgress(i);
    }
}
```

1.13 SpeedPickerFrag.java

```
 package com.example.snoee.myapplistview.fragments;

import android.app.Activity;
import android.app.DialogFragment;
import android.content.Context;
import android.content.res.TypedArray;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.util.AttributeSet;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;

import com.example.snoee.myapplistview.R;
import com.example.snoee.myapplistview.interfaces.Communicator;

public class SpeedPickerFrag extends DialogFragment implements View.OnClickListener {
    Button ok;
    Button cancel;
    Communicator communicator;
    TextView pumpSpeed;
    SeekBar pumpSlider;
    int seekPin;
```

```
    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        //isto nao vai funcionar se a nossa actividade nao implementar interface
        communicator = (Communicator) activity;
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        //
        if (themeResId != NO_CUSTOM_THEME) {
            inflater = inflater.cloneInContext(
                   new ContextThemeWrapper(getActivity(), themeResId)
            );
        }
        //
        View view = inflater.inflate(R.layout.fragment_speed_picker, null);
        pumpSpeed = view.findViewById(R.id.pumpSpeed);
        pumpSlider = view.findViewById(R.id.pumpSlider);
        ok = (Button) view.findViewById(R.id.ok_btn);
        cancel = (Button) view.findViewById(R.id.canel_btn);
        ok.setOnClickListener(this);
        cancel.setOnClickListener(this);
        pumpSliderInit();
        return view;
    }

    public void pumpSliderInit() {

        pumpSlider.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
//              Toast.makeText(getActivity(),
// "seekbar progress: " +
// progress, Toast.LENGTH_SHORT).show();
                seekPin = progress;
                pumpSpeed.setText("Pump Speed : " + progress);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
//              Toast.makeText(getActivity(), "seekbar touch started!",
// Toast.LENGTH_SHORT).show();
            }
```

```
            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {
//                Toast.makeText(getActivity(), "seekbar touch stopped!",
Toast.LENGTH_SHORT).show();
//                communicator.onDialogMessage("SpeedPickerFrag", "" + seekPin);
            }
        });
    }

    @Override
    public void onInflate(
            @NonNull Context context,
            AttributeSet attrs,
            Bundle savedInstanceState
    ) {
        super.onInflate(context, attrs, savedInstanceState);
        TypedArray a = context.obtainStyledAttributes(
                attrs,
                R.styleable.ChildFragment
        );
        themeResId = a.getResourceId(
                R.styleable.ChildFragment_customTheme,
                NO_CUSTOM_THEME
        );
        a.recycle();
    }

    public void setSeekPinPs(int i) {
        pumpSlider.setProgress(i);
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.ok_btn:
                //Display the newly selected value from picker
                communicator.onDialogMessage("SpeedPickerFrag",seekPin+"");
                dismiss();
                break;
            case R.id.canel_btn:
                dismiss();
                break;
        }
    }
    @Override
    public void onResume() {
        super.onResume();
        getDialog().getWindow().setLayout( getResources()
        .getDimensionPixelSize(R.dimen.dialog_width), getResources()
        .getDimensionPixelSize(R.dimen.dialog_height));
```

```
    }
}
```

1.14 SpeedPickerFromSetupFrag.java

```
package com.example.snoee.myapplistview.fragments;

import android.app.Activity;
import android.app.DialogFragment;
import android.content.Context;
import android.content.res.TypedArray;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.util.AttributeSet;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.SeekBar;
import android.widget.TextView;

import com.example.snoee.myapplistview.R;
import com.example.snoee.myapplistview.interfaces.Communicator;

public class SpeedPickerFromSetupFrag extends DialogFragment implements View.OnClickListener {
    Button ok;
    Button cancel;
    Communicator communicator;
    TextView pumpSpeed;
    SeekBar pumpSlider;
    int seekPin;
    private @StyleRes
    int themeResId;
    private static final int NO_CUSTOM_THEME = 0;

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        //isto nao vai funcionar se a nossa actividade nao implementar interface
        communicator = (Communicator) activity;
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
    Bundle savedInstanceState) {
        //
```

```
        if (themeResId != NO_CUSTOM_THEME) {
            inflater = inflater.cloneInContext(
                    new ContextThemeWrapper(getActivity(), themeResId)
            );
        }
        //
        View view = inflater.inflate(R.layout.fragment_speed_picker_from_setup, null);
        pumpSpeed = view.findViewById(R.id.pumpSpeed);
        pumpSlider = view.findViewById(R.id.pumpSlider);
        ok = (Button) view.findViewById(R.id.ok_btn);
        cancel = (Button) view.findViewById(R.id.canel_btn);
        ok.setOnClickListener(this);
        cancel.setOnClickListener(this);
        pumpSliderInit();
        return view;
    }

    public void pumpSliderInit() {

        pumpSlider.setOnSeekBarChangeListener(new SeekBar.OnSeekBarChangeListener() {
            @Override
            public void onProgressChanged(SeekBar seekBar, int progress, boolean fromUser) {
//              Toast.makeText(getActivity(), "seekbar progress: " + progress,
Toast.LENGTH_SHORT).show();
                seekPin = progress;
                pumpSpeed.setText("Pump Speed : " + progress);
            }

            @Override
            public void onStartTrackingTouch(SeekBar seekBar) {
//              Toast.makeText(getActivity(), "seekbar touch started!", Toast.LENGTH_SHORT).show();
            }

            @Override
            public void onStopTrackingTouch(SeekBar seekBar) {
//              Toast.makeText(getActivity(), "seekbar touch stopped!", Toast.LENGTH_SHORT).show();
//              communicator.onDialogMessage("SpeedPickerFromSetupFrag", "" + seekPin);
            }
        });
    }

    @Override
    public void onInflate(
            @NonNull Context context,
            AttributeSet attrs,
            Bundle savedInstanceState
    ) {
        super.onInflate(context, attrs, savedInstanceState);
        TypedArray a = context.obtainStyledAttributes(
                attrs,
```

```
                R.styleable.ChildFragment
        );
        themeResId = a.getResourceId(
                R.styleable.ChildFragment_customTheme,
                NO_CUSTOM_THEME
        );
        a.recycle();
    }

    public void setSeekPinPs(int i) {
        pumpSlider.setProgress(i);
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.ok_btn:
                //Display the newly selected value from picker
                communicator.onDialogMessage("SpeedPickerFromSetupFrag",seekPin+"");
                dismiss();
                break;
            case R.id.canel_btn:
                dismiss();
                break;
        }
    }
    @Override
    public void onResume() {
        super.onResume();
        getDialog().getWindow().setLayout( getResources()
        .getDimensionPixelSize(R.dimen.dialog_width), getResources()
        .getDimensionPixelSize(R.dimen.dialog_height));
    }
}
```

1.15 StatisticsFrag.java

```
package com.example.snoee.myapplistview.fragments;

import android.os.Bundle;
import android.support.annotation.Nullable;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import com.example.snoee.myapplistview.BaseFragment;
import com.example.snoee.myapplistview.R;
```

```
public class StatisticsFrag extends BaseFragment {
    View view;

    @Override
    public View provideYourFragmentView(LayoutInflater inflater,
    ViewGroup parent, Bundle savedInstanceState) {
        View view = inflater.inflate(R.layout.fragment_statistics, parent, false);
        return view;
    }

    @Override
    public String getFragmentName() {
        return null;
    }

    @Override
    public String getWindDirection() {
        return null;
    }

    @Override
    public void setBattery(int bat) {

    }

    @Override
    public void setSeekPin(int i) {

    }

    @Override
    public void setphValue(double i) {

    }

    @Override
    public void setphStatus(String i) {

    }

    @Override
    public void setLiquidLevelTv(double ll) {

    }

    @Override
    public void displayScheduleTv(String str) {

    }
```

```
}
```

1.16 WeatherFrag.java

```
 package com.example.snoee.myapplistview.fragments;

import android.content.res.TypedArray;
import android.support.annotation.NonNull;
import android.support.annotation.StyleRes;
import android.support.v4.app.Fragment;
import android.content.Context;
import android.graphics.Typeface;
import android.net.ConnectivityManager;
import android.os.AsyncTask;
import android.os.Bundle;
import android.text.Html;
import android.util.AttributeSet;
import android.util.Log;
import android.view.ContextThemeWrapper;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.LinearLayout;
import android.widget.ProgressBar;
import android.widget.TextView;
import android.widget.Toast;

import com.example.snoee.myapplistview.R;

import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.InputStream;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;
import java.text.DateFormat;
import java.util.Date;
import java.util.Locale;

public class WeatherFrag extends Fragment {
    TextView cityField, detailsField, currentTemperatureField, humidity_field,
    weatherIcon, windDirection, windSpeed;
    ProgressBar loader;
    Typeface weatherFont;
    String direction = "none";
```

```
private static String city = "Leicester,England";
/* Please Put your API KEY here */
String OPEN_WEATHER_MAP_API = "efdb3bd9059a5aed2397aa57362e9327";
/* Please Put your API KEY here */

private @StyleRes
int themeResId;
private static final int NO_CUSTOM_THEME = 0;

@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {
    //
    if (themeResId != NO_CUSTOM_THEME) {
        inflater = inflater.cloneInContext(
                new ContextThemeWrapper(getActivity(), themeResId)
        );
    }
    //
    View view = inflater.inflate(R.layout.fragment_weather, container, false);

    loader = (ProgressBar) view.findViewById(R.id.loader);
    cityField = (TextView) view.findViewById(R.id.city_field);
    windDirection = (TextView) view.findViewById(R.id.wind_direction);
    windSpeed = (TextView) view.findViewById(R.id.wind_speed);
    detailsField = (TextView) view.findViewById(R.id.details_field);
    currentTemperatureField = (TextView) view.findViewById(R.id.current_temperature_field);
    humidity_field = (TextView) view.findViewById(R.id.humidity_field);
    weatherIcon = (TextView) view.findViewById(R.id.weather_icon);
    weatherFont = Typeface.createFromAsset(getActivity().getAssets(),
    "fonts/weathericons-regular-webfont.ttf");
    weatherIcon.setTypeface(weatherFont);

    taskLoadUp(city);


    Log.d("VIVZ", "Fragment A onCreateView");
    return view;
}

public void taskLoadUp(String query) {
    if (isNetworkAvailable(getActivity())) {
        DownloadWeather task = new DownloadWeather();
        task.execute(query);
    } else {
        Toast.makeText(getActivity(), "No Internet Connection", Toast.LENGTH_LONG).show();
    }
}

class DownloadWeather extends AsyncTask<String, Void, String> {
```

```
    @Override
    protected void onPreExecute() {
        super.onPreExecute();
        loader.setVisibility(View.VISIBLE);

    }

    protected String doInBackground(String... args) {
        String xml = excuteGet("http://api.openweathermap.org/data/2.5/weather?q=" + args[0] +
                "&units=metric&appid=" + OPEN_WEATHER_MAP_API);
        return xml;
    }

    @Override
    protected void onPostExecute(String xml) {

        try {
            JSONObject json = new JSONObject(xml);
            if (json != null) {
                JSONObject details = json.getJSONArray("weather").getJSONObject(0);
                JSONObject main = json.getJSONObject("main");
                JSONObject wind = json.getJSONObject("wind");
                DateFormat df = DateFormat.getDateTimeInstance();

                Log.i("test", wind.getString("speed"));
                Log.i("test", wind.getString("deg"));
                direction = degToDir(wind.getDouble("deg"));
                cityField.setText(json.getString("name").toUpperCase(Locale.US)
                + ", " + json.getJSONObject("sys").getString("country"));
                detailsField.setText(details.getString("description")
                .toLowerCase(Locale.US));
                currentTemperatureField.setText(
                String.format("%.2f", main.getDouble("temp")) + "C");
                humidity_field.setText("Humidity: " + main.getString("humidity") + "%");
                windDirection.setText(direction);
                windSpeed.setText("at " + wind.getString("speed") + " m/s");
                weatherIcon.setText(Html.fromHtml(setWeatherIcon(details.getInt("id"),
                        json.getJSONObject("sys").getLong("sunrise") * 1000,
                        json.getJSONObject("sys").getLong("sunset") * 1000)));

                loader.setVisibility(View.GONE);

            }
        } catch (JSONException e) {
            Toast.makeText(getActivity(), "Error, Check City", Toast.LENGTH_SHORT).show();
        }

    }

}
```

```
    public static boolean isNetworkAvailable(Context context) {
        return ((ConnectivityManager) context.getSystemService(Context.CONNECTIVITY_SERVICE))
        .getActiveNetworkInfo() != null;
    }

    public static String excuteGet(String targetURL) {
        URL url;
        HttpURLConnection connection = null;
        try {
            //Create connection
            url = new URL(targetURL);
            connection = (HttpURLConnection) url.openConnection();
            connection.setRequestProperty("content-type", "application/json;
charset=utf-8");
            connection.setRequestProperty("Content-Language", "en-US");
            connection.setUseCaches(false);
            connection.setDoInput(true);
            connection.setDoOutput(false);
            InputStream is;
            int status = connection.getResponseCode();
            if (status != HttpURLConnection.HTTP_OK)
                is = connection.getErrorStream();
            else
                is = connection.getInputStream();
            BufferedReader rd = new BufferedReader(new InputStreamReader(is));
            String line;
            StringBuffer response = new StringBuffer();
            while ((line = rd.readLine()) != null) {
                response.append(line);
                response.append('\r');
            }
            rd.close();
            return response.toString();
        } catch (Exception e) {
            return null;
        } finally {
            if (connection != null) {
                connection.disconnect();
            }
        }
    }

    public static String setWeatherIcon(int actualId, long sunrise, long sunset) {
        int id = actualId / 100;
        String icon = "";
        if (actualId == 800) {
            long currentTime = new Date().getTime();
            if (currentTime >= sunrise && currentTime < sunset) {
```

```
            icon = "&#xf00d;";
        } else {
            icon = "&#xf02e;";
        }
    } else {
        switch (id) {
            case 2:
                icon = "&#xf01e;";
                break;
            case 3:
                icon = "&#xf01c;";
                break;
            case 7:
                icon = "&#xf014;";
                break;
            case 8:
                icon = "&#xf013;";
                break;
            case 6:
                icon = "&#xf01b;";
                break;
            case 5:
                icon = "&#xf019;";
                break;
        }
    }
    return icon;
}

public String degToDir(double x) {
    String directions[] = {"North", "N-E", "East", "S-E", "South", "S-W", "West", "N-W", "North"};
    return directions[(int) Math.round((((double) x % 360) / 45))];
}

@Override
public void onInflate(
        @NonNull Context context,
        AttributeSet attrs,
        Bundle savedInstanceState
) {
    super.onInflate(context, attrs, savedInstanceState);
    TypedArray a = context.obtainStyledAttributes(
            attrs,
            R.styleable.ChildFragment
    );
    themeResId = a.getResourceId(
            R.styleable.ChildFragment_customTheme,
            NO_CUSTOM_THEME
    );
    a.recycle();
```

```
    }

    public static void updateCity(String city) {
        WeatherFrag.city = city;
    }

    public String getWindDirection() {
        return direction;
    }
}
```

1.17 AutoCutDialogFrag.java

```
 package com.example.snoee.myapplistview;

import android.app.Activity;
import android.app.DialogFragment;
import android.content.Context;
import android.net.Uri;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.NumberPicker;
import android.widget.Toast;

import com.example.snoee.myapplistview.interfaces.Communicator;


public class AutoCutDialogFrag extends DialogFragment implements View.OnClickListener {
    Button sendFeedback;
    EditText feedbackEt;

    Button ok;
    Button skip;
    Button cancel;

    private int selectedValue=1;
    Communicator communicator;

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        communicator = (Communicator) activity;
    }
```

```
@Override
public View onCreateView(LayoutInflater inflater, ViewGroup container,
                         Bundle savedInstanceState) {
    View v = inflater.inflate(R.layout.fragment_auto_cut_dialog, container, false);
    ok = (Button) v.findViewById(R.id.ok_btn);
    skip = (Button) v.findViewById(R.id.skip_btn);
    cancel = (Button) v.findViewById(R.id.canel_btn);
    ok.setOnClickListener(this);
    skip.setOnClickListener(this);
    cancel.setOnClickListener(this);

    NumberPicker np = (NumberPicker) v.findViewById(R.id.np);
    //Populate NumberPicker values from minimum and maximum value range
    //Set the minimum value of NumberPicker
    np.setMinValue(1);
    //Specify the maximum value/number of NumberPicker
    np.setMaxValue(10);
    //Gets whether the selector wheel wraps when reaching the min/max value.
    np.setWrapSelectorWheel(true);
    //Set a value change listener for NumberPicker
    //Set a value change listener for NumberPicker
    np.setOnValueChangedListener(new NumberPicker.OnValueChangeListener() {
        @Override
        public void onValueChange(NumberPicker picker, int oldVal, int newVal) {
            //Display the newly selected value from picker

            selectedValue = newVal;
        }
    });
    return v;
}

@Override
public void onClick(View view) {
    switch (view.getId()) {
        case R.id.ok_btn:
            //Display the newly selected value from picker
            communicator.onDialogMessage("AutoCutDialogFrag","" + selectedValue);
            dismiss();
            break;
        case R.id.skip_btn:
            dismiss();
            break;
        case R.id.canel_btn:
            dismiss();
            break;
    }
}
```

```
}
```

1.18 BaseFragment.java

```
package com.example.snoee.myapplistview;

import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

public abstract class BaseFragment extends Fragment {

    public String[] outputValues;

    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }

    public View onCreateView(LayoutInflater inflater, ViewGroup parent, Bundle savedInstanseState) {
        View view = provideYourFragmentView(inflater, parent, savedInstanseState);
        return view;
    }

    public abstract View provideYourFragmentView(LayoutInflater inflater, ViewGroup parent,
    Bundle savedInstanceState);

    public abstract String getFragmentName();

    public abstract String getWindDirection();


    public abstract void setBattery(int bat);

    public abstract void setSeekPin(int i);

    public abstract void setphValue(double i);

    public abstract void setphStatus(String status);

    public abstract void setLiquidLevelTv(double ll) ;

    public abstract void displayScheduleTv(String str);

}
```

1.19 CityWeatherFrag.java

```
 package com.example.snoee.myapplistview;

import android.app.Activity;
import android.app.DialogFragment;
import android.content.Context;
import android.net.Uri;
import android.os.Bundle;
import android.support.v4.app.Fragment;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.EditText;
import android.widget.NumberPicker;
import android.widget.TextView;

import com.example.snoee.myapplistview.interfaces.Communicator;


public class CityWeatherFrag extends DialogFragment implements View.OnClickListener {

    TextView currentCityTv;
    EditText changeCityTv;
    Button ok;
    Button skip;
    Button cancel;

    Communicator communicator;

    @Override
    public void onAttach(Activity activity) {
        super.onAttach(activity);
        communicator = (Communicator) activity;
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        View v = inflater.inflate(R.layout.fragment_city_weather, container, false);
        currentCityTv = (TextView) v.findViewById(R.id.current_city_id);
        changeCityTv = (EditText) v.findViewById(R.id.change_city_id);
        ok = (Button) v.findViewById(R.id.ok_btn);
        skip = (Button) v.findViewById(R.id.skip_btn);
        cancel = (Button) v.findViewById(R.id.canel_btn);
        ok.setOnClickListener(this);
        skip.setOnClickListener(this);
        cancel.setOnClickListener(this);
```

```
        return v;
    }

    @Override
    public void onClick(View view) {
        switch (view.getId()) {
            case R.id.ok_btn:
                //Display the newly selected value from picker
                communicator.onDialogMessage("CityWeatherFrag", ""+changeCityTv.getText());
                dismiss();
                break;
            case R.id.skip_btn:
                dismiss();
                break;
            case R.id.canel_btn:
                dismiss();
                break;
        }
    }

}
```

1.20 activity_control_panel.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".activities.ControlPanel">

    <android.support.v7.widget.Toolbar
        android:id="@+id/control_panel_toolbar"
        android:layout_width="match_parent"
        android:layout_height="?attr/actionBarSize"
         app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar" />

    <android.support.design.widget.TabLayout
        android:id="@+id/tabLayout_id"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="#000000"
        app:tabMaxWidth="0dp"
        app:tabGravity="fill"
        app:tabMode="fixed"
        app:tabSelectedTextColor="@color/colorAccent"
```

```
        app:tabTextColor="#FFFFFF" />

    <android.support.v4.view.ViewPager
        android:id="@+id/viewPager_id"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_below="@id/tabLayout_id" />


</LinearLayout>
```

1.21 activity_main.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="5dp"
    android:background="@drawable/main_screen_big"
    tools:context=".activities.MainActivity">

    <ListView
        android:id="@+id/category_list"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="5dp"
        android:padding="5dp" />
</RelativeLayout>
```

1.22 categories_listview.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="10dp"
    android:background="@drawable/linear_gradient_drawable"
    android:padding="10dp">

    <ImageView
        android:id="@+id/category_image"
        android:layout_width="50dp"
        android:layout_height="50dp"
        android:padding="15dp"
```

```
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        />

    <TextView
        android:id="@+id/category_title"
        android:layout_width="wrap_content"
        android:layout_height="30dp"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:layout_toRightOf="@id/category_image"
        android:gravity="center_horizontal"
        android:text="title"
        android:textColor="#FFFFFF"
        android:textSize="20sp"
        android:textStyle="bold" />

    <TextView
        android:id="@+id/category_description"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignBottom="@id/category_image"
        android:layout_alignParentRight="true"
        android:layout_below="@id/category_title"
        android:layout_toRightOf="@id/category_image"
        android:gravity="center_horizontal"
        android:text="description"
        android:textColor="#FFFFFF"
        android:textStyle="italic" />

</RelativeLayout>
```

1.23 fragment_battery.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="80dp"
    android:layout_marginLeft="5dp"
    android:layout_marginTop="10dp"
    android:layout_marginRight="5dp"
    android:layout_marginBottom="5dp"
    android:orientation="horizontal"
    android:padding="5dp"
    tools:context="com.example.snoee.myapplistview.fragments.BatteryFrag">

    <FrameLayout
```

```
    android:layout_width="100dp"
    android:layout_height="50dp"
    android:layout_gravity="center"
    android:layout_marginLeft="5dp"
    android:background="@android:color/transparent"
    android:orientation="horizontal">

    <ImageView
        android:id="@+id/battery_icon_id"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center"
        android:background="@android:color/transparent"
        android:src="@drawable/battery_icon" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:background="@android:color/transparent"
        android:paddingLeft="11dp"
        android:paddingRight="16dp">

        <ImageView
            android:id="@+id/batt_level_id"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_gravity="center"
            android:background="@android:color/transparent"
            android:src="@drawable/battery_level_gradient_" />
    </LinearLayout>

    <TextView
        android:id="@+id/batt"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:background="@android:color/transparent"
        android:gravity="center"
        android:text=""
        android:textAppearance="@style/MText"
        android:textStyle="italic|bold" />
</FrameLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:layout_marginLeft="20dp"
    android:background="@android:color/transparent"
    android:orientation="vertical">
```

```
        <TextView
            android:id="@+id/backup_status_id"
            android:layout_width="wrap_content"
            android:layout_height="0dp"
android:gravity="bottom"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:text="Power Backup Is UnPlugged"
            android:textAppearance="@style/SText" />

        <TextView
            android:id="@+id/estimated_running_time_id"
            android:layout_width="wrap_content"
            android:layout_height="0dp"
             android:layout_weight="1"
            android:gravity="top"
            android:background="@android:color/transparent"
            android:text="Estimated Time : unavailable"
            android:textAppearance="@style/SText" />
    </LinearLayout>
</LinearLayout>
```

1.24 fragment_city_weather.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#EDEDEA"
    android:orientation="vertical"
    tools:context=".fragments.PatternTimeGapPickerFrag">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:text="Select The City Of The Machine"
        android:textStyle="bold" />

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Current city is unavailable"
        android:id="@+id/current_city_id"
        android:textAlignment="center" />
```

```
    <EditText
        android:layout_width="match_parent"
        android:layout_height="300dp"
        android:hint="Change City"
        android:id="@+id/change_city_id"
        android:textAlignment="center" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="right"
        android:orientation="horizontal">

        <Button
            android:id="@+id/canel_btn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="cancel" />

        <Button
            android:id="@+id/skip_btn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="skip" />

        <Button
            android:id="@+id/ok_btn"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Finish" />
    </LinearLayout>
</LinearLayout>
```

1.25 fragment_control_panel.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/control_panel_big"
    android:orientation="vertical"
    tools:context="com.example.snoee.myapplistview.fragments.ControlPanelFrag">

    <ScrollView
        android:layout_width="match_parent"
        android:layout_height="match_parent">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <fragment
        android:id="@+id/batt_id"
        android:name="com.example.snoee.myapplistview.fragments.BatteryFrag"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/fragment_margin"
        android:padding="@dimen/fragment_padding"
        app:customTheme="@style/AppTheme.Agro"
        tools:layout="@layout/fragment_battery" />

    <fragment
        android:id="@+id/weath_id"
        android:name="com.example.snoee.myapplistview.fragments.WeatherFrag"
        android:layout_width="match_parent"
        android:layout_height="@dimen/agro_frag_height"
        android:layout_margin="@dimen/fragment_margin"
        android:padding="@dimen/fragment_padding"
        app:customTheme="@style/AppTheme.Agro"
        tools:layout="@layout/fragment_weather" />

    <fragment
        android:id="@+id/ph_id"
        android:name="com.example.snoee.myapplistview.fragments.phFrag"
        android:layout_width="match_parent"
        android:layout_height="@dimen/agro_frag_height"
        android:layout_margin="@dimen/fragment_margin"
        android:padding="@dimen/fragment_padding"
        app:customTheme="@style/AppTheme.Agro"
        tools:layout="@layout/fragment_ph" />

    <fragment
        android:id="@+id/liquid_level_id"
        android:name="com.example.snoee.myapplistview.fragments.LiquidLevelFrag"
        android:layout_width="match_parent"
        android:layout_height="@dimen/agro_frag_height"
        android:layout_margin="@dimen/fragment_margin"
        android:padding="@dimen/fragment_padding"
        app:customTheme="@style/AppTheme.Agro"
        tools:layout="@layout/fragment_liquid_level" />

    <fragment
        android:id="@+id/ps_id"
        android:name="com.example.snoee.myapplistview.fragments.PumpSliderFrag"
        android:layout_width="match_parent"
```

```
                android:layout_height="@dimen/agro_frag_height"
                android:layout_margin="@dimen/fragment_margin"
                android:padding="@dimen/fragment_padding"
                app:customTheme="@style/AppTheme.Agro"
                tools:layout="@layout/fragment_pump_slider" />

        </LinearLayout>
    </ScrollView>
</LinearLayout>
```

1.26 fragment_dev.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".fragments.DevFrag">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:scaleType="centerCrop"
        android:src="@drawable/circuit" />

    <RelativeLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:padding="15dp">

        <de.hdodenhof.circleimageview.CircleImageView
            android:id="@+id/face"
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:contentDescription="Inventrepreneur"
            android:scaleType="centerCrop"
            android:src="@drawable/face"
            app:civ_border_color="#FFFFFF"
            app:civ_border_overlay="true"
            app:civ_border_width="2dp" />

        <TextView
            android:id="@+id/name"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/face"
            android:padding="3dp"
```

```
            android:text="Arpit Sharma"
            android:textAppearance="@style/MText"
            android:textStyle="bold|italic" />

        <TextView
            android:id="@+id/email"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/name"
            android:padding="3dp"
            android:text="sharma.arpit956@gmail.com"
            android:textAppearance="@style/MText"
            android:textStyle="bold|italic" />

        <TextView
            android:id="@+id/contact_number"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/email"
            android:padding="3dp"
            android:text="+44 7882113088"
            android:textAppearance="@style/MText"
            android:textStyle="bold|italic" />

        <EditText
            android:id="@+id/feedback"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/contact_number"
            android:layout_marginTop="10dp"
            android:hint="enter feedback here"
            android:padding="3dp"
            android:textAppearance="@style/MText"
            android:textColor="@color/white"
            android:textColorHint="#DDDDDD"
            android:textStyle="italic" />

        <Button
            android:id="@+id/send_feedback"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_below="@id/feedback"
            android:padding="10dp"
            android:background="@drawable/linear_gradient_drawable"
            android:text="send"
            android:textAppearance="@style/MText"
            android:textStyle="bold|italic" />
    </RelativeLayout>

</FrameLayout>
```

1.27 fragment_liquid_level.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="80dp"
    android:layout_margin="5dp"
    android:orientation="horizontal"
    tools:context=".fragments.phFrag">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:orientation="vertical">

        <TextView
            android:id="@+id/liquid_level"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|bottom"
            android:text="Loading..."
            android:textAppearance="@style/LText" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|top"
            android:text="Liquid Level (in mm)"
            android:textAppearance="@style/MText" />
    </LinearLayout>

<!--    <LinearLayout-->
<!--        android:layout_width="0dp"-->
<!--        android:layout_height="match_parent"-->
<!--        android:layout_weight="1"-->
<!--        android:background="@android:color/transparent"-->
<!--        android:orientation="vertical">-->

<!--        <TextView-->
```

```
<!--           android:id="@+id/ph_status"-->
<!--           android:layout_width="match_parent"-->
<!--           android:layout_height="0dp"-->
<!--           android:layout_weight="1"-->
<!--           android:background="@android:color/transparent"-->
<!--           android:gravity="center|bottom"-->
<!--           android:text="Loading..."-->
<!--           android:textAppearance="@style/LText" />-->

<!--       <TextView-->
<!--           android:layout_width="match_parent"-->
<!--           android:layout_height="0dp"-->
<!--           android:layout_weight="1"-->
<!--           android:background="@android:color/transparent"-->
<!--           android:gravity="center|top"-->
<!--           android:text="Status"-->
<!--           android:textAppearance="@style/MText" />-->
<!--   </LinearLayout>-->
</LinearLayout>
```

1.28 fragment_ph.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="80dp"
    android:layout_margin="5dp"
    android:orientation="horizontal"
    tools:context=".fragments.phFrag">

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:orientation="vertical">

        <TextView
            android:id="@+id/ph_value"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|bottom"
            android:text="Loading..."
            android:textAppearance="@style/LText" />
```

```
        <TextView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|top"
            android:text="pH Value"
            android:textAppearance="@style/MText" />
    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="match_parent"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:orientation="vertical">

        <TextView
            android:id="@+id/ph_status"
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|bottom"
            android:text="Loading..."
            android:textAppearance="@style/LText" />

        <TextView
            android:layout_width="match_parent"
            android:layout_height="0dp"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="center|top"
            android:text="Status"
            android:textAppearance="@style/MText" />
    </LinearLayout>
</LinearLayout>
```

1.29 fragment_pump_slider.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">

    <LinearLayout
```

```
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:layout_margin="5dp"
        android:background="@android:color/transparent"
        android:orientation="vertical"
        android:padding="5dp">

        <TextView
            android:id="@+id/pumpSpeed"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:background="@android:color/transparent"
            android:text="Pump Speed : 0"
            android:textAppearance="@style/MText" />

        <SeekBar
            android:id="@+id/pumpSlider"
            android:layout_width="match_parent"
            android:layout_height="53dp"
            android:background="@android:color/transparent" />
    </LinearLayout>
</LinearLayout>
```

1.30 fragment_speed_picker.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="5dp"
    android:background="@android:color/transparent"
    android:padding="5dp">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_centerInParent="true"
    android:orientation="vertical"
    android:layout_height="wrap_content">
    <TextView
        android:id="@+id/pumpSpeed"
        android:layout_width="match_parent"
        android:layout_marginLeft="10dp"
        android:layout_height="wrap_content"
        android:background="@android:color/transparent"
        android:text="Pump Speed : 0" />

    <SeekBar
```

```
        android:id="@+id/pumpSlider"
        android:layout_width="match_parent"
        android:layout_height="53dp"
        android:layout_below="@id/pumpSpeed"
        android:background="@android:color/transparent" />
</LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:gravity="right"
        android:orientation="horizontal">

        <Button
            android:id="@+id/canel_btn"
            style="?android:attr/borderlessButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="cancel" />

        <Button
            android:id="@+id/ok_btn"
            style="?android:attr/borderlessButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Next" />


    </LinearLayout>

</RelativeLayout>
```

1.31 fragment_speed_picker_from_setup.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="5dp"
    android:background="@android:color/transparent"
    android:padding="5dp">

<LinearLayout
    android:layout_width="match_parent"
    android:layout_centerInParent="true"
    android:orientation="vertical"
    android:layout_height="wrap_content">
```

```
    <TextView
        android:id="@+id/pumpSpeed"
        android:layout_width="match_parent"
        android:layout_marginLeft="10dp"
        android:layout_height="wrap_content"
        android:background="@android:color/transparent"
        android:text="Pump Speed : 0" />

    <SeekBar
        android:id="@+id/pumpSlider"
        android:layout_width="match_parent"
        android:layout_height="53dp"
        android:layout_below="@id/pumpSpeed"
        android:background="@android:color/transparent" />
</LinearLayout>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:gravity="right"
        android:orientation="horizontal">

        <Button
            android:id="@+id/canel_btn"
            style="?android:attr/borderlessButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="cancel" />

        <Button
            android:id="@+id/ok_btn"
            style="?android:attr/borderlessButtonStyle"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Finish" />


    </LinearLayout>

</RelativeLayout>
```

1.32 fragment_statistics.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
    android:background="@drawable/statistics_big"
    android:orientation="vertical"
    tools:context="com.example.snoee.myapplistview.fragments.StatisticsFrag">

</LinearLayout>
```

1.33 fragment_weather.xml

```
<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_margin="5dp"
    android:padding="5dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_margin="5dp"

        android:background="@android:color/transparent"
        android:padding="5dp">

        <LinearLayout
            android:layout_width="0dp"
            android:layout_height="match_parent"
            android:layout_gravity="center"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:orientation="vertical">

            <TextView
                android:id="@+id/weather_icon"
                android:layout_width="match_parent"
                android:layout_height="0dp"
                android:layout_weight="1"
                android:background="@android:color/transparent"
                android:gravity="bottom|center"
                android:textAppearance="@style/LText" />

            <TextView
                android:id="@+id/details_field"
                android:layout_width="match_parent"
                android:layout_height="0dp"
                android:layout_weight="1"
                android:background="@android:color/transparent"
                android:gravity="top|center"
```

```
        android:textAppearance="@style/MText" />
</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_weight="1"
    android:background="@android:color/transparent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/wind_direction"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:gravity="bottom|center"
        android:textAppearance="@style/LText" />

    <TextView
        android:id="@+id/wind_speed"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:gravity="top|center"
        android:textAppearance="@style/MText" />
</LinearLayout>

<LinearLayout
    android:layout_width="0dp"
    android:layout_height="match_parent"
    android:layout_gravity="center"
    android:layout_weight="1"
    android:background="@android:color/transparent"
    android:orientation="vertical">

    <TextView
        android:id="@+id/current_temperature_field"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:background="@android:color/transparent"
        android:gravity="bottom|center"
        android:textAppearance="@style/LText" />

    <TextView
        android:id="@+id/humidity_field"
        android:layout_width="match_parent"
        android:layout_height="0dp"
```

```
            android:layout_centerHorizontal="true"
            android:layout_weight="1"
            android:background="@android:color/transparent"
            android:gravity="top|center"
            android:textAppearance="@style/MText" />
        </LinearLayout>
    </LinearLayout>

    <TextView
        android:id="@+id/city_field"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:background="@android:color/transparent"
        android:textAppearance="@style/CityName" />

    <ProgressBar
        android:id="@+id/loader"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:background="@android:color/transparent" />
</FrameLayout>
```

1.34 navigation_drawer.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <include layout="@layout/toolbar_layout" />

    <android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:id="@+id/drawer_layout"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="?attr/actionBarSize"
        android:fitsSystemWindows="false"
        tools:openDrawer="start">

        <include layout="@layout/activity_main" />

        <android.support.design.widget.NavigationView
            android:id="@+id/navigation_view"
```

```
            android:layout_width="wrap_content"
            android:layout_height="match_parent"
            android:layout_gravity="start"
            android:fitsSystemWindows="false"
            app:headerLayout="@layout/navigation_header"
            app:menu="@menu/navigation_menu" />
    </android.support.v4.widget.DrawerLayout>
</RelativeLayout>
```

1.35 navigation_header.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="160dp"
    android:background="@drawable/linear_gradient_drawable"
    android:gravity="bottom"
    android:orientation="vertical"
    android:padding="16dp"
    android:theme="@style/ThemeOverlay.AppCompat.Dark">

    <de.hdodenhof.circleimageview.CircleImageView
        android:layout_width="56dp"
        android:layout_height="56dp"
        android:contentDescription="my_image"
        android:scaleType="centerCrop"
        android:src="@drawable/face" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="8dp"
        android:text="Arpit Sharma"
        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textStyle="bold" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="4dp"
        android:text="sharma.arpit956@gmail.com"
        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textStyle="bold|italic" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:paddingTop="4dp"
```

```
        android:text="+44 7882113088"
        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textStyle="bold|italic" />
</LinearLayout>
```

1.36 toolbar_layout.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.Toolbar android:layout_width="match_parent"
    android:layout_height="?attr/actionBarSize"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:id="@+id/toolbar"
     app:theme="@style/ThemeOverlay.AppCompat.Dark.ActionBar"
    android:background="@color/Black"
    xmlns:android="http://schemas.android.com/apk/res/android"/>
```

1.37 menu_main.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context=".activities.BaseActivity">
    <item
        android:id="@+id/menu_settings_id"
        android:icon="@drawable/settings"
        android:title="Settings" />
    <item
        android:id="@+id/menu_developer_id"
        android:icon="@drawable/developer"
        android:title="Developer" />
    <item
        android:id="@+id/item3_id"
        android:icon="@drawable/power"
        android:title="Close" />
</menu>
```

1.38 navigation_menu.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <group android:checkableBehavior="single"></group>
```

```
    <item android:title="">
        <menu>
            <item
                android:id="@+id/settings_id"
                android:icon="@drawable/gear"
                android:title="Settings" />
            <item
                android:id="@+id/status_check_id"
                android:icon="@drawable/status_check"
                android:title="Check Status" />
            <item
                android:id="@+id/developer_id"
                android:icon="@drawable/developer"
                android:title="Developer" />
        </menu>
    </item>
</menu>
```

1.39 attrs.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<resources>
    <attr name="backgroundColor" format="color"/>
    <!--          -->
    <declare-styleable name="ChildFragment">
        <attr name="customTheme" format="reference" />
    </declare-styleable>
    <!--          -->
</resources>
```

1.40 colors.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#FFFFFF</color>
    <color name="colorPrimaryDark">#000000</color>
    <color name="colorAccent">#000000</color>

    <color name="colorPrimary_list_view">#515151</color>
    <color name="colorPrimaryDark_list_view">#000000</color>


    <color name="ColorPrimary_controlPanel">#33203A43</color>
    <color name="ColorPrimaryDark_controlPanel">#D80F2027</color>
    <color name="ColorAccent_controlPanel">#CFD8DC</color>
    <color name="ColorPrimaryLight_controlPanel">#D82C5364</color>
```

```
    <color name="ColorPrimary_statistics">#33ff9d00</color>
    <color name="ColorPrimaryDark_statistics">#ff8800</color>
    <color name="ColorAccent_statistics">#ffcc00</color>
    <color name="ColorPrimaryLight_statistics">#ffae00</color>

    <color name="ColorPrimary_agriculturalSpray">#33203A43</color>
    <color name="ColorPrimaryDark_agriculturalSpray">#D80F2027</color>
    <color name="ColorAccent_agriculturalSpray">#CFD8DC</color>
    <color name="ColorPrimaryLight_agriculturalSpray">#D82C5364</color>


    <color name="white">#FFFFFF</color>
    <color name="Black">#000000</color>

    <color name="translucentWhite1">#33FFFFFF</color>
    <color name="translucentWhite2">#66FFFFFF</color>

</resources>
```

1.41 dimens.xml

```
 <?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="agro_frag_height">80dp</dimen>
    <dimen name="fragment_margin">5dp</dimen>
    <dimen name="fragment_padding">5dp</dimen>

    <dimen name="dialog_width">325dp</dimen>
    <dimen name="dialog_height">415dp</dimen>


</resources>
```

1.42 strings.xml

```
 <resources>
    <string name="app_name">Intelligent Hydroponic System</string>
    <string-array name="titles">


        <item>Control Panel</item>
        <item>Statistics</item>
    </string-array>
    <string-array name="description">
```

```
        <item>Current status and manual control </item>
        <item>The graphs for the last month</item>
    </string-array>

    <string name="open_drawer">Open navigation drawer</string>
    <string name="close_drawer">Close navigation drawer</string>


</resources>
```

1.43 styles.xml

```
<resources>
    <!-- Base application theme. -->
    <style name="AppTheme" parent="Theme.AppCompat.Light.NoActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
        <item name="popupTheme">@style/PopupTheme</item>
         <item name="android:navigationBarColor">@color/colorPrimaryDark</item>
        <item name="android:textColor">@color/white</item>
    </style>

    <style name="PopupTheme" parent="Theme.AppCompat.Light">
        <item name="android:background">@color/colorPrimaryDark</item>
        <item name="android:textColor">@color/white</item>
    </style>

    <style name="AppTheme.NoActionBar">
        <item name="android:windowActionBar">false</item>
        <item name="android:windowNoTitle">true</item>
        <item name="android:windowDrawsSystemBarBackgrounds">true</item>
    </style>

    <style name="AppTheme.Agro" parent="Theme.AppCompat.Light.NoActionBar">
        <item name="colorPrimary">@color/ColorPrimary_agriculturalSpray</item>
        <item name="android:textColor">@color/white</item>
        <item name="colorPrimaryDark">@color/ColorPrimaryDark_agriculturalSpray</item>
        <item name="colorAccent">@color/ColorAccent_agriculturalSpray</item>
        <item name="android:background">@drawable/linear_gradient_drawable_agro</item>
     </style>

    <style name="SText" parent="@android:style/TextAppearance.Small">
        <item name="android:textSize">12sp</item>
        <item name="android:textColor">@color/white</item>
        <item name="android:textStyle">bold</item>
        <item name="android:typeface">sans</item>
```

```
    </style>

    <style name="MText" parent="@android:style/TextAppearance.Medium">
        <item name="android:textSize">15sp</item>
        <item name="android:textColor">@color/white</item>
        <item name="android:textStyle">bold</item>
        <item name="android:typeface">sans</item>
    </style>

    <style name="LText" parent="@android:style/TextAppearance.Large">
        <item name="android:textSize">18sp</item>
        <item name="android:textColor">@color/white</item>
        <item name="android:textStyle">bold</item>
        <item name="android:typeface">sans</item>
    </style>

    <style name="CityName" parent="@android:style/TextAppearance.Small">
        <item name="android:textSize">9sp</item>
        <item name="android:textColor">@color/white</item>
        <item name="android:textStyle">bold</item>
        <item name="android:typeface">sans</item>
    </style>

</resources>
```

2 Arduino Code

The code used for programming the micro-controller using the Arduino IDE is split in different files for readability as follows:

2.1 Main arduino Code for pre-trained model

1. ai.ino

```
#include <TensorFlowLite_ESP32.h>
#include "main_functions.h"
#include "constants.h"
#include "output_handler.h"
#include "ph_model_data.h"
#include "tensorflow/lite/experimental/micro/kernels/all_ops_resolver.h"
#include "tensorflow/lite/experimental/micro/micro_error_reporter.h"
#include "tensorflow/lite/experimental/micro/micro_interpreter.h"
#include "tensorflow/lite/schema/schema_generated.h"
#include "tensorflow/lite/version.h"

// Globals, used for compatibility with Arduino-style sketches.
```

```
namespace {
tflite::ErrorReporter* error_reporter = nullptr;
const tflite::Model* model = nullptr;
tflite::MicroInterpreter* interpreter = nullptr;
TfLiteTensor* input = nullptr;
TfLiteTensor* output = nullptr;
int inference_count = 0;

// Create an area of memory to use for input, output, and intermediate arrays.
// Finding the minimum value for your model may require some trial and error.
constexpr int kTensorArenaSize = 3.2 * 1024;
uint8_t tensor_arena[kTensorArenaSize];
}  // namespace

void ai_setup() {
  // Set up logging. Google style is to avoid globals or statics because of
  // lifetime uncertainty, but since this has a trivial destructor it's okay.
  // NOLINTNEXTLINE(runtime-global-variables)
  static tflite::MicroErrorReporter micro_error_reporter;
  error_reporter = &micro_error_reporter;

  // Map the model into a usable data structure. This doesn't involve any
  // copying or parsing, it's a very lightweight operation.
  model = tflite::GetModel(g_sine_model_data);
  if (model->version() != TFLITE_SCHEMA_VERSION) {
    error_reporter->Report(
      "Model provided is schema version %d not equal "
      "to supported version %d.",
      model->version(), TFLITE_SCHEMA_VERSION);
    return;
  }

  // This pulls in all the operation implementations we need.
  // NOLINTNEXTLINE(runtime-global-variables)
  static tflite::ops::micro::AllOpsResolver resolver;

  // Build an interpreter to run the model with.
  static tflite::MicroInterpreter static_interpreter(
    model, resolver, tensor_arena, kTensorArenaSize, error_reporter);
  interpreter = &static_interpreter;

  // Allocate memory from the tensor_arena for the model's tensors.
  TfLiteStatus allocate_status = interpreter->AllocateTensors();
  if (allocate_status != kTfLiteOk) {
    error_reporter->Report("AllocateTensors() failed");
    return;
  }

  // Obtain pointers to the model's input and output tensors.
  input = interpreter->input(0);
```

```
  output = interpreter->output(0);

  // Keep track of how many inferences we have performed.
  inference_count = 0;
}

void ai_loop() {

  // Calculate input to feed to the model, which is the value read by the pH sensor
  float x_val = pHValue;
  // Copy value to input buffer (tensor)
  input->data.f[0] = x_val;

  // Run inference
  TfLiteStatus invoke_status = interpreter->Invoke();
  if (invoke_status != kTfLiteOk) {
    error_reporter->Report("Invoke failed on input: %f\n", x_val);
  }

  // Read predicted y value from output buffer (tensor), which is the pump speed
  float y_val = output->data.f[0];

  // The output is made equal to the pump speed
  pumpSpeed = y_val;
  // The pump speed is converted into an ADC value for the pump to be controlled by PWM
  convertedPumpSpeed = map(pumpSpeed, 0, 100, 4, 9 );

}
```

2. battery.ino

```
 #include "RunningMedian.h" // A library for getting the running median of the sensor values for
#include "avdweb_VirtualDelay.h" // The delay library for delays without blocking.
default = millis
RunningMedian batterySamples = RunningMedian(10); // A runnng median of 10 consecutive samples
VirtualDelay batteryDelay;

void send_batteryLevel(int interval) {
  interval = interval * 1000;
  float batteryCorrectionFactor = 12.52 / 13.15;
  //  float batteryCorrectionFactor = 1;
  batteryDelay.start(interval); // calls while running are ignored
  if (batteryDelay.elapsed()) {
    int x = analogRead(A0);
    batterySamples.add(x);
    float adcValue = (float)((batterySamples.getMedian()) * (3.300 / 1024.000));
    float battVoltage = adcValue * batteryCorrectionFactor * 10;
    //    Serial.println();
    //    Serial.print("Read Voltage is : "); Serial.println(adcValue);
```

```
    //   Serial.print("Battery voltage is : "); Serial.print(battVoltage); Serial.println(" V ")
    int battPercentage = 100; // The value is hard coded, as it is just for being sent to the clo

    if  (! batteryFeed.publish(battPercentage)) {
      Serial.println(F("batery value publish Failed"));
    } else {
      Serial.print(F("batery value published: "));
      Serial.println(battPercentage);
    }
  }
}
```

3. constants.cpp

```
#include "constants.h"
// This is a small number so that it's easy to read the logs
const int kInferencesPerCycle = 1000;
```

4. constants.h

```
#ifndef TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_CONSTANTS_H_
#define TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_CONSTANTS_H_



// This constant represents the range of x values our model was trained on,
// which is from 0 to 14 for the pH value.
const float kXrange = 14.00f;

// This constant determines the number of inferences to perform across the range
// of x values defined above. Since each inference takes time, the higher this
// number, the more time it will take to run through the entire range. The value
// of this constant can be tuned so that one full cycle takes a desired amount
// of time. Since different devices take different amounts of time to perform
// inference, this value should be defined per-device.
extern const int kInferencesPerCycle;

#endif // TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_CONSTANTS_H_
```

5. getPumpSpeed.ino

```
// Get the pump speed via IOT in case the pump is being controlled by the pump fragment on the a
void get_pumpSpeed() {
  //  Check continously after every 50 milliseconds for a new value published to the feed
  Adafruit_MQTT_Subscribe *subscription;
  while ((subscription = mqtt.readSubscription(50))) {
    if (subscription == &pumpSpeedFeed) {
      int tempSpeed = atol((char *)pumpSpeedFeed.lastread);
```

```
      // If a new value is published, make it equal to the pump speed and print it on the serial
      if (pumpSpeed != tempSpeed) {
        pumpSpeed = tempSpeed;
        convertedPumpSpeed = map(pumpSpeed, 0, 100, 4, 9 );
        Serial.print(F("Got pump speed: "));
        Serial.println(pumpSpeed);
        Serial.println();
      }
    }
  }
}
```

6. getTemperature.ino

```
 #include "avdweb_VirtualDelay.h"
VirtualDelay weatherTemperatureDelay; // default = millis
void get_weatherTemperature(double interval) {
  interval = interval * 1000;
  weatherTemperatureDelay.start(interval); // calls while running are ignored
  if (weatherTemperatureDelay.elapsed()) {
    //  Check continously after every 50 milliseconds for a new value published to the feed
    Adafruit_MQTT_Subscribe *subscription;
    while ((subscription = mqtt.readSubscription(50))) {
      // If a new value is published, print it on the serial monitor
      if (subscription == &weatherTemperatureFeed) {
        Serial.print(F("Got weather temperature: "));
        double weatherTemperature = atof((char *)weatherTemperatureFeed.lastread);
        Serial.println(weatherTemperature);

      }
    }
  }
}
```

7. main.ino

```
 // for ESP 32 microcontroller
#include <WiFi.h>
#include <analogWrite.h>
// for NodeMCU microcontroller
//#include <ESP8266WiFi.h>

// pins
const int pHSensor_pin = 33; // pin for the pH sensor
int pump_pin = 17; // pin for the pump
const int trig_pin = 5; //trigger pin for the ultrasonic level sensor
const int echo_pin = 18; // Echo pin for the for ultrasonic level sensor
```

```
// Variables for actually printing stuff
double pumpSpeed = 0; // variable for the pump speed
int waterLevelValue = 0; // variable for the water level in the nutrient tank
double pHValue; // variable for the pH value
char  pHStatus[20] = ""; // Variable for storing pH status
constexpr float period = 2000; // Period (milliseconds)
int convertedPumpSpeed = 0; // ADC value for the pump speed after conversion

void setup() {
  Serial.begin(9600); // Begin the serial communication at 9600 baud rate
  pinMode(pump_pin, OUTPUT); // sets the pin as outputoutput
  digitalWrite(pump_pin, LOW); // set the pump initially off before the experiment
  pinMode(trig_pin, OUTPUT); // Sets the trig_pin as an Output
  pinMode(echo_pin, INPUT); // Sets the echo_pin as an Input
  delay(10); // delay 10 milliseconds
  MQTT_setup(); // Setup the MQTT
  ai_setup(); // Setup the AI
}

void loop() {

  MQTT_loop(); // Conduct MQTT operations continously
  send_pH(16); // Send the pH sensor reading to the cloud via IOT. The parameter is the seconds a
  send_liquidLevel(17); // Send the liquid level sensor reading to the cloud via IOT. The paramet
  send_batteryLevel(18); // Send the battery level to the cloud via IOT. The parameter is the sec
  get_weatherTemperature(19); // Get the temperature from the cloud. The parameter is the seconds
  calculatepH(); // Calculate the pH value

//  // For calculating the time taken per inference by the AI
//  unsigned long start_timestamp = micros();
//  // Get current timestamp and modulo with period
//  unsigned long timestamp = micros();
//  timestamp = timestamp % (unsigned long)period;
  ai_loop();
//  Serial.print("Time for inference (us): ");
//  Serial.println(micros() - start_timestamp);

  get_pumpSpeed(); // Get the pump speed via IOT in case the pump is being controlled by the pump
  send_pumpSpeed(20); // Send the pump speed calculated by the AI to the cloud via IOT. The param
  set_pumpSpeed(convertedPumpSpeed); // Control the speed of the pump using PWM
}
```

8. main_functions.h

```
/* Copyright 2019 The TensorFlow Authors. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at
```

```
    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
==============================================================================*/

#ifndef TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_MAIN_FUNCTIONS_H_
#define TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_MAIN_FUNCTIONS_H_

// Initializes all data needed for the example. The name is important, and needs
// to be setup() for Arduino compatibility.
void setup();

// Runs one iteration of data gathering and inference. This should be called
// repeatedly from the application code. The name needs to be loop() for Arduino
// compatibility.
void loop();

#endif // TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_MAIN_FUNCTIONS_H_
```

9. MQTT.ino

```
 // To subscribe to a new feed ctrl F --> addSubscription
// To publish to a new feed ctrl F --> addPublish

#include "Adafruit_MQTT.h"
#include "Adafruit_MQTT_Client.h"


/************************* WiFi Access Point *********************************/

#define WLAN_SSID     "Connectify-J6"
#define WLAN_PASS     "33333333"

/************************* Adafruit.io Setup *********************************/

#define AIO_SERVER    "io.adafruit.com"
#define AIO_SERVERPORT 1883                   // use 8883 for SSL
#define AIO_USERNAME  "arpitsmscproject"
#define AIO_KEY       "aio_GPgg95Ao3NOndl8EpBI9k2E3HxNJ"

// Create an ESP 32 WiFiClient class to connect to the MQTT server.
WiFiClient client;
// Setup the MQTT client class by passing in the WiFi client and MQTT server and login details.
Adafruit_MQTT_Client mqtt(&client, AIO_SERVER, AIO_SERVERPORT, AIO_USERNAME, AIO_KEY);
```

```
// addSubscription
// Subscribe to Feeds
Adafruit_MQTT_Subscribe weatherTemperatureFeed = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/fe
Adafruit_MQTT_Subscribe pumpSpeedFeed = Adafruit_MQTT_Subscribe(&mqtt, AIO_USERNAME "/feeds/pumpS

// addPublish
// Publish to Feeds
Adafruit_MQTT_Publish soilMoistureStatusFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/
Adafruit_MQTT_Publish soilMoistureInPercentageFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/
Adafruit_MQTT_Publish pumpSpeedPublishFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/pu
Adafruit_MQTT_Publish pHStatusFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/pHStatus")
Adafruit_MQTT_Publish pHFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/pH");

Adafruit_MQTT_Publish batteryFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/battery");

Adafruit_MQTT_Publish liquidLevelFeed = Adafruit_MQTT_Publish(&mqtt, AIO_USERNAME "/feeds/liquidL

void MQTT_loop() {
  int8_t ret;
  // Stop if already connected.
  if (mqtt.connected()) {
    return;
  }
  Serial.print("Connecting to MQTT... ");
  uint8_t retries = 3;
  while ((ret = mqtt.connect()) != 0) { // connect will return 0 for connected
    Serial.println(mqtt.connectErrorString(ret));
    Serial.println("Retrying MQTT connection in 5 seconds...");
    mqtt.disconnect();
    delay(5000); // wait 5 seconds
    retries--;
    if (retries == 0) {
      // basically die and wait for WDT to reset me
      while (1);
    }
  }
  Serial.println("MQTT Connected!");
  Serial.println();
  Serial.println();
  Serial.println();
}

void MQTT_setup() {
  Serial.println(F("Adafruit MQTT demo"));
  // Connect to WiFi access point.
  Serial.println(); Serial.println();
  Serial.print("Connecting to ");
  Serial.println(WLAN_SSID);
  WiFi.begin(WLAN_SSID, WLAN_PASS);
```

```
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println();
  Serial.println("WiFi connected");
  Serial.println("IP address: "); Serial.println(WiFi.localIP());


  // addSubscription
  mqtt.subscribe(&weatherTemperatureFeed);
  mqtt.subscribe(&pumpSpeedFeed);
}
```

10. output_handler.cpp

```
/* Copyright 2019 The TensorFlow Authors. All Rights Reserved.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
  ==============================================================================*/

#include "output_handler.h"

#include "Arduino.h"
#include "constants.h"

// The pin of the Arduino's built-in LED
#if defined(LED_BUILTIN)
int led = LED_BUILTIN;
#elif defined(ARDUINO_M5Stick_C)
int led = 10;
#else
int led = -1;
#endif

// Track whether the function has run at least once
bool initialized = false;

// Animates a dot across the screen to represent the current x and y values
```

```
void HandleOutput(tflite::ErrorReporter* error_reporter, float x_value,
                  float y_value) {
  // Do this only once
  if (!initialized) {
    // Set the LED pin to output
    if (led != -1) {
      pinMode(led, OUTPUT);
      ledcSetup(0, 1200, 8);
      ledcAttachPin(led, 0);
    }
    initialized = true;
  }

  // Calculate the brightness of the LED such that y=-1 is fully off
  // and y=1 is fully on. The LED's brightness can range from 0-255.
  int brightness = (int)(127.5f * (y_value + 1));

  // Set the brightness of the LED. If the specified pin does not support PWM,
  // this will result in the LED being on when y > 127, off otherwise.
  if (led != -1) {
    ledcWrite(0, brightness);
  }

  // Log the current brightness value for display in the Arduino plotter
  error_reporter->Report("%d", brightness);
}
```

11. output_handler.h

```
 /* Copyright 2019 The TensorFlow Authors. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
==============================================================================*/

#ifndef TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_OUTPUT_HANDLER_H_
#define TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_OUTPUT_HANDLER_H_

#include "tensorflow/lite/c/c_api_internal.h"
#include "tensorflow/lite/experimental/micro/micro_error_reporter.h"
```

```
// Called by the main loop to produce some output based on the x and y values
void HandleOutput(tflite::ErrorReporter* error_reporter, float x_value,
                  float y_value);

#endif // TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_OUTPUT_HANDLER_H_
```

12. ph_model_data.cpp

```
/* Copyright 2019 The TensorFlow Authors. All Rights Reserved.

  Licensed under the Apache License, Version 2.0 (the "License");
  you may not use this file except in compliance with the License.
  You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

  Unless required by applicable law or agreed to in writing, software
  distributed under the License is distributed on an "AS IS" BASIS,
  WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
  See the License for the specific language governing permissions and
  limitations under the License.
  ==============================================================================*/

// Automatically created from a TensorFlow Lite flatbuffer using the command:
// xxd -i sine_model.tflite > sine_model_data.cc
// See the README for a full description of the creation process.

#include "ph_model_data.h"

// We need to keep the data array aligned on some architectures.
#ifdef __has_attribute
#define HAVE_ATTRIBUTE(x) __has_attribute(x)
#else
#define HAVE_ATTRIBUTE(x) 0
#endif
#if HAVE_ATTRIBUTE(aligned) || (defined(__GNUC__) && !defined(__clang__))
#define DATA_ALIGN_ATTRIBUTE __attribute__((aligned(4)))
#else
#define DATA_ALIGN_ATTRIBUTE
#endif

const unsigned char g_sine_model_data[] DATA_ALIGN_ATTRIBUTE = {
  0x1c, 0x00, 0x00, 0x00, 0x54, 0x46, 0x4c, 0x33, 0x14, 0x00, 0x20, 0x00,
  0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00,
  0x18, 0x00, 0x1c, 0x00, 0x14, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00,
  0x18, 0x00, 0x00, 0x00, 0x24, 0x00, 0x00, 0x00, 0xdc, 0x00, 0x00, 0x00,
  0x24, 0x00, 0x00, 0x00, 0x84, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x00,
  0x03, 0x00, 0x00, 0x00, 0xa4, 0x07, 0x00, 0x00, 0xac, 0x06, 0x00, 0x00,
```

```
0xbc, 0x01, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0xdc, 0x00, 0x00, 0x00,
0x17, 0x00, 0x00, 0x00, 0xf4, 0x56, 0x00, 0x00, 0xf0, 0x56, 0x00, 0x00,
0xf0, 0x55, 0x00, 0x00, 0x3c, 0x55, 0x00, 0x00, 0xc4, 0x52, 0x00, 0x00,
0x34, 0x51, 0x00, 0x00, 0xbc, 0x30, 0x00, 0x00, 0x2c, 0x2e, 0x00, 0x00,
0xb4, 0x0d, 0x00, 0x00, 0x24, 0x0c, 0x00, 0x00, 0xac, 0x09, 0x00, 0x00,
0xfc, 0x08, 0x00, 0x00, 0x7c, 0x08, 0x00, 0x00, 0xe8, 0x07, 0x00, 0x00,
0xbc, 0x56, 0x00, 0x00, 0xb8, 0x56, 0x00, 0x00, 0xb4, 0x56, 0x00, 0x00,
0xb0, 0x56, 0x00, 0x00, 0xac, 0x56, 0x00, 0x00, 0xa8, 0x56, 0x00, 0x00,
0xa4, 0x56, 0x00, 0x00, 0xa0, 0x56, 0x00, 0x00, 0x3c, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x08, 0x00, 0x0c, 0x00, 0x04, 0x00, 0x08, 0x00, 0x08, 0x00, 0x00, 0x00,
0x08, 0x00, 0x00, 0x00, 0x16, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x6d, 0x69, 0x6e, 0x5f, 0x72, 0x75, 0x6e, 0x74, 0x69, 0x6d, 0x65, 0x5f,
0x76, 0x65, 0x72, 0x73, 0x69, 0x6f, 0x6e, 0x00, 0xa2, 0xaa, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x31, 0x2e, 0x31, 0x34,
0x2e, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x0f, 0x00, 0x00, 0x00, 0x4d, 0x4c, 0x49, 0x52, 0x20, 0x43, 0x6f, 0x6e,
0x76, 0x65, 0x72, 0x74, 0x65, 0x64, 0x2e, 0x00, 0x00, 0x00, 0x0e, 0x00,
0x18, 0x00, 0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x14, 0x00,
0x0e, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x68, 0x00, 0x00, 0x00,
0x6c, 0x00, 0x00, 0x00, 0x70, 0x00, 0x00, 0x00, 0x90, 0x00, 0x00, 0x00,
0x15, 0x00, 0x00, 0x00, 0xb8, 0x55, 0x00, 0x00, 0x28, 0x55, 0x00, 0x00,
0x7c, 0x54, 0x00, 0x00, 0xe4, 0x53, 0x00, 0x00, 0x54, 0x51, 0x00, 0x00,
0xdc, 0x4f, 0x00, 0x00, 0x4c, 0x2f, 0x00, 0x00, 0xd4, 0x2c, 0x00, 0x00,
0x44, 0x0c, 0x00, 0x00, 0xcc, 0x0a, 0x00, 0x00, 0x3c, 0x08, 0x00, 0x00,
0xa4, 0x07, 0x00, 0x00, 0x0c, 0x07, 0x00, 0x00, 0x8c, 0x06, 0x00, 0x00,
0x80, 0x05, 0x00, 0x00, 0x6c, 0x04, 0x00, 0x00, 0x70, 0x03, 0x00, 0x00,
0x84, 0x02, 0x00, 0x00, 0x98, 0x01, 0x00, 0x00, 0xf8, 0x00, 0x00, 0x00,
0x7c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x06, 0x00, 0x00, 0xfc, 0x04, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00,
0x04, 0x03, 0x00, 0x00, 0x18, 0x02, 0x00, 0x00, 0x2c, 0x01, 0x00, 0x00,
0x90, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x6d, 0x61, 0x69, 0x6e, 0x00, 0x00, 0x0a, 0x00, 0x10, 0x00, 0x04, 0x00,
0x08, 0x00, 0x0c, 0x00, 0x0a, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x14, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x2c, 0xfa, 0xff, 0xff, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06,
0x02, 0x00, 0x00, 0x00, 0x28, 0xab, 0xff, 0xff, 0x14, 0x00, 0x00, 0x00,
0x15, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x30, 0x00, 0x00, 0x00,
0x20, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x49, 0x64, 0x65, 0x6e,
0x74, 0x69, 0x74, 0x79, 0x00, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0xff, 0xff, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0x0c, 0xab, 0xff, 0xff,
0xae, 0xfc, 0xff, 0xff, 0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00,
0x18, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x28, 0xab, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x03, 0x00, 0x00, 0x00, 0x12, 0x00, 0x00, 0x00, 0x0b, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x98, 0xfa, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09,
0x14, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00,
```

```
0x30, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x09, 0x00, 0x00, 0x00,
0x49, 0x64, 0x65, 0x6e, 0x74, 0x69, 0x74, 0x79, 0x31, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00,
0x14, 0xac, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0xe1, 0x50, 0xc8, 0x3e, 0x01, 0x00, 0x00, 0x00,
0x80, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x46, 0xfd, 0xff, 0xff,
0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x3e, 0xfc, 0xff, 0xff,
0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00, 0x12, 0x00, 0x00, 0x00,
0x03, 0x00, 0x00, 0x00, 0x11, 0x00, 0x00, 0x00, 0x09, 0x00, 0x00, 0x00,
0x0a, 0x00, 0x00, 0x00, 0x34, 0xfb, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09,
0x14, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00,
0x7c, 0x00, 0x00, 0x00, 0x6c, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x55, 0x00, 0x00, 0x00,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x34, 0x2f, 0x4d, 0x61,
0x74, 0x4d, 0x75, 0x6c, 0x3b, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74,
0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f,
0x33, 0x34, 0x2f, 0x52, 0x65, 0x6c, 0x75, 0x3b, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x34, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64,
0x64, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0xfc, 0xac, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0xa2, 0xd0, 0x92, 0x3e,
0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x2e, 0xfe, 0xff, 0xff, 0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x26, 0xfd, 0xff, 0xff, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00,
0x11, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x07, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x1c, 0xfc, 0xff, 0xff,
0x00, 0x00, 0x00, 0x09, 0x14, 0x00, 0x00, 0x00, 0x12, 0x00, 0x00, 0x00,
0x18, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x00, 0x6c, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00,
0x55, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x33, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c, 0x3b, 0x73, 0x65, 0x71,
0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65,
0x6e, 0x73, 0x65, 0x5f, 0x33, 0x33, 0x2f, 0x52, 0x65, 0x6c, 0x75, 0x3b,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x33, 0x2f, 0x42, 0x69,
0x61, 0x73, 0x41, 0x64, 0x64, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0xff, 0xff, 0xff, 0xff, 0x40, 0x00, 0x00, 0x00, 0xe4, 0xad, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x65, 0xce, 0x8f, 0x3d, 0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0x16, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x08,
0x01, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x04, 0x00, 0x00, 0x00, 0x0e, 0xfe, 0xff, 0xff, 0x00, 0x00, 0x00, 0x01,
0x01, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00,
0x0f, 0x00, 0x00, 0x00, 0x05, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00,
```

```
0x04, 0xfd, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09, 0x14, 0x00, 0x00, 0x00,
0x11, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x00,
0x6c, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x80, 0x00, 0x00, 0x00, 0x55, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x32, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c,
0x3b, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f,
0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x32, 0x2f, 0x52,
0x65, 0x6c, 0x75, 0x3b, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x32, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64, 0x64, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xff, 0x80, 0x00, 0x00, 0x00,
0xcc, 0xae, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x10, 0xfa, 0xb5, 0x3c, 0x01, 0x00, 0x00, 0x00,
0x80, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x0e, 0x00,
0x18, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x07, 0x00, 0x14, 0x00,
0x0e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x06, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x01, 0x01, 0x00, 0x00, 0x00,
0x0f, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00, 0x0e, 0x00, 0x00, 0x00,
0x03, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0xfc, 0xfd, 0xff, 0xff,
0x00, 0x00, 0x00, 0x09, 0x14, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x18, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x00, 0x6c, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00,
0x55, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x31, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c, 0x3b, 0x73, 0x65, 0x71,
0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65,
0x6e, 0x73, 0x65, 0x5f, 0x33, 0x31, 0x2f, 0x52, 0x65, 0x6c, 0x75, 0x3b,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x31, 0x2f, 0x42, 0x69,
0x61, 0x73, 0x41, 0x64, 0x64, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0xff, 0xff, 0xff, 0xff, 0x40, 0x00, 0x00, 0x00, 0xc4, 0xaf, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x22, 0x59, 0xda, 0x3c, 0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff,
0xff, 0xff, 0xff, 0xff, 0x00, 0x00, 0x0e, 0x00, 0x1a, 0x00, 0x08, 0x00,
0x0c, 0x00, 0x10, 0x00, 0x07, 0x00, 0x14, 0x00, 0x0e, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00, 0x24, 0x00, 0x00, 0x00,
0x18, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x00,
0x08, 0x00, 0x07, 0x00, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x01,
0x01, 0x00, 0x00, 0x00, 0x0e, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00,
0x0d, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x18, 0xff, 0xff, 0xff, 0x09, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x09,
0x04, 0x00, 0x00, 0x00, 0x0c, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09,
0x14, 0x00, 0x00, 0x00, 0x0f, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00,
0x7c, 0x00, 0x00, 0x00, 0x6c, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x55, 0x00, 0x00, 0x00,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x30, 0x2f, 0x4d, 0x61,
0x74, 0x4d, 0x75, 0x6c, 0x3b, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74,
```

```
0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f,
0x33, 0x30, 0x2f, 0x52, 0x65, 0x6c, 0x75, 0x3b, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x30, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64,
0x64, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0xd4, 0xb0, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x11, 0xfd, 0x09, 0x3d,
0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x00, 0x00, 0x0a, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x04, 0x00, 0x08, 0x00,
0x0a, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x0d, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x0b, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x04, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x72, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x72, 0x01, 0x00, 0x00, 0x00, 0x14, 0x00, 0x1c, 0x00,
0x08, 0x00, 0x07, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00,
0x00, 0x00, 0x18, 0x00, 0x14, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x09,
0x14, 0x00, 0x00, 0x00, 0x0e, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00,
0x34, 0x00, 0x00, 0x00, 0x24, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x74, 0x66, 0x6c, 0x2e, 0x71, 0x75, 0x61, 0x6e, 0x74, 0x69, 0x7a, 0x65,
0x00, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0xff, 0xff, 0xff, 0xff,
0x01, 0x00, 0x00, 0x00, 0x94, 0xb1, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0xe1, 0xe0, 0x60, 0x3d,
0x01, 0x00, 0x00, 0x00, 0x80, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x2a, 0xb2, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x04, 0x00, 0x00, 0x00, 0x30, 0x00, 0x00, 0x00, 0x1e, 0xb2, 0xff, 0xff,
0x00, 0x00, 0x00, 0x02, 0x10, 0x00, 0x00, 0x00, 0x0d, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x35, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64,
0x64, 0x2f, 0x52, 0x65, 0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62,
0x6c, 0x65, 0x4f, 0x70, 0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63,
0x65, 0x00, 0x00, 0x00, 0x24, 0xb2, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0xe2, 0x38, 0x9d, 0x3b,
0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xba, 0xb2, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x08, 0x00, 0x00, 0x00, 0xe1, 0x27, 0xe9, 0x1a, 0xc7, 0xd4, 0x2f, 0x81,
0xb2, 0xb2, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09, 0x10, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x35, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c, 0x00, 0x00, 0x00, 0x00,
0xa4, 0xb2, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0xd1, 0x12, 0x89, 0x3c, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x36, 0xb3, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xbc, 0x01, 0x00, 0x00, 0xcf, 0xff, 0xff, 0xff, 0x6d, 0x00, 0x00, 0x00,
0x58, 0xff, 0xff, 0xff, 0xc7, 0xff, 0xff, 0xff, 0xb7, 0x01, 0x00, 0x00,
```

```
0x1e, 0x01, 0x00, 0x00, 0x46, 0xb3, 0xff, 0xff, 0x00, 0x00, 0x00, 0x02,
0x10, 0x00, 0x00, 0x00, 0x0b, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x34, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64, 0x64, 0x2f, 0x52, 0x65,
0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62, 0x6c, 0x65, 0x4f, 0x70,
0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63, 0x65, 0x00, 0x00, 0x00,
0x4c, 0xb3, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0xed, 0x7c, 0x0f, 0x3a, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xe2, 0xb3, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00,
0x1c, 0x06, 0x19, 0xe5, 0xf5, 0xe0, 0x02, 0xe4, 0xe3, 0xe6, 0xe0, 0x01,
0xde, 0x0a, 0xdd, 0xea, 0x1e, 0xeb, 0xec, 0x13, 0xf1, 0xfb, 0xed, 0xf4,
0x06, 0xf6, 0x15, 0xec, 0x00, 0x18, 0xed, 0xf2, 0x13, 0x15, 0xf8, 0xe9,
0xe5, 0xf3, 0xdf, 0x11, 0x1b, 0xe5, 0xe1, 0x18, 0x09, 0x1a, 0xe5, 0x1d,
0xf1, 0xea, 0xe2, 0x0f, 0xe4, 0xf1, 0x19, 0xdf, 0xe6, 0xf7, 0xee, 0x05,
0x01, 0xde, 0xde, 0xfb, 0xf6, 0x21, 0xe1, 0x0c, 0xed, 0x0b, 0x0a, 0x30,
0x16, 0xe5, 0x20, 0xed, 0xe5, 0xd8, 0xec, 0x34, 0x18, 0xf0, 0x21, 0xf2,
0xfd, 0x3c, 0x2e, 0x24, 0xff, 0x2e, 0x1e, 0x2e, 0xfc, 0x21, 0x0d, 0x53,
0x1d, 0x26, 0x11, 0x47, 0xf9, 0x1a, 0xdc, 0xe0, 0x14, 0x22, 0x06, 0x90,
0xf1, 0x14, 0x08, 0x2c, 0x1c, 0x31, 0x18, 0xe5, 0x2c, 0x34, 0x50, 0xde,
0x0e, 0xec, 0x10, 0x1c, 0x25, 0x1e, 0xbd, 0x24, 0xee, 0xe7, 0x17, 0xf7,
0x23, 0x0f, 0x09, 0x04, 0xf8, 0xe4, 0xda, 0xe6, 0x03, 0x21, 0x0e, 0xd7,
0xe9, 0x15, 0x04, 0x1d, 0x22, 0x0f, 0xfa, 0xf4, 0xe4, 0x15, 0xec, 0xff,
0xe0, 0xdb, 0xff, 0xe7, 0x12, 0xfb, 0xe9, 0x15, 0x14, 0x22, 0xdc, 0xf1,
0x19, 0xec, 0x18, 0xe6, 0x04, 0x09, 0xdf, 0x04, 0x0c, 0xfe, 0xdf, 0x1a,
0x17, 0xeb, 0x10, 0x0e, 0xe6, 0xe3, 0x18, 0xf6, 0xdc, 0x11, 0xe8, 0xf0,
0xfa, 0x04, 0x01, 0xe4, 0x1b, 0xfa, 0xf1, 0xce, 0x0d, 0xf8, 0xf1, 0x1f,
0xfd, 0xee, 0x1c, 0xfa, 0xec, 0x1d, 0xdd, 0x0c, 0xe8, 0xec, 0x18, 0xd4,
0xea, 0xda, 0xde, 0xe9, 0x0e, 0xe5, 0xe7, 0xe5, 0xdd, 0xcb, 0x0e, 0xfe,
0xf4, 0xf1, 0xe2, 0x23, 0x1d, 0xff, 0xf9, 0x0e, 0xe7, 0x16, 0xf4, 0xac,
0x04, 0x10, 0x0d, 0x0d, 0x0e, 0x07, 0x02, 0xff, 0x03, 0xe3, 0x11, 0xfe,
0xe3, 0x05, 0x18, 0xf9, 0xfd, 0xaf, 0x08, 0xf4, 0x01, 0xd3, 0xea, 0x0a,
0xe7, 0x21, 0xa8, 0x23, 0xfe, 0xea, 0x0b, 0xe8, 0x1a, 0xe7, 0x03, 0x21,
0xeb, 0x20, 0x22, 0x17, 0x0f, 0xf7, 0xfa, 0xbd, 0xf6, 0xe5, 0xea, 0xe6,
0xf9, 0x1f, 0xf1, 0x0a, 0xf4, 0x20, 0x1f, 0x00, 0x1a, 0x0e, 0x14, 0x3f,
0x1d, 0xf7, 0x0a, 0xf2, 0x05, 0x22, 0x10, 0xf6, 0x16, 0xeb, 0xda, 0xdc,
0xef, 0xef, 0x07, 0x18, 0x0e, 0x07, 0x29, 0xf1, 0x1e, 0xe4, 0x12, 0xef,
0x11, 0xe6, 0x19, 0xda, 0xed, 0xef, 0xd8, 0xeb, 0xe1, 0xff, 0xe8, 0xf7,
0xf5, 0xf7, 0x1c, 0xf0, 0x02, 0xee, 0xfd, 0xf0, 0x0a, 0xff, 0x06, 0x15,
0x0f, 0xe7, 0x03, 0xe7, 0x22, 0xed, 0xdc, 0x00, 0xf6, 0xe0, 0xe9, 0x0f,
0x0a, 0xe0, 0xf3, 0xdd, 0x06, 0xde, 0x17, 0xf5, 0xe9, 0x0d, 0x08, 0xff,
0x18, 0xec, 0x05, 0xf4, 0xeb, 0xfe, 0x10, 0x1d, 0xed, 0x12, 0xf5, 0xf9,
0xde, 0xfc, 0xe3, 0x23, 0x0e, 0x2b, 0xde, 0x2f, 0x0a, 0xf9, 0x12, 0x04,
0xe0, 0x1a, 0x07, 0x1a, 0x16, 0xf2, 0xdf, 0x0f, 0x09, 0x31, 0x1e, 0x15,
0xed, 0x23, 0x00, 0x48, 0xfd, 0xf7, 0x1d, 0x24, 0x10, 0x35, 0x0f, 0x1f,
0x07, 0xf5, 0xf6, 0xe8, 0x08, 0xfb, 0xf6, 0x9f, 0xe6, 0xf4, 0xe6, 0x2a,
0xde, 0x2a, 0xe9, 0xe2, 0x34, 0x1e, 0x1f, 0xf3, 0x0b, 0xf3, 0x39, 0x3d,
0xe2, 0xe9, 0xaa, 0x2a, 0xdf, 0xa3, 0xf8, 0xf4, 0xfc, 0x9d, 0x14, 0x23,
```

```
0xf5, 0xec, 0x8f, 0x0c, 0x1f, 0xef, 0xf2, 0xfc, 0xed, 0xf4, 0x0e, 0xf1,
0xf8, 0xee, 0xe1, 0xf4, 0x01, 0xfb, 0xf0, 0x81, 0xfa, 0x0a, 0x11, 0xc2,
0xdc, 0xf4, 0x0e, 0x26, 0xfb, 0xec, 0x16, 0xf5, 0x05, 0xdb, 0xe7, 0x43,
0x1d, 0x18, 0x07, 0x14, 0xdf, 0x25, 0x04, 0xef, 0x1e, 0x00, 0xb2, 0x18,
0x15, 0xe3, 0x29, 0x22, 0x16, 0x1c, 0x40, 0xf0, 0xd2, 0xb5, 0xff, 0xff,
0x00, 0x00, 0x00, 0x09, 0x10, 0x00, 0x00, 0x00, 0x0a, 0x00, 0x00, 0x00,
0x14, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x08, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x34, 0x2f, 0x4d, 0x61,
0x74, 0x4d, 0x75, 0x6c, 0x00, 0x00, 0x00, 0x00, 0xc4, 0xb5, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0xf9, 0x6e, 0xff, 0x3b, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x56, 0xb6, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xea, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x40, 0x01, 0x00, 0x00, 0xd6, 0xff, 0xff, 0xff,
0x5f, 0x00, 0x00, 0x00, 0xe1, 0xff, 0xff, 0xff, 0xde, 0x01, 0x00, 0x00,
0xd3, 0xff, 0xff, 0xff, 0xd8, 0xff, 0xff, 0xff, 0x4f, 0x01, 0x00, 0x00,
0xe1, 0xff, 0xff, 0xff, 0xcb, 0xff, 0xff, 0xff, 0xcb, 0xff, 0xff, 0xff,
0xdc, 0xff, 0xff, 0xff, 0x98, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x11, 0x00, 0x00, 0x00, 0xe4, 0xff, 0xff, 0xff, 0xcc, 0xff, 0xff, 0xff,
0xf2, 0xff, 0xff, 0xff, 0x57, 0x01, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x4b, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x79, 0x01, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x43, 0x00, 0x00, 0x00, 0xcb, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0xd3, 0xff, 0xff, 0xff, 0x54, 0x01, 0x00, 0x00,
0xcb, 0xff, 0xff, 0xff, 0x81, 0x01, 0x00, 0x00, 0xd7, 0xff, 0xff, 0xff,
0xfe, 0x01, 0x00, 0x00, 0xf8, 0xff, 0xff, 0xff, 0xf1, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0xec, 0xff, 0xff, 0xff, 0xa8, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x9b, 0xfc, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0xbc, 0xff, 0xff, 0xff, 0xcb, 0xff, 0xff, 0xff,
0xbb, 0x01, 0x00, 0x00, 0xf7, 0xff, 0xff, 0xff, 0x8a, 0x01, 0x00, 0x00,
0xf2, 0xff, 0xff, 0xff, 0xd3, 0xff, 0xff, 0xff, 0xe5, 0x01, 0x00, 0x00,
0xa3, 0x01, 0x00, 0x00, 0xa9, 0x01, 0x00, 0x00, 0xcb, 0xff, 0xff, 0xff,
0xbd, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x7f, 0x01, 0x00, 0x00,
0xb9, 0x01, 0x00, 0x00, 0xf7, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x95, 0xfd, 0xff, 0xff, 0x80, 0x01, 0x00, 0x00, 0x46, 0xb7, 0xff, 0xff,
0x00, 0x00, 0x00, 0x02, 0x10, 0x00, 0x00, 0x00, 0x09, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x40, 0x00, 0x00, 0x00, 0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x33, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64,
0x64, 0x2f, 0x52, 0x65, 0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62,
0x6c, 0x65, 0x4f, 0x70, 0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63,
0x65, 0x00, 0x00, 0x00, 0x4c, 0xb7, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x4a, 0xe2, 0x1b, 0x3a,
0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xe2, 0xb7, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x00, 0x20, 0x00, 0x00, 0xfe, 0x01, 0x06, 0xfb, 0x03, 0x05, 0xfd, 0x01,
0x02, 0xfd, 0x00, 0x00, 0xfc, 0xfe, 0x00, 0xfa, 0x02, 0x05, 0x00, 0xfa,
0x05, 0xfd, 0x03, 0xfa, 0x00, 0x03, 0x00, 0xfe, 0xfb, 0x04, 0xfd, 0x03,
```

```
0xfa, 0x01, 0xfe, 0x02, 0xfb, 0xfa, 0x03, 0xff, 0xff, 0x00, 0x03, 0xfd,
0xfb, 0xfe, 0x00, 0xfa, 0x01, 0xfd, 0xfb, 0x04, 0x05, 0xfa, 0x03, 0x04,
0x01, 0x00, 0x00, 0xfd, 0x05, 0xfc, 0x06, 0xfe, 0x04, 0x04, 0x02, 0x05,
0x00, 0xff, 0xfe, 0xfe, 0xfd, 0xfb, 0x06, 0xfc, 0x00, 0x02, 0xfd, 0x05,
0x02, 0xff, 0xfb, 0x04, 0xfc, 0xfb, 0xfc, 0xfa, 0x03, 0x04, 0x01, 0x02,
0xfb, 0x01, 0x06, 0x02, 0x04, 0x01, 0x05, 0x04, 0xff, 0x02, 0xff, 0xfd,
0xfe, 0x02, 0xfe, 0xff, 0xff, 0x05, 0x00, 0xfe, 0xfa, 0xfc, 0xfb, 0x05,
0xfd, 0xfc, 0xfe, 0xff, 0x00, 0x05, 0x06, 0x04, 0xfb, 0x02, 0xff, 0x03,
0xf5, 0x09, 0x05, 0xde, 0xff, 0xfd, 0x08, 0xf9, 0xfe, 0x02, 0xfe, 0x00,
0x08, 0x04, 0x06, 0x00, 0x06, 0xfe, 0xff, 0x02, 0x00, 0x02, 0x00, 0x04,
0x02, 0xfa, 0x05, 0x09, 0x05, 0xfb, 0x05, 0xfd, 0x07, 0x06, 0x01, 0x08,
0xfc, 0x03, 0x08, 0x03, 0x03, 0xe2, 0x04, 0x05, 0x05, 0x07, 0xff, 0xfe,
0xfa, 0x06, 0x03, 0x02, 0xfe, 0x03, 0xfe, 0xfd, 0x06, 0x03, 0x06, 0x01,
0xff, 0x00, 0xfe, 0xff, 0x02, 0xfa, 0xcf, 0x06, 0x0b, 0x04, 0xfe, 0xfe,
0x08, 0xff, 0x01, 0x02, 0x05, 0xc4, 0xff, 0x01, 0x01, 0x02, 0xfd, 0xfd,
0x02, 0x04, 0xff, 0x01, 0x02, 0x03, 0x05, 0x04, 0x01, 0xfc, 0xfd, 0xfb,
0xfd, 0x00, 0x05, 0x01, 0x05, 0x02, 0x0e, 0x04, 0x04, 0xfc, 0x04, 0x05,
0xfa, 0xfd, 0xff, 0x09, 0xfb, 0xff, 0x01, 0xfb, 0xfc, 0xfd, 0x02, 0x03,
0x01, 0x03, 0x02, 0x03, 0xff, 0x06, 0x01, 0x04, 0x06, 0xfb, 0xfc, 0xfd,
0xfe, 0x02, 0xfd, 0x00, 0x03, 0xfd, 0x04, 0xfa, 0xfb, 0xff, 0xfc, 0xfe,
0xff, 0xfb, 0xfb, 0xfe, 0xfc, 0xfc, 0x01, 0xfd, 0xfb, 0xfa, 0x02, 0xfb,
0x02, 0x04, 0x02, 0xfe, 0x00, 0xfe, 0x03, 0x01, 0xfa, 0xfd, 0x02, 0x00,
0xfd, 0xfe, 0x05, 0xfd, 0xfd, 0xfe, 0x00, 0x03, 0xfa, 0xfd, 0xff, 0xfe,
0xfb, 0x03, 0x00, 0x05, 0x03, 0x00, 0x00, 0x00, 0x05, 0xfa, 0x06, 0x06,
0x05, 0xfd, 0x01, 0xf9, 0xfa, 0x03, 0x02, 0x06, 0x05, 0xfc, 0x04, 0x01,
0x06, 0x02, 0x05, 0x02, 0xfe, 0x02, 0x05, 0xff, 0x03, 0xfb, 0x00, 0x01,
0xfc, 0xfe, 0x00, 0xfa, 0xfa, 0xfe, 0xfb, 0x05, 0x06, 0x00, 0x03, 0xfb,
0xfa, 0x00, 0xfb, 0x01, 0x04, 0x07, 0x01, 0xfb, 0xfb, 0x00, 0xfa, 0x04,
0xfa, 0xfe, 0x00, 0xfa, 0x04, 0x06, 0x01, 0x04, 0xfd, 0xfb, 0xfb, 0x02,
0x03, 0x01, 0xfe, 0xff, 0xfe, 0xfc, 0x1c, 0xd5, 0x04, 0x05, 0x01, 0xfd,
0x00, 0x12, 0x05, 0xff, 0x04, 0x03, 0x01, 0x03, 0xfd, 0xfb, 0x06, 0x00,
0x06, 0xfe, 0xfb, 0x03, 0x03, 0x06, 0x00, 0x16, 0xfd, 0xfe, 0xfb, 0xfe,
0x06, 0x06, 0xfb, 0x09, 0x02, 0xfe, 0xfe, 0x01, 0x06, 0xd2, 0xfd, 0xfe,
0x04, 0x02, 0x02, 0x01, 0x05, 0xfc, 0xfe, 0x04, 0x00, 0xfe, 0x02, 0xfe,
0x04, 0x03, 0xfc, 0xfd, 0x01, 0x00, 0x01, 0xff, 0x04, 0xfd, 0xf1, 0x01,
0x05, 0xfc, 0xfc, 0x04, 0x07, 0x06, 0xff, 0xfe, 0x02, 0xf4, 0x00, 0x03,
0x06, 0xff, 0xfd, 0x00, 0x08, 0xfb, 0xfb, 0x01, 0x05, 0xfb, 0x04, 0x06,
0x00, 0x07, 0x08, 0x05, 0xfa, 0x01, 0x04, 0x02, 0xfb, 0x08, 0x08, 0xfc,
0xfe, 0x04, 0x02, 0xfd, 0xfb, 0x04, 0xff, 0x00, 0x00, 0x01, 0x04, 0xfe,
0xfe, 0x04, 0xfe, 0x05, 0x04, 0x05, 0xfb, 0x04, 0xfd, 0x04, 0x05, 0xfa,
0x01, 0x05, 0xfb, 0xfa, 0x02, 0x03, 0x04, 0xfd, 0x01, 0xf9, 0xfe, 0xfb,
0x00, 0xfd, 0xfa, 0xfe, 0xff, 0x00, 0x05, 0x01, 0xff, 0xfa, 0x04, 0x01,
0xf9, 0xfc, 0x01, 0xf9, 0xfa, 0x06, 0xfa, 0x01, 0x02, 0x01, 0xfa, 0x04,
0xfc, 0x01, 0xff, 0x06, 0x02, 0xfb, 0x05, 0xfe, 0x05, 0xf9, 0x02, 0x03,
0x05, 0xfd, 0xfe, 0xf9, 0x00, 0x02, 0x02, 0x02, 0xfc, 0x01, 0xfe, 0xfb,
0xfd, 0xf9, 0x04, 0x05, 0xff, 0x01, 0x07, 0x01, 0x03, 0x02, 0xfd, 0x03,
0xfe, 0x05, 0xfc, 0xff, 0xf9, 0x06, 0xfb, 0x04, 0x00, 0xfb, 0xfa, 0x01,
0x02, 0xfe, 0x06, 0xfb, 0x06, 0xff, 0x05, 0x05, 0xfc, 0x04, 0xfa, 0xfb,
0xfa, 0xfb, 0xfb, 0x03, 0x02, 0xf8, 0xfa, 0x06, 0x01, 0x02, 0xfc, 0xfa,
0xff, 0x05, 0x06, 0xfb, 0xfa, 0x01, 0x06, 0x02, 0x06, 0xfd, 0xfb, 0xfc,
```

```
0xfb, 0xfd, 0xfc, 0xf8, 0xfe, 0x01, 0xfd, 0x02, 0xfc, 0xfe, 0x18, 0xcc,
0xfe, 0x03, 0x06, 0xf6, 0xfd, 0x10, 0x01, 0x00, 0x07, 0x03, 0x04, 0xfc,
0xfe, 0x01, 0xfb, 0xfb, 0xfc, 0x07, 0x05, 0x00, 0x04, 0xfa, 0x00, 0x14,
0xfb, 0x05, 0x04, 0xfe, 0xfb, 0xff, 0xfc, 0x0d, 0x02, 0x00, 0x00, 0x02,
0x05, 0xd8, 0xfd, 0xfe, 0x05, 0x03, 0xfb, 0xfe, 0x03, 0xff, 0x03, 0x01,
0x02, 0x05, 0x00, 0x03, 0x04, 0x04, 0x08, 0x06, 0xfd, 0x06, 0xfb, 0x06,
0x03, 0xfd, 0xd0, 0xff, 0x12, 0xff, 0x03, 0xfd, 0x02, 0xfc, 0x01, 0x06,
0xfe, 0xcc, 0xfb, 0x04, 0x02, 0xfd, 0xff, 0x03, 0xff, 0x06, 0x06, 0x02,
0x03, 0xfe, 0xfe, 0xff, 0xfa, 0x04, 0xfd, 0x03, 0x06, 0x02, 0x06, 0x04,
0x05, 0x04, 0x14, 0x04, 0xfe, 0xf1, 0x06, 0xfa, 0xf1, 0xfd, 0xfe, 0x06,
0x05, 0x02, 0x00, 0x02, 0xff, 0x05, 0xfa, 0x05, 0x02, 0x03, 0xfa, 0x02,
0xfd, 0x06, 0x02, 0xfb, 0xfa, 0x00, 0x06, 0xfc, 0xfa, 0xfc, 0x01, 0xf9,
0xf9, 0xfa, 0xfe, 0x02, 0xfa, 0xf9, 0xfa, 0x03, 0x02, 0x04, 0xfd, 0x04,
0xf9, 0xfc, 0x05, 0xfe, 0x01, 0x00, 0xfd, 0xfa, 0xff, 0xfb, 0x02, 0x03,
0x04, 0xfd, 0xfe, 0xfb, 0xfa, 0x05, 0xfc, 0x03, 0xfe, 0xfe, 0x05, 0x02,
0xfd, 0xfd, 0x01, 0x02, 0x06, 0xfe, 0x06, 0x03, 0x03, 0x01, 0x03, 0x03,
0x05, 0xfe, 0xf9, 0xfc, 0xfa, 0x03, 0xff, 0x00, 0x06, 0x02, 0x03, 0x05,
0x03, 0xff, 0x05, 0xf9, 0x03, 0xff, 0xfa, 0xfe, 0xfc, 0x01, 0xfd, 0xfc,
0xfd, 0x02, 0x04, 0x03, 0x03, 0xfe, 0xfe, 0xff, 0xfa, 0xfe, 0x02, 0x06,
0xff, 0xfb, 0xfb, 0x00, 0x03, 0xfc, 0x05, 0xfc, 0x04, 0xfd, 0x00, 0x04,
0x01, 0xf9, 0x02, 0x06, 0xfc, 0xfd, 0xfb, 0xf9, 0x00, 0x00, 0x05, 0x04,
0x02, 0xff, 0x03, 0xfc, 0xfd, 0x01, 0x05, 0xfb, 0x05, 0x04, 0x07, 0x02,
0xf9, 0x08, 0x26, 0xb5, 0xfe, 0x05, 0x02, 0x00, 0xfb, 0x1c, 0xfd, 0x05,
0xfe, 0x01, 0x00, 0x02, 0xfe, 0x09, 0xfb, 0xfe, 0x05, 0xfe, 0x07, 0x03,
0x05, 0x06, 0xfe, 0x2a, 0x02, 0x01, 0xfb, 0xf9, 0x02, 0x09, 0x02, 0x0f,
0x01, 0x01, 0x03, 0x00, 0xf9, 0x98, 0x04, 0xfc, 0x02, 0xfd, 0x01, 0xfa,
0x06, 0x09, 0xff, 0x09, 0x01, 0xf9, 0x07, 0x00, 0x05, 0xfc, 0x04, 0xfd,
0x08, 0x07, 0xff, 0x03, 0x06, 0x03, 0xf7, 0x08, 0x0b, 0x09, 0x01, 0x04,
0x08, 0x0b, 0xff, 0xfd, 0x08, 0xfe, 0x01, 0x09, 0xff, 0x05, 0xfc, 0xfa,
0x05, 0x06, 0x04, 0x05, 0xfb, 0x04, 0xfe, 0x06, 0xfa, 0xff, 0x0a, 0xfb,
0xfb, 0x02, 0x01, 0xfd, 0xfd, 0x00, 0x03, 0x00, 0x03, 0xfb, 0x07, 0xfb,
0xfe, 0x04, 0x04, 0x05, 0xfa, 0xfe, 0x00, 0x02, 0xfa, 0x03, 0x05, 0xf9,
0xfb, 0xfb, 0xfc, 0x08, 0x08, 0x05, 0xff, 0x06, 0x03, 0x02, 0xfb, 0xff,
0x03, 0x05, 0x02, 0xff, 0x05, 0xfc, 0xf9, 0x03, 0xfb, 0xfd, 0xff, 0xfd,
0xfa, 0x02, 0xfc, 0x00, 0x00, 0xfb, 0xfa, 0x03, 0xfe, 0xfb, 0x04, 0x04,
0x01, 0x05, 0x01, 0xfa, 0xf9, 0x04, 0x01, 0x03, 0x00, 0x01, 0xf9, 0x03,
0x01, 0x04, 0xfd, 0x05, 0x02, 0x01, 0x00, 0xfc, 0xfe, 0xfc, 0xfd, 0xf9,
0xfe, 0x01, 0x03, 0x03, 0x05, 0x01, 0xfa, 0x03, 0x04, 0x03, 0xfc, 0x02,
0xfb, 0xfa, 0xf9, 0xf9, 0xfa, 0xff, 0xfe, 0x03, 0x05, 0xfb, 0x02, 0xff,
0xfd, 0xfc, 0x04, 0xfb, 0xff, 0x06, 0x02, 0xfa, 0xf8, 0x03, 0x04, 0x05,
0xfc, 0xfc, 0x04, 0x06, 0xfb, 0x06, 0x02, 0xff, 0x04, 0xfb, 0x05, 0xfa,
0x04, 0xfb, 0x00, 0x03, 0x02, 0xfd, 0x04, 0xfd, 0xf9, 0x03, 0xfd, 0x01,
0x02, 0xfa, 0x01, 0x04, 0xf9, 0xfc, 0x05, 0x06, 0x01, 0xfb, 0x05, 0x03,
0xfa, 0x02, 0xff, 0xfc, 0x04, 0xf9, 0xff, 0x01, 0xff, 0x02, 0xfe, 0xfb,
0x02, 0x01, 0xfd, 0x03, 0x04, 0x02, 0x05, 0x01, 0xf9, 0x05, 0x03, 0x01,
0x05, 0xf9, 0xff, 0xfc, 0x02, 0x02, 0xff, 0xfd, 0x05, 0x04, 0x04, 0xff,
0x00, 0xfb, 0xfc, 0x01, 0xfb, 0xff, 0x04, 0xfc, 0x04, 0xfc, 0xfa, 0xfc,
0x02, 0x00, 0x03, 0x01, 0x05, 0x01, 0xff, 0xfa, 0xf9, 0x05, 0xfa, 0xfe,
0xff, 0x04, 0xfa, 0xfc, 0x00, 0x06, 0xfb, 0x05, 0x00, 0xfe, 0x03, 0x03,
0x00, 0xfe, 0x03, 0x03, 0x02, 0x04, 0xfa, 0x03, 0x04, 0x05, 0xfc, 0xfe,
```

```
0x00, 0x00, 0x02, 0xfc, 0x03, 0xff, 0x00, 0xfa, 0xfb, 0x02, 0x01, 0x00,
0x00, 0x01, 0x03, 0x04, 0x03, 0x06, 0x03, 0x03, 0xfd, 0x00, 0x04, 0xff,
0xff, 0xfc, 0x02, 0xfc, 0x05, 0x00, 0x04, 0xfb, 0x06, 0x04, 0x03, 0x02,
0x04, 0xfc, 0x01, 0xfe, 0x05, 0x00, 0xfb, 0x01, 0xff, 0x02, 0x05, 0xfe,
0xf2, 0x04, 0x14, 0xd5, 0x01, 0x04, 0xfc, 0xfe, 0x01, 0x07, 0x03, 0x06,
0xff, 0x01, 0xfb, 0xfc, 0x04, 0x06, 0xff, 0x01, 0x06, 0x07, 0x01, 0x06,
0x04, 0xff, 0x03, 0x15, 0xfa, 0x04, 0x00, 0x03, 0xfd, 0xff, 0x04, 0x0e,
0x00, 0xfb, 0xfd, 0x00, 0x00, 0xdc, 0xfb, 0xfc, 0x00, 0x01, 0x01, 0x04,
0xfd, 0xfd, 0x00, 0xfe, 0x00, 0x04, 0x01, 0xfd, 0x00, 0xfb, 0xff, 0x02,
0x07, 0xfd, 0x00, 0xfe, 0x05, 0xfb, 0xca, 0xff, 0x0a, 0xff, 0xfd, 0x07,
0xfc, 0x03, 0x03, 0x04, 0x02, 0xc8, 0xfc, 0x05, 0x03, 0x00, 0x03, 0x07,
0xff, 0xfe, 0x05, 0xfe, 0x01, 0x03, 0xff, 0xff, 0xfc, 0xfc, 0x09, 0xff,
0x00, 0x03, 0x04, 0x01, 0x05, 0xfe, 0x10, 0xfd, 0xfb, 0xf8, 0x06, 0xfb,
0xfd, 0x02, 0xfc, 0x08, 0x00, 0x03, 0x06, 0xfb, 0x00, 0x01, 0x01, 0xfd,
0xfe, 0x05, 0x03, 0x08, 0xfd, 0x04, 0xfa, 0xfd, 0x02, 0x00, 0x00, 0x00,
0x02, 0x02, 0xfd, 0xfa, 0x06, 0xfe, 0xfe, 0xfd, 0x00, 0xfd, 0x00, 0x04,
0x05, 0x02, 0xfc, 0xff, 0xff, 0x00, 0xfc, 0xfb, 0x02, 0x05, 0x04, 0x06,
0xfe, 0x06, 0xff, 0xfe, 0xfa, 0xfa, 0x02, 0x05, 0xff, 0xfe, 0x03, 0x01,
0xfc, 0xfa, 0x01, 0xfd, 0x04, 0xf9, 0x04, 0x01, 0xfe, 0xf9, 0x01, 0x03,
0x03, 0xfc, 0xfa, 0xfe, 0x00, 0xfb, 0xfb, 0x03, 0xfe, 0x00, 0x03, 0x05,
0x02, 0xfb, 0x04, 0x03, 0xfc, 0x01, 0x02, 0x00, 0x00, 0x02, 0x00, 0x03,
0x04, 0x03, 0xfc, 0xfe, 0x00, 0x02, 0xf9, 0xfe, 0xfc, 0xfb, 0x04, 0xff,
0x04, 0x00, 0xfe, 0xfa, 0xfd, 0xfb, 0x03, 0x04, 0xfb, 0x03, 0xfb, 0xfc,
0x05, 0xfe, 0x05, 0x05, 0xfe, 0xfe, 0xfe, 0x03, 0x06, 0x05, 0x00, 0xfe,
0xfc, 0xfc, 0xff, 0x05, 0x04, 0xfe, 0x03, 0x01, 0x01, 0xfe, 0xfb, 0xfb,
0x00, 0xfa, 0x02, 0x03, 0xfb, 0x01, 0x05, 0x01, 0xfe, 0x00, 0xfa, 0xfb,
0xfc, 0x04, 0x01, 0x04, 0xfd, 0xfe, 0xfe, 0xf9, 0x02, 0xfb, 0xfa, 0x02,
0xfb, 0xf9, 0xfb, 0x01, 0x00, 0xfa, 0xff, 0xf9, 0xfe, 0x05, 0xff, 0x01,
0x03, 0x05, 0x06, 0x02, 0xfb, 0xfb, 0x02, 0xfd, 0xf9, 0x02, 0x03, 0xff,
0xfa, 0xfd, 0x05, 0xfd, 0xfa, 0xff, 0xfa, 0xfd, 0xfe, 0x04, 0x02, 0xfa,
0xff, 0x04, 0x02, 0x06, 0xfa, 0xfb, 0x06, 0xf9, 0x00, 0xfd, 0xfc, 0xff,
0x04, 0x05, 0x03, 0x05, 0xf8, 0xfe, 0xfe, 0xfb, 0x04, 0xfe, 0xff, 0x04,
0x05, 0x04, 0x03, 0xf9, 0xf9, 0xfb, 0x04, 0xfd, 0x01, 0x02, 0xfe, 0xf9,
0x04, 0x06, 0xf9, 0xfe, 0xfc, 0x06, 0xfa, 0x04, 0x00, 0x01, 0xfc, 0x05,
0xfa, 0x05, 0xf8, 0x01, 0x04, 0xfb, 0x04, 0xfe, 0xfc, 0x01, 0xfb, 0xfd,
0x05, 0x05, 0xf9, 0x06, 0x01, 0xff, 0xfb, 0x05, 0xfb, 0xf8, 0x04, 0x02,
0x04, 0x03, 0xf9, 0x04, 0xfb, 0x01, 0xfa, 0xfa, 0xff, 0x03, 0xfd, 0xfc,
0xfe, 0x04, 0xfd, 0xfb, 0xfc, 0xff, 0x04, 0xfd, 0xfe, 0xfc, 0xfc, 0xfb,
0xf9, 0xfb, 0xfb, 0xfe, 0x03, 0xfe, 0xfa, 0x04, 0x05, 0xfb, 0xff, 0xff,
0x06, 0xfc, 0x01, 0xfe, 0x04, 0x03, 0x06, 0xff, 0xfc, 0x03, 0x06, 0x06,
0x04, 0xfe, 0x00, 0xfa, 0xfd, 0xfe, 0x01, 0x01, 0xfb, 0xfb, 0x01, 0xff,
0xfc, 0xfc, 0xfc, 0x01, 0x05, 0x02, 0xfb, 0xfa, 0x03, 0x04, 0x03, 0x06,
0x05, 0x02, 0x01, 0xfe, 0xfe, 0xfc, 0x04, 0xfe, 0xfe, 0xff, 0xff, 0xfc,
0x02, 0x00, 0x06, 0x00, 0x00, 0x04, 0x03, 0x04, 0xfb, 0xfd, 0xfd, 0xfa,
0x02, 0xff, 0x03, 0xfa, 0x05, 0xfa, 0x03, 0xfc, 0x02, 0x03, 0xfd, 0xfd,
0x05, 0x05, 0xfc, 0xfb, 0x04, 0x02, 0xfc, 0x00, 0x03, 0xfb, 0x05, 0xfe,
0x02, 0x02, 0xfc, 0x00, 0xfe, 0xfb, 0xfb, 0x02, 0x03, 0x04, 0x03, 0xfe,
0x04, 0xfc, 0xfb, 0x01, 0x03, 0xff, 0x06, 0xfd, 0xfa, 0xfa, 0xf9, 0x01,
0x01, 0x04, 0x05, 0xfc, 0xfe, 0xfd, 0x05, 0x02, 0xfd, 0xff, 0x00, 0xfb,
0x04, 0x03, 0x01, 0xfb, 0x03, 0xfc, 0xfa, 0x04, 0x02, 0x00, 0x00, 0x01,
```

```
0xfc, 0x01, 0x03, 0x03, 0x01, 0xfd, 0x03, 0xfc, 0x04, 0xfd, 0xfe, 0xfb,
0xfd, 0x06, 0x03, 0x03, 0xfb, 0xfb, 0xfe, 0xfe, 0x05, 0x01, 0x06, 0x02,
0x02, 0xfb, 0x06, 0x01, 0x00, 0x00, 0xf9, 0x01, 0xfa, 0x02, 0x05, 0x06,
0xf9, 0xf9, 0xfb, 0xfe, 0xfa, 0x00, 0x00, 0xfd, 0x05, 0xfb, 0x04, 0x02,
0x03, 0xfe, 0xfa, 0x01, 0xff, 0xfa, 0xff, 0x02, 0xfe, 0x05, 0xfa, 0xfc,
0x03, 0xfe, 0x05, 0x00, 0xfe, 0xfd, 0x04, 0x00, 0x03, 0xfc, 0x03, 0xfb,
0x05, 0xfc, 0x05, 0x06, 0xfa, 0x02, 0x01, 0x05, 0xfa, 0xfd, 0x01, 0x02,
0x05, 0x01, 0x04, 0x05, 0xf9, 0x02, 0x22, 0xce, 0x05, 0x08, 0xff, 0xfd,
0xfa, 0x19, 0x00, 0xfe, 0x06, 0xfb, 0xfa, 0x07, 0xff, 0xfe, 0xfc, 0xfd,
0xfe, 0x08, 0x05, 0x04, 0x04, 0x06, 0xff, 0x1c, 0x01, 0x04, 0x03, 0x03,
0x01, 0x07, 0xfd, 0x07, 0xfd, 0xfc, 0x02, 0xfb, 0xfe, 0xc7, 0xfd, 0xfc,
0x04, 0xff, 0xfa, 0x03, 0x06, 0xfd, 0x02, 0x08, 0x05, 0xfd, 0x01, 0x01,
0x05, 0x02, 0x04, 0x05, 0x05, 0x01, 0x04, 0xfd, 0xff, 0x02, 0xf1, 0x05,
0x12, 0x06, 0x03, 0xfb, 0x00, 0x00, 0xfc, 0x00, 0xfb, 0xf2, 0x03, 0x04,
0xfc, 0x03, 0xfb, 0xfe, 0x05, 0x05, 0xfa, 0xfd, 0xfe, 0xfc, 0xfe, 0x08,
0x03, 0x00, 0x07, 0xfb, 0xfe, 0xfb, 0x02, 0x00, 0xfc, 0x07, 0x04, 0xfb,
0xff, 0xfb, 0x02, 0x05, 0xfb, 0xfa, 0x03, 0x00, 0x02, 0xfc, 0x06, 0xfc,
0xfc, 0x00, 0x05, 0x02, 0xfa, 0x06, 0xfb, 0x00, 0xff, 0xfd, 0xff, 0xfa,
0x04, 0xff, 0x03, 0xfd, 0xfe, 0x05, 0xfc, 0xfc, 0x03, 0x04, 0xfc, 0x04,
0xfd, 0xff, 0x03, 0xff, 0x00, 0x01, 0xfa, 0x01, 0x01, 0xfd, 0x04, 0xfd,
0xfe, 0x02, 0xfd, 0xfb, 0x02, 0xfb, 0xfd, 0xfe, 0xfb, 0xfe, 0xfe, 0xff,
0x04, 0x01, 0x00, 0xfd, 0xfd, 0xfc, 0x03, 0x05, 0xfd, 0xfd, 0xfc, 0x02,
0x05, 0xff, 0xff, 0x05, 0xfd, 0x02, 0xfb, 0x03, 0x01, 0x02, 0xff, 0xfd,
0x02, 0xfa, 0x06, 0xfb, 0xff, 0xff, 0x01, 0x06, 0xff, 0xff, 0x03, 0x02,
0xfc, 0xff, 0xfc, 0xfb, 0x03, 0xfa, 0xfb, 0x01, 0x02, 0xfc, 0xfa, 0x05,
0x03, 0x05, 0x01, 0xfd, 0xfb, 0xfd, 0xfe, 0xfe, 0x05, 0xfd, 0xfb, 0xff,
0x05, 0x04, 0x02, 0xfa, 0xfc, 0xfe, 0xff, 0xfd, 0xfe, 0x00, 0x05, 0xfe,
0xfe, 0xfc, 0x01, 0x05, 0x05, 0x05, 0x05, 0x00, 0xfd, 0x04, 0xfa, 0x02,
0x04, 0x03, 0xfc, 0x01, 0xfe, 0x02, 0x03, 0x04, 0xfe, 0x01, 0x04, 0xfd,
0xfd, 0x02, 0x06, 0xfa, 0xfc, 0x05, 0xff, 0x05, 0x00, 0xfe, 0xfc, 0x04,
0x03, 0xff, 0xfb, 0x01, 0xfb, 0xff, 0xf9, 0xff, 0xfc, 0x04, 0xfc, 0x00,
0xfc, 0x01, 0x02, 0x01, 0xfe, 0x03, 0x06, 0xfc, 0xff, 0xff, 0xfa, 0x01,
0xfc, 0x05, 0x03, 0xff, 0xff, 0x00, 0x00, 0x00, 0xfc, 0xfb, 0x06, 0xf9,
0xfe, 0x06, 0xf9, 0x02, 0x06, 0x03, 0xfd, 0xfb, 0x02, 0x03, 0xfb, 0xfc,
0xfc, 0x03, 0x00, 0x01, 0xf9, 0xfd, 0xfd, 0x00, 0x02, 0x01, 0x05, 0xfe,
0x00, 0xfe, 0x06, 0xfb, 0x06, 0xfa, 0x03, 0x04, 0xfa, 0xfe, 0xff, 0x02,
0x06, 0x05, 0x00, 0xfa, 0xfe, 0xff, 0x02, 0xfd, 0xfa, 0xff, 0xfd, 0x03,
0xfa, 0xfd, 0xfc, 0x06, 0x05, 0xfb, 0x03, 0x01, 0x01, 0x03, 0xfa, 0x02,
0xfe, 0xfe, 0xfb, 0xff, 0x00, 0x03, 0x02, 0xfa, 0xff, 0x06, 0xfb, 0x04,
0xfc, 0x03, 0x00, 0xff, 0xf9, 0x01, 0xfe, 0x01, 0x02, 0x01, 0xff, 0x04,
0x02, 0xff, 0xfa, 0x02, 0x04, 0x03, 0x03, 0xfc, 0xfd, 0xff, 0x05, 0x03,
0x00, 0xfb, 0xff, 0xff, 0xfa, 0x02, 0xff, 0xfb, 0x01, 0x01, 0xfb, 0x05,
0xff, 0x03, 0x04, 0x05, 0x05, 0x00, 0xf9, 0xfe, 0xfa, 0x01, 0x02, 0x01,
0xfc, 0x02, 0x02, 0x01, 0x01, 0x02, 0x07, 0xfb, 0x04, 0x01, 0xfc, 0x06,
0x01, 0xfe, 0x03, 0x00, 0x02, 0x01, 0xfc, 0x00, 0xfe, 0x00, 0x04, 0xf9,
0xfb, 0x03, 0x02, 0xfb, 0x00, 0x04, 0x04, 0x03, 0xff, 0x01, 0xfa, 0x01,
0x01, 0xff, 0xfe, 0xfb, 0x05, 0xfd, 0xfc, 0x06, 0xfd, 0xfe, 0xfe, 0xff,
0xfd, 0x06, 0xff, 0x02, 0x05, 0x04, 0xfe, 0x00, 0x05, 0x00, 0xfe, 0xff,
0x02, 0xfc, 0x05, 0xfa, 0xf9, 0xfe, 0xff, 0xfd, 0xfa, 0x05, 0xfc, 0x03,
0x03, 0x02, 0xfc, 0x02, 0xfa, 0xfe, 0x01, 0x00, 0x00, 0xfc, 0xfd, 0xfc,
```

```
0x06, 0xfa, 0x04, 0xfd, 0x05, 0x01, 0xfb, 0x01, 0xfa, 0x00, 0xfe, 0xfc,
0x02, 0xf9, 0x00, 0x00, 0x00, 0xfb, 0x05, 0xfc, 0xff, 0xfa, 0x03, 0x04,
0x01, 0x03, 0xfb, 0x05, 0xfa, 0xff, 0x05, 0xfc, 0x02, 0x02, 0x03, 0xfc,
0x02, 0xfd, 0xfc, 0x05, 0x04, 0x02, 0xfd, 0xfe, 0x04, 0xfc, 0x04, 0x04,
0xfe, 0xfb, 0xfe, 0xfa, 0xfb, 0xff, 0xfa, 0xff, 0xfd, 0xfc, 0xfc, 0x06,
0xfe, 0xfe, 0x02, 0x01, 0x04, 0xfd, 0xfa, 0x00, 0x02, 0x00, 0xf9, 0xfc,
0x05, 0x01, 0xfb, 0x00, 0xfa, 0x00, 0xff, 0xfe, 0xfd, 0x00, 0x00, 0x04,
0xf9, 0xfa, 0x05, 0x00, 0x01, 0x03, 0x02, 0xfe, 0xfe, 0x05, 0x01, 0xfc,
0x03, 0xfc, 0xff, 0xfe, 0x03, 0xf9, 0xfd, 0x00, 0xff, 0x05, 0xfc, 0x06,
0xfa, 0xfa, 0x02, 0xfb, 0xfe, 0xff, 0x00, 0xfc, 0x06, 0x07, 0x01, 0xfb,
0x00, 0x02, 0x04, 0x00, 0x01, 0xfe, 0x00, 0xfd, 0xf9, 0xfd, 0x03, 0x01,
0x01, 0x05, 0xfb, 0xff, 0xfa, 0xff, 0xfd, 0xf9, 0x06, 0xfb, 0xff, 0x04,
0x05, 0x01, 0x00, 0x01, 0xfb, 0x01, 0xfd, 0x04, 0xff, 0x03, 0x05, 0x01,
0x02, 0xfc, 0xfc, 0x01, 0xfa, 0xfb, 0xfe, 0x04, 0xfd, 0xfc, 0x03, 0x03,
0xfe, 0xff, 0xfe, 0xfb, 0x06, 0x05, 0xfc, 0xfe, 0xfa, 0x00, 0x04, 0xfe,
0x06, 0x06, 0x02, 0xff, 0x00, 0xfd, 0xfb, 0xfa, 0xfc, 0x02, 0x01, 0x03,
0x01, 0x06, 0xf9, 0x02, 0xfe, 0xff, 0xfa, 0xfd, 0xfc, 0xff, 0xfd, 0x04,
0xfd, 0x06, 0x05, 0xfd, 0x06, 0x01, 0x04, 0xfd, 0xfc, 0x02, 0xfc, 0xfe,
0xfe, 0xff, 0xfe, 0x00, 0xfe, 0x05, 0xfb, 0xfc, 0x00, 0x05, 0xfa, 0x01,
0xf9, 0x00, 0x02, 0x03, 0x05, 0xff, 0x05, 0x00, 0xff, 0x00, 0x06, 0xfa,
0x04, 0x00, 0x06, 0x03, 0x04, 0x01, 0x05, 0xff, 0xfd, 0x06, 0xfa, 0xfb,
0x02, 0x06, 0xfd, 0xfb, 0xf7, 0x09, 0x1b, 0x9e, 0x04, 0x01, 0xfc, 0x01,
0x03, 0x17, 0x01, 0xfb, 0xfd, 0x07, 0x02, 0x00, 0xfe, 0x07, 0xfc, 0xfa,
0x03, 0x01, 0x04, 0x03, 0x05, 0x00, 0xff, 0x18, 0xfb, 0xfc, 0x05, 0xfc,
0xfe, 0xfd, 0xfe, 0x02, 0xfd, 0x06, 0x03, 0xfc, 0x00, 0x81, 0xff, 0xfa,
0xfb, 0x00, 0xff, 0x01, 0xfc, 0x02, 0x04, 0x08, 0xfb, 0xff, 0x08, 0x01,
0xfd, 0xfd, 0x01, 0xfb, 0x00, 0x05, 0x03, 0xff, 0xfa, 0xfe, 0xf0, 0x06,
0x0d, 0x08, 0xfd, 0xfe, 0x08, 0x05, 0x04, 0x00, 0x08, 0xf1, 0x06, 0x05,
0xfc, 0xfb, 0xfc, 0xff, 0x01, 0x06, 0xfc, 0xfa, 0x03, 0xfc, 0x04, 0xfa,
0xfe, 0x05, 0xff, 0x01, 0xfb, 0xfe, 0xf9, 0xfa, 0xfd, 0x04, 0x03, 0x01,
0x00, 0x00, 0xfd, 0xfa, 0xff, 0x04, 0xfa, 0x08, 0xfc, 0xfd, 0x02, 0x01,
0xff, 0x00, 0x03, 0x06, 0x00, 0x00, 0x06, 0xff, 0x03, 0x03, 0x05, 0xfa,
0xc4, 0x00, 0x02, 0x12, 0xfd, 0xfd, 0x04, 0xe5, 0xfa, 0x02, 0x02, 0x04,
0xfd, 0xf1, 0x06, 0xfb, 0xfd, 0xf9, 0xfb, 0x03, 0x01, 0xfb, 0xfc, 0xfa,
0x03, 0xfd, 0xea, 0x07, 0xfc, 0x05, 0xfd, 0x05, 0x04, 0xfc, 0xfe, 0x00,
0xff, 0x03, 0x03, 0xfd, 0xfc, 0x0e, 0x03, 0x00, 0x04, 0xf9, 0xfc, 0x05,
0x01, 0xfd, 0x03, 0xfd, 0xfe, 0x06, 0xff, 0xfe, 0x02, 0x01, 0xf8, 0x05,
0xff, 0xfa, 0xfe, 0x01, 0x01, 0xff, 0x10, 0xfe, 0x05, 0xfb, 0xf9, 0x03,
0xff, 0x02, 0x03, 0xfb, 0x01, 0x0e, 0xff, 0x00, 0xfd, 0xf9, 0x04, 0xfd,
0x00, 0xfc, 0xff, 0xf9, 0x05, 0xfa, 0x05, 0xfb, 0x03, 0xfe, 0xfe, 0xf9,
0x04, 0xfb, 0xff, 0xff, 0x03, 0xfe, 0x04, 0x00, 0x01, 0xff, 0x02, 0x03,
0xd1, 0xfc, 0xff, 0x03, 0x01, 0x05, 0x06, 0xff, 0x00, 0xfe, 0xfc, 0xff,
0xfa, 0x00, 0xff, 0xf9, 0xfb, 0x00, 0x03, 0xfd, 0xfc, 0x03, 0x18, 0xc6,
0x00, 0xfa, 0x04, 0x00, 0x01, 0x1a, 0x06, 0x00, 0x03, 0x05, 0xfc, 0x05,
0x05, 0xf9, 0x01, 0x07, 0x03, 0x03, 0x03, 0x04, 0xfc, 0x01, 0x00, 0x1a,
0x03, 0xfe, 0x02, 0x06, 0x04, 0xfd, 0x05, 0x0f, 0x00, 0xfb, 0x02, 0xfa,
0xfc, 0xcb, 0xfa, 0x04, 0xfe, 0xfd, 0xff, 0x03, 0xfd, 0x02, 0x04, 0xfc,
0xfd, 0xfb, 0x06, 0xfe, 0x06, 0xff, 0x01, 0x03, 0xfb, 0x02, 0xfa, 0xfb,
0xfb, 0x05, 0xef, 0x01, 0x09, 0x07, 0x07, 0x05, 0x02, 0x04, 0xfc, 0x06,
0x06, 0xee, 0xff, 0x03, 0x05, 0x02, 0xfd, 0x03, 0xfe, 0x05, 0x05, 0xf8,
```

```
0xfd, 0x04, 0x05, 0x01, 0xfa, 0xff, 0x06, 0x04, 0x02, 0xfb, 0xfc, 0x06,
0x01, 0x06, 0x00, 0x00, 0x01, 0xfe, 0x07, 0xfa, 0xf4, 0x05, 0x04, 0x04,
0x06, 0x01, 0xfb, 0x01, 0xff, 0xff, 0xfa, 0x05, 0xfa, 0xfb, 0x06, 0x07,
0xff, 0xfe, 0x02, 0x06, 0x05, 0xfb, 0xfb, 0xfe, 0x04, 0x03, 0xfe, 0x04,
0xff, 0xfd, 0x05, 0x05, 0x04, 0xfe, 0x04, 0xfb, 0xfb, 0xfa, 0xfa, 0xfe,
0xfe, 0xfc, 0xfd, 0x04, 0x06, 0x04, 0x00, 0x03, 0xfc, 0x03, 0xfe, 0xfd,
0x04, 0xff, 0xff, 0xfa, 0xfc, 0x06, 0xfe, 0x00, 0x01, 0x06, 0xff, 0xfc,
0xfe, 0x02, 0x01, 0xfa, 0x03, 0x06, 0xfe, 0xfc, 0x01, 0x00, 0xfc, 0x05,
0x05, 0xff, 0x03, 0x06, 0xfa, 0xfa, 0x02, 0x01, 0xfd, 0xfd, 0x03, 0x05,
0x02, 0xff, 0xfa, 0xfc, 0x02, 0xfc, 0xfb, 0x01, 0xfc, 0x04, 0x03, 0xfd,
0x04, 0x01, 0x04, 0xfc, 0x04, 0x01, 0x06, 0xfd, 0xfa, 0xfa, 0x06, 0xff,
0xff, 0x06, 0x00, 0x05, 0x04, 0xfb, 0x02, 0x02, 0xfa, 0xfc, 0xff, 0xfe,
0x05, 0xfb, 0x01, 0x07, 0xfd, 0x04, 0xff, 0x06, 0x03, 0x01, 0x00, 0xfe,
0x04, 0x03, 0x05, 0x00, 0x00, 0x03, 0xfd, 0x00, 0xff, 0xfb, 0xfa, 0x00,
0xfb, 0x01, 0x1c, 0xc6, 0xff, 0x03, 0x07, 0x01, 0xfb, 0x1a, 0xfd, 0xff,
0x01, 0x04, 0xff, 0xff, 0x01, 0x03, 0xfc, 0xff, 0x00, 0x03, 0x03, 0x02,
0x06, 0xfe, 0xf9, 0x21, 0xff, 0x06, 0x04, 0x03, 0x06, 0x00, 0x01, 0x10,
0x07, 0xfd, 0x05, 0x06, 0xfa, 0xc3, 0x05, 0xfc, 0x02, 0x01, 0xfc, 0x02,
0x00, 0xfb, 0x02, 0x03, 0xfe, 0x02, 0xfe, 0xf9, 0x05, 0xfb, 0x01, 0xfb,
0xfd, 0x01, 0xfc, 0xfe, 0x03, 0x04, 0xed, 0xfd, 0x0e, 0x07, 0x08, 0xfa,
0x01, 0x08, 0x06, 0x06, 0x06, 0xf5, 0xfc, 0xfc, 0x06, 0x02, 0xfb, 0x04,
0x05, 0x01, 0x00, 0x00, 0xfa, 0x06, 0x00, 0xff, 0xfc, 0xfb, 0x01, 0x00,
0xfe, 0xfc, 0xff, 0x02, 0x01, 0xfd, 0x08, 0x01, 0x00, 0xfd, 0x00, 0xfb,
0xfa, 0x05, 0xff, 0x06, 0xfe, 0x06, 0x04, 0x03, 0xfb, 0xfd, 0xfa, 0x06,
0x03, 0x00, 0xfd, 0x07, 0x03, 0xfd, 0xfc, 0x00, 0x01, 0xfe, 0x02, 0xfc,
0xfa, 0xfe, 0x06, 0xfb, 0xfb, 0xfd, 0xfa, 0x03, 0xff, 0x06, 0x03, 0xfe,
0x04, 0x02, 0xfc, 0x02, 0xff, 0x05, 0x03, 0xfc, 0xfe, 0xfd, 0xfa, 0xfd,
0xfc, 0x01, 0x05, 0xff, 0x02, 0x06, 0x00, 0xff, 0xff, 0xff, 0x02, 0xf9,
0xfe, 0x04, 0xfd, 0xfc, 0x00, 0x05, 0x04, 0xfc, 0x02, 0x05, 0x07, 0xfb,
0xfe, 0x01, 0xfc, 0xfb, 0x04, 0x00, 0x05, 0x04, 0xfc, 0x03, 0xff, 0x06,
0xfa, 0x05, 0x01, 0xfa, 0xfc, 0xfc, 0x05, 0x06, 0xfb, 0xff, 0xfb, 0xfc,
0xfa, 0xff, 0xfa, 0xfa, 0x01, 0x06, 0x00, 0x00, 0x03, 0xfb, 0xfd, 0x04,
0x07, 0xff, 0xff, 0xfc, 0x01, 0x01, 0x05, 0x00, 0x03, 0x01, 0x02, 0xfe,
0x02, 0xfd, 0x01, 0x07, 0xf9, 0xfb, 0x03, 0x05, 0x02, 0xff, 0x01, 0xff,
0x04, 0xfb, 0xfd, 0x00, 0x00, 0x05, 0xfd, 0xf9, 0xfe, 0x06, 0xfc, 0xfe,
0xff, 0x02, 0x04, 0x00, 0xf4, 0x03, 0x1d, 0xcf, 0x01, 0x00, 0x01, 0xfd,
0x02, 0x20, 0x02, 0x06, 0x06, 0xfe, 0x06, 0x02, 0xff, 0x05, 0x02, 0xff,
0xfb, 0x01, 0xfd, 0x03, 0x00, 0xfa, 0x03, 0x1a, 0x01, 0x02, 0x00, 0x02,
0xfd, 0x05, 0xff, 0x05, 0xfa, 0x01, 0x05, 0xfa, 0x01, 0xcc, 0xff, 0xfd,
0xfb, 0x04, 0xfa, 0xff, 0x04, 0x06, 0xfd, 0xfb, 0x01, 0x03, 0xfa, 0x02,
0x00, 0xfa, 0xfc, 0xfe, 0x06, 0x04, 0xff, 0x04, 0xfb, 0xfe, 0xca, 0xff,
0x0a, 0xfb, 0x07, 0xfa, 0x07, 0x03, 0x00, 0xfc, 0xfc, 0xcf, 0x04, 0x04,
0x06, 0x00, 0xfd, 0xfd, 0x06, 0x00, 0x01, 0xfb, 0xfa, 0xfb, 0x01, 0x03,
0x00, 0x05, 0x01, 0x02, 0xfd, 0xfd, 0xfb, 0xfe, 0x03, 0xfc, 0x0c, 0x06,
0x06, 0xc8, 0xfa, 0x01, 0xf2, 0xfd, 0x03, 0xfd, 0xfb, 0x00, 0xfd, 0xfa,
0x00, 0x05, 0x06, 0xfa, 0xff, 0x02, 0x04, 0x00, 0x05, 0x02, 0xfe, 0x06,
0x03, 0x04, 0xf9, 0x01, 0x02, 0xfb, 0xf9, 0x05, 0xfd, 0x04, 0xfe, 0x04,
0x03, 0x03, 0xfb, 0x02, 0x05, 0xf8, 0x00, 0xfa, 0xfb, 0xfc, 0xfb, 0x00,
0xfa, 0xff, 0xfb, 0xf9, 0x04, 0xfb, 0x06, 0xfc, 0xfe, 0x01, 0xfd, 0x04,
0x03, 0xfd, 0x01, 0xfa, 0xfd, 0x01, 0x05, 0x03, 0x01, 0xfe, 0xfb, 0x04,
```

```
0x04, 0x00, 0x03, 0xfc, 0x02, 0xfd, 0xfe, 0xff, 0x00, 0x00, 0xfc, 0xfc,
0x02, 0x01, 0xfd, 0x03, 0xfb, 0xfa, 0xfb, 0xfc, 0x03, 0xfd, 0x01, 0xfe,
0x00, 0x05, 0xfb, 0xfc, 0xfc, 0x03, 0xfc, 0xfd, 0x02, 0xfa, 0x02, 0xfe,
0xf9, 0xfb, 0xfb, 0xfd, 0xfc, 0x03, 0xf9, 0x06, 0x05, 0x04, 0xfc, 0xfd,
0xff, 0x02, 0x04, 0x03, 0xfd, 0xfa, 0xfb, 0x05, 0x01, 0xfa, 0x01, 0x06,
0x00, 0xfe, 0xfc, 0xfd, 0xff, 0x05, 0xfd, 0xfb, 0xfc, 0xfb, 0xfc, 0xfb,
0xfa, 0x01, 0x06, 0xfd, 0x05, 0xfb, 0xfe, 0x04, 0xff, 0xff, 0xff, 0x00,
0xfe, 0xfd, 0xfe, 0x02, 0xff, 0xfe, 0x03, 0xfb, 0x01, 0x06, 0xfb, 0x05,
0xfd, 0x03, 0xfa, 0xfc, 0x00, 0xfe, 0x01, 0x05, 0x01, 0x02, 0xf9, 0xfc,
0xfa, 0xfa, 0xfc, 0x07, 0xfe, 0x02, 0xfa, 0xfb, 0x00, 0x02, 0x01, 0x03,
0x01, 0xff, 0xfb, 0xfa, 0x01, 0x03, 0x03, 0xfd, 0x00, 0x04, 0xfd, 0xfc,
0xfd, 0xff, 0x06, 0x05, 0x01, 0xfc, 0xff, 0x04, 0x00, 0x06, 0x01, 0x03,
0x03, 0xfb, 0xf9, 0xfa, 0xf9, 0xfd, 0xfa, 0xff, 0xff, 0xfe, 0xfb, 0xfe,
0xfb, 0xfc, 0x02, 0xfc, 0x01, 0x04, 0x00, 0xfc, 0xff, 0x03, 0xfb, 0xff,
0xfb, 0xfc, 0x06, 0x00, 0xff, 0xfb, 0xfc, 0xfc, 0x04, 0x05, 0x03, 0xfa,
0x03, 0x01, 0xfb, 0xff, 0x04, 0xfa, 0x00, 0xfc, 0xff, 0x06, 0x05, 0xfe,
0x03, 0xff, 0xfe, 0xfa, 0x06, 0x01, 0xfa, 0x05, 0xf9, 0x04, 0xfc, 0xfe,
0xfb, 0x04, 0xfc, 0xfb, 0x02, 0x05, 0xfc, 0xfd, 0x01, 0xfb, 0xfa, 0xff,
0xfb, 0xf8, 0xfa, 0x05, 0x05, 0x00, 0x04, 0x03, 0x06, 0x05, 0x04, 0x00,
0xff, 0xf9, 0x02, 0xf9, 0xfc, 0xfa, 0x01, 0x04, 0xfb, 0xfd, 0xfc, 0x02,
0x04, 0xf9, 0x02, 0xfb, 0xfb, 0x06, 0xff, 0x06, 0x05, 0x01, 0xfb, 0x02,
0x01, 0xfa, 0xff, 0x04, 0x06, 0xff, 0x02, 0x04, 0x04, 0xff, 0xfd, 0x00,
0x04, 0x06, 0x02, 0x02, 0xf9, 0xfd, 0xfb, 0x01, 0x04, 0x07, 0x04, 0xf9,
0xfd, 0xfb, 0xff, 0xfb, 0xff, 0xfc, 0x00, 0xfd, 0x01, 0xfc, 0xfb, 0x05,
0x06, 0x06, 0x00, 0x03, 0xfa, 0x04, 0x05, 0xfa, 0xfd, 0x00, 0xff, 0xfb,
0xff, 0x01, 0x04, 0xfc, 0x01, 0xff, 0xfe, 0x00, 0xfd, 0xfa, 0xfb, 0xfc,
0xfa, 0xfd, 0x02, 0xff, 0x00, 0xfc, 0xff, 0xfa, 0x02, 0x03, 0xff, 0x01,
0xfc, 0x03, 0x02, 0xfa, 0xfa, 0xfa, 0xfc, 0xfd, 0xfa, 0xfb, 0x02, 0xfc,
0xf7, 0x03, 0x26, 0xbd, 0xfb, 0x06, 0xfc, 0xf8, 0xff, 0x1f, 0xfe, 0x04,
0x03, 0xfb, 0xff, 0x05, 0xfc, 0x01, 0x03, 0xfb, 0xfd, 0x00, 0x00, 0xfc,
0x04, 0x02, 0xf9, 0x24, 0xfa, 0xff, 0xff, 0xfc, 0xfc, 0x06, 0xfa, 0x0d,
0x01, 0x04, 0x03, 0xff, 0x05, 0xb2, 0x01, 0xfb, 0x05, 0xfb, 0x06, 0x00,
0xfb, 0x01, 0x05, 0x04, 0xff, 0x05, 0xfe, 0x05, 0x03, 0x01, 0x05, 0xfd,
0xff, 0x05, 0x00, 0xfe, 0x03, 0xfc, 0xc8, 0xfe, 0x15, 0x02, 0x05, 0x06,
0x05, 0x05, 0x02, 0x00, 0xfb, 0xd4, 0xfe, 0x00, 0xfe, 0xf9, 0xfd, 0x06,
0x02, 0xf9, 0x06, 0xfb, 0x06, 0x04, 0x03, 0x01, 0xff, 0x06, 0x02, 0xfd,
0x02, 0xfd, 0xfa, 0xfc, 0xfc, 0xfb, 0x0a, 0x06, 0xfb, 0xd9, 0xfb, 0xfc,
0xf4, 0xfd, 0x00, 0x08, 0xfe, 0x00, 0x06, 0x03, 0xfe, 0xfa, 0x02, 0xfe,
0xfd, 0xfe, 0x06, 0x06, 0x04, 0x01, 0xff, 0x06, 0x03, 0xfa, 0x05, 0xfc,
0xf9, 0x03, 0xfe, 0x04, 0x00, 0xf9, 0xff, 0xfc, 0x00, 0xfd, 0xfc, 0xff,
0x01, 0xfd, 0xfc, 0x04, 0x00, 0xff, 0xfb, 0xfc, 0x00, 0x00, 0xff, 0x00,
0x05, 0x06, 0x04, 0xfe, 0xff, 0xfb, 0x02, 0x04, 0xfd, 0x04, 0xf9, 0xfb,
0x07, 0x00, 0x05, 0x04, 0xff, 0x01, 0x02, 0x04, 0x04, 0xff, 0x05, 0x02,
0xfc, 0xfb, 0xf8, 0x02, 0xfd, 0xfd, 0x01, 0xfe, 0xfc, 0xfe, 0x04, 0xfb,
0x05, 0x05, 0xfd, 0xfb, 0xfc, 0x00, 0x00, 0x01, 0xf9, 0x02, 0x02, 0xff,
0xfc, 0xfb, 0x04, 0x03, 0xfa, 0x03, 0xfe, 0x01, 0x00, 0xfe, 0x00, 0x05,
0x05, 0xff, 0x04, 0xfe, 0xff, 0xfa, 0x03, 0xfe, 0xfe, 0xfd, 0x02, 0xfe,
0xff, 0xf9, 0xfd, 0xfe, 0x03, 0xfc, 0xfb, 0x06, 0xfd, 0xfa, 0x05, 0x01,
0xfd, 0xfb, 0xfc, 0xfa, 0x04, 0xfb, 0x00, 0x05, 0x02, 0xfd, 0xfa, 0x00,
0x03, 0xfc, 0x03, 0x02, 0xf9, 0x06, 0x1a, 0xc1, 0x04, 0xfe, 0x04, 0xf9,
```

```
0xfa, 0x1a, 0x08, 0x04, 0x01, 0xfc, 0x04, 0x03, 0xfe, 0xfb, 0x02, 0x02,
0x01, 0x02, 0x05, 0xfb, 0x04, 0xfa, 0x01, 0x20, 0x06, 0x05, 0xfe, 0x05,
0xfb, 0x08, 0x00, 0x04, 0xfc, 0xfb, 0x02, 0xfe, 0x01, 0xbc, 0x04, 0x00,
0x05, 0xfc, 0x05, 0x04, 0xfe, 0x06, 0x01, 0x05, 0x06, 0x02, 0x00, 0x02,
0x06, 0xfd, 0x07, 0xfe, 0xfd, 0x04, 0x04, 0x03, 0x04, 0xfe, 0xf2, 0x08,
0x0a, 0xfd, 0x06, 0x00, 0x06, 0xfe, 0xfa, 0x03, 0xfb, 0xee, 0xfb, 0x00,
0x03, 0x06, 0x05, 0x06, 0x05, 0x05, 0x04, 0xfb, 0x04, 0xfd, 0xf9, 0x05,
0xfb, 0xfb, 0x07, 0xfc, 0x06, 0x04, 0x01, 0xfc, 0xfc, 0xfb, 0x08, 0xff,
0xfc, 0xfa, 0x06, 0x05, 0x00, 0xfd, 0xfb, 0x06, 0x03, 0xfa, 0xff, 0xfe,
0xff, 0xff, 0x04, 0x04, 0x04, 0x00, 0xfa, 0xff, 0x04, 0x07, 0x04, 0xfb,
0xfa, 0x03, 0x04, 0x02, 0xfb, 0xfc, 0xfe, 0x02, 0xfb, 0x01, 0xfa, 0x04,
0xfa, 0xfa, 0xfe, 0xfd, 0xfc, 0xfa, 0x07, 0x04, 0x01, 0x04, 0x00, 0xfd,
0xfb, 0xfa, 0x01, 0xfd, 0x04, 0xff, 0x03, 0xfc, 0x00, 0xfb, 0xfe, 0x05,
0x06, 0x06, 0x01, 0x01, 0x00, 0x05, 0xfd, 0xfd, 0xfd, 0xff, 0x02, 0x00,
0xfe, 0xfc, 0xfa, 0xfa, 0xf9, 0xff, 0xfc, 0xfd, 0x05, 0x05, 0xfa, 0x05,
0x05, 0xf8, 0x04, 0x02, 0x02, 0x03, 0x01, 0xff, 0xfc, 0x00, 0x02, 0x03,
0xf9, 0x00, 0xff, 0x00, 0xfa, 0x05, 0xff, 0xfd, 0xfe, 0x02, 0x05, 0x02,
0xfc, 0x04, 0x05, 0x02, 0x05, 0xff, 0x06, 0xfa, 0x00, 0x02, 0xfd, 0x02,
0xfe, 0x05, 0xff, 0xff, 0x00, 0x02, 0x01, 0x02, 0x03, 0xfe, 0xfd, 0x02,
0xfc, 0xfb, 0xfc, 0x02, 0xff, 0x06, 0x02, 0xfe, 0x02, 0xfc, 0x02, 0x04,
0x02, 0x01, 0x02, 0x00, 0x04, 0xff, 0x04, 0xfc, 0xfa, 0x0a, 0x30, 0xb2,
0x06, 0x06, 0x05, 0x05, 0xfd, 0x24, 0x08, 0x01, 0x09, 0x02, 0x05, 0x01,
0x07, 0xff, 0x06, 0x03, 0x09, 0x08, 0x02, 0x05, 0x02, 0x04, 0x04, 0x2e,
0xfb, 0xff, 0x01, 0xfa, 0x06, 0xff, 0x01, 0x13, 0xfb, 0x07, 0xff, 0x00,
0xfc, 0xa2, 0x06, 0x05, 0x02, 0x00, 0xfd, 0x03, 0x04, 0x00, 0x03, 0xff,
0x02, 0x02, 0x08, 0x03, 0x04, 0x03, 0xff, 0x02, 0x01, 0x05, 0x04, 0x00,
0x02, 0x03, 0xfc, 0x04, 0x14, 0x00, 0x0a, 0x06, 0x06, 0x0a, 0x05, 0x03,
0x01, 0x05, 0x05, 0x01, 0x01, 0xfb, 0x07, 0xfa, 0x06, 0x01, 0xfb, 0xfc,
0xfd, 0xfb, 0xff, 0x05, 0x02, 0x03, 0x04, 0x07, 0x02, 0xfe, 0xfe, 0xff,
0x04, 0x00, 0xff, 0xfb, 0x02, 0xfc, 0x01, 0x06, 0xf5, 0xfb, 0xfe, 0x05,
0x04, 0xfe, 0x04, 0xfd, 0x06, 0xfe, 0x05, 0x02, 0x06, 0xfb, 0xfd, 0x06,
0x04, 0x03, 0x05, 0x01, 0x04, 0x03, 0x02, 0x02, 0xfc, 0x04, 0x01, 0xfc,
0xfc, 0xff, 0x04, 0xfa, 0x00, 0xfe, 0x05, 0xff, 0x00, 0x03, 0x02, 0xfd,
0x00, 0xfc, 0x01, 0xfd, 0xfc, 0x04, 0x02, 0xfe, 0x05, 0xfb, 0x04, 0xfb,
0x00, 0x05, 0xfd, 0x01, 0xfd, 0xfc, 0xfc, 0x01, 0xfd, 0xfc, 0x04, 0x01,
0xfe, 0xfb, 0x00, 0xff, 0x02, 0xfd, 0xff, 0x02, 0x01, 0x03, 0x02, 0xfb,
0xfd, 0xfe, 0xfa, 0x01, 0x02, 0xfa, 0x03, 0x01, 0x02, 0x02, 0xfb, 0xfc,
0x04, 0xfd, 0x05, 0x01, 0xfd, 0xfd, 0x07, 0xfb, 0xfe, 0x00, 0xff, 0x01,
0xfd, 0xfd, 0x03, 0x03, 0x04, 0x02, 0xfe, 0xff, 0x05, 0x00, 0x05, 0x05,
0xfd, 0x02, 0xfa, 0x05, 0x01, 0xfa, 0xfe, 0x02, 0xfe, 0x02, 0xfc, 0x06,
0x04, 0x01, 0x06, 0xfb, 0xfb, 0x03, 0xfe, 0xfc, 0xfd, 0x00, 0x00, 0x00,
0xfe, 0x01, 0x06, 0x05, 0x06, 0x06, 0x06, 0xfe, 0x00, 0xff, 0xfa, 0x06,
0xfc, 0xfc, 0x04, 0x00, 0xfe, 0x02, 0x05, 0xfc, 0xfd, 0xff, 0x01, 0xfc,
0xfe, 0x04, 0x04, 0x04, 0xfb, 0x01, 0xfc, 0xfa, 0x06, 0xfe, 0xfe, 0xfe,
0xfa, 0x04, 0x05, 0xfa, 0x04, 0x03, 0xfe, 0x01, 0xfc, 0x00, 0xfa, 0xfa,
0xfb, 0x00, 0xfe, 0xfb, 0xfd, 0xfa, 0x05, 0xfc, 0x03, 0x05, 0x02, 0x01,
0xfd, 0xfd, 0x05, 0xfb, 0x01, 0x01, 0xfd, 0x02, 0x00, 0xfb, 0x03, 0x05,
0xfa, 0x03, 0x01, 0xfb, 0x04, 0xfe, 0x05, 0xfc, 0x07, 0x04, 0xfa, 0x01,
0x00, 0x06, 0x01, 0xfb, 0x05, 0x05, 0x01, 0x04, 0xff, 0xfc, 0x04, 0x00,
0x00, 0x06, 0x05, 0xfe, 0x02, 0xfe, 0x04, 0xfd, 0xff, 0x05, 0xfb, 0x02,
```

```
0xfb, 0xff, 0xfb, 0x01, 0x06, 0x05, 0xfd, 0xff, 0xfb, 0x04, 0xfc, 0x02,
0x04, 0x01, 0xff, 0x02, 0xfe, 0xff, 0xfc, 0xfb, 0xfa, 0x02, 0x01, 0xfd,
0x04, 0xfd, 0xfd, 0xfd, 0x04, 0x06, 0xfa, 0xfd, 0x05, 0xff, 0xff, 0xff,
0x03, 0x03, 0x00, 0xfb, 0x04, 0x02, 0x00, 0xf9, 0xfa, 0x06, 0x00, 0xfd,
0x04, 0x01, 0x02, 0xfb, 0xfa, 0x04, 0xfc, 0xfe, 0xfd, 0x06, 0xff, 0x06,
0xfd, 0xfa, 0xfe, 0xfb, 0x02, 0xfc, 0x06, 0xfa, 0xfb, 0x01, 0x01, 0x02,
0xff, 0xfb, 0x06, 0x03, 0xfa, 0xfc, 0x03, 0xfc, 0xff, 0xfc, 0xfe, 0x06,
0xfd, 0xfc, 0xfa, 0xff, 0xfa, 0x04, 0x05, 0xfe, 0xfa, 0xfb, 0xfe, 0x03,
0x05, 0xff, 0x05, 0xfd, 0x05, 0x01, 0xfe, 0x03, 0x02, 0xfe, 0xfb, 0x01,
0xff, 0x05, 0xfb, 0x00, 0x01, 0x01, 0x01, 0x00, 0x04, 0x03, 0xfa, 0x06,
0xfb, 0x03, 0x03, 0xfa, 0xfa, 0x03, 0xfe, 0xfd, 0x02, 0x06, 0x06, 0x05,
0xfc, 0xff, 0xfb, 0x02, 0x01, 0xfd, 0xfd, 0xff, 0x03, 0xfe, 0xfa, 0xfe,
0x05, 0xfb, 0x02, 0x03, 0x06, 0x04, 0xfe, 0xfe, 0xfc, 0xfb, 0x04, 0x03,
0x06, 0x01, 0x05, 0x05, 0xfc, 0xfe, 0x02, 0xfe, 0xfd, 0xfc, 0x05, 0xff,
0x06, 0xfe, 0xff, 0xff, 0x02, 0x00, 0x02, 0xfe, 0x03, 0xff, 0x04, 0xfb,
0xfa, 0x03, 0xfd, 0x05, 0x06, 0xfb, 0x03, 0x06, 0x00, 0x04, 0x03, 0x04,
0x04, 0xff, 0xfc, 0x03, 0x02, 0x01, 0xfa, 0xfb, 0xfb, 0xfa, 0x00, 0xfb,
0xfd, 0xfa, 0x06, 0x06, 0xfc, 0x02, 0xfd, 0xff, 0x05, 0x06, 0x03, 0x04,
0x03, 0xff, 0x04, 0xfe, 0xff, 0xfa, 0xfa, 0x01, 0x03, 0x00, 0xfa, 0x05,
0x02, 0xfd, 0xfb, 0xfd, 0xfe, 0xfb, 0x04, 0xfc, 0x01, 0x03, 0x00, 0x04,
0x03, 0x02, 0x06, 0xfc, 0x01, 0xfe, 0x06, 0x01, 0xfc, 0x02, 0xfa, 0xfc,
0x03, 0x02, 0xf9, 0x05, 0xfc, 0x06, 0xfb, 0xfb, 0x05, 0x05, 0x05, 0xfc,
0x06, 0x04, 0xfa, 0x05, 0xfb, 0xf9, 0x04, 0xfb, 0xff, 0x01, 0x06, 0x02,
0x04, 0x01, 0xfe, 0xfb, 0xfb, 0x04, 0x04, 0x01, 0x04, 0x00, 0xff, 0xfd,
0xff, 0x01, 0x04, 0x02, 0xfa, 0x00, 0x05, 0xfe, 0x00, 0x03, 0x00, 0x00,
0xf9, 0xfa, 0xfe, 0xfc, 0xfe, 0xf8, 0xff, 0x00, 0xff, 0x02, 0xfa, 0x01,
0x04, 0xfa, 0x01, 0xf9, 0x02, 0x03, 0xfa, 0x04, 0x06, 0x02, 0xfc, 0xfd,
0x03, 0x02, 0x00, 0x06, 0x06, 0xfe, 0x05, 0xff, 0x05, 0xfd, 0xfc, 0xfa,
0x02, 0xfa, 0x03, 0xfd, 0x01, 0xfe, 0xfd, 0x04, 0x03, 0x05, 0xfa, 0x04,
0xff, 0x03, 0x02, 0x06, 0xfe, 0x00, 0xfc, 0x04, 0xfe, 0xfb, 0x03, 0xfb,
0x04, 0xf9, 0xfa, 0x06, 0x00, 0xfb, 0x05, 0xf8, 0xfc, 0xfc, 0x04, 0x00,
0xfa, 0xff, 0xfc, 0x04, 0xfb, 0x03, 0x00, 0xfa, 0x03, 0x02, 0xfc, 0xfb,
0xfd, 0x02, 0x01, 0xfd, 0x01, 0xf9, 0xfc, 0x00, 0x00, 0xfd, 0xfb, 0x05,
0x03, 0x06, 0xfe, 0x03, 0xfd, 0x00, 0xfa, 0x02, 0x01, 0x03, 0xf8, 0xfe,
0x00, 0x04, 0x06, 0xf8, 0xfe, 0x02, 0xfb, 0x06, 0x02, 0x04, 0x00, 0xff,
0xfb, 0x03, 0xfe, 0x00, 0xfd, 0xfe, 0x06, 0xfb, 0xfd, 0xfc, 0xfc, 0x03,
0x04, 0x03, 0xff, 0xfa, 0x04, 0x05, 0x00, 0x04, 0xfa, 0x04, 0xfb, 0x05,
0x01, 0xfb, 0xfd, 0x02, 0x02, 0xfa, 0x04, 0x00, 0xfe, 0x06, 0x02, 0x02,
0x05, 0xfa, 0xfb, 0x01, 0xfd, 0xfb, 0x04, 0x02, 0x01, 0x04, 0x00, 0xfb,
0xfd, 0xfa, 0xfa, 0x01, 0x06, 0x02, 0xfa, 0x04, 0x00, 0xfd, 0xfb, 0xfe,
0x06, 0x01, 0x01, 0xfc, 0xfa, 0x01, 0x01, 0x04, 0xfa, 0xfd, 0xfa, 0xfb,
0x05, 0xfb, 0xfa, 0x01, 0x00, 0xfd, 0xfb, 0xfe, 0xfe, 0x05, 0x05, 0xfe,
0x06, 0x00, 0xfd, 0xff, 0x05, 0xfe, 0x02, 0xfd, 0x04, 0xfd, 0xfb, 0x05,
0x01, 0x02, 0xf9, 0xfa, 0x05, 0xff, 0xfc, 0x04, 0x02, 0x05, 0xfc, 0xfb,
0xff, 0xff, 0x03, 0x03, 0x05, 0xfd, 0xff, 0x00, 0xfc, 0x06, 0xff, 0xfb,
0x06, 0xfc, 0xff, 0x02, 0x00, 0xfc, 0xfe, 0xfa, 0xfb, 0x02, 0xfa, 0xfa,
0x04, 0xff, 0x01, 0x04, 0xfc, 0x03, 0x00, 0xfc, 0xfa, 0xfe, 0xfc, 0x04,
0x02, 0x04, 0xfe, 0x00, 0x03, 0x00, 0x06, 0xfa, 0xfc, 0xfe, 0x00, 0xfc,
0x04, 0x03, 0x06, 0x01, 0x04, 0xfa, 0xfa, 0xfa, 0xff, 0x02, 0x06, 0x00,
0x00, 0xfa, 0x02, 0x00, 0xfa, 0xfa, 0x03, 0x02, 0xfe, 0xfe, 0xff, 0xfa,
```

```
0xfc, 0xfd, 0xfd, 0xfe, 0xfb, 0xfc, 0x05, 0x06, 0x01, 0x00, 0xfa, 0xfa,
0x03, 0x00, 0x00, 0xff, 0x06, 0x01, 0x00, 0xfe, 0x03, 0x05, 0x00, 0x01,
0xfa, 0xfd, 0xff, 0x01, 0xff, 0xfb, 0xfb, 0x06, 0xff, 0x01, 0x01, 0xfa,
0xfb, 0xfc, 0x02, 0xfd, 0xfe, 0xfe, 0xfa, 0x04, 0x01, 0xfe, 0xf9, 0x01,
0x03, 0x04, 0xfc, 0x04, 0x05, 0x06, 0xfb, 0xfb, 0xfb, 0x04, 0xff, 0x00,
0xfd, 0x00, 0xfc, 0x05, 0x06, 0x01, 0x02, 0x01, 0xfb, 0x02, 0x00, 0xfe,
0x0c, 0xfc, 0xde, 0x39, 0xfc, 0x00, 0x04, 0x0e, 0x04, 0xe0, 0x02, 0x03,
0xfb, 0x01, 0x03, 0xfd, 0x07, 0x07, 0x05, 0xfc, 0x05, 0x07, 0x04, 0x05,
0x01, 0xff, 0x06, 0xe2, 0x02, 0xff, 0xff, 0xff, 0x02, 0x01, 0x02, 0xf2,
0x04, 0x01, 0x00, 0xff, 0xf9, 0x3b, 0x02, 0xfe, 0x02, 0xfc, 0x03, 0x01,
0xfe, 0x06, 0x02, 0x06, 0x02, 0xfc, 0x04, 0xfd, 0x06, 0xff, 0x00, 0x01,
0x06, 0xfb, 0x06, 0xfa, 0xfa, 0xfc, 0x09, 0x06, 0xec, 0x00, 0x05, 0xfe,
0x04, 0xfe, 0x02, 0x05, 0x07, 0x11, 0xfc, 0xfb, 0xfa, 0x06, 0x00, 0xfc,
0x00, 0x03, 0x06, 0xfc, 0xfe, 0xfd, 0x05, 0xf9, 0x03, 0xf9, 0xfc, 0x00,
0xfd, 0x02, 0x04, 0x01, 0x02, 0xfb, 0xfb, 0x01, 0xfc, 0xf7, 0xfc, 0x01,
0x0e, 0x00, 0x05, 0xfe, 0xfa, 0xfc, 0xfb, 0xfa, 0x02, 0x02, 0xfd, 0x04,
0xff, 0x09, 0x04, 0x09, 0x03, 0x02, 0x06, 0xfb, 0xfb, 0x02, 0xfc, 0x03,
0x05, 0x03, 0x00, 0x05, 0x02, 0x03, 0xfd, 0xfe, 0xfe, 0xfb, 0xfd, 0xfb,
0x01, 0xfb, 0xff, 0xff, 0xfa, 0x04, 0x00, 0x01, 0xfd, 0x05, 0xfa, 0x00,
0xfa, 0x00, 0x01, 0x04, 0x00, 0xfa, 0xfd, 0x01, 0x03, 0x01, 0xfc, 0x03,
0x05, 0x00, 0x00, 0x00, 0xff, 0x02, 0xfb, 0xfe, 0x05, 0x00, 0xff, 0xfa,
0x06, 0xfb, 0x01, 0x03, 0xff, 0xff, 0x03, 0x05, 0x01, 0x06, 0xfa, 0x06,
0xfb, 0xfc, 0xff, 0xfe, 0xfc, 0x02, 0xfe, 0x01, 0xff, 0x06, 0xfe, 0xfa,
0x03, 0xff, 0x00, 0xfb, 0x00, 0xfb, 0x06, 0xff, 0xfc, 0x01, 0xfc, 0x02,
0x03, 0xff, 0x00, 0xff, 0xfe, 0xfc, 0xfb, 0x04, 0x02, 0xff, 0x04, 0xfa,
0x03, 0x01, 0xfd, 0xfb, 0x06, 0xff, 0xfa, 0x04, 0xfc, 0x04, 0x06, 0x01,
0x05, 0x01, 0x02, 0x00, 0x03, 0x02, 0xfa, 0xfc, 0x03, 0xfe, 0x02, 0x02,
0xfc, 0x05, 0x03, 0xfd, 0xf8, 0x00, 0x05, 0xf9, 0xfd, 0xff, 0x00, 0xf9,
0x05, 0x05, 0xfd, 0x00, 0xfe, 0xf9, 0x04, 0x02, 0x04, 0xff, 0xfa, 0xfa,
0xf8, 0x03, 0xff, 0x05, 0x05, 0x04, 0xfa, 0xf9, 0x00, 0xfb, 0x04, 0x05,
0x02, 0x04, 0x05, 0x04, 0xfa, 0xfe, 0x01, 0xfb, 0xfe, 0xfd, 0xff, 0x01,
0x00, 0x05, 0xfc, 0x04, 0xff, 0x01, 0x02, 0xfa, 0xfa, 0xfe, 0xfa, 0xfa,
0x06, 0xf9, 0xfc, 0xfc, 0xfd, 0xfc, 0x01, 0x00, 0x06, 0x06, 0xfa, 0xfa,
0xff, 0x03, 0xfc, 0x03, 0x04, 0xf9, 0x02, 0xfa, 0xf8, 0x00, 0x06, 0xfc,
0x02, 0xfc, 0x05, 0x00, 0x03, 0x06, 0xff, 0x02, 0xfd, 0xfe, 0xfa, 0xff,
0xfc, 0x00, 0x00, 0x01, 0xfb, 0x03, 0xfa, 0xfc, 0x00, 0xfb, 0x03, 0x04,
0xfa, 0xfe, 0xff, 0x04, 0xfd, 0x04, 0x04, 0x04, 0xfe, 0x05, 0x00, 0xff,
0xfc, 0xfd, 0xfe, 0xfe, 0x03, 0xff, 0xfe, 0xfd, 0xfb, 0xf9, 0xfc, 0xf9,
0xfd, 0xfc, 0x02, 0x04, 0x03, 0xf9, 0xfd, 0xfd, 0xff, 0xfa, 0x05, 0x03,
0x01, 0x02, 0xfe, 0x02, 0xfc, 0x04, 0xfc, 0x05, 0xfb, 0xf9, 0xfa, 0x05,
0xfc, 0xfb, 0xfe, 0x01, 0xff, 0xfe, 0x01, 0x02, 0xfc, 0xfd, 0x05, 0x03,
0x06, 0x00, 0xfe, 0x01, 0xfe, 0xfd, 0xff, 0x02, 0xfa, 0x06, 0x06, 0x03,
0xfa, 0x05, 0x06, 0x02, 0xff, 0x00, 0xff, 0x00, 0x00, 0xfd, 0xfc, 0x06,
0xfa, 0xfc, 0x06, 0x02, 0x03, 0x02, 0xfd, 0x00, 0x05, 0xfa, 0x02, 0xfd,
0xf9, 0xfc, 0xfa, 0x02, 0xfe, 0x01, 0x00, 0xfe, 0x06, 0x05, 0xfe, 0x01,
0xff, 0xfe, 0xfc, 0xfc, 0xff, 0xfd, 0x03, 0x03, 0xfe, 0xfa, 0xff, 0x04,
0xfc, 0x05, 0x03, 0xfc, 0xfa, 0xfa, 0xf9, 0x01, 0x05, 0xff, 0xfe, 0xfe,
0xff, 0x02, 0x03, 0xfc, 0xfc, 0xfc, 0xfa, 0xfa, 0xfb, 0xfe, 0x05, 0x05,
0xfb, 0xf8, 0x03, 0xfc, 0xff, 0xfe, 0x04, 0xfd, 0xf6, 0x03, 0x28, 0xb7,
0x01, 0x06, 0x00, 0xf8, 0xfe, 0x26, 0x07, 0xfc, 0x01, 0x04, 0x00, 0x09,
```

```
0x03, 0x04, 0x07, 0x06, 0x01, 0x07, 0xfe, 0x07, 0x03, 0xfa, 0x00, 0x1e,
0xfc, 0x00, 0xfd, 0xfd, 0x06, 0x07, 0x02, 0x09, 0x05, 0x01, 0xfc, 0xfa,
0xfc, 0xa9, 0xfe, 0x01, 0x01, 0x02, 0x02, 0x02, 0x00, 0xfe, 0x03, 0x08,
0x04, 0x00, 0x04, 0x05, 0xff, 0xfb, 0x07, 0x01, 0x03, 0x03, 0xfd, 0x06,
0x04, 0x06, 0xe6, 0x03, 0x07, 0xfe, 0x06, 0x00, 0x08, 0x02, 0x03, 0x01,
0x03, 0xe8, 0x01, 0x05, 0xfe, 0x03, 0xff, 0x04, 0xfc, 0x01, 0xfa, 0xfe,
0x00, 0x05, 0xfd, 0xfc, 0xfd, 0x05, 0x07, 0xff, 0x01, 0x01, 0xfb, 0xff,
0xfd, 0x08, 0xfc, 0x00, 0x06, 0xfe, 0x01, 0x03, 0xfb, 0x01, 0x06, 0x05,
0x05, 0x03, 0x02, 0x03, 0x03, 0xfd, 0x05, 0xfd, 0x06, 0xfb, 0x04, 0x03,
0xfe, 0x02, 0xff, 0xfd, 0xfb, 0xfc, 0xfe, 0x02, 0x00, 0xfa, 0x05, 0x01,
0x05, 0xfe, 0x03, 0xff, 0x00, 0x05, 0x03, 0x06, 0x04, 0xfa, 0x04, 0x03,
0x03, 0x04, 0xfa, 0xfd, 0x03, 0xfd, 0xfc, 0xff, 0x06, 0xfe, 0x05, 0xfb,
0x01, 0xfc, 0x01, 0xfd, 0xfe, 0x03, 0xfd, 0x06, 0x03, 0x06, 0x02, 0xfc,
0xfe, 0xfa, 0xfe, 0xfc, 0x02, 0xfd, 0xfd, 0xfa, 0x03, 0x01, 0xfd, 0x03,
0x05, 0x00, 0xfd, 0x02, 0xfc, 0xfc, 0x03, 0xfd, 0xfc, 0x06, 0x05, 0xf9,
0xff, 0x04, 0x01, 0x02, 0x03, 0x05, 0x02, 0x00, 0x02, 0x08, 0xfe, 0xff,
0xff, 0xfb, 0x04, 0x02, 0xfc, 0xfb, 0xfe, 0x04, 0x05, 0x01, 0xff, 0xfc,
0xfe, 0x03, 0x02, 0xfa, 0xfa, 0x02, 0xfd, 0xfb, 0x01, 0x03, 0xfe, 0x01,
0xfe, 0xfe, 0x04, 0x04, 0xf9, 0x01, 0xfd, 0xf9, 0xfb, 0x00, 0x02, 0xfc,
0xfe, 0x03, 0xfc, 0x01, 0x02, 0xfb, 0xfa, 0x00, 0xff, 0xfc, 0xff, 0x00,
0xfd, 0x08, 0x2b, 0xb8, 0x03, 0x09, 0x08, 0x02, 0xfd, 0x24, 0x07, 0xfb,
0x08, 0x01, 0x01, 0x04, 0xfd, 0xfc, 0xfe, 0x02, 0x00, 0x00, 0x04, 0x09,
0x09, 0x00, 0xfe, 0x24, 0x05, 0xfd, 0xfa, 0xfd, 0x01, 0x09, 0xf9, 0x11,
0xfb, 0xfe, 0xff, 0x02, 0x03, 0xa0, 0xfa, 0xff, 0x04, 0x01, 0x04, 0xff,
0x05, 0x05, 0xfd, 0x08, 0xfc, 0x00, 0x02, 0x06, 0x00, 0xfb, 0x07, 0xfe,
0xfb, 0xfd, 0xfd, 0xfc, 0x04, 0xff, 0xe7, 0x07, 0x12, 0x09, 0x05, 0x03,
0x01, 0x09, 0x05, 0x03, 0x02, 0xfb, 0xfb, 0x03, 0x05, 0xff, 0x00, 0xfc,
0x08, 0xff, 0x01, 0x04, 0x04, 0x00, 0x03, 0x00, 0x00, 0x05, 0x03, 0x00,
0x02, 0xfe, 0xfa, 0xfc, 0xfe, 0x05, 0x04, 0xff, 0xfe, 0x01, 0x07, 0xfb,
0x03, 0xfa, 0x06, 0x09, 0xfb, 0xfe, 0x05, 0xff, 0xfe, 0xfd, 0x04, 0x05,
0x00, 0x05, 0xfd, 0x02, 0x05, 0x03, 0x05, 0x04, 0xff, 0x04, 0x04, 0xfa,
0xf9, 0x05, 0xfe, 0x06, 0xfd, 0x04, 0xfc, 0xfd, 0x00, 0x05, 0x05, 0x06,
0xff, 0xff, 0xfa, 0x06, 0xf9, 0x04, 0xfc, 0x04, 0xfa, 0xfb, 0xff, 0xfb,
0x03, 0x04, 0x04, 0xfb, 0xfb, 0x05, 0x03, 0x05, 0xfa, 0x05, 0x04, 0xfc,
0x02, 0x02, 0xfe, 0xfe, 0xff, 0xfc, 0xfb, 0x05, 0xff, 0xff, 0xff, 0x03,
0xfe, 0x04, 0x00, 0x01, 0xfb, 0x02, 0xfc, 0x01, 0x02, 0x03, 0xfb, 0xfd,
0x01, 0x05, 0x00, 0xf9, 0xfa, 0x03, 0xfd, 0x06, 0xfa, 0x00, 0xfb, 0x01,
0x00, 0xfe, 0xfb, 0xfc, 0x02, 0xfa, 0x03, 0xfe, 0xfa, 0x03, 0x06, 0x04,
0x01, 0x05, 0xff, 0xff, 0xfa, 0xfe, 0xfb, 0x03, 0x06, 0x01, 0xff, 0x00,
0x02, 0x04, 0x03, 0xfa, 0xfb, 0xfd, 0x00, 0xfb, 0xfd, 0x03, 0x03, 0xfa,
0xfa, 0xfa, 0x05, 0xfc, 0xfd, 0xfc, 0x03, 0xfe, 0x00, 0xfe, 0xfa, 0x01,
0xff, 0xff, 0xfd, 0x06, 0x02, 0x00, 0xf9, 0x03, 0x04, 0x02, 0x01, 0x03,
0xfd, 0x03, 0x04, 0xfa, 0xfa, 0xfe, 0x06, 0xfa, 0xf8, 0xfd, 0x02, 0xfa,
0xfb, 0xfb, 0xfd, 0x05, 0xfc, 0x01, 0x05, 0xf9, 0x05, 0x01, 0x00, 0x06,
0xfd, 0x01, 0x04, 0xfb, 0x05, 0xfa, 0x00, 0xfb, 0x00, 0xfd, 0x02, 0xfe,
0xfa, 0x04, 0xfc, 0x06, 0x04, 0xfd, 0x02, 0xff, 0xfc, 0x01, 0x06, 0x06,
0x02, 0x07, 0xfd, 0xfa, 0x04, 0xfb, 0xfc, 0xfc, 0xfc, 0x02, 0x03, 0xfa,
0x05, 0xfa, 0xff, 0xff, 0x01, 0xff, 0x00, 0x06, 0xfc, 0x01, 0x02, 0xf9,
0x02, 0x06, 0x00, 0x03, 0x02, 0x01, 0x04, 0x04, 0x04, 0xff, 0x06, 0x00,
0x04, 0x00, 0xfa, 0xff, 0x06, 0xff, 0x02, 0x03, 0xfc, 0x02, 0x02, 0xfb,
```

```
0x01, 0xfd, 0x04, 0xfe, 0xfc, 0xff, 0xfd, 0x02, 0x03, 0x07, 0x03, 0xfa,
0xff, 0x02, 0xfe, 0xff, 0x04, 0x02, 0xfb, 0xfa, 0x03, 0xff, 0x01, 0xfe,
0xf6, 0x07, 0x2b, 0xb2, 0x03, 0x06, 0x09, 0xfb, 0xfc, 0x2a, 0xff, 0x02,
0x0a, 0x00, 0x00, 0x08, 0xfb, 0x05, 0x03, 0x04, 0xfc, 0x09, 0x02, 0x07,
0x02, 0x00, 0x02, 0x2c, 0x06, 0x00, 0x00, 0x04, 0xfd, 0x03, 0x02, 0x0b,
0x03, 0xfc, 0x07, 0x03, 0x02, 0xa2, 0x05, 0x00, 0x03, 0x02, 0xfb, 0xfb,
0x00, 0xff, 0x06, 0x06, 0xff, 0xfc, 0x03, 0x03, 0xfb, 0x02, 0x04, 0xfb,
0x03, 0x02, 0xfb, 0x00, 0xfa, 0xfb, 0xf0, 0x08, 0x13, 0xfd, 0x01, 0xfc,
0x06, 0x07, 0x01, 0xfb, 0xfd, 0x00, 0x00, 0x08, 0x03, 0x02, 0x02, 0xfe,
0x0a, 0xfc, 0xfc, 0x04, 0xfa, 0x02, 0x01, 0x02, 0x06, 0x02, 0x01, 0xfb,
0x01, 0xff, 0xfc, 0x05, 0xfb, 0x04, 0x00, 0x03, 0x01, 0x01, 0x07, 0x03,
0xf7, 0x00, 0xfc, 0x00, 0xfe, 0x04, 0xfa, 0x04, 0xfc, 0xfd, 0xfc, 0xfc,
0xff, 0x05, 0x03, 0xfe, 0x05, 0x01, 0x01, 0x05, 0xf8, 0xfc, 0x21, 0xc8,
0xfd, 0x03, 0x03, 0xf7, 0x07, 0x1a, 0x03, 0x02, 0x05, 0xff, 0x01, 0x09,
0x01, 0x06, 0xff, 0xfe, 0xfb, 0x05, 0x08, 0x03, 0x00, 0xfb, 0x02, 0x22,
0x06, 0x07, 0xfc, 0x02, 0x03, 0x06, 0x05, 0x0a, 0x05, 0x05, 0x01, 0xfa,
0x05, 0xc9, 0xff, 0x01, 0xfd, 0x04, 0x01, 0xff, 0xfc, 0x03, 0xfb, 0x06,
0xfa, 0x04, 0x03, 0xfc, 0x00, 0x06, 0xfd, 0xfc, 0x00, 0x08, 0x03, 0x06,
0xfc, 0xfc, 0xef, 0x00, 0x0b, 0x04, 0x02, 0xfa, 0x01, 0x06, 0x04, 0xfb,
0x07, 0xf9, 0x02, 0x06, 0x05, 0x04, 0x06, 0x01, 0xfe, 0x02, 0x04, 0xfc,
0x00, 0x06, 0x06, 0xfc, 0xfa, 0x04, 0x08, 0x08, 0x00, 0xff, 0xfa, 0xff,
0x07, 0xfc, 0xfd, 0x01, 0x05, 0xff, 0x05, 0xfb, 0xf7, 0xff, 0x02, 0x00,
0xfa, 0xfe, 0x04, 0xfb, 0x04, 0x03, 0xfe, 0xfd, 0x00, 0xff, 0x05, 0xff,
0x07, 0x02, 0xfe, 0x01, 0xfb, 0xfd, 0x25, 0xca, 0x03, 0x02, 0x01, 0xf6,
0xff, 0x1e, 0xfe, 0xfa, 0xfe, 0xfb, 0x05, 0xfd, 0x03, 0xfe, 0x02, 0xfa,
0x00, 0x02, 0x03, 0x00, 0x04, 0x05, 0xfd, 0x1d, 0xfd, 0x04, 0xfc, 0x03,
0xfb, 0xff, 0x02, 0x08, 0xff, 0x01, 0xfd, 0x00, 0x04, 0xc5, 0x03, 0x04,
0xfa, 0x01, 0x00, 0xfa, 0xff, 0x02, 0x01, 0x05, 0xfe, 0xfc, 0xfa, 0x06,
0x03, 0x04, 0x05, 0xff, 0xfb, 0x07, 0xfb, 0x04, 0xfc, 0x04, 0xc8, 0x03,
0x0d, 0xfb, 0x06, 0x07, 0x01, 0xfc, 0x02, 0xfb, 0x02, 0xc5, 0x01, 0x06,
0xfc, 0xfc, 0xfc, 0x05, 0xff, 0xfb, 0x00, 0x05, 0x04, 0x02, 0xfb, 0x04,
0xf9, 0x02, 0x05, 0xff, 0xfe, 0x03, 0x05, 0xff, 0xfe, 0x02, 0x09, 0x01,
0xfe, 0xf6, 0xfe, 0x03, 0xf3, 0x06, 0x06, 0x07, 0x05, 0x00, 0xfd, 0x01,
0xf9, 0x04, 0x01, 0xfe, 0xff, 0x00, 0x02, 0xff, 0xfe, 0x01, 0x06, 0xff,
0x01, 0x00, 0x03, 0x04, 0x03, 0xff, 0xfb, 0x02, 0xfd, 0x01, 0xfe, 0xf9,
0x00, 0xff, 0xfd, 0xfa, 0xfb, 0x00, 0x03, 0xfd, 0x00, 0xfe, 0xfb, 0xff,
0xff, 0xfc, 0x01, 0xfb, 0xfc, 0xfa, 0xfc, 0x00, 0xfb, 0x05, 0x02, 0x04,
0xfd, 0xfa, 0x04, 0x01, 0xfa, 0x00, 0x06, 0xfa, 0x01, 0xff, 0x02, 0x01,
0x05, 0x00, 0x06, 0xfd, 0xfd, 0x02, 0xfc, 0x02, 0xfe, 0x01, 0xfc, 0x06,
0xff, 0xfd, 0xfa, 0x06, 0x02, 0x06, 0xfd, 0xfc, 0x03, 0xfe, 0xfa, 0x01,
0x01, 0xf9, 0x02, 0xfd, 0x03, 0x00, 0xfd, 0xfa, 0xff, 0x05, 0xfc, 0x05,
0x04, 0xfc, 0xfc, 0x03, 0xfe, 0xff, 0x04, 0x04, 0xfa, 0xff, 0xff, 0x00,
0xfa, 0xff, 0x04, 0x06, 0x04, 0xfb, 0x02, 0x01, 0xfa, 0x03, 0xfa, 0xfb,
0xff, 0xfa, 0xfb, 0xfd, 0xfd, 0x07, 0xfc, 0xfe, 0x02, 0x00, 0x01, 0x00,
0xff, 0x04, 0x05, 0x01, 0xfc, 0xfb, 0xfc, 0xfb, 0x04, 0xfd, 0xfd, 0x03,
0xfc, 0xff, 0xfa, 0x04, 0xfc, 0xff, 0x01, 0x06, 0xfe, 0xfc, 0xfa, 0xf9,
0x01, 0xfd, 0x00, 0xfd, 0xfa, 0xfa, 0xff, 0xfa, 0xf9, 0x06, 0xfb, 0xf9,
0xfb, 0x03, 0x05, 0x02, 0x01, 0xfd, 0x06, 0xfe, 0x00, 0x06, 0x02, 0xfa,
0x01, 0x02, 0xfd, 0xfc, 0xfc, 0x03, 0x00, 0xfe, 0x02, 0xfc, 0x03, 0x00,
0x04, 0x06, 0xfa, 0xff, 0x00, 0x03, 0x03, 0xfd, 0x03, 0x05, 0xfe, 0xfd,
```

```
0x00, 0x06, 0xfb, 0x03, 0x02, 0x05, 0xff, 0xfb, 0x05, 0xfe, 0xfe, 0xfa,
0x02, 0xff, 0xfe, 0xfe, 0x02, 0xfa, 0xfb, 0x06, 0x04, 0x01, 0xfa, 0xfc,
0x06, 0x06, 0xfd, 0xfe, 0x00, 0x01, 0x02, 0x05, 0xff, 0x01, 0xff, 0x05,
0xfa, 0xfb, 0x02, 0xfc, 0x03, 0xff, 0xf9, 0x05, 0xfc, 0x01, 0x06, 0xff,
0x00, 0xff, 0xff, 0x03, 0x03, 0x04, 0xf9, 0xfb, 0xff, 0x01, 0xfa, 0x04,
0xfe, 0x03, 0x05, 0xfd, 0xfc, 0xfd, 0xfb, 0x02, 0x01, 0xfe, 0x00, 0x04,
0x06, 0x06, 0xff, 0xfc, 0x00, 0x00, 0x07, 0x06, 0x00, 0x04, 0xfa, 0xfe,
0x06, 0xfe, 0x04, 0xfa, 0xfc, 0x03, 0xfb, 0x04, 0x00, 0x06, 0x02, 0xfe,
0xfe, 0xfe, 0x04, 0xfe, 0xfc, 0xfe, 0xfa, 0x07, 0xfc, 0xfd, 0xfb, 0x01,
0xfc, 0x02, 0x03, 0xff, 0xfe, 0x00, 0xfb, 0xfd, 0xfc, 0xfb, 0xfc, 0x02,
0xfc, 0x06, 0xfe, 0x02, 0xfc, 0xff, 0xfa, 0x06, 0xf9, 0x00, 0x01, 0x06,
0xfe, 0xff, 0xfa, 0x04, 0xfc, 0x04, 0x03, 0xfe, 0x01, 0x01, 0x03, 0xfc,
0x01, 0xfb, 0xfb, 0xfb, 0xfb, 0xff, 0x02, 0x04, 0x01, 0x05, 0x05, 0xfa,
0x03, 0xff, 0xfc, 0xfa, 0x02, 0x05, 0xff, 0x04, 0x01, 0xfe, 0xfe, 0x03,
0x02, 0x05, 0xff, 0x04, 0xfd, 0xfa, 0x05, 0x00, 0x02, 0xfb, 0x00, 0x02,
0xfb, 0x06, 0x00, 0xfd, 0xfa, 0xfa, 0xfe, 0x03, 0xfc, 0xff, 0xfc, 0x04,
0xfc, 0x08, 0x25, 0xbb, 0x03, 0x03, 0x04, 0x02, 0x03, 0x21, 0xfe, 0xfb,
0x03, 0x05, 0x04, 0x03, 0xff, 0x06, 0xff, 0xfb, 0x01, 0x06, 0xfb, 0x01,
0xfd, 0xfb, 0xfa, 0x29, 0x00, 0x05, 0xfa, 0xfe, 0x06, 0x00, 0xff, 0x11,
0xfc, 0x00, 0x07, 0x02, 0x02, 0xbb, 0xfb, 0x00, 0x03, 0x06, 0x04, 0xfd,
0x02, 0x07, 0xfc, 0x08, 0x03, 0x02, 0x00, 0x04, 0x01, 0xfd, 0x03, 0x03,
0x03, 0xfd, 0xfb, 0xfd, 0xfe, 0x03, 0xec, 0x01, 0x13, 0x02, 0x09, 0xfc,
0x05, 0x06, 0xfd, 0xff, 0x08, 0xf7, 0x06, 0x06, 0x02, 0xff, 0x03, 0xfb,
0xfe, 0xfd, 0xfa, 0x05, 0x06, 0xfa, 0xfb, 0x01, 0xfc, 0x02, 0x07, 0x01,
0xfe, 0x06, 0x00, 0xff, 0x01, 0x00, 0xfd, 0x07, 0xfe, 0x03, 0x02, 0x05,
0xf8, 0x05, 0x00, 0x07, 0x02, 0x05, 0x03, 0x01, 0x04, 0xfc, 0x08, 0x00,
0x01, 0x02, 0x03, 0x01, 0x04, 0x05, 0x00, 0xff, 0xfe, 0x06, 0x2a, 0xb0,
0x07, 0x00, 0xff, 0x01, 0xff, 0x20, 0x03, 0x06, 0x03, 0xfe, 0xf9, 0xfe,
0x04, 0x01, 0xfb, 0x02, 0x08, 0x02, 0xfc, 0x08, 0x08, 0xfa, 0x05, 0x2b,
0xfe, 0x02, 0x03, 0x04, 0x09, 0x07, 0xfb, 0x0f, 0xfc, 0x04, 0x00, 0xfa,
0xf9, 0x96, 0xfb, 0xfe, 0xfc, 0x03, 0xfa, 0x04, 0xf9, 0x02, 0x05, 0x02,
0x00, 0xff, 0x05, 0x06, 0xfb, 0xff, 0x03, 0x02, 0x01, 0x02, 0xff, 0xfd,
0x04, 0xff, 0xf4, 0x04, 0x12, 0x08, 0x0a, 0xf9, 0xfd, 0x08, 0xfd, 0x05,
0x08, 0xf7, 0x03, 0x09, 0x04, 0x03, 0x01, 0xfa, 0x09, 0x00, 0x02, 0x04,
0xfd, 0x05, 0x04, 0xff, 0x06, 0x01, 0x07, 0x07, 0xfc, 0xfb, 0x01, 0x04,
0xfc, 0x06, 0xff, 0x02, 0x01, 0x00, 0x07, 0x06, 0xf9, 0xfb, 0x04, 0x08,
0x02, 0xfe, 0x07, 0xfe, 0x01, 0x05, 0xfc, 0xfb, 0xfe, 0x07, 0xfe, 0x01,
0x07, 0x09, 0x01, 0x00, 0xfe, 0x02, 0xff, 0x06, 0x06, 0x05, 0x06, 0xf9,
0xff, 0xfa, 0x01, 0xfe, 0xfb, 0x01, 0xfa, 0xfd, 0xfc, 0xfd, 0x03, 0x06,
0xfa, 0xfe, 0x02, 0xfd, 0x00, 0x01, 0x06, 0x04, 0x04, 0xfe, 0x02, 0x05,
0x00, 0xfa, 0x04, 0x02, 0xff, 0x06, 0xfc, 0xfb, 0x06, 0x03, 0xfe, 0xfc,
0xfa, 0x06, 0x06, 0x01, 0x07, 0xfb, 0xfb, 0x01, 0x01, 0xfb, 0xff, 0xff,
0xfc, 0x05, 0xff, 0x01, 0xfe, 0xff, 0xfb, 0xfc, 0x03, 0x02, 0xfc, 0xff,
0xfa, 0x01, 0x04, 0xfd, 0xfd, 0xff, 0x06, 0xfc, 0x04, 0x01, 0xfa, 0xfd,
0x01, 0xfa, 0xfc, 0xfe, 0x01, 0xfd, 0xfe, 0xfd, 0x00, 0x01, 0xfe, 0xfd,
0xfe, 0x02, 0xf9, 0xfe, 0xff, 0x03, 0x00, 0xfa, 0xfe, 0x04, 0xf9, 0xff,
0xfe, 0xf9, 0x06, 0xfa, 0x03, 0xfc, 0xfb, 0xfb, 0xff, 0x04, 0x04, 0xff,
0x05, 0xff, 0x05, 0xff, 0xfb, 0xfb, 0x02, 0xfb, 0xfe, 0x04, 0x06, 0x06,
0xfc, 0xfc, 0x00, 0x04, 0xfd, 0xfc, 0xff, 0xfe, 0xfe, 0x00, 0xfb, 0x00,
0xfa, 0x01, 0xfb, 0x00, 0xfe, 0x00, 0x01, 0x03, 0x04, 0x06, 0x05, 0x01,
```

```
0xfa, 0x04, 0x03, 0xff, 0x06, 0xfe, 0xff, 0xfa, 0x01, 0xfc, 0xfc, 0x03,
0xfd, 0xfd, 0x05, 0xfe, 0x02, 0xfe, 0xfc, 0x02, 0xfc, 0xfa, 0x06, 0x05,
0x05, 0xfe, 0xfb, 0xfd, 0x01, 0xfb, 0xfd, 0x06, 0x05, 0x07, 0xfd, 0xfe,
0xfe, 0x05, 0x06, 0xfa, 0x03, 0xff, 0xff, 0xfd, 0x06, 0x02, 0x02, 0xfd,
0x02, 0x05, 0xfe, 0x04, 0x03, 0x00, 0x02, 0x04, 0xff, 0x05, 0xfe, 0x01,
0xfc, 0x01, 0x04, 0xff, 0x03, 0xfb, 0x03, 0x02, 0x04, 0xfd, 0xfd, 0xfc,
0x00, 0x03, 0xfd, 0x04, 0x00, 0xfa, 0xfd, 0xff, 0x00, 0x00, 0xfa, 0x05,
0x01, 0xfc, 0x03, 0x02, 0x05, 0xff, 0x04, 0x03, 0x03, 0x06, 0xfa, 0x01,
0xfd, 0xfd, 0xfb, 0xfb, 0xfd, 0xfc, 0xff, 0x06, 0x07, 0xfb, 0xe3, 0x33,
0x06, 0x07, 0x02, 0x0c, 0x05, 0xe5, 0x05, 0x05, 0xfc, 0x09, 0xfa, 0x00,
0xfe, 0x04, 0xfa, 0x03, 0xfe, 0x04, 0x07, 0x00, 0xfc, 0xfe, 0x04, 0xe1,
0xfe, 0x03, 0x06, 0xfe, 0xfe, 0xfc, 0xfa, 0xcf, 0xfe, 0xff, 0x09, 0x05,
0xfb, 0x3e, 0x04, 0x04, 0xfe, 0xfc, 0xfb, 0xfe, 0x03, 0xff, 0xff, 0x02,
0xff, 0x01, 0x00, 0xfa, 0xfe, 0xfd, 0x04, 0xfd, 0x00, 0x02, 0xfb, 0x04,
0xfb, 0xfa, 0x0a, 0xff, 0xed, 0xfd, 0xfb, 0x02, 0xfe, 0x06, 0x04, 0xfa,
0x05, 0x18, 0xfe, 0x05, 0xf9, 0xfb, 0x00, 0x01, 0x08, 0x02, 0xfb, 0x03,
0xfe, 0xfc, 0x02, 0xfa, 0x05, 0x02, 0x02, 0xfa, 0xfe, 0x03, 0xfb, 0x00,
0xfa, 0x04, 0x02, 0x01, 0xff, 0x0f, 0x08, 0x03, 0x0f, 0x03, 0xfc, 0x04,
0xfc, 0x03, 0xfe, 0xff, 0xfe, 0xff, 0x06, 0xfe, 0x02, 0x02, 0x05, 0xfe,
0x00, 0xfc, 0xff, 0x04, 0x00, 0x02, 0x1c, 0xd0, 0xfd, 0x03, 0x03, 0x04,
0x00, 0x1a, 0x05, 0x02, 0x05, 0xfb, 0xff, 0x06, 0x06, 0x07, 0xfd, 0xfd,
0x00, 0xfc, 0x06, 0xfd, 0x02, 0x01, 0xff, 0x16, 0x00, 0x00, 0x00, 0x04,
0xfc, 0xff, 0x00, 0x09, 0xfe, 0xff, 0x04, 0x02, 0xfa, 0xc6, 0x03, 0x05,
0xfd, 0xff, 0x06, 0x06, 0xfe, 0x03, 0x06, 0x06, 0x06, 0x02, 0xff, 0x02,
0xfc, 0xff, 0x01, 0x04, 0x07, 0x02, 0xfd, 0xfe, 0x05, 0xfb, 0xf4, 0xfe,
0x0c, 0x04, 0x06, 0x01, 0x06, 0x04, 0x01, 0xfd, 0x02, 0xf0, 0x02, 0x04,
0xff, 0x01, 0x07, 0xfe, 0x03, 0xfc, 0xfc, 0x02, 0x02, 0xfd, 0xfb, 0xfd,
0x05, 0xfb, 0x01, 0xfd, 0x01, 0xfa, 0x04, 0xff, 0x02, 0x01, 0xff, 0xfd,
0x03, 0x02, 0xfd, 0xfb, 0xf6, 0x05, 0x04, 0x02, 0x02, 0xfb, 0x05, 0xfc,
0xfe, 0x04, 0xfc, 0xfe, 0x03, 0xfe, 0xfa, 0x08, 0x07, 0xfe, 0x06, 0x01,
0xd2, 0xd7, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09, 0x10, 0x00, 0x00, 0x00,
0x08, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x33, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c, 0x00, 0x00, 0x00, 0x00,
0xc4, 0xd7, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x14, 0x4b, 0xdb, 0x3c, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x56, 0xd8, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x6d, 0xf7, 0xff, 0xff,
0xc0, 0x02, 0x00, 0x00, 0xf1, 0x09, 0x00, 0x00, 0x5f, 0x0f, 0x00, 0x00,
0x89, 0x00, 0x00, 0x00, 0xd3, 0x01, 0x00, 0x00, 0x0c, 0x01, 0x00, 0x00,
0x3c, 0xf8, 0xff, 0xff, 0xe2, 0xff, 0xff, 0xff, 0x27, 0x0e, 0x00, 0x00,
0x84, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x31, 0x02, 0x00, 0x00,
0x53, 0xfb, 0xff, 0xff, 0xbb, 0xff, 0xff, 0xff, 0x06, 0x02, 0x00, 0x00,
0xbd, 0xfe, 0xff, 0xff, 0x5f, 0xff, 0xff, 0xff, 0x8f, 0xff, 0xff, 0xff,
0x98, 0xff, 0xff, 0xff, 0x84, 0x00, 0x00, 0x00, 0x0c, 0x01, 0x00, 0x00,
0x0a, 0x00, 0x00, 0x00, 0xbf, 0x01, 0x00, 0x00, 0x51, 0x02, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x0d, 0xfb, 0xff, 0xff, 0x52, 0x0b, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x97, 0xff, 0xff, 0xff, 0xeb, 0xff, 0xff, 0xff,
```

```
0xb3, 0xff, 0xff, 0xff, 0xf2, 0x00, 0x00, 0x00, 0xaa, 0x02, 0x00, 0x00,
0xb8, 0xff, 0xff, 0xff, 0x5b, 0x05, 0x00, 0x00, 0x99, 0xff, 0xff, 0xff,
0x84, 0xff, 0xff, 0xff, 0xda, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0xb6, 0x00, 0x00, 0x00, 0xe4, 0x0a, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x1b, 0x00, 0x00, 0x00, 0xb3, 0xff, 0xff, 0xff,
0xb0, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x95, 0xff, 0xff, 0xff,
0xc1, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3c, 0x02, 0x00, 0x00,
0xe9, 0xff, 0xff, 0xff, 0xb9, 0xff, 0xff, 0xff, 0x1c, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xb0, 0xff, 0xff, 0xff,
0xfa, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x3f, 0xfe, 0xff, 0xff,
0xa4, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xc3, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x53, 0x05, 0x00, 0x00,
0xd6, 0x00, 0x00, 0x00, 0x6c, 0x09, 0x00, 0x00, 0xab, 0x01, 0x00, 0x00,
0x56, 0x02, 0x00, 0x00, 0xf2, 0xff, 0xff, 0xff, 0xcf, 0x01, 0x00, 0x00,
0x55, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x09, 0xfe, 0xff, 0xff, 0x95, 0x04, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xaf, 0x02, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xbe, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x9d, 0x01, 0x00, 0x00,
0xc0, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x8b, 0xff, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xa7, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x94, 0xff, 0xff, 0xff,
0xfd, 0x02, 0x00, 0x00, 0x7e, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xaa, 0xff, 0xff, 0xff, 0xc6, 0x00, 0x00, 0x00, 0x20, 0x02, 0x00, 0x00,
0xa9, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0xa8, 0x03, 0x00, 0x00,
0xf1, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x53, 0xf8, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0xce, 0x02, 0x00, 0x00,
0xe5, 0xff, 0xff, 0xff, 0x95, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xd5, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x0b, 0x00, 0x00, 0x00, 0x8f, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x8c, 0xfe, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0xda, 0x00, 0x00, 0x00,
0xb8, 0x00, 0x00, 0x00, 0x36, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x46, 0xda, 0xff, 0xff, 0x00, 0x00, 0x00, 0x02,
0x10, 0x00, 0x00, 0x00, 0x07, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x80, 0x00, 0x00, 0x00,
0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x32, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64, 0x64, 0x2f, 0x52, 0x65,
0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62, 0x6c, 0x65, 0x4f, 0x70,
0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63, 0x65, 0x00, 0x00, 0x00,
0x4c, 0xda, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x4d, 0x5e, 0x9a, 0x39, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xe2, 0xda, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00,
0x19, 0x17, 0xf7, 0xf3, 0x0a, 0x00, 0xfe, 0x08, 0x16, 0xff, 0x04, 0x10,
0x0a, 0x06, 0x0b, 0x0f, 0x0d, 0x23, 0x0c, 0x05, 0xf6, 0x05, 0xfb, 0x07,
0x00, 0x18, 0x06, 0xfc, 0xf2, 0xf7, 0xf4, 0x02, 0x01, 0x03, 0x02, 0x07,
0x11, 0x15, 0x05, 0x10, 0x18, 0x06, 0x03, 0x04, 0x00, 0xfc, 0x01, 0x0f,
0x04, 0x09, 0x01, 0xf0, 0x08, 0x07, 0x09, 0x13, 0x03, 0x0d, 0x09, 0x0a,
0xf4, 0x02, 0xe9, 0x04, 0xf4, 0xfc, 0x09, 0xfc, 0x01, 0xf2, 0x06, 0xf7,
```

```
0xf7, 0x0a, 0xff, 0xf3, 0xfe, 0x09, 0xf8, 0x00, 0x04, 0xe7, 0xfa, 0x0e,
0x17, 0xfa, 0x05, 0xfa, 0x06, 0xfc, 0x0a, 0xfe, 0x0b, 0xfe, 0x0a, 0x08,
0x06, 0xfe, 0x10, 0x0d, 0xf6, 0x0d, 0x0a, 0xfd, 0xea, 0xf3, 0x02, 0xf5,
0xfb, 0xfe, 0x0e, 0x09, 0x12, 0xf4, 0xf5, 0xfe, 0x10, 0x01, 0x09, 0xf8,
0x03, 0x0d, 0xee, 0x0d, 0x04, 0x10, 0x06, 0x0f, 0xef, 0xef, 0x07, 0xf2,
0xfa, 0xfe, 0x10, 0x05, 0xf3, 0x05, 0xf1, 0x01, 0xff, 0xff, 0xfd, 0xf0,
0x07, 0xcd, 0x02, 0x0c, 0x05, 0xf9, 0xf5, 0xd0, 0xfc, 0xda, 0x01, 0x01,
0x0a, 0xf8, 0x00, 0x02, 0x05, 0x0f, 0xf2, 0x0f, 0xd1, 0xfb, 0xf2, 0xfe,
0xcb, 0xea, 0x08, 0xfa, 0x0d, 0x07, 0xf7, 0xe0, 0x09, 0x04, 0x0e, 0xfb,
0xea, 0xf6, 0x0b, 0xcc, 0xfa, 0x0b, 0xe1, 0x09, 0x0f, 0x02, 0x32, 0x0d,
0xf5, 0xe6, 0x03, 0xf4, 0xfc, 0x0a, 0x07, 0xfd, 0xf8, 0x00, 0xfc, 0xf0,
0x06, 0xf5, 0xf9, 0xf8, 0xf7, 0xb0, 0x07, 0x0c, 0x14, 0x0f, 0xf3, 0x9b,
0x0e, 0xd1, 0x13, 0x05, 0xf6, 0x09, 0x0e, 0xf3, 0xf4, 0x0a, 0x01, 0xf7,
0xc1, 0xf9, 0x09, 0x0f, 0xbd, 0xd7, 0x0b, 0x00, 0xf3, 0x03, 0x07, 0xa9,
0xff, 0xf2, 0x05, 0x01, 0x04, 0xfb, 0xf9, 0xca, 0xf2, 0xfa, 0xbe, 0x00,
0x0c, 0x0a, 0x39, 0xf5, 0x07, 0x07, 0x0a, 0x0c, 0xf1, 0xf1, 0x03, 0x10,
0xfd, 0x0b, 0x0d, 0xff, 0x0b, 0x09, 0x00, 0x09, 0x05, 0x01, 0x0b, 0xfe,
0xf9, 0x11, 0x03, 0x01, 0xf9, 0xf1, 0xf3, 0xf2, 0x01, 0xf4, 0xfb, 0xf7,
0xfc, 0xf6, 0xf9, 0x09, 0xe4, 0xf4, 0xfd, 0x01, 0xfb, 0xf9, 0x0c, 0x0b,
0xf8, 0x03, 0x0f, 0x09, 0x04, 0xfb, 0x0c, 0x07, 0xf7, 0xfe, 0xff, 0xfe,
0xf9, 0x0a, 0x11, 0x09, 0xfa, 0x09, 0xee, 0x0c, 0xfd, 0x03, 0xfc, 0x01,
0x0c, 0xf5, 0x12, 0x0d, 0xfb, 0x0f, 0x0e, 0x06, 0x04, 0x07, 0xf5, 0xf4,
0x09, 0xec, 0xf9, 0x06, 0x0a, 0xfd, 0x07, 0xe7, 0xf7, 0xf6, 0xf8, 0x05,
0x0f, 0xfa, 0xfb, 0x04, 0x0b, 0x08, 0x00, 0x08, 0xe6, 0xf4, 0xff, 0xf8,
0xf7, 0x01, 0x0f, 0x07, 0x01, 0xfe, 0x03, 0xff, 0x14, 0xfc, 0x08, 0x0b,
0xf5, 0x02, 0xf6, 0xfb, 0x0a, 0xfa, 0x0a, 0x03, 0xff, 0x0f, 0xee, 0x0d,
0x00, 0xff, 0xf2, 0xfc, 0x01, 0x0d, 0x0b, 0x04, 0xee, 0x11, 0x00, 0xfd,
0x0d, 0xf6, 0x0a, 0xf8, 0x0c, 0xee, 0x00, 0xfd, 0xff, 0x12, 0xf8, 0xf8,
0x0f, 0xe4, 0x0a, 0x0f, 0xfa, 0xfc, 0xfc, 0xfb, 0x02, 0x11, 0x09, 0xf7,
0xf4, 0x00, 0x08, 0xf3, 0xfd, 0xf3, 0x0c, 0x0d, 0xf4, 0x08, 0x09, 0xff,
0xfc, 0x0e, 0x08, 0x07, 0xff, 0x0f, 0xf5, 0xfa, 0x04, 0xf7, 0xed, 0xfa,
0x0f, 0xf7, 0xfc, 0x03, 0x0a, 0x03, 0x16, 0x05, 0xf7, 0xf6, 0xff, 0x04,
0x02, 0x0e, 0xfd, 0x01, 0xf4, 0x04, 0x05, 0x04, 0x02, 0x13, 0x09, 0xfa,
0xfb, 0xf6, 0xf8, 0x17, 0xf5, 0x1e, 0x12, 0xf3, 0xff, 0xf2, 0xfb, 0x0a,
0x0c, 0x00, 0x0a, 0x0e, 0x1a, 0x0c, 0x0a, 0xf3, 0x2a, 0x03, 0x0e, 0xf0,
0x06, 0xfb, 0xf5, 0x1a, 0xfc, 0x0b, 0x0f, 0x04, 0xfd, 0x0e, 0x0d, 0x1c,
0xf8, 0x02, 0x16, 0xfa, 0xf1, 0xfa, 0xdd, 0x00, 0x0a, 0xf5, 0xf1, 0xfe,
0xf2, 0xf2, 0xf0, 0x0c, 0x0c, 0xfa, 0xf1, 0xf6, 0x05, 0x03, 0x0b, 0xf7,
0x03, 0xfd, 0x02, 0xfc, 0xf3, 0x08, 0x04, 0xfc, 0xfb, 0x01, 0x02, 0xfa,
0xfd, 0x0e, 0x05, 0xf5, 0xff, 0x07, 0x09, 0x09, 0x0d, 0x02, 0x01, 0x0f,
0x0e, 0xf9, 0x03, 0x0f, 0xf5, 0xf1, 0x0f, 0x0e, 0xf5, 0x01, 0xff, 0x09,
0xf8, 0xfd, 0xf5, 0x08, 0xff, 0xf1, 0x02, 0xfb, 0xfc, 0xf9, 0x09, 0xf4,
0x06, 0xef, 0xff, 0x0a, 0xfa, 0xff, 0xf6, 0xf5, 0x02, 0x05, 0xef, 0xfc,
0xf9, 0xf7, 0xf4, 0x02, 0xf8, 0xdb, 0xf8, 0x05, 0xfe, 0x13, 0x0c, 0xe1,
0x06, 0xda, 0x14, 0x0d, 0x06, 0x04, 0x07, 0xf7, 0x0a, 0x0d, 0x0f, 0xf5,
0xd5, 0xf8, 0xf5, 0xff, 0xe2, 0xf7, 0x00, 0x09, 0xf8, 0xf9, 0xf1, 0xf4,
0xfe, 0x0b, 0x07, 0x03, 0x0c, 0x02, 0xf8, 0xea, 0x04, 0x08, 0xfa, 0xfc,
0x00, 0x10, 0x2c, 0x0a, 0x00, 0xfc, 0xf0, 0x07, 0x04, 0x05, 0xf5, 0x03,
0xf5, 0x07, 0x01, 0xfe, 0x11, 0xfb, 0xff, 0xf4, 0xf3, 0xee, 0xf9, 0x0b,
0xf4, 0xfd, 0xf8, 0xee, 0xfc, 0xed, 0x0b, 0x0f, 0xf4, 0xfe, 0x0e, 0xf3,
```

```
0x0a, 0xf3, 0xf9, 0x05, 0xe6, 0x09, 0xf4, 0x05, 0xfd, 0x0a, 0x0f, 0x07,
0xf1, 0x02, 0x0c, 0xf7, 0xfd, 0x03, 0x0a, 0x0d, 0x00, 0xf3, 0xf1, 0xea,
0x03, 0x07, 0x0f, 0xff, 0x0d, 0x02, 0xf4, 0xf6, 0xff, 0xf8, 0xff, 0xf9,
0x03, 0x0f, 0xf9, 0x05, 0xf7, 0x0c, 0x00, 0xff, 0x0c, 0xfc, 0x02, 0xf3,
0xff, 0x0c, 0xf5, 0xf3, 0xf4, 0xfd, 0xfd, 0x05, 0x07, 0xf7, 0x01, 0xf2,
0x06, 0x05, 0xf5, 0xff, 0xfe, 0x03, 0xfd, 0xfa, 0x07, 0x04, 0x03, 0x09,
0xf1, 0x07, 0xfa, 0x05, 0x0d, 0x0c, 0xf4, 0x0a, 0x06, 0x09, 0x0f, 0xf3,
0xf8, 0x0d, 0xfc, 0xf4, 0xfd, 0xfb, 0xf9, 0xfe, 0xf8, 0x00, 0x02, 0x06,
0xf9, 0x09, 0x04, 0x08, 0x01, 0xf2, 0x14, 0xfd, 0xfb, 0x13, 0x08, 0xfd,
0x04, 0xf9, 0x08, 0xfe, 0x13, 0xf9, 0xfd, 0x14, 0xf9, 0xfe, 0xf3, 0x01,
0xfd, 0xe4, 0x02, 0x0a, 0x07, 0x0a, 0xf6, 0xf3, 0xfc, 0x0f, 0x0c, 0x0a,
0xe1, 0x0d, 0xf9, 0x0a, 0xe6, 0x05, 0xf4, 0x08, 0xf5, 0xf1, 0x02, 0xec,
0xf7, 0xfd, 0xfc, 0x02, 0xfc, 0xf2, 0xfd, 0xf4, 0x0b, 0x12, 0x02, 0xf4,
0x01, 0xfa, 0x02, 0x04, 0x01, 0x0b, 0xf6, 0x00, 0xfe, 0xf1, 0xf8, 0xfc,
0x0f, 0x01, 0x08, 0x0a, 0xf3, 0x10, 0xf7, 0x0e, 0x0b, 0x09, 0x05, 0x09,
0x07, 0x08, 0x04, 0x02, 0xf6, 0xfd, 0xf4, 0xf8, 0xf9, 0xfa, 0x04, 0xf6,
0xff, 0xfd, 0xff, 0x0f, 0x15, 0xfd, 0x0a, 0x07, 0x11, 0xf2, 0x09, 0xf3,
0x0a, 0x02, 0x0d, 0xf1, 0x07, 0x05, 0x0a, 0xfa, 0x00, 0x09, 0x04, 0x0c,
0xfd, 0x0f, 0xf2, 0x10, 0xfe, 0xf4, 0xf4, 0x0b, 0x0b, 0xee, 0x09, 0x0b,
0xfe, 0xfc, 0xf9, 0x04, 0x01, 0xfa, 0xf8, 0xf2, 0xf3, 0x05, 0xf8, 0x06,
0x02, 0xf4, 0x02, 0x01, 0xfd, 0xfd, 0xfc, 0x0a, 0x00, 0x08, 0x09, 0xf2,
0x0e, 0xfa, 0x08, 0xf8, 0xf8, 0x0a, 0x04, 0xf0, 0xf4, 0xfa, 0xf8, 0xf7,
0xfc, 0xf5, 0x0c, 0xfb, 0x04, 0xf2, 0x04, 0xfa, 0xfc, 0x0d, 0xfa, 0xf6,
0xf4, 0x0b, 0x0f, 0xff, 0x0d, 0xf7, 0xfe, 0xfe, 0xf9, 0xf8, 0x0c, 0xf6,
0x0c, 0xfc, 0x07, 0x0f, 0x04, 0xf5, 0x0d, 0xfa, 0x00, 0x0b, 0xfb, 0x0e,
0xfd, 0x0b, 0xfb, 0xf2, 0x07, 0x00, 0x09, 0xfe, 0xfb, 0x09, 0x08, 0xf4,
0xf6, 0xe4, 0x0a, 0x0a, 0xfe, 0xf6, 0xfb, 0xf7, 0xf8, 0x12, 0x04, 0x0f,
0xf5, 0x08, 0xf8, 0x03, 0xf8, 0xef, 0x0b, 0xf1, 0x0e, 0xf1, 0xf6, 0xf7,
0x0a, 0x02, 0x06, 0x02, 0xf4, 0x07, 0x0d, 0xe6, 0x04, 0x10, 0xf4, 0xf6,
0xf3, 0x04, 0xf0, 0xf1, 0xf7, 0x01, 0x08, 0x0d, 0xf3, 0x0a, 0x05, 0x01,
0xf2, 0x0b, 0xfc, 0xfa, 0x08, 0xf3, 0xfd, 0x05, 0xfe, 0xea, 0x0d, 0x01,
0x0c, 0xfc, 0xfb, 0xe8, 0xf8, 0xf3, 0xf9, 0xf3, 0x08, 0x0c, 0x0c, 0x02,
0xfb, 0xff, 0xf9, 0x00, 0xf2, 0xfa, 0xf1, 0x0f, 0xf8, 0x03, 0xf8, 0xf8,
0xf1, 0x02, 0xf4, 0xec, 0xff, 0xfe, 0xf4, 0xfd, 0x09, 0xf8, 0x04, 0xf2,
0x0d, 0x09, 0xf7, 0x0f, 0xf1, 0xf5, 0xf1, 0xf3, 0x00, 0xfb, 0x01, 0x0d,
0xff, 0x0a, 0x0d, 0x08, 0xfd, 0x0b, 0xf5, 0x0b, 0xfc, 0x0a, 0xf2, 0xf6,
0x06, 0xf5, 0x05, 0x08, 0x0e, 0xf3, 0x0a, 0xff, 0x0b, 0x02, 0xfa, 0x04,
0x09, 0xfa, 0xf7, 0xf9, 0xfd, 0x0f, 0xfd, 0x0d, 0xe6, 0x02, 0x09, 0x0f,
0xff, 0xfd, 0xf1, 0x0c, 0xfb, 0x02, 0x03, 0xf8, 0xf8, 0xfb, 0xf3, 0x0f,
0xfb, 0x01, 0xf4, 0xef, 0x0d, 0x06, 0xf9, 0xf1, 0x08, 0xfe, 0xfa, 0xfc,
0x07, 0xf3, 0x0c, 0x05, 0xfc, 0x05, 0xf6, 0xf1, 0x09, 0x01, 0x0b, 0x01,
0x04, 0xfc, 0x0b, 0xf3, 0xf5, 0xf3, 0xf7, 0xf2, 0x04, 0x02, 0x0b, 0x05,
0x08, 0xf5, 0x04, 0xfa, 0xf6, 0x00, 0xfd, 0x07, 0x0a, 0x0a, 0xfe, 0x0d,
0x0c, 0xf6, 0xf0, 0x01, 0xfc, 0xfa, 0x04, 0xf1, 0xf4, 0x07, 0xfb, 0x01,
0x03, 0xf6, 0xf2, 0xf4, 0x01, 0x02, 0xfe, 0x08, 0x03, 0xed, 0xf8, 0x0e,
0x07, 0x08, 0x02, 0xf1, 0xf4, 0x01, 0xf4, 0x04, 0x0a, 0x0e, 0xfa, 0xf9,
0x08, 0xfc, 0x01, 0x10, 0xf1, 0xf4, 0x04, 0x07, 0xf5, 0x05, 0xf9, 0x04,
0x07, 0x05, 0x09, 0x0f, 0xfb, 0x0e, 0x05, 0xf8, 0xf4, 0xf2, 0xfc, 0x0c,
0x08, 0xfd, 0xfd, 0x05, 0x01, 0xfe, 0xf1, 0x10, 0xf4, 0xf0, 0xfb, 0x0c,
0xfa, 0xff, 0x03, 0xf2, 0xf0, 0x02, 0xef, 0xf1, 0xff, 0x00, 0x07, 0xfd,
```

```
0xf9, 0xf1, 0x09, 0x0f, 0x0a, 0x0c, 0xfe, 0xf7, 0x07, 0xeb, 0x0e, 0x0b,
0x04, 0x05, 0xf5, 0x0c, 0xef, 0xf9, 0xff, 0xf6, 0x0a, 0x06, 0xfe, 0xff,
0x08, 0xeb, 0xff, 0xfc, 0x06, 0xff, 0xf2, 0xee, 0x09, 0xf6, 0xff, 0x06,
0x0a, 0xff, 0xf3, 0x0f, 0x0e, 0x03, 0xf3, 0xf3, 0xee, 0xed, 0x07, 0xfb,
0xe6, 0x04, 0xfb, 0xfd, 0x0f, 0xff, 0x01, 0xf4, 0x00, 0xf8, 0x03, 0x05,
0x08, 0x0f, 0x0b, 0xda, 0x0e, 0xf1, 0xfb, 0xf5, 0xf4, 0x0d, 0x05, 0xf6,
0x0a, 0xed, 0xf8, 0xf7, 0xff, 0xfd, 0x09, 0xff, 0xfd, 0x10, 0xf3, 0xf6,
0x0c, 0x07, 0xf5, 0x04, 0x0a, 0x02, 0xf6, 0x05, 0x16, 0x0f, 0xff, 0xe7,
0x00, 0xec, 0x0b, 0xfc, 0x0c, 0x09, 0xfc, 0xf7, 0xf3, 0x08, 0xff, 0xfa,
0xed, 0x00, 0x00, 0x01, 0x02, 0xf7, 0x03, 0x0a, 0x0e, 0x04, 0x06, 0x02,
0xfe, 0x03, 0x0c, 0x19, 0x04, 0x0f, 0x0c, 0xf4, 0xfc, 0x03, 0xfd, 0x02,
0xfc, 0x07, 0xea, 0xf9, 0x0b, 0x08, 0x08, 0xf9, 0x0b, 0x0e, 0x02, 0x0d,
0x05, 0xfb, 0x0c, 0xf5, 0x0c, 0xfa, 0x0a, 0x07, 0x09, 0xf3, 0xf7, 0xfc,
0x0b, 0xf9, 0xf1, 0xf6, 0x00, 0x04, 0x0f, 0x00, 0x0d, 0x03, 0x00, 0x0a,
0x09, 0xfa, 0xf4, 0x0a, 0xfc, 0x06, 0xf4, 0x05, 0x07, 0x04, 0xf6, 0x09,
0xfc, 0xf9, 0x0f, 0xf9, 0x0a, 0xf1, 0x00, 0x18, 0x0b, 0xf5, 0x03, 0xfc,
0x09, 0xfc, 0xfe, 0xf4, 0x0a, 0xf6, 0xef, 0xf6, 0xf7, 0x08, 0x00, 0xf6,
0x05, 0xf5, 0x08, 0x11, 0x0a, 0x0c, 0x09, 0x09, 0x0f, 0xf2, 0xf6, 0x08,
0x0d, 0xf6, 0x0a, 0x06, 0x10, 0x08, 0xf6, 0xf6, 0x00, 0xfc, 0xfe, 0x08,
0xf7, 0xfa, 0x02, 0x0f, 0x0d, 0x00, 0xfe, 0x10, 0xef, 0x01, 0xf8, 0xfc,
0x02, 0xf0, 0x02, 0x03, 0x0f, 0x10, 0x04, 0x08, 0x11, 0x02, 0x0d, 0x12,
0x09, 0xf9, 0x02, 0xe4, 0x0b, 0x0b, 0x05, 0x00, 0x0a, 0x13, 0xfe, 0x0a,
0xfc, 0x00, 0xf6, 0x04, 0x0c, 0x0d, 0x16, 0x12, 0xfc, 0xff, 0x02, 0x0b,
0x08, 0xf8, 0x05, 0x01, 0x01, 0xf9, 0x04, 0x0b, 0x06, 0x09, 0x08, 0xee,
0xfa, 0x01, 0xf8, 0x09, 0xf6, 0xf9, 0xfa, 0x0c, 0x04, 0x09, 0x00, 0x0f,
0xf2, 0xf9, 0x04, 0xfc, 0xe5, 0x08, 0x00, 0xf4, 0x09, 0xfa, 0x02, 0x01,
0x10, 0x03, 0xf3, 0x08, 0xfd, 0x0a, 0xfd, 0xe3, 0xf2, 0x08, 0xf8, 0x10,
0xfe, 0x05, 0xef, 0xf7, 0x0d, 0x03, 0xf8, 0x06, 0xf7, 0xf3, 0x09, 0x02,
0x0b, 0xf9, 0x0b, 0xf9, 0x0f, 0x0b, 0x06, 0xf5, 0xf1, 0xf9, 0xfc, 0x07,
0x0f, 0xf3, 0xf6, 0xf9, 0x09, 0xf8, 0xf9, 0x01, 0x0c, 0x0d, 0xfd, 0xfd,
0x08, 0x07, 0xfa, 0x06, 0x00, 0xfc, 0x02, 0xf7, 0xfc, 0xf5, 0xf8, 0xfc,
0xf8, 0x06, 0x04, 0xf7, 0xf9, 0x05, 0xf3, 0xf5, 0xf8, 0xfe, 0xf4, 0xf7,
0xfe, 0xf9, 0xfe, 0x00, 0xff, 0xfd, 0xf3, 0xfe, 0x11, 0x05, 0x03, 0x01,
0x0a, 0xfa, 0x08, 0xfa, 0x11, 0x0e, 0x0b, 0xf3, 0x01, 0x0b, 0x03, 0xfc,
0x0c, 0x0b, 0xf2, 0x01, 0x03, 0x10, 0x0c, 0xf9, 0x03, 0x12, 0xff, 0x08,
0xf8, 0xf4, 0x0f, 0xf3, 0xf3, 0xf4, 0x03, 0xfb, 0x0c, 0x03, 0x0a, 0x0e,
0x04, 0xf9, 0x09, 0xfe, 0x04, 0xfe, 0xf4, 0xf4, 0x0b, 0xf5, 0xfc, 0x02,
0xf8, 0xf1, 0xf8, 0x15, 0x0f, 0x04, 0xf9, 0xfb, 0xf5, 0x10, 0xd8, 0xfb,
0xf1, 0xec, 0x0e, 0x03, 0xf5, 0xf8, 0xfd, 0x08, 0x02, 0x08, 0xeb, 0x00,
0xfa, 0x07, 0x10, 0xf3, 0x05, 0xd9, 0xf7, 0x09, 0x0a, 0x14, 0xf9, 0xd2,
0x09, 0xc9, 0x01, 0xf7, 0x01, 0xf9, 0x04, 0xf0, 0x09, 0x0a, 0xf7, 0x07,
0xd5, 0xef, 0x0c, 0x0d, 0xdc, 0xea, 0xf2, 0xff, 0xfa, 0xf1, 0x0e, 0xe7,
0x02, 0x08, 0x04, 0xf6, 0x06, 0xfa, 0x02, 0xda, 0xef, 0x01, 0xf3, 0x03,
0x07, 0x10, 0x32, 0xf2, 0xf6, 0xf4, 0x03, 0x03, 0x0e, 0xf3, 0xfe, 0x0e,
0xf3, 0x04, 0xf6, 0xfe, 0x04, 0x10, 0x0f, 0x07, 0xfa, 0xff, 0x0f, 0x04,
0xfb, 0xfa, 0x03, 0xf8, 0xfd, 0x05, 0xfc, 0x00, 0x04, 0xfc, 0x10, 0xf1,
0x03, 0x02, 0xfe, 0x05, 0x06, 0x00, 0x00, 0x0c, 0xf7, 0xf1, 0xff, 0xff,
0xf1, 0x10, 0xfa, 0x09, 0xf4, 0x06, 0x08, 0x0a, 0x0a, 0x02, 0xf5, 0x04,
0xf5, 0xff, 0xf3, 0xf6, 0x01, 0xfd, 0xff, 0xfc, 0xff, 0xf7, 0xee, 0xf2,
0x01, 0x0c, 0xf5, 0xf5, 0xfa, 0xf1, 0x08, 0xf2, 0xf9, 0xfc, 0xf1, 0xf3,
```

```
0xef, 0x03, 0xf6, 0xf2, 0xf5, 0x01, 0xf2, 0xfe, 0xfc, 0xfd, 0x0a, 0x08,
0xf2, 0x07, 0x02, 0xf1, 0xff, 0x0c, 0xf6, 0x01, 0x0b, 0xee, 0x06, 0x04,
0x05, 0xff, 0xfa, 0xfc, 0x01, 0xf6, 0x00, 0x09, 0x0b, 0xf6, 0x0b, 0xfb,
0xee, 0xfd, 0xf5, 0xff, 0x04, 0x04, 0xff, 0xf3, 0xf4, 0x08, 0xfc, 0xfc,
0xf6, 0x06, 0x0f, 0x09, 0xf1, 0x02, 0x00, 0xf6, 0xf8, 0x02, 0xf5, 0x09,
0xfd, 0x07, 0xf3, 0x08, 0x02, 0x09, 0x0d, 0xf3, 0xf7, 0xf7, 0xf6, 0x0f,
0x05, 0x09, 0x05, 0xf6, 0xff, 0xf2, 0x0e, 0x0e, 0xfc, 0xf6, 0xff, 0x05,
0xff, 0x02, 0x06, 0x0f, 0x09, 0xf3, 0x07, 0x03, 0x09, 0xfd, 0xf9, 0xfb,
0x03, 0x0c, 0xf2, 0xf8, 0xfa, 0x01, 0x01, 0x0c, 0xf4, 0xfa, 0x06, 0x07,
0xf8, 0x09, 0x05, 0x06, 0x02, 0xf2, 0x05, 0xfc, 0x0b, 0x07, 0xfe, 0xf8,
0x02, 0xfb, 0xf2, 0xf6, 0xf6, 0x06, 0x0b, 0x01, 0xf0, 0x01, 0x0e, 0x0c,
0xf9, 0xf0, 0x0f, 0xf3, 0xfa, 0xf6, 0x00, 0xfd, 0xf6, 0xfa, 0x10, 0x0e,
0x03, 0x08, 0x0b, 0xfc, 0xf7, 0x08, 0x07, 0xf4, 0xf0, 0xfd, 0x0e, 0x0d,
0xfd, 0x04, 0x09, 0x07, 0x0a, 0x01, 0x0a, 0x02, 0xf1, 0x09, 0xfc, 0xfe,
0xfc, 0xfa, 0x05, 0x01, 0x02, 0xf3, 0x05, 0x09, 0xee, 0xf6, 0x07, 0x0e,
0x05, 0x08, 0xf6, 0x03, 0x01, 0x09, 0xf4, 0x08, 0x09, 0x0d, 0x0d, 0xf4,
0x02, 0xeb, 0xf4, 0x04, 0x00, 0x07, 0x0e, 0xe9, 0xf5, 0xf1, 0x07, 0xf3,
0xf2, 0x06, 0xf7, 0x0f, 0xfb, 0xf8, 0xff, 0xfa, 0xe7, 0x02, 0x02, 0x01,
0xe4, 0xee, 0x02, 0xf9, 0x0b, 0xf9, 0x07, 0x09, 0x04, 0xf8, 0x04, 0x09,
0xfe, 0xf5, 0x0c, 0xdb, 0xfa, 0x00, 0xee, 0xfc, 0x0e, 0x0d, 0xf4, 0x0e,
0xfe, 0x04, 0x08, 0xf6, 0x02, 0xf2, 0x04, 0x01, 0x09, 0x07, 0x06, 0xf5,
0x0c, 0x02, 0xfb, 0xf1, 0x04, 0x03, 0xf1, 0xfd, 0x13, 0x0f, 0x00, 0xf4,
0x07, 0xf2, 0x10, 0x01, 0x06, 0xfa, 0x0d, 0xfb, 0xfb, 0xff, 0xf4, 0xf7,
0xe3, 0x03, 0xf5, 0xf5, 0xf0, 0xef, 0x06, 0x03, 0xfc, 0xff, 0xf5, 0xf6,
0x12, 0x08, 0x0d, 0xf6, 0x07, 0xf2, 0x02, 0xe4, 0x03, 0x00, 0x05, 0xf4,
0x00, 0x0e, 0xf2, 0xf9, 0x0b, 0x08, 0xf5, 0x07, 0xfd, 0x0b, 0xfe, 0x06,
0xfd, 0x00, 0xf9, 0xf6, 0xf5, 0x08, 0xf8, 0x07, 0xf8, 0x02, 0xf2, 0xfb,
0x06, 0xfc, 0x05, 0xee, 0x01, 0xf0, 0x0b, 0xf1, 0x0e, 0x09, 0xfa, 0x10,
0x06, 0xf8, 0x00, 0xfd, 0x05, 0xf0, 0xf4, 0x05, 0x02, 0xfa, 0x0d, 0x03,
0x0d, 0xf5, 0xf2, 0x05, 0xef, 0xfd, 0x04, 0x04, 0xf1, 0x0e, 0xfd, 0xfb,
0x06, 0xef, 0xf1, 0x09, 0xf3, 0x02, 0x09, 0x05, 0xfa, 0xff, 0xfe, 0xf5,
0xfc, 0x06, 0x06, 0xef, 0xfc, 0xfd, 0xf9, 0xf3, 0x00, 0xfe, 0xff, 0x05,
0xf6, 0xd7, 0xf7, 0x0c, 0x12, 0xfc, 0xf3, 0xfe, 0xf4, 0xe3, 0x10, 0xf6,
0x07, 0x0b, 0x03, 0x06, 0x0a, 0x0c, 0xfe, 0xf0, 0xf4, 0x07, 0x0c, 0x03,
0xf3, 0xe5, 0x06, 0xfd, 0x03, 0xff, 0x08, 0xf5, 0x08, 0xff, 0xff, 0x0c,
0x01, 0x04, 0xf6, 0xdb, 0xfd, 0x03, 0xe3, 0xf8, 0xf5, 0x0d, 0x2d, 0x04,
0x01, 0xfe, 0xf8, 0x06, 0xfd, 0xf4, 0x06, 0xfa, 0x0b, 0xf7, 0x08, 0x10,
0xfe, 0xff, 0x03, 0x08, 0xfd, 0xf8, 0x06, 0xf8, 0x03, 0x0d, 0xfb, 0xf3,
0x0a, 0x07, 0xff, 0xf6, 0xf6, 0xfb, 0xf7, 0xf8, 0x0c, 0x08, 0xff, 0xf5,
0xf0, 0x00, 0xf7, 0x09, 0x02, 0x01, 0xf5, 0x05, 0x07, 0xf1, 0x05, 0x0c,
0xfa, 0xf5, 0x06, 0xf9, 0xf0, 0xf4, 0x08, 0x0c, 0xf4, 0x0d, 0xf8, 0xfc,
0xfc, 0xfd, 0x03, 0xf1, 0xfa, 0x0d, 0xee, 0xf5, 0x0a, 0x09, 0x0b, 0xff,
0x07, 0xf8, 0x08, 0xf8, 0xf4, 0x07, 0x0f, 0xfb, 0x01, 0xf0, 0xff, 0x06,
0xef, 0x02, 0x09, 0xf8, 0xfb, 0x02, 0xfd, 0x00, 0x0c, 0xf9, 0x0b, 0xff,
0x0e, 0x0c, 0xfa, 0x06, 0x03, 0xf4, 0x07, 0xf3, 0xf8, 0xf6, 0x08, 0xf9,
0xf3, 0xf8, 0x0c, 0xef, 0x00, 0x07, 0x01, 0xf4, 0xfd, 0x0f, 0x04, 0xfe,
0xef, 0xf2, 0xf9, 0x05, 0xfe, 0x04, 0xe8, 0xfb, 0xf2, 0x05, 0xff, 0x07,
0xfb, 0x02, 0xf8, 0xfe, 0x08, 0x00, 0x03, 0x09, 0x03, 0x03, 0xfc, 0xf4,
0x11, 0x04, 0xfc, 0x01, 0x08, 0x11, 0x10, 0xec, 0x0c, 0xfc, 0x0d, 0xf3,
0x05, 0x0f, 0x0a, 0x02, 0x0a, 0xfa, 0xf5, 0xf4, 0xfb, 0xf8, 0x03, 0xf6,
```

```
0xf5, 0x03, 0xfa, 0x09, 0x00, 0x07, 0xf1, 0x01, 0x0a, 0xf4, 0x06, 0x10,
0xfe, 0xf1, 0xfc, 0xe8, 0xf9, 0x0c, 0xff, 0xf2, 0x08, 0x0e, 0xfd, 0xf7,
0x0d, 0x03, 0xf4, 0x00, 0x05, 0xf6, 0x02, 0xf7, 0xfe, 0xf3, 0xf2, 0x0a,
0xff, 0x08, 0xfe, 0x0b, 0xf6, 0x05, 0x07, 0xfd, 0xf7, 0x09, 0xfb, 0x09,
0x07, 0x03, 0xfc, 0xfc, 0xfa, 0xfe, 0x0b, 0x0f, 0xf8, 0xf8, 0xff, 0x04,
0xf8, 0x0d, 0x0a, 0xf8, 0x0e, 0xfc, 0xfe, 0x0e, 0x02, 0xf5, 0xf4, 0x01,
0x00, 0x00, 0x09, 0xf2, 0x0f, 0x09, 0x0b, 0x0d, 0xf5, 0x04, 0xf6, 0x07,
0x05, 0x0e, 0xf6, 0xf7, 0xf0, 0x05, 0xef, 0x02, 0xf6, 0x00, 0x0e, 0xf9,
0x04, 0xfa, 0xf9, 0xf5, 0x03, 0x05, 0x07, 0x02, 0xff, 0xfe, 0xf5, 0xf1,
0x0d, 0x0c, 0x09, 0x0a, 0xff, 0xfe, 0xf4, 0x06, 0x01, 0x08, 0x0c, 0x06,
0xfa, 0x0c, 0xfa, 0x09, 0xfd, 0xf1, 0xf6, 0x09, 0xf8, 0x00, 0xfb, 0x09,
0xf5, 0x10, 0x01, 0x09, 0xf5, 0xf8, 0xf2, 0x3b, 0xfe, 0xf1, 0xf6, 0xef,
0xf3, 0xfe, 0xf7, 0xfe, 0x01, 0x06, 0x04, 0xf2, 0xf3, 0xee, 0xf2, 0x0e,
0xff, 0x09, 0x14, 0xfc, 0xe7, 0x07, 0xf7, 0x09, 0x02, 0x07, 0x09, 0xef,
0x09, 0x9a, 0x00, 0x16, 0x01, 0x0e, 0xff, 0xb1, 0xf0, 0xbb, 0x0e, 0x01,
0xfd, 0x0d, 0x05, 0xf5, 0xf5, 0x07, 0xfb, 0xfc, 0xcd, 0xfa, 0x04, 0x0d,
0xce, 0xd5, 0xf9, 0x08, 0xfe, 0x0b, 0xf8, 0xaf, 0x12, 0x06, 0xfe, 0xeb,
0xfb, 0xf9, 0xf7, 0xb9, 0xfd, 0xfd, 0xae, 0x0a, 0xf6, 0x03, 0x31, 0x0d,
0x03, 0xfa, 0x08, 0xf4, 0x0c, 0x08, 0xf4, 0x04, 0x07, 0xff, 0xf7, 0x0a,
0x03, 0xf5, 0xf9, 0xfb, 0x04, 0xfa, 0x07, 0x04, 0x04, 0xfe, 0x06, 0xfe,
0xf3, 0x0d, 0x02, 0x06, 0x09, 0xff, 0x08, 0xfc, 0xf3, 0x09, 0x03, 0xf6,
0xfd, 0xf8, 0x06, 0xfa, 0xf0, 0xf1, 0xfa, 0x07, 0x0c, 0x08, 0xf9, 0xf6,
0x0c, 0xf1, 0xfa, 0x09, 0x0d, 0x0f, 0xf5, 0x01, 0xff, 0xfb, 0x00, 0xfb,
0xfc, 0xfb, 0xf2, 0x09, 0xff, 0x01, 0xfa, 0x0a, 0x05, 0x09, 0x03, 0xfc,
0x00, 0x03, 0x03, 0x0d, 0xf7, 0xf8, 0x0e, 0x0c, 0x00, 0x09, 0x03, 0xfd,
0x09, 0x06, 0xf9, 0x07, 0x09, 0xfa, 0xfd, 0xfc, 0x0a, 0xf6, 0x07, 0x0d,
0x01, 0x02, 0x0d, 0xf5, 0x04, 0xfd, 0xff, 0x02, 0xf3, 0x09, 0x06, 0xf8,
0xf2, 0xf5, 0xf2, 0xfe, 0x09, 0x02, 0xff, 0x03, 0xf7, 0x02, 0x0f, 0xf5,
0xf5, 0xf9, 0x08, 0xf1, 0x0e, 0x02, 0xfc, 0x02, 0xf1, 0x09, 0x07, 0xf0,
0xfc, 0x02, 0xf4, 0x10, 0x0b, 0x0c, 0x05, 0xf4, 0xfd, 0x04, 0xfa, 0xff,
0x0c, 0xed, 0x07, 0xf7, 0xf7, 0x01, 0xf1, 0xf8, 0xf5, 0xea, 0x04, 0x00,
0xf2, 0x0a, 0x00, 0x03, 0xf9, 0xfe, 0xfd, 0x0c, 0xfe, 0x0c, 0x05, 0xfe,
0xf9, 0xfa, 0x0e, 0x0f, 0xf1, 0x02, 0x04, 0xee, 0xfd, 0xf5, 0xf2, 0x11,
0x0a, 0x10, 0x03, 0xef, 0x05, 0x00, 0x08, 0x02, 0x0f, 0x00, 0xf4, 0x0b,
0x06, 0x0e, 0x00, 0x09, 0xf3, 0xf2, 0xf3, 0xfe, 0x09, 0x03, 0x08, 0xfb,
0xf1, 0x0d, 0xf8, 0x03, 0xef, 0x0c, 0xf5, 0xfd, 0x0d, 0x0b, 0xfe, 0xfb,
0x0a, 0x0b, 0x06, 0x0a, 0xf3, 0xfa, 0x0d, 0xf0, 0x00, 0xf1, 0xf3, 0xf6,
0xf9, 0x04, 0x03, 0x10, 0xfb, 0xfd, 0x08, 0xf9, 0x0a, 0x06, 0xf0, 0xf5,
0x06, 0xf3, 0x08, 0xf7, 0xff, 0x02, 0xf5, 0xfb, 0x08, 0x04, 0xef, 0x06,
0x00, 0x0b, 0xf7, 0xfe, 0xfb, 0xfd, 0xf8, 0x0b, 0xf5, 0x0b, 0x0c, 0xf4,
0xf3, 0x0c, 0xfc, 0x05, 0x0c, 0x0c, 0xf1, 0xee, 0x0d, 0xf4, 0xfb, 0xfb,
0xf6, 0xf8, 0x03, 0x08, 0xfe, 0xfe, 0xfa, 0x05, 0x0a, 0x04, 0xf3, 0xff,
0xf8, 0x02, 0xf7, 0xfa, 0xfa, 0xf2, 0x09, 0xf9, 0x0b, 0x01, 0x0e, 0xf2,
0x01, 0x01, 0x02, 0xf5, 0xf6, 0xf4, 0xf7, 0xfc, 0xfb, 0xf4, 0x06, 0x0f,
0xf1, 0x0b, 0xfb, 0xfd, 0xfd, 0x0d, 0xfb, 0x01, 0x01, 0xfd, 0xf3, 0x0b,
0x0e, 0x09, 0x0d, 0x03, 0xf8, 0xf2, 0x07, 0xfc, 0x0f, 0xf0, 0xfe, 0xf6,
0x02, 0x08, 0x07, 0xff, 0xf1, 0x06, 0x0b, 0x05, 0xf1, 0x07, 0xfe, 0x0d,
0xfd, 0x06, 0xf8, 0x0b, 0x0d, 0x10, 0x09, 0xf0, 0xff, 0xf4, 0x04, 0xf8,
0x04, 0x09, 0x04, 0x00, 0x04, 0xfe, 0x07, 0x00, 0xf2, 0x0d, 0xf7, 0x0c,
0xfc, 0x00, 0x09, 0x03, 0x05, 0x01, 0xf9, 0xfd, 0x08, 0x06, 0xfb, 0xfc,
```

```
0xf9, 0xfe, 0xf5, 0x04, 0x0a, 0xf1, 0x04, 0xf9, 0x09, 0x06, 0x01, 0x0e,
0xfc, 0xfb, 0x0b, 0xf8, 0xf9, 0x08, 0xfe, 0xf5, 0xf4, 0x09, 0xf4, 0x07,
0x09, 0xf6, 0xfb, 0xf4, 0xf1, 0x01, 0xf0, 0x03, 0xf5, 0xfd, 0xfd, 0xf3,
0x0e, 0x02, 0xfa, 0xf5, 0x0b, 0xf3, 0xf7, 0x0c, 0xfb, 0x06, 0x0a, 0xf3,
0x05, 0xf4, 0x05, 0xf8, 0xf7, 0x0c, 0x02, 0xf5, 0xf8, 0x06, 0xf4, 0x08,
0x03, 0xfe, 0x01, 0x09, 0xfb, 0x04, 0x0a, 0xf9, 0x0b, 0x0b, 0x10, 0x08,
0x01, 0xf7, 0x04, 0xfe, 0x03, 0x0b, 0x0a, 0xfa, 0xf8, 0xeb, 0xfd, 0xfb,
0xfe, 0xfb, 0x0f, 0xf3, 0x09, 0xf8, 0x0b, 0xf0, 0x09, 0x04, 0x04, 0x00,
0xf4, 0xfc, 0xfb, 0x04, 0xe7, 0x0a, 0x02, 0xf8, 0xfb, 0x05, 0xfc, 0x03,
0xf3, 0x07, 0xf5, 0xf1, 0x09, 0x08, 0x0d, 0x01, 0xf9, 0xf1, 0x03, 0xfe,
0x0b, 0x11, 0xef, 0xff, 0x08, 0x06, 0xfa, 0x03, 0xfe, 0xf2, 0xfa, 0x0e,
0xfc, 0x0c, 0xfe, 0xf2, 0x08, 0xf5, 0xfb, 0x0f, 0x04, 0x02, 0xf0, 0xf6,
0xf4, 0xf3, 0x0d, 0x06, 0xf3, 0x01, 0x02, 0xfb, 0xfc, 0xf2, 0xfa, 0x0f,
0x09, 0xf6, 0xf9, 0x0d, 0xf3, 0x0d, 0x08, 0xf6, 0x06, 0x0c, 0xfd, 0x05,
0xf2, 0xf1, 0x01, 0x03, 0xf6, 0xf5, 0xf6, 0xfd, 0x0c, 0x0d, 0x0d, 0xf3,
0x0e, 0xfa, 0xf8, 0x00, 0xfa, 0x03, 0x08, 0xf5, 0x0c, 0x0a, 0xfd, 0x0e,
0x0f, 0x01, 0x02, 0x08, 0xf7, 0x0f, 0xf6, 0x0e, 0x0c, 0x0a, 0x04, 0xf8,
0x03, 0x09, 0xf6, 0x0f, 0x0b, 0xfe, 0x08, 0x05, 0x0a, 0x04, 0xff, 0xf5,
0xf4, 0xee, 0x15, 0x0f, 0x04, 0x0d, 0xfe, 0x10, 0xf3, 0x15, 0x0a, 0x14,
0xec, 0xff, 0xf5, 0x01, 0x06, 0xf1, 0xf6, 0xfe, 0x01, 0xf9, 0xfa, 0x03,
0x10, 0x0e, 0x0c, 0xf8, 0x10, 0xf4, 0x08, 0xf0, 0x0d, 0x03, 0xf5, 0x06,
0x0f, 0xfb, 0xee, 0xf8, 0xee, 0xef, 0xf6, 0x02, 0xfb, 0x07, 0xf3, 0x06,
0xf0, 0x0d, 0x01, 0xfa, 0x0b, 0xf2, 0x03, 0x05, 0x00, 0xfa, 0xf0, 0x05,
0x09, 0xfc, 0xfe, 0xea, 0x0b, 0xf8, 0x04, 0x09, 0xf1, 0xfd, 0x07, 0x0a,
0x0c, 0x00, 0xfc, 0x0b, 0xff, 0xf1, 0x03, 0xf3, 0xff, 0x07, 0x01, 0x0e,
0x0f, 0xf3, 0xfd, 0xfc, 0x05, 0xf5, 0xf2, 0xfb, 0xfd, 0xf7, 0x06, 0xfd,
0xf4, 0xf2, 0x0b, 0x0d, 0xff, 0x06, 0xf0, 0x0d, 0xef, 0x00, 0xf4, 0xfb,
0x0c, 0xf8, 0x0e, 0x03, 0xfb, 0x0d, 0x0b, 0x09, 0x06, 0xf1, 0xf1, 0x07,
0xfc, 0xf4, 0x09, 0x01, 0xf1, 0xf6, 0xf7, 0xf1, 0xfd, 0xf1, 0x0c, 0x0a,
0xfc, 0x02, 0x0a, 0xfc, 0xf5, 0xfb, 0x0d, 0xfc, 0xf1, 0x01, 0x09, 0x06,
0xff, 0xf2, 0xf2, 0x06, 0x03, 0xfc, 0xf0, 0xf7, 0xfd, 0x04, 0xf3, 0xfc,
0xf3, 0x06, 0xf5, 0x02, 0xf5, 0x01, 0xfc, 0xfe, 0x0a, 0xf4, 0xf9, 0xfe,
0x0a, 0x0c, 0x04, 0xf3, 0xf1, 0x0b, 0xf5, 0x11, 0xf7, 0x09, 0x0c, 0xf7,
0xf3, 0xf5, 0xf6, 0x07, 0x05, 0x06, 0xfc, 0x07, 0xf7, 0x0b, 0x01, 0xf4,
0xf9, 0xf5, 0xf8, 0xff, 0x09, 0xfc, 0x09, 0x0d, 0x00, 0x0e, 0x0a, 0x07,
0x03, 0x02, 0xfd, 0xf3, 0x11, 0xf3, 0xfe, 0x08, 0xfc, 0x0f, 0x07, 0x0c,
0x02, 0x0d, 0xff, 0xfa, 0x0c, 0x0d, 0x05, 0x09, 0xfb, 0x11, 0xf2, 0x0a,
0x01, 0x03, 0xfe, 0x00, 0x04, 0x01, 0x10, 0xf9, 0xf3, 0xf3, 0x02, 0x09,
0xf8, 0xfb, 0xf9, 0x0e, 0xf0, 0x10, 0xf4, 0xf5, 0x07, 0x00, 0xf7, 0xfc,
0x01, 0xf8, 0xfc, 0xf3, 0xf3, 0x0c, 0xfb, 0xf9, 0x09, 0xfd, 0x06, 0x03,
0xf6, 0xf8, 0x0d, 0x06, 0x04, 0xf5, 0x05, 0x0a, 0x0e, 0xf1, 0x0d, 0x07,
0xfd, 0xfb, 0xfb, 0x06, 0x07, 0xf1, 0x01, 0x09, 0xfd, 0xfe, 0xf5, 0xf6,
0xf5, 0xfd, 0xf6, 0x00, 0xf5, 0xf3, 0xf5, 0xfa, 0xff, 0xfc, 0x09, 0xff,
0x00, 0xfb, 0xf0, 0xff, 0xf3, 0xfa, 0x03, 0xf9, 0xf7, 0x00, 0xf2, 0x0c,
0xf0, 0x0b, 0xf5, 0x0e, 0x0c, 0x0b, 0x0e, 0xf1, 0x0f, 0xf7, 0xf0, 0x08,
0xfa, 0x01, 0x0d, 0xf1, 0x0f, 0xf3, 0x0b, 0xf8, 0x0e, 0x0d, 0xf7, 0x07,
0x0f, 0x08, 0xf1, 0xf2, 0x0f, 0x0a, 0xff, 0xfa, 0xf2, 0xfb, 0xf3, 0xff,
0xf9, 0x03, 0x04, 0xf9, 0x00, 0xf7, 0xf5, 0xf6, 0x09, 0x04, 0x0f, 0xf4,
0xf5, 0xf1, 0x03, 0xf3, 0xf5, 0x04, 0xf8, 0x0a, 0xfd, 0x0b, 0x06, 0x02,
0xfd, 0x02, 0xff, 0xf5, 0x0a, 0x08, 0xf3, 0x0d, 0xf0, 0xf6, 0x02, 0xf3,
```

```
0xff, 0x05, 0xf2, 0xfe, 0xf7, 0x09, 0x00, 0xfa, 0x04, 0x03, 0xf9, 0xfd,
0x0a, 0xf7, 0xf6, 0x01, 0xfd, 0xfa, 0xf4, 0x05, 0x0a, 0xf9, 0x07, 0x02,
0x08, 0xfe, 0x03, 0xf3, 0x02, 0xf8, 0x00, 0xf2, 0xf9, 0x00, 0x07, 0x09,
0x04, 0x03, 0xf7, 0xfa, 0x03, 0xed, 0xf3, 0x04, 0x07, 0x0d, 0x01, 0xf8,
0xfb, 0x10, 0xf6, 0xf2, 0xff, 0xfd, 0x00, 0x09, 0xfe, 0x03, 0xf4, 0x07,
0x10, 0xfb, 0xf6, 0xf2, 0x0c, 0xed, 0x0d, 0xf3, 0x0e, 0xf5, 0x0d, 0x0e,
0xfc, 0x00, 0x0f, 0x09, 0xed, 0xf0, 0xfe, 0x08, 0xed, 0x07, 0x09, 0xf9,
0xf5, 0x0c, 0xf4, 0xfc, 0x0a, 0x0e, 0xf7, 0xff, 0xf9, 0x0a, 0xf3, 0xf1,
0x0d, 0x00, 0x00, 0xf8, 0xf2, 0x06, 0xf3, 0x00, 0x01, 0x07, 0xf9, 0x06,
0xff, 0xfd, 0xf7, 0x01, 0xf3, 0x0c, 0xfe, 0x10, 0x01, 0x0a, 0x0c, 0x03,
0x08, 0xfd, 0x09, 0xf6, 0xfb, 0x04, 0x00, 0xfc, 0xfd, 0xf7, 0xf8, 0x01,
0x02, 0xf4, 0x01, 0x0c, 0x0d, 0xf1, 0xfc, 0x0d, 0xf9, 0x0a, 0x00, 0x09,
0x08, 0xfa, 0xf6, 0xff, 0xf7, 0x0e, 0xfd, 0x02, 0xf5, 0x09, 0x09, 0xf7,
0x04, 0x00, 0x07, 0xf6, 0xf8, 0xf1, 0xfd, 0xf6, 0xf8, 0xf9, 0x0c, 0xf4,
0x04, 0x02, 0xfc, 0xfb, 0x09, 0xf6, 0x0d, 0xfa, 0xfc, 0xfe, 0xf4, 0x04,
0x07, 0x07, 0x0e, 0x08, 0xef, 0xee, 0xf5, 0xfd, 0xf5, 0x0d, 0x0c, 0xe0,
0xfa, 0x03, 0x0d, 0xf2, 0x03, 0xf6, 0xf5, 0x05, 0x10, 0xf7, 0x05, 0x0c,
0x03, 0x00, 0x04, 0xf3, 0xed, 0xea, 0xf8, 0xfb, 0x05, 0xf2, 0xf2, 0xfc,
0xf0, 0x06, 0x06, 0x01, 0xf4, 0xf9, 0x02, 0xee, 0xfd, 0x08, 0x03, 0x0d,
0xf3, 0xfe, 0xf3, 0x01, 0xf2, 0xee, 0x03, 0x09, 0x02, 0xf8, 0x09, 0xfc,
0x03, 0x09, 0x0d, 0xf6, 0x0d, 0xf3, 0xf4, 0x0b, 0x01, 0xfc, 0x0f, 0x11,
0x08, 0x0a, 0xfe, 0xf3, 0x0d, 0xf0, 0x10, 0xfd, 0x0e, 0x0e, 0x02, 0x07,
0xff, 0xfc, 0xf8, 0x02, 0xed, 0x04, 0x0a, 0x0c, 0xe8, 0x08, 0xfe, 0xfd,
0xf6, 0x0d, 0x09, 0x06, 0x0d, 0x00, 0x07, 0x12, 0x00, 0xf1, 0x0f, 0xed,
0x0c, 0x05, 0xfc, 0xfd, 0x00, 0x07, 0xf8, 0x06, 0x02, 0xf3, 0xfc, 0x0a,
0xf9, 0x04, 0x01, 0xf8, 0xfd, 0x05, 0x00, 0x0f, 0x02, 0xfa, 0xf2, 0xf9,
0xf4, 0x09, 0x0a, 0x07, 0x03, 0x0c, 0xf1, 0xf8, 0x0c, 0x0d, 0xf1, 0xf6,
0x09, 0xf1, 0x08, 0xf7, 0x07, 0xf2, 0x04, 0xf5, 0xfc, 0xff, 0x04, 0x01,
0x08, 0xf3, 0x0a, 0xf9, 0xfa, 0x08, 0xff, 0xf2, 0xf7, 0x0d, 0x02, 0xf5,
0x05, 0xfb, 0xfd, 0x0b, 0xf6, 0xfd, 0x0f, 0x0d, 0x05, 0x09, 0xf2, 0xfc,
0xfa, 0x01, 0x01, 0x0d, 0xfa, 0xf3, 0xf5, 0x0c, 0xf9, 0x01, 0xf9, 0x0b,
0x0b, 0x0f, 0x03, 0x09, 0xee, 0x03, 0xf0, 0x09, 0x05, 0xef, 0x0c, 0x02,
0x0a, 0xf2, 0x04, 0x03, 0xfe, 0x01, 0xf6, 0x08, 0x0e, 0xf4, 0xf2, 0xf9,
0xf8, 0x05, 0x0a, 0xf2, 0x00, 0x08, 0xf4, 0xf4, 0x0a, 0x01, 0xf6, 0xf3,
0x02, 0xfc, 0x09, 0x04, 0xf9, 0x06, 0x0a, 0xf9, 0xfd, 0x0d, 0xef, 0xf1,
0x08, 0x02, 0x05, 0x06, 0xfc, 0x0f, 0xf9, 0x0f, 0xff, 0x0d, 0xfb, 0x0e,
0xfd, 0xf9, 0x0c, 0xf0, 0xf1, 0x0b, 0x0f, 0xff, 0x04, 0x00, 0xf6, 0xf2,
0xf7, 0xfe, 0xf4, 0x0d, 0x02, 0xf5, 0x0b, 0xfd, 0xfa, 0xf0, 0xf8, 0x0f,
0xf8, 0x07, 0xf1, 0xf1, 0x08, 0xf5, 0xff, 0x05, 0xf6, 0x02, 0x09, 0xf5,
0xf4, 0xf6, 0xf7, 0x0f, 0x09, 0x03, 0xf3, 0x07, 0x07, 0x02, 0x0b, 0xf0,
0x09, 0xf6, 0xfe, 0xf4, 0xfe, 0x0f, 0x00, 0x01, 0xff, 0x0c, 0x02, 0xfc,
0x0d, 0xf5, 0xf2, 0x0c, 0x0f, 0x06, 0x0b, 0xf1, 0xf8, 0x0b, 0x0f, 0xf4,
0xf7, 0x07, 0xf7, 0x08, 0xfe, 0xf1, 0x08, 0xf8, 0x06, 0x05, 0x07, 0x04,
0xf0, 0x07, 0x04, 0x0d, 0x00, 0xff, 0x0d, 0xf1, 0xf9, 0xf9, 0xfe, 0x0c,
0xfa, 0xfc, 0xfc, 0x07, 0x10, 0x05, 0x0f, 0x0c, 0xf3, 0xf1, 0xff, 0x0d,
0x05, 0xf4, 0x04, 0xfc, 0xfa, 0xf4, 0xfb, 0xf7, 0xf5, 0x03, 0x0b, 0x09,
0xf3, 0xfb, 0xf9, 0xf5, 0x03, 0x02, 0xf1, 0x09, 0xed, 0xf4, 0xe9, 0xf7,
0x03, 0x07, 0xfd, 0xec, 0x01, 0xca, 0x02, 0x04, 0x0b, 0xed, 0xf1, 0xe9,
0x0a, 0xad, 0x03, 0xf4, 0x07, 0x10, 0xf6, 0x0d, 0xff, 0xff, 0x0e, 0xfd,
0xc2, 0xe6, 0xfe, 0x02, 0xbe, 0x01, 0xfe, 0xfa, 0x00, 0x0f, 0x0a, 0xe9,
```

```
0x07, 0xf2, 0xf8, 0xfd, 0xea, 0x05, 0xf7, 0xae, 0xfb, 0xf8, 0xfc, 0x0e,
0x0f, 0xff, 0x1c, 0xfe, 0xf6, 0x0c, 0x07, 0xf1, 0xf5, 0x10, 0xfd, 0x10,
0xf2, 0x0f, 0x0b, 0x07, 0x00, 0x0f, 0xf8, 0x0b, 0x11, 0xeb, 0x0c, 0xfa,
0x13, 0xfe, 0x0b, 0xfc, 0xf4, 0xf4, 0xf6, 0x0c, 0xfb, 0xf3, 0x05, 0xf7,
0xf9, 0x15, 0xfd, 0xfc, 0xf5, 0x04, 0xf9, 0xf4, 0x02, 0x0d, 0x05, 0x02,
0x09, 0x01, 0x0f, 0xfd, 0x0a, 0xf7, 0x0e, 0xfb, 0x11, 0x06, 0xff, 0xf5,
0x09, 0x09, 0xf7, 0xfc, 0x01, 0xfd, 0xf5, 0xf6, 0xee, 0x03, 0xf1, 0x07,
0xf7, 0xf8, 0x05, 0xfc, 0xeb, 0xf9, 0xfc, 0x07, 0x12, 0x0b, 0x03, 0x02,
0xf3, 0xee, 0xf9, 0x14, 0x15, 0x10, 0x0a, 0xdf, 0xff, 0xe2, 0x0e, 0x08,
0x10, 0xf3, 0xfe, 0x08, 0xf9, 0x13, 0xfd, 0x01, 0xd7, 0x09, 0x0d, 0xfd,
0xcc, 0xe4, 0x04, 0x03, 0xf5, 0x00, 0xf0, 0xe5, 0x09, 0x0f, 0x05, 0x20,
0xf6, 0x03, 0x07, 0xdd, 0xf2, 0xf8, 0x01, 0xfe, 0xfb, 0xfc, 0x27, 0xfb,
0x01, 0xfb, 0x08, 0xfa, 0xff, 0x0b, 0x10, 0x07, 0x06, 0xf3, 0x06, 0xfb,
0x0f, 0xf8, 0x05, 0x01, 0xf7, 0xf5, 0x01, 0x0a, 0x0f, 0x04, 0x0e, 0x03,
0x05, 0x01, 0x0e, 0x06, 0xf0, 0x0b, 0xf6, 0x08, 0xf9, 0x01, 0x0b, 0x09,
0x02, 0x0f, 0x08, 0xf3, 0xe8, 0x00, 0xf9, 0xfd, 0xf6, 0x01, 0xf1, 0xf7,
0xfb, 0x06, 0xf8, 0xfa, 0xf3, 0xfc, 0xf6, 0xf8, 0x0b, 0xff, 0xf8, 0x0b,
0x01, 0x11, 0xeb, 0x0b, 0xf8, 0x07, 0x10, 0x0c, 0xf2, 0x05, 0x02, 0x13,
0xf7, 0x12, 0x09, 0x0b, 0x16, 0xf7, 0x05, 0x01, 0xf7, 0xea, 0xfa, 0x14,
0x14, 0x07, 0x03, 0x04, 0x00, 0xe6, 0xfe, 0xf4, 0x04, 0x10, 0xf9, 0xf8,
0xf8, 0x16, 0x0b, 0x06, 0xe8, 0x0d, 0xf0, 0x0e, 0xfd, 0xfe, 0x0f, 0x07,
0xf3, 0x08, 0xf2, 0xee, 0x12, 0xff, 0x10, 0x03, 0xf7, 0x0b, 0xf2, 0xe0,
0xfc, 0x0f, 0x0c, 0x0c, 0xf9, 0xf6, 0x0d, 0x08, 0x02, 0x00, 0x08, 0x00,
0xf1, 0xf6, 0x0f, 0xff, 0xf7, 0xef, 0xfb, 0x0b, 0xf9, 0x0b, 0xf9, 0x06,
0xf4, 0x04, 0xff, 0x0f, 0xfb, 0x0b, 0x10, 0xfe, 0x0b, 0xfd, 0xf9, 0xf1,
0x03, 0xf8, 0xfa, 0xfe, 0x0f, 0xf7, 0x06, 0x05, 0xef, 0xfa, 0xf6, 0xfd,
0x09, 0xf2, 0xfe, 0xf2, 0xfc, 0xf4, 0xfd, 0xf7, 0xfd, 0x0d, 0xfa, 0x12,
0xfe, 0x03, 0x05, 0xf3, 0x02, 0xf1, 0xfb, 0x0e, 0xf2, 0xf4, 0x0b, 0x04,
0x08, 0xf5, 0x0d, 0xfe, 0xf8, 0xf8, 0x05, 0xf7, 0xfc, 0x0a, 0x0b, 0x10,
0xf8, 0x0a, 0x0a, 0x08, 0x03, 0xf2, 0x03, 0x14, 0x13, 0x13, 0xf0, 0xf8,
0xfa, 0x00, 0x15, 0xf6, 0x05, 0xf5, 0xf7, 0x08, 0x06, 0xfb, 0xfd, 0xfe,
0xf8, 0xf1, 0xf8, 0x0c, 0xed, 0xef, 0xf7, 0x0e, 0xf7, 0x02, 0x07, 0xfa,
0x14, 0x10, 0x11, 0x03, 0x08, 0x06, 0x0e, 0xf6, 0xf7, 0x00, 0x0a, 0x03,
0x08, 0x05, 0xee, 0x0c, 0xfc, 0xf7, 0x12, 0xf8, 0xfe, 0xf7, 0x0b, 0x12,
0xff, 0x0f, 0xfb, 0x0e, 0x04, 0xfb, 0xf1, 0xf3, 0x0b, 0xf8, 0xfb, 0x13,
0x09, 0x12, 0x0d, 0xf8, 0xfc, 0xf2, 0x0b, 0x0a, 0xfa, 0x0c, 0xf6, 0xfe,
0x0b, 0xf8, 0xf2, 0xfb, 0xee, 0x0a, 0xf8, 0x02, 0xff, 0x01, 0xfc, 0xf0,
0x09, 0xf2, 0x08, 0xf3, 0x11, 0xfd, 0xf6, 0x03, 0x05, 0x05, 0xf6, 0xf0,
0xfd, 0x0b, 0x10, 0x09, 0x05, 0xfe, 0xfc, 0x00, 0xfa, 0x04, 0xfd, 0x07,
0xf0, 0xf6, 0xf5, 0xf4, 0x01, 0x0f, 0xf6, 0x08, 0x0f, 0xff, 0x0f, 0x05,
0xfe, 0xf2, 0x0f, 0x02, 0xf8, 0x0e, 0xfb, 0xf5, 0xf4, 0xf5, 0x02, 0xf2,
0x0e, 0x09, 0xf8, 0x01, 0xfa, 0x0d, 0x02, 0xf2, 0xf7, 0xf2, 0x04, 0x07,
0x09, 0x03, 0xf5, 0x0d, 0xfe, 0xfb, 0xfe, 0x10, 0xf1, 0x0e, 0xf4, 0x09,
0xfc, 0xf5, 0xfd, 0xf8, 0xfa, 0x06, 0x07, 0x02, 0x03, 0xf5, 0xfd, 0xf1,
0xf6, 0x09, 0xf7, 0xf8, 0xf3, 0xf4, 0xf3, 0x0c, 0x05, 0xfb, 0xf8, 0x01,
0xfe, 0xf7, 0xfe, 0x02, 0xf6, 0xf8, 0x0c, 0xff, 0x05, 0x0a, 0x0c, 0xf2,
0x0a, 0xf1, 0xfa, 0x08, 0x0c, 0x0c, 0xf5, 0xfc, 0xf3, 0x0e, 0x09, 0xf1,
0x0b, 0xff, 0xf7, 0xf9, 0x05, 0x04, 0x0e, 0xf0, 0x0f, 0x0f, 0xfe, 0x09,
0x0d, 0x03, 0xf8, 0x05, 0x0a, 0xfa, 0xf9, 0xfe, 0xfa, 0x0b, 0xfe, 0xfd,
0xf2, 0xfa, 0x00, 0x0c, 0x0b, 0x01, 0x03, 0xf4, 0xf2, 0x05, 0x06, 0x0e,
```

```
0x06, 0x09, 0x03, 0xff, 0xfe, 0x03, 0x05, 0xfb, 0xf6, 0xee, 0x09, 0xf9,
0x02, 0xff, 0x0b, 0x01, 0x00, 0x02, 0x0e, 0xf4, 0xfb, 0x04, 0x02, 0xfb,
0xf9, 0x0e, 0x10, 0x12, 0x11, 0x03, 0xf6, 0xfe, 0x00, 0x0f, 0xf1, 0x0e,
0x00, 0x02, 0x04, 0x02, 0xf3, 0x07, 0xfb, 0xfe, 0xf9, 0xf4, 0xfc, 0x08,
0xff, 0x0b, 0x10, 0x03, 0xf2, 0xfe, 0xee, 0x05, 0xe6, 0xf3, 0xf4, 0x0b,
0xf0, 0x00, 0xff, 0x05, 0xee, 0x00, 0xf9, 0x09, 0xf5, 0x0e, 0xf5, 0xe7,
0x07, 0xb4, 0xf7, 0x0b, 0x04, 0x06, 0x05, 0xbd, 0xf7, 0x8f, 0xf6, 0xf3,
0xf8, 0x0a, 0x05, 0x08, 0x07, 0x06, 0xf5, 0xfd, 0x81, 0xe7, 0xf2, 0xfb,
0x96, 0xeb, 0x0d, 0xff, 0x01, 0x04, 0xf2, 0xe5, 0xf4, 0x03, 0xf6, 0x02,
0x00, 0xf1, 0x0b, 0x8b, 0xea, 0xf1, 0xfd, 0xf1, 0x0f, 0x05, 0x14, 0xf7,
0xf6, 0x0e, 0xf6, 0xfd, 0x03, 0x04, 0x0c, 0xf2, 0xfc, 0x07, 0x07, 0xf1,
0xf2, 0x0b, 0xfe, 0x08, 0x0b, 0xf0, 0x0b, 0xf3, 0xfe, 0x05, 0xf8, 0x0c,
0x0f, 0x02, 0x03, 0x0b, 0x02, 0x08, 0x08, 0xf7, 0x00, 0x04, 0xfa, 0xf5,
0x0e, 0x03, 0x06, 0xf9, 0x06, 0xf4, 0xf4, 0x0e, 0xf1, 0xff, 0x00, 0x01,
0x09, 0x0e, 0xf8, 0x09, 0x07, 0x10, 0x03, 0xf3, 0x0d, 0xf7, 0xfe, 0x08,
0x0e, 0xf6, 0xfb, 0x0b, 0xf8, 0xfa, 0x06, 0xf1, 0xf2, 0xfb, 0x15, 0x0d,
0xed, 0xfb, 0x03, 0xfd, 0x12, 0x0c, 0x0d, 0x08, 0x12, 0xf5, 0x03, 0x15,
0xfc, 0x08, 0xf7, 0xf2, 0x0a, 0xed, 0x0c, 0x00, 0xff, 0x0e, 0x09, 0x06,
0xfc, 0x04, 0x04, 0x00, 0xe9, 0x04, 0xf6, 0x0b, 0xe9, 0xf4, 0xf8, 0x0b,
0xf1, 0x02, 0x04, 0xf2, 0x15, 0x0e, 0x0f, 0xff, 0xfc, 0xf8, 0x0e, 0xf0,
0x04, 0x0a, 0xf8, 0xf8, 0x0e, 0x10, 0xf5, 0x0f, 0xf8, 0x0f, 0x0b, 0x0c,
0x0d, 0xf5, 0xf2, 0x0c, 0xfb, 0x05, 0x08, 0xf4, 0xf3, 0xfb, 0xfe, 0x05,
0xfb, 0xfd, 0xf8, 0x09, 0x0e, 0xf8, 0x0a, 0xfe, 0x07, 0xf1, 0xfe, 0x0a,
0xf2, 0x0c, 0x0e, 0x00, 0xfc, 0xf5, 0x09, 0xf8, 0x04, 0xfa, 0xf3, 0x0c,
0x09, 0xf4, 0xfd, 0x08, 0x0f, 0xfd, 0x0c, 0x05, 0xf7, 0xf4, 0x0e, 0xf8,
0xf1, 0x0e, 0x05, 0xfb, 0x0b, 0xf9, 0xfc, 0xf7, 0x02, 0x02, 0x09, 0x09,
0xf4, 0xf4, 0xf1, 0xff, 0xf3, 0x0f, 0xf1, 0xfb, 0xf4, 0xff, 0xfb, 0xf1,
0xf9, 0x0c, 0xfe, 0xfd, 0xf7, 0xf5, 0x04, 0xf1, 0xfe, 0xf7, 0xf6, 0x0f,
0x0a, 0x00, 0xfe, 0x04, 0x0c, 0xf1, 0x0e, 0xfe, 0x0a, 0xf2, 0xf9, 0x0e,
0xfb, 0xf0, 0x02, 0x09, 0x04, 0xf8, 0xfa, 0x0a, 0x01, 0x0f, 0xf0, 0xf6,
0xfb, 0x05, 0xf9, 0x08, 0x0d, 0x0d, 0x0b, 0x01, 0x0b, 0xf6, 0xfe, 0xfc,
0x0b, 0xff, 0x0e, 0x09, 0x09, 0x00, 0xf9, 0xfc, 0xf7, 0xf7, 0x0d, 0x0e,
0xf6, 0xef, 0x03, 0xfc, 0xf3, 0xf1, 0xfd, 0x0e, 0x0b, 0xf5, 0xf2, 0x03,
0xf7, 0x06, 0x07, 0xf8, 0x04, 0xf4, 0xee, 0x01, 0xf5, 0x09, 0xf7, 0xf3,
0x08, 0xf9, 0x0d, 0x0a, 0xff, 0xff, 0xfc, 0xf9, 0x0a, 0xf8, 0x0d, 0xf3,
0x0a, 0x0e, 0xf4, 0xf8, 0xf6, 0x08, 0x08, 0xf8, 0x07, 0x03, 0x03, 0xfa,
0x0a, 0x08, 0x06, 0xf4, 0xf9, 0x04, 0xf6, 0xff, 0x0a, 0x04, 0xfa, 0x04,
0xfa, 0xf0, 0xff, 0xf8, 0xf9, 0xf9, 0xf8, 0xf6, 0x06, 0xf6, 0x0c, 0x00,
0xf7, 0xf7, 0x0f, 0xf1, 0x07, 0xf5, 0xfe, 0x0e, 0x04, 0x0f, 0xfe, 0xfd,
0xf2, 0xf1, 0xf9, 0xf6, 0xf7, 0xf7, 0xfe, 0xf5, 0xf4, 0x09, 0x00, 0x02,
0xf5, 0x03, 0xf3, 0xfb, 0x0c, 0xf5, 0xf7, 0xf4, 0x03, 0xf7, 0x02, 0x0e,
0xf5, 0x05, 0x02, 0x09, 0x10, 0x00, 0x03, 0xf1, 0x09, 0x02, 0x07, 0xf3,
0x09, 0x01, 0x0f, 0x05, 0xf1, 0x04, 0x0b, 0x06, 0xf4, 0xfd, 0xfe, 0xf6,
0x00, 0xf5, 0xf7, 0x05, 0x13, 0x05, 0xfa, 0xf7, 0xfc, 0x02, 0xf2, 0xfe,
0x0b, 0xe7, 0x06, 0x0c, 0x07, 0x0d, 0xf5, 0x01, 0x04, 0x0f, 0xfe, 0x01,
0xe4, 0x06, 0x0f, 0x06, 0xe8, 0x01, 0x04, 0x0d, 0xfc, 0x0a, 0xf2, 0xf3,
0x11, 0xf3, 0x0a, 0x0b, 0x0d, 0xf3, 0x0f, 0xfe, 0x0b, 0xfb, 0xf0, 0x00,
0xff, 0x06, 0x02, 0xf3, 0x09, 0x09, 0xfe, 0xf9, 0xfb, 0x09, 0x08, 0xff,
0xf6, 0x0d, 0xf0, 0xfa, 0x02, 0xf1, 0xf1, 0x06, 0xf3, 0x09, 0x0e, 0xf4,
0x00, 0x07, 0x00, 0xf8, 0xf1, 0x03, 0x07, 0x0c, 0x06, 0x05, 0x03, 0xf2,
```

```
0x0b, 0x0a, 0x04, 0xf0, 0x0b, 0x0a, 0x01, 0xfb, 0x0f, 0xf1, 0xf7, 0x08,
0x0f, 0x05, 0x00, 0xf2, 0xf8, 0x03, 0x01, 0xef, 0xfe, 0x0f, 0xf2, 0xfc,
0xfe, 0x01, 0x0d, 0x06, 0x04, 0xf9, 0xfd, 0x03, 0xf6, 0x0f, 0xf3, 0xf7,
0x0f, 0xf5, 0xff, 0x03, 0x0e, 0xf4, 0xfb, 0xfe, 0xf0, 0x07, 0xfa, 0x04,
0xf2, 0xf9, 0xf7, 0xff, 0xfd, 0x0d, 0xf5, 0xf6, 0x08, 0x0a, 0xf9, 0x05,
0xfe, 0xfe, 0x05, 0x07, 0xf7, 0x09, 0x03, 0xf4, 0x05, 0x06, 0x05, 0x05,
0xff, 0xf1, 0x09, 0x08, 0x07, 0x0a, 0x06, 0x05, 0xf9, 0xf2, 0x01, 0x0e,
0xf5, 0x00, 0xfe, 0xff, 0x05, 0xff, 0xf5, 0xf7, 0xff, 0x0f, 0x0c, 0xf0,
0xf2, 0x07, 0xee, 0xfa, 0xf4, 0x09, 0x09, 0x09, 0x06, 0xef, 0x0b, 0x0d,
0x04, 0xf5, 0x04, 0x0b, 0xf1, 0xf3, 0x0c, 0xfd, 0x07, 0x03, 0xf4, 0xf3,
0xf6, 0xf9, 0xfb, 0xf6, 0x0f, 0xfb, 0x06, 0x03, 0x07, 0xf5, 0xfb, 0x08,
0xf7, 0xf1, 0x03, 0xf9, 0xf5, 0xf8, 0x04, 0x10, 0x10, 0xf5, 0x0f, 0x09,
0xed, 0x09, 0xef, 0x07, 0xff, 0xfa, 0x09, 0x0b, 0xfb, 0xf4, 0xf6, 0xfa,
0x0c, 0x07, 0x07, 0x01, 0x0d, 0x07, 0xf8, 0x0e, 0x0a, 0xf3, 0xfb, 0xff,
0x0a, 0x0a, 0xfc, 0x05, 0x0e, 0x06, 0xf4, 0x09, 0x0a, 0x00, 0x0b, 0x01,
0xfe, 0x0c, 0xf7, 0x10, 0x06, 0xf1, 0xf5, 0x07, 0xfe, 0x03, 0x0e, 0xfb,
0xf5, 0xf5, 0x09, 0xf6, 0x0e, 0xfa, 0x03, 0x02, 0x01, 0xf0, 0x06, 0x0c,
0xf9, 0xfd, 0xf0, 0xf8, 0xfb, 0xfd, 0xf8, 0x01, 0xfb, 0x0b, 0x0a, 0x09,
0xf4, 0x0b, 0xf0, 0x03, 0xf9, 0x01, 0x0f, 0xff, 0x0a, 0x03, 0xf3, 0xfa,
0x07, 0xfd, 0xf4, 0x0f, 0xf5, 0xf2, 0xf7, 0x07, 0xf8, 0xfb, 0xf3, 0xfd,
0xfd, 0x07, 0x10, 0xf2, 0x01, 0x07, 0x02, 0xf9, 0x05, 0x07, 0xfe, 0x03,
0xf9, 0xf4, 0x09, 0x0f, 0xf4, 0xf8, 0x07, 0xf9, 0xf2, 0xf7, 0xf2, 0x03,
0xf1, 0xf3, 0xfc, 0xf8, 0xf3, 0x09, 0xf2, 0xf9, 0xf7, 0x0d, 0x06, 0xfa,
0x0e, 0xfc, 0xf2, 0x10, 0xf5, 0x0e, 0x07, 0xfe, 0x03, 0xfd, 0xf2, 0xfd,
0x05, 0xf8, 0xfb, 0xff, 0xfb, 0x00, 0x09, 0xf2, 0xf9, 0xf7, 0xf2, 0x10,
0xf8, 0x0d, 0x0a, 0x0e, 0xf1, 0xfc, 0xfd, 0x01, 0x04, 0xf8, 0xf9, 0x0a,
0xf3, 0xfc, 0xf2, 0x0a, 0x0e, 0xfe, 0x0a, 0xf3, 0x01, 0x00, 0xf9, 0x04,
0xfb, 0x07, 0xfc, 0xf6, 0xf0, 0xf1, 0x07, 0xf8, 0xf5, 0x0c, 0x05, 0xf7,
0xf7, 0xf6, 0xf1, 0x0c, 0x03, 0xfa, 0x0f, 0xf7, 0x0d, 0x01, 0x0f, 0xf1,
0xf8, 0xf9, 0x0d, 0xf1, 0xfa, 0xfa, 0xf6, 0xf3, 0x04, 0x07, 0x00, 0xfa,
0xf6, 0x03, 0x0b, 0xfe, 0xfa, 0xf1, 0x05, 0x01, 0xf4, 0xfe, 0x07, 0xf9,
0x02, 0xfe, 0xf7, 0xfe, 0xfc, 0x01, 0x00, 0x06, 0xfb, 0x03, 0xfb, 0x07,
0x05, 0x08, 0x03, 0xf4, 0xf7, 0xf7, 0x0d, 0xf3, 0xf3, 0xf4, 0xf9, 0xf5,
0x03, 0xf0, 0x0c, 0xfc, 0x01, 0xfa, 0xf3, 0x10, 0xfa, 0x0b, 0xf3, 0x07,
0xf6, 0x00, 0x08, 0xf5, 0x02, 0x0b, 0xfa, 0xf7, 0xf5, 0x0c, 0x09, 0xfb,
0x08, 0xff, 0xf1, 0x07, 0x06, 0xfc, 0x09, 0x08, 0x07, 0xfe, 0xfe, 0xf3,
0xf0, 0x00, 0xf1, 0x10, 0xf5, 0xf2, 0xfd, 0xfc, 0xff, 0xf9, 0xf5, 0x0a,
0x0f, 0xf6, 0x07, 0x0c, 0xf2, 0xf6, 0x08, 0xf8, 0x0a, 0x0f, 0x0e, 0xf7,
0x08, 0xf3, 0x0e, 0xf7, 0x01, 0x0d, 0x04, 0xf7, 0xf0, 0x03, 0xf2, 0x00,
0x07, 0x0c, 0xf3, 0x0b, 0xfb, 0xfb, 0x0c, 0xfb, 0xfe, 0xf0, 0x06, 0x05,
0xf3, 0x06, 0xf3, 0x05, 0x05, 0xf2, 0x06, 0x02, 0xfb, 0xfa, 0x03, 0x0f,
0xf7, 0xf1, 0xf9, 0x10, 0xf6, 0x0d, 0x04, 0x05, 0xf3, 0x07, 0x05, 0x00,
0x07, 0xff, 0xf9, 0xf1, 0x02, 0x05, 0x05, 0x0f, 0x03, 0xf4, 0x09, 0xff,
0x07, 0x07, 0x0d, 0xff, 0x0b, 0x00, 0xf9, 0xf9, 0x08, 0x0d, 0x06, 0xfb,
0xf8, 0x06, 0xf4, 0xf7, 0x06, 0xf8, 0x01, 0xff, 0xf3, 0xf6, 0x03, 0xf6,
0x06, 0x09, 0x08, 0x0b, 0x05, 0x02, 0xf9, 0xf4, 0xfb, 0xf8, 0x0e, 0x03,
0xf0, 0x0e, 0x0e, 0xfa, 0x00, 0x05, 0x0f, 0xf9, 0x12, 0xf0, 0xff, 0x10,
0x12, 0x13, 0x03, 0xe4, 0x0c, 0xed, 0xff, 0x08, 0x05, 0xfa, 0xf7, 0xfd,
0x00, 0x0d, 0x05, 0xf5, 0xde, 0xf0, 0xfe, 0xfd, 0xec, 0xfe, 0xfb, 0xf7,
0xf8, 0xf9, 0xf0, 0xfd, 0xfd, 0xfc, 0x0a, 0xf7, 0x05, 0xff, 0xf8, 0xf6,
```

```
0x11, 0x04, 0xf3, 0x08, 0xf6, 0x0a, 0xfd, 0xf8, 0xf7, 0xfd, 0xed, 0x05,
0x07, 0x03, 0xff, 0x0b, 0xf1, 0xf7, 0x07, 0x0d, 0xed, 0x09, 0x0e, 0x05,
0xf2, 0x08, 0xf3, 0xf5, 0x0a, 0x03, 0x09, 0xfc, 0xf5, 0xff, 0x0b, 0x05,
0xf6, 0x03, 0xfc, 0xfb, 0xf3, 0x0a, 0xf7, 0xf0, 0x01, 0x0c, 0x0a, 0xf4,
0x0c, 0xf0, 0xf4, 0x03, 0x03, 0x08, 0x06, 0x0b, 0xf1, 0xf2, 0x04, 0xfd,
0xf2, 0x0a, 0xf7, 0x08, 0xff, 0xf8, 0x04, 0x09, 0xfb, 0xf4, 0x0b, 0xf2,
0x07, 0xfb, 0xf1, 0x0c, 0xff, 0xfa, 0x01, 0xf0, 0x06, 0x00, 0xff, 0xf0,
0xfd, 0xfe, 0x01, 0xf8, 0x00, 0x05, 0xfb, 0xf7, 0xfd, 0x0d, 0x0d, 0x02,
0x0b, 0x03, 0x02, 0x0c, 0x0a, 0xf8, 0x01, 0x06, 0x0f, 0x07, 0x0f, 0xf4,
0xf8, 0x09, 0x01, 0xf4, 0xf8, 0xfd, 0x04, 0x08, 0x06, 0x0b, 0xf1, 0x06,
0xfa, 0xf4, 0x01, 0xf9, 0xf6, 0x06, 0xfa, 0x05, 0x07, 0x00, 0x06, 0x10,
0x00, 0xf2, 0x07, 0xfb, 0xfa, 0x03, 0x07, 0xfd, 0xf1, 0x0f, 0xf0, 0x01,
0x02, 0xf8, 0xf2, 0x06, 0x01, 0x09, 0xfa, 0xfa, 0xf1, 0xf9, 0xfe, 0xf9,
0x08, 0x01, 0x06, 0x0c, 0x0f, 0xfe, 0xfc, 0x02, 0xfa, 0xf8, 0xf1, 0xfc,
0x01, 0x07, 0xf8, 0xf7, 0xf1, 0x0a, 0x04, 0xfc, 0x03, 0x0f, 0xfb, 0xf6,
0xff, 0xf6, 0xf1, 0x02, 0xfa, 0x05, 0x0f, 0xf2, 0x0f, 0xfe, 0xfd, 0x04,
0x0f, 0xfa, 0x0f, 0xfd, 0x00, 0xf1, 0xff, 0xfc, 0x09, 0xfd, 0xf1, 0x06,
0x02, 0x0c, 0xf4, 0xf1, 0x08, 0x0e, 0xf1, 0x0e, 0x0b, 0x0d, 0xf8, 0xf6,
0x00, 0x09, 0x05, 0x02, 0xf5, 0xf2, 0xf3, 0x0e, 0x0f, 0xf3, 0xf7, 0x0a,
0xf3, 0xfa, 0x0a, 0xff, 0x0a, 0x0e, 0xff, 0xf3, 0xfd, 0x0f, 0xfc, 0x0c,
0xf4, 0x03, 0x04, 0x08, 0xf0, 0xfd, 0xf3, 0xfe, 0xf9, 0x10, 0x03, 0xf4,
0xf9, 0x02, 0x06, 0x09, 0xf3, 0xf3, 0xf6, 0x03, 0x04, 0xf2, 0x09, 0x09,
0xfe, 0x05, 0xf3, 0x08, 0x03, 0xf3, 0xf4, 0xf3, 0xf4, 0xfb, 0x09, 0x0b,
0x0b, 0xf5, 0xf9, 0xf3, 0x05, 0x08, 0x02, 0xf7, 0x03, 0xf2, 0x0c, 0xf3,
0x0d, 0xf3, 0xf6, 0xf6, 0xf5, 0xf0, 0xf9, 0x0a, 0x00, 0x0e, 0x03, 0xf4,
0xf6, 0xf0, 0xf7, 0x0a, 0xf8, 0xf8, 0xf8, 0x0b, 0xfa, 0xf9, 0x07, 0x0e,
0x07, 0xfe, 0x0f, 0xf5, 0x0b, 0xfc, 0x04, 0x05, 0xfc, 0xf6, 0x0c, 0x02,
0x02, 0x0f, 0x0d, 0x05, 0x07, 0x08, 0x07, 0x10, 0x0c, 0xf2, 0x05, 0xf3,
0xfe, 0x03, 0xf5, 0x00, 0x08, 0xfb, 0xfc, 0x00, 0x0c, 0x09, 0x0d, 0xee,
0xfd, 0x04, 0x0b, 0x04, 0x00, 0x0b, 0xfe, 0x04, 0xf3, 0x09, 0x05, 0x06,
0xf0, 0x08, 0x0f, 0xfc, 0x0c, 0x01, 0xf3, 0xf5, 0x0b, 0x09, 0x01, 0x10,
0xf5, 0x0c, 0xf2, 0xf8, 0x01, 0x0e, 0xef, 0xf6, 0x04, 0xf0, 0x09, 0xf0,
0xf6, 0xf9, 0x07, 0x09, 0xf4, 0xf7, 0xfb, 0x0f, 0xf0, 0x0c, 0xfa, 0xf1,
0x02, 0xf1, 0x09, 0x03, 0xff, 0x0d, 0x04, 0xff, 0x03, 0xf8, 0x0c, 0x09,
0xfe, 0xf4, 0x01, 0x0c, 0xfe, 0x0d, 0xff, 0xf9, 0xf0, 0xfc, 0x0a, 0xfb,
0x01, 0x0f, 0x07, 0xf2, 0x05, 0xf8, 0xf5, 0x0d, 0xf4, 0xfd, 0xfa, 0xfa,
0xf2, 0xf9, 0xfa, 0x07, 0xf3, 0x00, 0xff, 0xfa, 0xf9, 0x07, 0xfe, 0x0b,
0x0c, 0x04, 0xf5, 0xed, 0xfe, 0x01, 0xf4, 0xf6, 0x0b, 0xf5, 0xe7, 0x06,
0xe6, 0xe6, 0xef, 0xf1, 0xfa, 0x02, 0x0c, 0xfb, 0xc6, 0x0c, 0xd9, 0x0c,
0x09, 0x00, 0x08, 0xdd, 0x0a, 0xee, 0x05, 0x08, 0x0b, 0x08, 0x0b, 0xf9,
0xfc, 0xdf, 0xf0, 0xfc, 0x0f, 0x0e, 0x05, 0xf6, 0x09, 0x03, 0x0e, 0xf5,
0xc1, 0xd1, 0x0f, 0xff, 0xdd, 0xf0, 0x02, 0x07, 0x0b, 0x02, 0xfa, 0x0c,
0x0f, 0x08, 0xfc, 0xf0, 0xf9, 0xfa, 0xfb, 0xdf, 0xf2, 0x0b, 0x05, 0x09,
0x03, 0xfa, 0xf4, 0xf0, 0xfa, 0x02, 0x0b, 0xf0, 0xfd, 0x08, 0x05, 0xee,
0x06, 0x02, 0x0a, 0xf7, 0xef, 0x09, 0xf8, 0xfc, 0x07, 0xf4, 0xfc, 0x02,
0x05, 0xfb, 0xff, 0xf8, 0x0f, 0x0a, 0xf3, 0x0c, 0x0b, 0x02, 0x08, 0xf9,
0xf0, 0x01, 0x0d, 0xff, 0xfc, 0xff, 0xff, 0x03, 0xf9, 0x03, 0xfc, 0x0f,
0x09, 0x05, 0x00, 0x0b, 0x03, 0x0d, 0x08, 0x07, 0x03, 0xf1, 0xfb, 0xf2,
0xf8, 0x09, 0x0d, 0x0a, 0x0f, 0xf3, 0x05, 0xf4, 0x05, 0x0e, 0x0f, 0x0a,
0xf5, 0x0a, 0xf5, 0x09, 0x0a, 0xfe, 0xf4, 0xfc, 0xfc, 0xf2, 0xf3, 0xf9,
```

```
0xf8, 0x07, 0x0d, 0x10, 0xf3, 0x07, 0xfc, 0x09, 0xf1, 0xf3, 0xf6, 0x0e,
0x06, 0xf6, 0x08, 0x02, 0x01, 0xf9, 0xf8, 0x0d, 0xf6, 0xf3, 0x0d, 0x01,
0xf2, 0xfe, 0xf1, 0xf9, 0xfd, 0xf3, 0x0d, 0xf1, 0xf1, 0x05, 0x07, 0xf5,
0x02, 0xf7, 0x0c, 0xf4, 0xf0, 0xf5, 0xfc, 0xf1, 0x06, 0x09, 0x00, 0xf8,
0xe6, 0x01, 0xfd, 0x08, 0xfc, 0xf4, 0x00, 0xf8, 0xf5, 0xec, 0xf0, 0xf2,
0xf9, 0xfc, 0xf3, 0xe8, 0x02, 0xcc, 0xff, 0x00, 0x05, 0x00, 0xf8, 0xdd,
0x01, 0xac, 0x06, 0x09, 0xf2, 0x0e, 0xf5, 0xf2, 0x05, 0xf9, 0x04, 0xff,
0xab, 0xee, 0x05, 0x07, 0xae, 0x06, 0x02, 0x08, 0xfa, 0xfc, 0x0b, 0xee,
0x09, 0x06, 0xed, 0x05, 0xff, 0x07, 0x06, 0xa5, 0xfd, 0xf8, 0xfb, 0xf5,
0x05, 0xf8, 0x26, 0xfb, 0x03, 0xf4, 0x0e, 0x06, 0xfb, 0xf9, 0x08, 0xfe,
0xee, 0x04, 0xf4, 0x04, 0x0e, 0xf3, 0x07, 0xf9, 0x11, 0xed, 0x01, 0x0c,
0x12, 0x05, 0x0c, 0xfb, 0xff, 0xf9, 0x10, 0x03, 0x09, 0x00, 0x0c, 0x03,
0x0e, 0x07, 0x07, 0x02, 0xf8, 0x00, 0x0e, 0xff, 0xeb, 0x07, 0xfd, 0x0c,
0x09, 0x0d, 0xf1, 0xfd, 0x06, 0xfd, 0x08, 0x0b, 0xfd, 0x10, 0xf1, 0xf2,
0x0d, 0xfa, 0x01, 0x06, 0xfb, 0x0d, 0xee, 0x05, 0xfa, 0x0f, 0x0f, 0x08,
0x0d, 0xfe, 0xff, 0xff, 0xfc, 0xfb, 0x05, 0xf2, 0xfd, 0xfd, 0x06, 0xfe,
0xf0, 0xfd, 0xf4, 0x00, 0x0e, 0xf4, 0x05, 0xf7, 0xf4, 0x03, 0x00, 0xf2,
0x00, 0xff, 0x0f, 0x00, 0x0b, 0xf2, 0xf4, 0xf1, 0xff, 0xfc, 0x00, 0xf6,
0xf5, 0xf8, 0xfd, 0xf5, 0xf8, 0x0d, 0xf3, 0xfb, 0x0f, 0x02, 0xfa, 0xf7,
0xfe, 0x0f, 0xfe, 0xfe, 0x0f, 0x06, 0x0a, 0xf9, 0xff, 0xf3, 0x0e, 0x05,
0x15, 0x15, 0x03, 0x02, 0x03, 0x0f, 0x03, 0x0e, 0x1d, 0xf8, 0x09, 0x0f,
0xff, 0xff, 0xfc, 0x08, 0x0c, 0x22, 0xf9, 0xf2, 0xfc, 0x00, 0xfd, 0x13,
0xf2, 0x1a, 0x0d, 0x03, 0xfa, 0xf0, 0x09, 0xfb, 0xf7, 0xf9, 0xfb, 0x0f,
0x2e, 0x1a, 0x09, 0x06, 0x22, 0x00, 0x0a, 0xff, 0xf1, 0x0d, 0x00, 0x0e,
0x0f, 0x0e, 0x0a, 0xfd, 0x03, 0xf7, 0xf5, 0x19, 0x0f, 0x07, 0x1a, 0xff,
0xf7, 0x00, 0xe2, 0x0f, 0x04, 0x03, 0xf1, 0xf9, 0x0a, 0xf2, 0xf5, 0x01,
0xf3, 0x0b, 0xf5, 0x05, 0xfa, 0x0a, 0xf4, 0xfd, 0xf3, 0x03, 0xfe, 0x02,
0xf4, 0xf0, 0x0d, 0x06, 0xf7, 0x08, 0x0a, 0x03, 0x02, 0xf2, 0x06, 0x01,
0xff, 0x10, 0xf5, 0xfd, 0xf4, 0x00, 0xfa, 0x00, 0xf5, 0xfc, 0x0b, 0x09,
0x0e, 0xf6, 0x01, 0x0d, 0xfb, 0x0c, 0xf4, 0x00, 0x04, 0x00, 0x03, 0x05,
0x00, 0xf9, 0xf3, 0x10, 0xf7, 0xf8, 0x0a, 0xf4, 0x03, 0x0a, 0xf1, 0xf1,
0x06, 0xf5, 0xf0, 0xfe, 0xf7, 0x04, 0xf7, 0xfc, 0xf5, 0xfb, 0xf8, 0x0a,
0x0f, 0x0e, 0xff, 0x03, 0xf9, 0xfc, 0x0a, 0xfb, 0xf2, 0x09, 0x05, 0x0e,
0x06, 0xf6, 0xf7, 0x06, 0x05, 0xff, 0x02, 0xf4, 0x00, 0x0d, 0xf5, 0xf6,
0xff, 0x0f, 0x09, 0x09, 0xfc, 0xff, 0x0f, 0xf3, 0xf2, 0xfc, 0xfa, 0x00,
0x00, 0xf5, 0xfc, 0x04, 0x0d, 0xfc, 0xf3, 0xff, 0x0f, 0x0c, 0x09, 0xf7,
0xf5, 0xf6, 0x03, 0x0d, 0xf5, 0xf3, 0x02, 0xfa, 0xf4, 0x03, 0xf1, 0xf6,
0xf9, 0xf8, 0x04, 0xfb, 0x02, 0xe6, 0x0e, 0x13, 0x16, 0x06, 0xf8, 0xf0,
0xfb, 0xf4, 0x0e, 0xfc, 0x0b, 0x05, 0xf2, 0x0e, 0xf2, 0x15, 0x01, 0x0f,
0xfb, 0xf6, 0xf6, 0xf5, 0xf9, 0xf2, 0x0b, 0x0b, 0x07, 0xf8, 0xf6, 0xed,
0xff, 0x0b, 0x03, 0x0f, 0x08, 0x0d, 0xf4, 0xf8, 0xfa, 0x0a, 0xfe, 0xfe,
0xf9, 0x12, 0x06, 0xf7, 0xfe, 0x00, 0x0a, 0xfd, 0x06, 0xf9, 0x0d, 0xf4,
0xf2, 0x04, 0xff, 0x04, 0xff, 0x02, 0xf7, 0xfa, 0xf7, 0x07, 0x0f, 0xfb,
0x0a, 0x03, 0x0b, 0x0c, 0x04, 0xf8, 0x04, 0xfb, 0x02, 0xf5, 0x04, 0x09,
0xfe, 0xf0, 0xf8, 0x09, 0x0c, 0xf9, 0x06, 0x07, 0xf5, 0xf1, 0x09, 0xf1,
0xf7, 0x09, 0xfb, 0x03, 0xfe, 0xfe, 0xfb, 0xf5, 0x0d, 0xfd, 0xfe, 0xf1,
0x02, 0x0a, 0x06, 0x0c, 0xfa, 0xf1, 0xf3, 0x0c, 0xef, 0xf5, 0x01, 0xf8,
0x0a, 0x08, 0xef, 0xf0, 0xf1, 0xee, 0xf5, 0x0e, 0xef, 0x0b, 0xfd, 0x01,
0xf7, 0x0f, 0xf4, 0x0b, 0x0d, 0x0b, 0x05, 0xff, 0x0b, 0xf3, 0xf9, 0x06,
0xf2, 0x0f, 0x07, 0xfa, 0x07, 0x02, 0x07, 0xf4, 0x03, 0x03, 0xfe, 0x0e,
```

```
0xf7, 0x04, 0xf8, 0xfe, 0xf8, 0xf8, 0x0e, 0x0a, 0x02, 0xfa, 0x05, 0xff,
0xef, 0x0e, 0x02, 0x05, 0xef, 0xf4, 0x0b, 0xfd, 0xf0, 0x07, 0x03, 0x08,
0xf7, 0xfa, 0x04, 0x0b, 0x02, 0xfb, 0xf5, 0x0f, 0xfb, 0x0d, 0x0b, 0x0d,
0x04, 0x03, 0x0a, 0xfd, 0x04, 0x02, 0x04, 0xf4, 0xfc, 0x0e, 0x0e, 0x04,
0xf8, 0x04, 0x00, 0xf4, 0xfa, 0xfe, 0x05, 0xff, 0xfe, 0xf8, 0xf2, 0xf7,
0x0f, 0x10, 0xf6, 0xfe, 0xf2, 0xf4, 0x08, 0xf8, 0xf5, 0x0c, 0xf9, 0xf8,
0xf6, 0x00, 0x0c, 0x0f, 0x0d, 0x01, 0x0b, 0xf6, 0xf4, 0xfa, 0xf8, 0x0b,
0x02, 0x02, 0x02, 0xfc, 0xfb, 0x0f, 0x0b, 0x00, 0xf5, 0x0a, 0x06, 0x0d,
0xf5, 0xf2, 0x0e, 0xf9, 0xf3, 0xf2, 0xf9, 0xf1, 0x08, 0x03, 0x09, 0x0d,
0xf4, 0xf3, 0x0b, 0xf7, 0xfc, 0x0c, 0xf7, 0xf8, 0xff, 0xfd, 0x0e, 0x05,
0x0f, 0x05, 0xf6, 0xf4, 0xfb, 0xff, 0xfb, 0x06, 0x09, 0xf0, 0x00, 0x0b,
0x00, 0x07, 0xf6, 0xf2, 0x0b, 0x08, 0xf9, 0xfd, 0x02, 0x0c, 0x0f, 0xf5,
0x0b, 0xfb, 0x0b, 0x05, 0xfd, 0x02, 0xff, 0x06, 0xfb, 0xf2, 0x0e, 0x08,
0x00, 0xf7, 0xfa, 0xf4, 0x06, 0x0e, 0xef, 0xfe, 0x09, 0x0b, 0x06, 0xf9,
0xff, 0x04, 0xf7, 0x05, 0xfa, 0x01, 0x02, 0x08, 0xfc, 0x06, 0x06, 0xfa,
0xf6, 0x0c, 0xfe, 0x0c, 0xfe, 0x06, 0x01, 0xf8, 0xfd, 0xff, 0xf7, 0x03,
0x0a, 0xfe, 0x08, 0x08, 0x00, 0xf8, 0xf8, 0xff, 0x00, 0x03, 0xf2, 0x01,
0x01, 0x09, 0x08, 0xfd, 0xfd, 0xf1, 0x0e, 0x09, 0xf8, 0x0b, 0xf7, 0x07,
0xf7, 0x10, 0xf4, 0x0c, 0xf4, 0xf0, 0x00, 0xfd, 0x02, 0x0a, 0xfc, 0xfb,
0xfc, 0x06, 0x00, 0x07, 0x06, 0x05, 0x06, 0xf1, 0x01, 0x0d, 0xfe, 0x06,
0xff, 0xfb, 0xf3, 0x07, 0xfc, 0x01, 0xfa, 0x02, 0xf0, 0x0a, 0xf9, 0xf2,
0x02, 0x08, 0xf4, 0x06, 0x0d, 0xfb, 0x08, 0x0b, 0x0d, 0xf4, 0x09, 0xf5,
0xf7, 0x04, 0xf8, 0x08, 0x04, 0x0e, 0xf3, 0xff, 0x03, 0x00, 0x0f, 0xfc,
0xfa, 0x0d, 0x01, 0xf9, 0x09, 0xfe, 0xf4, 0xfe, 0xf8, 0xfb, 0xf7, 0x00,
0x05, 0xf2, 0x07, 0x0c, 0xf9, 0x03, 0x0b, 0x0d, 0x0d, 0xe6, 0xfe, 0xff,
0x06, 0x0d, 0xf7, 0x00, 0xfb, 0xef, 0x0c, 0x00, 0xfa, 0x0b, 0xf5, 0xf1,
0x01, 0x0c, 0xfd, 0xfe, 0xeb, 0xf2, 0xfc, 0xf9, 0xfb, 0xf9, 0xfc, 0x03,
0x00, 0xf4, 0xfc, 0x05, 0x02, 0xfc, 0x0d, 0x0e, 0x07, 0x10, 0xf2, 0xec,
0x03, 0xf5, 0xfc, 0xf3, 0xf5, 0x0d, 0xf5, 0xf9, 0xf5, 0xf9, 0x07, 0xf5,
0x03, 0xfe, 0xff, 0xff, 0xf1, 0x0a, 0x04, 0xfb, 0xf0, 0xf9, 0xf5, 0x0a,
0xfd, 0x06, 0x07, 0x06, 0xf9, 0x09, 0x0b, 0x0f, 0xff, 0xf7, 0xf5, 0x09,
0x0a, 0x02, 0xfa, 0x00, 0x05, 0xf8, 0xfa, 0xee, 0x04, 0xfb, 0xf8, 0x05,
0xf1, 0xf1, 0x04, 0x0b, 0x10, 0x01, 0x05, 0xf5, 0x0c, 0x06, 0xff, 0x06,
0xfe, 0xfd, 0x10, 0x07, 0xfb, 0x0b, 0x00, 0x00, 0x05, 0xfe, 0xfe, 0xf8,
0xf1, 0x03, 0x0b, 0x0b, 0x0f, 0x03, 0xf9, 0x05, 0x00, 0xf0, 0x06, 0x0a,
0xf1, 0x03, 0xf7, 0x06, 0x02, 0x01, 0x01, 0x02, 0x0e, 0x01, 0x0d, 0xf8,
0x04, 0xf6, 0xfa, 0xfe, 0x0d, 0x03, 0xf5, 0x05, 0x0f, 0xf6, 0xfc, 0xfd,
0x07, 0xf1, 0xf3, 0xf6, 0x05, 0xf9, 0x0a, 0xfa, 0x08, 0x02, 0xf7, 0xf1,
0x0a, 0x0d, 0x06, 0x0f, 0xf7, 0xf7, 0xff, 0xf3, 0xfb, 0xfa, 0x0f, 0x08,
0x0c, 0xf0, 0xfd, 0xf9, 0x0f, 0x06, 0xf2, 0xf3, 0xf2, 0xfb, 0xf4, 0x05,
0xf2, 0xf8, 0x06, 0xfb, 0xf4, 0x04, 0xf9, 0xf7, 0x0f, 0xf5, 0x02, 0x06,
0x01, 0xff, 0xf2, 0x00, 0xf1, 0x02, 0xfe, 0x06, 0xfa, 0x0e, 0x0d, 0x03,
0x0b, 0x0e, 0xfc, 0x06, 0xf0, 0xf0, 0x05, 0xfc, 0xf5, 0x03, 0xfc, 0xf3,
0xfc, 0x05, 0xf8, 0xfc, 0x00, 0xf6, 0xfa, 0xec, 0x0e, 0xf2, 0x02, 0xfb,
0xff, 0x04, 0x09, 0xfc, 0x0c, 0x10, 0xe7, 0x00, 0xf2, 0x10, 0xf0, 0xff,
0xf4, 0x09, 0xf3, 0x0e, 0xf5, 0xf2, 0xff, 0x07, 0xf4, 0xf7, 0x05, 0x0b,
0x03, 0x00, 0xfa, 0x0b, 0x0b, 0xff, 0xf6, 0xfc, 0xf3, 0xfb, 0xf6, 0x07,
0xfe, 0xf9, 0xfe, 0xfc, 0x08, 0x05, 0x0e, 0xf8, 0xfe, 0x04, 0x03, 0xfa,
0x0b, 0x04, 0xf9, 0x09, 0xfb, 0xf7, 0x08, 0x0c, 0xfa, 0xff, 0x05, 0x09,
0x00, 0x00, 0xfe, 0x0b, 0x04, 0xfe, 0xfe, 0xfa, 0xf1, 0xf2, 0x03, 0x03,
```

```
0xf3, 0xed, 0x0a, 0xf1, 0x06, 0x08, 0x05, 0x0f, 0xf3, 0x02, 0xee, 0xf1,
0x01, 0xf6, 0xfb, 0xfb, 0x05, 0xf6, 0xfb, 0x07, 0x10, 0x00, 0x0d, 0xf5,
0x0d, 0xe5, 0x04, 0x0c, 0xf3, 0x01, 0xfb, 0x02, 0x00, 0x01, 0x07, 0x07,
0xfc, 0x07, 0xf2, 0x0a, 0xf7, 0xf0, 0x09, 0x10, 0x10, 0x0a, 0x09, 0x08,
0x11, 0x04, 0xf8, 0x0a, 0x08, 0xfc, 0x06, 0xe9, 0x04, 0xfb, 0x00, 0x05,
0x0d, 0xf7, 0xfe, 0x00, 0xf4, 0xf6, 0xff, 0xfc, 0x01, 0xf7, 0x0a, 0x09,
0x05, 0x0b, 0xfd, 0xfd, 0x01, 0xff, 0x05, 0x0b, 0xf9, 0x09, 0xf9, 0x0a,
0x03, 0x10, 0x0d, 0xec, 0xf1, 0x00, 0x12, 0x0b, 0x0d, 0x01, 0x05, 0xf7,
0xf2, 0x06, 0x0c, 0xfe, 0xeb, 0x04, 0x06, 0xf5, 0xff, 0xf9, 0x08, 0x0f,
0x09, 0xfc, 0x0b, 0xf7, 0x04, 0xfb, 0x0d, 0x07, 0xf2, 0x08, 0x00, 0xee,
0xf6, 0xfb, 0xfc, 0xf1, 0xfd, 0x0d, 0xf2, 0x02, 0xf0, 0xf2, 0xff, 0xf2,
0x09, 0xf7, 0x02, 0x03, 0xfe, 0x0e, 0xf9, 0xf5, 0xf4, 0x07, 0x07, 0x0a,
0xf7, 0xe9, 0x0c, 0x07, 0xfc, 0x0b, 0x10, 0xfe, 0xf9, 0xef, 0x0f, 0x02,
0xf8, 0x0e, 0x03, 0x0f, 0xfe, 0xf9, 0xf2, 0x0c, 0xfa, 0x06, 0x0c, 0xfa,
0xfe, 0xfa, 0xfa, 0x0d, 0xfe, 0xf3, 0x0d, 0x00, 0x00, 0xf7, 0x0f, 0xfa,
0xef, 0x05, 0xf0, 0xdf, 0x05, 0x0e, 0xed, 0x0c, 0xfa, 0xfa, 0x02, 0x07,
0xf4, 0xf5, 0x09, 0xfc, 0x04, 0xfa, 0xfa, 0xfd, 0xfe, 0xf0, 0xfd, 0x07,
0x04, 0xf4, 0x0d, 0xf9, 0xf2, 0x00, 0x01, 0xf4, 0xff, 0x05, 0x01, 0x0e,
0x03, 0x10, 0x06, 0x0d, 0xfa, 0xf9, 0xf4, 0xfc, 0x09, 0x07, 0xfe, 0x01,
0xfb, 0xf7, 0xff, 0xf9, 0x08, 0x0f, 0x08, 0xf6, 0x08, 0xf5, 0x0b, 0x01,
0xf4, 0xfb, 0x02, 0xf1, 0xfd, 0xf7, 0x10, 0xfd, 0xf4, 0x00, 0xf4, 0xf5,
0xfb, 0xf6, 0xff, 0x01, 0x07, 0xf4, 0x0b, 0xf9, 0xf5, 0xf3, 0xf4, 0xf3,
0x0e, 0x0b, 0x0a, 0x01, 0xf4, 0xf3, 0x07, 0xfd, 0x05, 0xf2, 0xf1, 0xf5,
0xf9, 0xf8, 0xfe, 0xf5, 0x10, 0xff, 0xf4, 0x09, 0xf8, 0x0e, 0x02, 0xf5,
0xff, 0x0e, 0x03, 0x03, 0x00, 0x03, 0xf6, 0xf5, 0xfb, 0x0e, 0x0f, 0x0a,
0xfa, 0x0b, 0xff, 0xfc, 0xfa, 0xff, 0xf1, 0x0b, 0xf4, 0xf7, 0x02, 0x09,
0xf4, 0xf3, 0xf9, 0x09, 0x09, 0x06, 0x0f, 0x08, 0xd2, 0xfa, 0xff, 0xff,
0x00, 0x00, 0x00, 0x09, 0x10, 0x00, 0x00, 0x00, 0x06, 0x00, 0x00, 0x00,
0x14, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x80, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00,
0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35,
0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x32, 0x2f, 0x4d, 0x61,
0x74, 0x4d, 0x75, 0x6c, 0x00, 0x00, 0x00, 0x00, 0xc4, 0xfa, 0xff, 0xff,
0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0xd0, 0xfc, 0x34, 0x3c, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x56, 0xfb, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x00, 0x01, 0x00, 0x00, 0x79, 0xec, 0xff, 0xff, 0x9b, 0xed, 0xff, 0xff,
0x16, 0xfe, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x24, 0x07, 0x00, 0x00, 0x1f, 0xfc, 0xff, 0xff,
0x97, 0xf0, 0xff, 0xff, 0x7d, 0x02, 0x00, 0x00, 0x85, 0xf5, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x30, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x2d, 0xf4, 0xff, 0xff, 0x69, 0xfd, 0xff, 0xff,
0x8f, 0xe9, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x65, 0x0b, 0x00, 0x00,
0x2e, 0x08, 0x00, 0x00, 0xfc, 0x09, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xe4, 0xe4, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00, 0x5b, 0xec, 0xff, 0xff,
0x32, 0x0b, 0x00, 0x00, 0x3a, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x94, 0xff, 0xff, 0xff, 0x3b, 0x09, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0xf9, 0xfb, 0xff, 0xff, 0xaa, 0xe9, 0xff, 0xff, 0x7d, 0xf5, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x5d, 0xed, 0xff, 0xff,
```

```
0x08, 0xe5, 0xff, 0xff, 0x4a, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x10, 0xe6, 0xff, 0xff, 0xec, 0x05, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x42, 0x03, 0x00, 0x00, 0x5b, 0x01, 0x00, 0x00, 0xee, 0xf8, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x40, 0xec, 0xff, 0xff,
0xa3, 0xf4, 0xff, 0xff, 0xad, 0xfe, 0xff, 0xff, 0xc5, 0xe7, 0xff, 0xff,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x86, 0x07, 0x00, 0x00,
0xc1, 0x18, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x46, 0xfc, 0xff, 0xff,
0x00, 0x00, 0x00, 0x02, 0x10, 0x00, 0x00, 0x00, 0x05, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x40, 0x00, 0x00, 0x00, 0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x31, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64,
0x64, 0x2f, 0x52, 0x65, 0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62,
0x6c, 0x65, 0x4f, 0x70, 0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63,
0x65, 0x00, 0x00, 0x00, 0x4c, 0xfc, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x5a, 0x07, 0x25, 0x39,
0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0xe2, 0xfc, 0xff, 0xff, 0x04, 0x00, 0x00, 0x00,
0x00, 0x02, 0x00, 0x00, 0xfe, 0x33, 0x24, 0x15, 0x09, 0xc9, 0xd6, 0xe0,
0x17, 0x24, 0x18, 0xdc, 0xf5, 0x14, 0xf1, 0x34, 0xf3, 0x2c, 0x18, 0x21,
0xf3, 0x24, 0x2f, 0xed, 0xcf, 0xf3, 0x07, 0xe8, 0x04, 0xfd, 0xf9, 0xec,
0xca, 0xeb, 0xcf, 0xcd, 0x37, 0xcb, 0x0f, 0x15, 0xc8, 0xf8, 0xe0, 0xf7,
0x13, 0x0e, 0xfc, 0x18, 0xc3, 0xd1, 0x10, 0x3b, 0xfe, 0xe2, 0xf9, 0x2a,
0x2f, 0x32, 0x27, 0x16, 0xd1, 0xe2, 0x11, 0xee, 0xe9, 0x19, 0xf5, 0x24,
0xc3, 0xe1, 0xea, 0x06, 0xdd, 0x0c, 0x15, 0x38, 0x2e, 0xe1, 0xf8, 0x22,
0x15, 0x31, 0xd3, 0x11, 0x26, 0xfd, 0xd1, 0x3b, 0xe9, 0xff, 0xd0, 0xe7,
0x23, 0x25, 0x2d, 0xe6, 0xf7, 0xd5, 0xe5, 0x2b, 0x33, 0xed, 0xdb, 0x11,
0xc7, 0xd8, 0xfc, 0x08, 0xcb, 0x19, 0xe6, 0xe2, 0xf7, 0xd0, 0xdf, 0x2f,
0xfc, 0xdf, 0xfa, 0xe0, 0x54, 0xf0, 0xe1, 0x21, 0x36, 0x12, 0x1e, 0x38,
0xf7, 0x28, 0x0d, 0x21, 0xc5, 0xde, 0x0a, 0x1c, 0x7f, 0x43, 0xe2, 0xf0,
0xc5, 0x34, 0x19, 0x3e, 0xe1, 0x25, 0xf3, 0xd6, 0xff, 0x2a, 0x01, 0x0d,
0xd8, 0xe7, 0x45, 0x22, 0xd6, 0x30, 0x33, 0x2d, 0xb9, 0xe5, 0x43, 0x03,
0x35, 0x35, 0xd6, 0x12, 0x0a, 0xe6, 0x48, 0x00, 0xd9, 0x1c, 0xda, 0x1c,
0xd7, 0xca, 0x08, 0x03, 0xdc, 0xf7, 0x3a, 0xd3, 0x2f, 0x4b, 0xfa, 0xef,
0x1f, 0xcc, 0xdc, 0x24, 0xce, 0x0c, 0xf2, 0xf2, 0xd7, 0x01, 0x0d, 0xc8,
0x7a, 0x06, 0x35, 0xd8, 0x0d, 0xce, 0xce, 0x32, 0xef, 0xf3, 0x44, 0x3c,
0x37, 0x3c, 0xed, 0x3f, 0xc4, 0xd5, 0xeb, 0x17, 0x36, 0xfa, 0x11, 0xf1,
0x33, 0x17, 0xeb, 0xe6, 0x16, 0xef, 0x2a, 0xff, 0xe9, 0xfa, 0xd4, 0xc3,
0x00, 0xea, 0x2d, 0xc3, 0x2d, 0x05, 0xd6, 0xe1, 0xf5, 0x28, 0x39, 0xed,
0xff, 0xdf, 0xd6, 0xdf, 0x1a, 0x0d, 0xe9, 0xf0, 0x1a, 0x20, 0xf2, 0xdd,
0xc2, 0x09, 0xd4, 0x1e, 0x01, 0xc2, 0x1a, 0x40, 0x19, 0xf0, 0x3c, 0x2f,
0x3d, 0x39, 0xed, 0xdf, 0x26, 0x3a, 0xf3, 0xf6, 0xd7, 0x0b, 0x38, 0x34,
0xcf, 0x11, 0x12, 0x3a, 0x1a, 0x41, 0xf5, 0x0c, 0x25, 0x39, 0x0a, 0x32,
0xf6, 0x16, 0xd2, 0x30, 0x24, 0xcd, 0x15, 0x16, 0xdc, 0xcb, 0xdf, 0xfe,
0x10, 0x0b, 0xe9, 0x2c, 0xf3, 0x17, 0x24, 0xc5, 0x11, 0xcb, 0x32, 0xf7,
0x27, 0x46, 0xea, 0x3a, 0xde, 0xe5, 0xcf, 0xf8, 0x49, 0x12, 0xfb, 0x17,
0xf1, 0x0e, 0x06, 0x0f, 0xdc, 0xe0, 0x22, 0xe8, 0xde, 0x00, 0xfb, 0xf1,
0x33, 0xd7, 0xce, 0xfe, 0xfc, 0xd6, 0x30, 0xdd, 0x1b, 0xe9, 0xca, 0xc5,
0x2b, 0xde, 0xf1, 0xfb, 0xd9, 0xcc, 0xe2, 0xf3, 0xeb, 0x00, 0x31, 0xc3,
```

```
0x0b, 0xf5, 0xc9, 0x2f, 0x13, 0x14, 0xc8, 0xf2, 0x75, 0x2b, 0x08, 0xfb,
0xef, 0xec, 0x3c, 0x0e, 0xc3, 0xe0, 0x31, 0x11, 0xdb, 0x3a, 0xc4, 0x37,
0xec, 0xfc, 0xf2, 0x0c, 0x0d, 0x03, 0x1c, 0xc7, 0xbb, 0x37, 0xd3, 0x31,
0xd1, 0x25, 0x28, 0x38, 0xd2, 0xca, 0x1c, 0x0c, 0x19, 0x2d, 0x39, 0xc3,
0xc7, 0xe5, 0x0c, 0xee, 0xe8, 0x05, 0x32, 0x10, 0xd9, 0x2c, 0xd9, 0xcb,
0x29, 0x1e, 0xc3, 0x14, 0xd1, 0x2c, 0xd7, 0xf3, 0xdb, 0xda, 0x05, 0xe1,
0x4f, 0x4d, 0x2c, 0xf7, 0xd3, 0x01, 0x11, 0xed, 0x30, 0xe2, 0xfb, 0x32,
0x13, 0x28, 0x05, 0xea, 0x21, 0xd3, 0x1c, 0x28, 0xe4, 0x3c, 0xc3, 0x20,
0x63, 0xed, 0x2f, 0x08, 0xc7, 0xeb, 0xf5, 0xe7, 0x3b, 0xf3, 0xe6, 0x0e,
0xdd, 0xf5, 0xf3, 0xd8, 0x02, 0xfb, 0x1d, 0xe1, 0x26, 0x02, 0xf1, 0xd4,
0x00, 0xdc, 0x1c, 0x0b, 0xc4, 0xf4, 0xd5, 0x3f, 0x00, 0x13, 0x41, 0xdf,
0x3a, 0xf8, 0xc3, 0xea, 0xe4, 0x35, 0xcc, 0xe4, 0x1e, 0x26, 0x38, 0xed,
0xd2, 0xfe, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09, 0x10, 0x00, 0x00, 0x00,
0x04, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00,
0x02, 0x00, 0x00, 0x00, 0x40, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x1c, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x31, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c, 0x00, 0x00, 0x00, 0x00,
0xc4, 0xfe, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x34, 0x15, 0x99, 0x3b, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x56, 0xff, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00, 0x00, 0x7e, 0xf5, 0xff, 0xff,
0x60, 0xf4, 0xff, 0xff, 0x9a, 0x03, 0x00, 0x00, 0x45, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x6d, 0xfa, 0xff, 0xff, 0x66, 0xff, 0xff, 0xff, 0x00, 0x00, 0x00, 0x02,
0x10, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00,
0x48, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x35, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75, 0x65, 0x6e, 0x74, 0x69,
0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33,
0x30, 0x2f, 0x42, 0x69, 0x61, 0x73, 0x41, 0x64, 0x64, 0x2f, 0x52, 0x65,
0x61, 0x64, 0x56, 0x61, 0x72, 0x69, 0x61, 0x62, 0x6c, 0x65, 0x4f, 0x70,
0x2f, 0x72, 0x65, 0x73, 0x6f, 0x75, 0x72, 0x63, 0x65, 0x00, 0x00, 0x00,
0x6c, 0xff, 0xff, 0xff, 0x08, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x8a, 0x37, 0x96, 0x39, 0x01, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06, 0x00,
0x08, 0x00, 0x04, 0x00, 0x06, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x08, 0x00, 0x00, 0x00, 0x14, 0x45, 0x72, 0x6a, 0xd6, 0x81, 0xf8, 0x57,
0x00, 0x00, 0x0e, 0x00, 0x18, 0x00, 0x08, 0x00, 0x07, 0x00, 0x0c, 0x00,
0x10, 0x00, 0x14, 0x00, 0x0e, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x09,
0x10, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00,
0x40, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x1c, 0x00, 0x00, 0x00, 0x73, 0x65, 0x71, 0x75,
0x65, 0x6e, 0x74, 0x69, 0x61, 0x6c, 0x5f, 0x35, 0x2f, 0x64, 0x65, 0x6e,
0x73, 0x65, 0x5f, 0x33, 0x30, 0x2f, 0x4d, 0x61, 0x74, 0x4d, 0x75, 0x6c,
0x00, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x00, 0x00,
0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x83, 0x01, 0xab, 0x3b,
0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x00, 0x00, 0x00, 0x00, 0x14, 0x00, 0x18, 0x00, 0x04, 0x00, 0x00, 0x00,
0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x00, 0x00, 0x00, 0x00, 0x14, 0x00,
```

```
  0x14, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
  0x18, 0x00, 0x00, 0x00, 0x34, 0x00, 0x00, 0x00, 0x24, 0x00, 0x00, 0x00,
  0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
  0x0e, 0x00, 0x00, 0x00, 0x64, 0x65, 0x6e, 0x73, 0x65, 0x5f, 0x33, 0x30,
  0x5f, 0x69, 0x6e, 0x70, 0x75, 0x74, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
  0xff, 0xff, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0xfc, 0xff, 0xff, 0xff,
  0x04, 0x00, 0x04, 0x00, 0x04, 0x00, 0x00, 0x00
};

const int g_sine_model_data_len = 22352;
```

13. ph_model_data.h

```
/* Copyright 2019 The TensorFlow Authors. All Rights Reserved.

Licensed under the Apache License, Version 2.0 (the "License");
you may not use this file except in compliance with the License.
You may obtain a copy of the License at

    http://www.apache.org/licenses/LICENSE-2.0

Unless required by applicable law or agreed to in writing, software
distributed under the License is distributed on an "AS IS" BASIS,
WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
See the License for the specific language governing permissions and
limitations under the License.
==============================================================================*/

// This is a standard TensorFlow Lite model file that has been converted into a
// C data array, so it can be easily compiled into a binary for devices that
// don't have a file system. It was created using the command:
// xxd -i sine_model.tflite > sine_model_data.cc

#ifndef TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_SINE_MODEL_DATA_H_
#define TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_SINE_MODEL_DATA_H_

extern const unsigned char g_sine_model_data[];
extern const int g_sine_model_data_len;

#endif // TENSORFLOW_LITE_EXPERIMENTAL_MICRO_EXAMPLES_HELLO_WORLD_SINE_MODEL_DATA_H_
```

14. send_pumpSpeed.ino

```
#include "avdweb_VirtualDelay.h" // The delay library for delays without blocking.
default = millis
VirtualDelay pumpSpeedPublishDelay; // default = millis

// Send the pump speed calculated by the AI to the cloud via IOT after every 'interval' seonds
```

```
void send_pumpSpeed(int interval) {
  interval = interval * 1000;
  pumpSpeedPublishDelay.start(interval); // calls while running are ignored
  if (pumpSpeedPublishDelay.elapsed()) {
    if  (! pumpSpeedPublishFeed.publish(pumpSpeed)) {
      Serial.println(F("Pump Speed publish Failed"));
      Serial.println();
    } else {
      Serial.print(F("Pump Speed (from AI) got published: "));
      Serial.println(pumpSpeed);
      Serial.println();
    }
  }
}
```

15. sendLiquidLevel.ino

```
 #include "RunningMedian.h" // A library for getting the running median of the sensor values for
#include "avdweb_VirtualDelay.h" // The delay library for delays without blocking.
default = millis
RunningMedian liquidLevelSamples = RunningMedian(10); // A runnng median of 10 consecutive sample
VirtualDelay liquidLevelDelay; // default = millis
const int liquid_level_pin = 33;

#define SOUND_SPEED 0.034 // Constant variable for the sound speed in cm/uS
long duration; // variable for the duration after which the wave is received back

void send_liquidLevel(int interval) {
  interval = interval * 1000;
  float liquidLevelCorrectionFactor = 1;
  liquidLevelDelay.start(interval); // calls while running are ignored
  if (liquidLevelDelay.elapsed()) {

    // Clears the trig_pin
    digitalWrite(trig_pin, LOW);
    delayMicroseconds(2);
    // Sets the trig_pin on HIGH state for 10 micro seconds
    digitalWrite(trig_pin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trig_pin, LOW);

    // Reads the echo_pin, returns the sound wave travel time in microseconds
    duration = pulseIn(echo_pin, HIGH);
    liquidLevelSamples.add(duration);

    // Calculate the distance
    waterLevelValue = 1000 - 100 * liquidLevelSamples.getMedian() * SOUND_SPEED / 2;

    //Publish the latest liquid level to the cloud
```

```
    if  (! liquidLevelFeed.publish(waterLevelValue)) {
      Serial.println(F("Liquid level publish Failed"));
    } else {
      Serial.print(F("Liquid level published: "));
      Serial.println(waterLevelValue);
    }
  }
}
```

16. sendpH.ino

```
 #include "avdweb_VirtualDelay.h" // The delay library for delays without blocking.
default = millis
VirtualDelay pHDelay; // default = millis
unsigned long int avgValue; //Store the average value of the sensor feedback
int sensorValue = 0; // initialise the sensor value to 0
int buf[10], temp; // temporary variables

void send_pH( double interval) {
  interval = interval * 1000; // convert seconds to milliseconds to be used by the delay functior

  // addPublish
  pHDelay.start(interval); // calls while running are ignored
  // Senfd the pH value to the cloud after every 'interval' seconds
  if (pHDelay.elapsed()) {
    //   calculatepH();
    if  (! pHStatusFeed.publish(pHStatus)) {
      Serial.println(F("pH publish Failed"));

    } else {
      Serial.print(F("pH Status published: "));
      Serial.println(pHStatus);

    }
    // Empty the array again for next reding
    memset(pHStatus, 0, sizeof pHStatus);

    if  (! pHFeed.publish((pHValue ))) {
      Serial.println(F("pH Value publish Failed"));
      Serial.println();
    } else {
      Serial.print(F("pH Value value published: "));
      Serial.println(pHValue );
    }
  }
}

// Calculate the pH value based on the sensor value
void calculatepH() {
```

```
memset(pHStatus, 0, sizeof pHStatus);
sensorValue = analogRead(pHSensor_pin);  //put Sensor insert into soil
for (int i = 0; i < 10; i++) //Get 10 sample value from the sensor for smooth the value
{
  buf[i] = analogRead(pHSensor_pin);
  delay(10); // delay 10 milliseconds
}

//sort the analog from small to large
for (int i = 0; i < 9; i++)
{
  for (int j = i + 1; j < 10; j++)
  {
    if (buf[i] > buf[j])
    {
      temp = buf[i];
      buf[i] = buf[j];
      buf[j] = temp;
    }
  }
}
avgValue = 0;
for (int i = 2; i < 8; i++)           //take the average value of 6 center sample
  avgValue += buf[i];
pHValue = (float)avgValue * 5.0 / 4096 / 6; //convert the analog into millivolt
pHValue = 3.5 * pHValue + 1.63;             //convert the millivolt into pH value


// Make sure that the pH value is not out of boundaries
if (pHValue < 0) {
  pHValue = 0;
} else if (pHValue > 14) {
  pHValue = 14;

}

// According to the pH value, assign it a pH status as follows:
if ((pHValue > 6.5) && (pHValue < 7.5)) {
  strcat(pHStatus, "Almost Neutral");
}
else if (pHValue > 11) {
  strcat(pHStatus, "Very Basic");
}
else if (pHValue > 8.5) {
  strcat(pHStatus, "Basic");
}
else if (pHValue > 7.5) {
  strcat(pHStatus, "Mildly Basic");
}
else if (pHValue > 5.5) {
```

```
    strcat(pHStatus, "Mildly Acidic");
  }
  else if (pHValue > 3) {
    strcat(pHStatus, "Acidic");
  }
  else if (pHValue > 0) {
    strcat(pHStatus, "Very Acidic");
  }
  else {
    strcat(pHStatus, "NA");
  }
}
```

17. setPumpSpeed.ino

```
void set_pumpSpeed(int speed) {
  //  working range
  //  4-9 ADC
  int adcValue = map(speed, 0, 100, 0, 31 );
  analogWrite(pump_pin, adcValue);
}
```

2.2 Arduino code for collecting data

1. data_collection.ino

```
#include <analogWrite.h>

// pins
int potmeterPin = 35; // pin for the sliding potentiometer
const int pH_sensor_pin = 33; // pin for the pH sensor
int pump_pin = 17; // pin for the pump
const int trigPin = 5; //trigger pin for the ultrasonic level sensor
const int echoPin = 18; // Echo pin for the for ultrasonic level sensor

// variables:
int convertedPumpSpeed = 0; // ADC value for the pump speed after conversion
int delayTime = 10; // Delay after which next reading has to be taken
bool experimentationFlag = false; // The flag to stop the experiment by keyboard control on the s

// Variables for actually printing stuff
int timeWhileExperimenting = 0; // variable for storing the time while the experimenting
double pumpSpeed = 0; // variable for the pump speed
int waterLevelValue = 0; // variable for the water level in the nutrient tank
double pHValue; // variable for the pH value

char  command = 123; // variable for incoming serial data
```

```
void setup() {
  Serial.begin(9600);
  pinMode(pump_pin, OUTPUT); // sets the pin as an output pin
  digitalWrite(pump_pin, LOW); // set the pump initially off before the experiment
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an Output for the water level sensor
  pinMode(echoPin, INPUT); // Sets the echoPin as an Input for the water level sensor
}

void loop() {
  // If something is typed on the keyboard, read it
  if (Serial.available() > 0) {
    // read the incoming byte:
    command = Serial.read();
  }

  // If the key typed is '1', begin the experiment
  if (command == '1') {
    experimentationFlag = true;
  }

  // If the key typed is '0', stop the experiment
  if (command == '0') {
    experimentationFlag = false;
    timeWhileExperimenting = 0;
  }

  // If the key typed is 's', stop the pump (in case potentiometer stops working)
  if (command == 's') {
    pumpSpeed = 0;
  }

  ////////////////////////////////////////// READ THE SENSORS ////////////////////////////////
  read_potentiometer(); // Read the potentiometer
  set_pumpSpeed(pumpSpeed); // Set the
  read_ultrasonic_waterLevel(); // Read the water level sensor
  read_pH(); // Read the pH sensor
  ////////////////////////////////////////////////////////////////////////////////////////////


  // If the experiment is being conducted, print the values to the serial monitor
  if (experimentationFlag == true) {
    Serial.print(timeWhileExperimenting);
    Serial.print(" ");
    Serial.print(pumpSpeed);
    Serial.print(" ");
    Serial.print(waterLevelValue);
    Serial.print(" ");
    Serial.print(pHValue);
    Serial.println(" ");
    timeWhileExperimenting += delayTime; // increment the time while experimenting
```

```
  }

  // delay before the next reading
  delay(delayTime);
}
```

2. read_pH.ino

```
 unsigned long int avgValue; //Store the average value of the sensor feedback
int buf[10], temp; // temporary variables
char  pHStatus[20] = ""; // Variable for storing pH status

void read_pH() {

  // Get 10 sample value from the sensor to smooth the value
  for (int i = 0; i < 10; i++)
  {
    buf[i] = analogRead(pH_sensor_pin);
    delay(10); // delay 10 milliseconds
  }

  // Sort the analog value from small to large
  for (int i = 0; i < 9; i++)
  {
    for (int j = i + 1; j < 10; j++)
    {
      if (buf[i] > buf[j])
      {
        temp = buf[i];
        buf[i] = buf[j];
        buf[j] = temp;
      }
    }
  }

  // Take the average value of 6 center samples
  avgValue = 0;
  for (int i = 2; i < 8; i++)
    avgValue += buf[i];
  pHValue = (float)avgValue * 5.0 / 4096 / 6; //convert the analog into millivolt
  pHValue = 3.5 * pHValue + 1.63;          //convert the millivolt into pH value and add the err
}
```

3. read_potentiometer.ino

```
 // Read the potentiometer
void read_potentiometer() {
 // Read the analog value of the slide potentiometer
```

```
  int temp_adc_value = analogRead(potmeterPin);
  // Map the read value between 0 and 4095, to a decimal value between 0.0 and 100.0
  pumpSpeed = map(temp_adc_value, 0, 4095, 0, 1000) / 10.0;
  // Convert the read value to control the speed of the pump. The values depend on the nature of
  convertedPumpSpeed = map(pumpSpeed, 0, 100, 4, 9 );
}
```

4. read_ultrasonic_waterLevel.ino

```
 #define SOUND_SPEED 0.034 // Constant variable for the sound speed in cm/uS
long duration; // variable for the duration after which the wave is received back

void read_ultrasonic_waterLevel() {

  // Clears the trigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro seconds
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration = pulseIn(echoPin, HIGH);

  // Calculate the distance
  // The actual distance half of the distance travelled by the waves
  waterLevelValue = 1000 - 100 * duration * SOUND_SPEED / 2;

  // Prints the distance in the Serial Monitor
  //  Serial.print("Distance (mm): ");
  //  Serial.println(waterLevelValue);
}
```

5. write_pumpSpeed.ino

```
 void set_pumpSpeed(int speed) {
  //  working range
  //  4-9 ADC
  // Map the pump speed from 0 to 100 to an ADC value for controllling the pump using PWM
  int adcValue = map(speed, 0, 100, 0, 31 );
  // Use the value to control the pump using PWM
  analogWrite(pump_pin, adcValue);
}
```

3 Jupyter Notebook code used for training the model

Train the model for the 'Intelligent Hydroponic System' using TensorFlow Lite to be run on the microcontroller

The training of the neural model happens in this notebook, after fetching the data from the Arduino's serial monitor, the data-set is made by data collected from the serial monitor of the Arduino IDE by manually sliding the potentiometer to control the pump speed to achieve the desired pH value. The data set is stored in a CSV file which is used for training the neural Model.

Deep learning networks learn to model patterns in underlying data. Here, we're going to train a network to model data generated by the pH sensor to control the speed of the pump. This will result in a model that can take a pH value, `x`, and predict its corresponding pump speed, `y`.

Training is much faster using GPU acceleration. Before you proceed, ensure you are using a GPU runtime by going to **Runtime -> Change runtime type** and set **Hardware accelerator: GPU**.

Configure Defaults

```
1
```

```
1 # Define paths to model files
2 import os
3 MODELS_DIR = 'models/'
4
5 if not os.path.isdir(MODELS_DIR):
6   os.mkdir(MODELS_DIR)
7 MODEL_TF = MODELS_DIR + 'model.pb'
8 MODEL_NO_QUANT_TFLITE = MODELS_DIR + 'model_no_quant.tflite'
9 MODEL_TFLITE = MODELS_DIR + 'model.tflite'
10 MODEL_TFLITE_MICRO = MODELS_DIR + 'model.cc'
```

Setup Environment

Install Dependencies

```
1 # ! pip install -q tensorflow==2
```

Set Seed for Repeatable Results

```
# Set a "seed" value, so we get the same random numbers each time we run this
# notebook for reproducible results.
# Numpy is a math library
import numpy as np
np.random.seed(1) # numpy seed
# TensorFlow is an open source machine learning library
import tensorflow as tf
tf.random.set_seed(1) # tensorflow global random seed
```

Import libraries

```
# Keras is TensorFlow's high-level API for deep learning
from tensorflow import keras
# Matplotlib is a graphing library
import matplotlib.pyplot as plt
# Math is Python's math library
import math
import csv

import numpy as np
import pandas as pd
import tensorflow as tf
import requests
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn import metrics
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Activation
from tensorflow.keras.callbacks import EarlyStopping
from tensorflow.keras.optimizers import SGD
from tensorflow.keras.datasets import cifar10
from tensorflow.keras.layers import Dense, Flatten, Conv2D, MaxPooling2D
from tensorflow.keras.losses import sparse_categorical_crossentropy
from tensorflow.keras.optimizers import Adam
from sklearn.model_selection import KFold

from sklearn.preprocessing import MinMaxScaler
import numpy as np
```

Read the dataset file from the google drive

After importing all the important libraries, the data is read from the dataset file using a custom method, read-file. All the columns in the data set are stored in individual arrays. Then, (X, Y) pairs are formed to prepare the data for training.


```
from google.colab import drive
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mour
```

```
corpus_name = "tinyML"
corpus = os.path.join("/content/drive/My Drive/Colab Notebooks/data", corpus_name)
```

```
import csv
csvfile = os.path.join(corpus, "ph_dataset.csv")
feature_names = ['Time (in milliseconds)','Pump speed','Water level (in mm)','pH']
# time_array=[]
speed_array=[]
# water_level_array=[]
ph_array=[]
rows_array=[]
line_count = 0

def read_file():
  global line_count
  with open(csvfile, 'r', encoding='iso-8859-1') as csv_file:
    csv_reader = csv.reader(csv_file, delimiter=',')

    # rows_array.append(feature_names)
    for row in csv_reader:
          row_array=[]
          # row is a type list
          if line_count != 0:
            for field in row:
              row_array.append(field)
            rows_array.append(row_array)
          line_count += 1

    # becuase first line a column names
    line_count -= 1

    print(f'Processed {line_count} lines.')
    for row in rows_array:
      # time_array.append(float(row[0]))
      speed_array.append(float(row[0]))
      # water_level_array.append(float(row[2]))
      ph_array.append(float(row[1]))

read_file()

# print(rows_array)
# print(time_array)
print(speed_array)
# print(water_level_array)
print(ph_array)
```

```
Processed 1000 lines.
[20.0, 20.4, 20.8, 21.2, 21.6, 22.0, 22.4, 22.8, 23.2, 23.6, 24.0, 24.4, 24.8, 25.2, 25
[14.0, 13.99, 13.98, 13.97, 13.96, 13.95, 13.94, 13.93, 13.92, 13.91, 13.9, 13.89, 13.88
```

Dataset

1. Generate Data

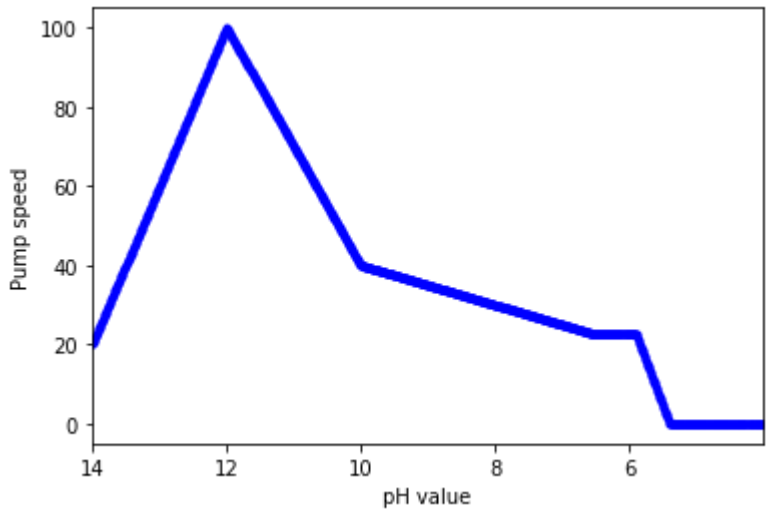
```
# Number of sample data points
SAMPLES = line_count

# FOR pH MONITORING
ph_array = np.array(ph_array, dtype=np.float32)
speed_array = np.array(speed_array, dtype=np.float32)

# Shuffle the values to guarantee they're not in order
xy=[]
for x, y in zip(ph_array, speed_array):
  xy.append([x,y])
np.random.shuffle(xy)

x_values=[]
y_values=[]
for pair in xy:
  x_values.append(pair[0])
  y_values.append(pair[1])


print(len(x_values))
print(len(y_values))
# print(x_values)
print(xy)

x=np.array(x_values)
y=np.array(y_values)

```

```
1000
1000
[[8.93, 34.65], [5.82, 18.9], [9.48, 37.4], [10.32, 49.6], [11.58, 87.4], [4.71, 0.0],
```

Plot

The data is plotted to visualize its nature.

```
# Plot our data. The 'b.' argument tells the library to print blue dots.
plt.plot(x_values, y_values, 'b.')
plt.xlabel('pH value')
plt.ylabel('Pump speed')
plt.xlim(max(x_values), min(x_values))
plt.show()
```



3. Split the Data

To evaluate the accuracy of the model we train, we'll need to compare its predictions to real data and check how well they match up. This evaluation happens during training (where it is referred to as validation) and after training (referred to as testing) It's important in both cases that we use fresh data that was not already used to train the model.

The data is split as follows:

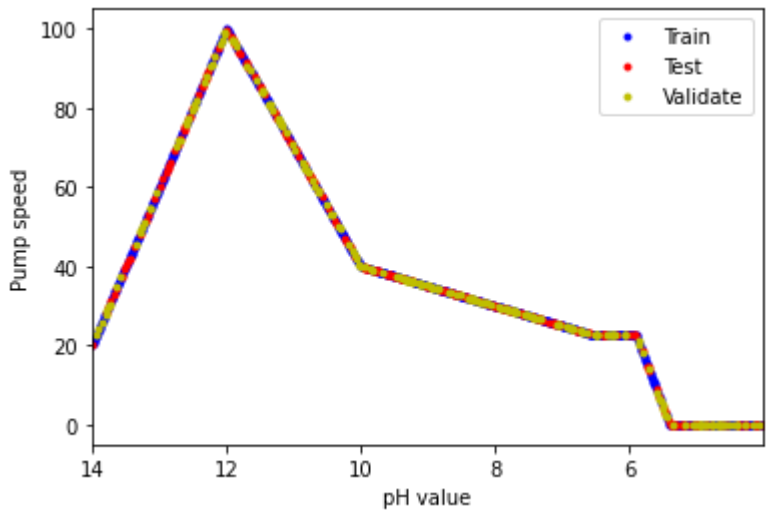
1. Training: 60%
2. Validation: 20%
3. Testing: 20%

It is made sure that the total number of samples is equal to their sum, to be on a safer side. All the sections of the datasets are individually plotted on the same graph to ensure the random assignment of data to each category.

The following code will split our data and then plots each set as a different color:

```
# We'll use 60% of our data for training and 20% for testing. The remaining 20%
# will be used for validation. Calculate the indices of each section.
TRAIN_SPLIT =  int(0.6 * SAMPLES)
TEST_SPLIT = int(0.2 * SAMPLES + TRAIN_SPLIT)
```

```
TEST_SPLIT = int(0.2 * SAMPLES + TRAIN_SPLIT)

# Use np.split to chop our data into three parts.
# The second argument to np.split is an array of indices where the data will be
# split. We provide two indices, so the data will be divided into three chunks.
x_train, x_test, x_validate = np.split(x_values, [TRAIN_SPLIT, TEST_SPLIT])
y_train, y_test, y_validate = np.split(y_values, [TRAIN_SPLIT, TEST_SPLIT])

# Double check that our splits add up correctly
assert (x_train.size + x_validate.size + x_test.size) ==  SAMPLES

# Plot the data in each partition in different colors:
plt.plot(x_train, y_train, 'b.', label="Train")
plt.plot(x_test, y_test, 'r.', label="Test")
plt.plot(x_validate, y_validate, 'y.', label="Validate")
plt.xlabel('pH value')
plt.ylabel('Pump speed')
plt.xlim(max(x_values), min(x_values))
plt.legend()
plt.show()

```



Training a Larger Model

1. Design the Model

To make our model bigger, let's add an additional layer of neurons. The following cell redefines our model in the same way as earlier, but with 16 neurons in the first layer and an additional layer of 16 neurons in the middle:

```
def sequential_model_builder():
    # # create an instance of sequential model
    # model = Sequential()
```

```
    # # Add layer with 'relu' activation function
    # # kernel_initializer, defines the way to set the initial random weights of hidden la
    # # set the initial weights according to random uniform distribtion of the neurons of
    # model.add(Dense(1,input_dim =1,activation = 'relu',kernel_initializer='random_unifor
    # # add another layer with 1 neuron, relu activation and random weight initialisation
    # model.add(Dense(8,activation='relu',kernel_initializer='random_uniform'))



    # # add last layer with output number of neurons and 'softmax' activation
    # model.add(Dense(1,activation='relu'))


    model = tf.keras.Sequential()

    # First layer takes a scalar input and feeds it through 16 "neurons". The
    # neurons decide whether to activate based on the 'relu' activation function.

    model.add(keras.layers.Dense(8, activation='relu', input_shape=(1,)))
    model.add(keras.layers.Dense(64, activation='relu', input_shape=(1,)))
    model.add(keras.layers.Dense(128, activation='relu', input_shape=(1,)))
    model.add(keras.layers.Dense(64, activation='relu', input_shape=(1,)))
    model.add(keras.layers.Dense(8, activation='relu', input_shape=(1,)))

    # Final layer is a single neuron, since we want to output a single value
    model.add(keras.layers.Dense(1))


    return model
```

K-Fold Cross Validation

```
# ADDED

# An array for storing accuracy per fold
mae_per_fold = []
# An array for storing loss per fold
loss_per_fold = []
# An variable for storing number of folds
num_folds=5
# An variable for storing epochs per fold
epoch_per_fold=300

# Concatenate the input values to be passed to the K-fold library
inputs = np.concatenate((x_train, x_test), axis=0)
# Concatenate the output values to be passed to the K-fold library
targets = np.concatenate((y_train, y_test), axis=0)
```

```
17 # Define the K-fold Cross Validator
18 k_fold = KFold(n_splits=num_folds, shuffle=False)
19
20 # K-fold counter
21 fold_count = 1
22
23 # Array for storing accuracies
24 accuracies_for_folds = []
25 # Array for storing all accuracies
26 losses_for_folds = []
27 # Array for storing all k accuracies
28 folds = []
29 # Define an array representing the entire k folds
30 entire_history_kfold = []
31
32
33 for train, test in k_fold.split(inputs, targets):
34
35   # model with crossentropy loss and adam optimizer
36   model = sequential_model_builder()
37   # model.compile(loss ='categorical_crossentropy',optimizer = 'adam',metrics = ['accuracy
38   model.compile(optimizer='adam', loss='mse', metrics=['mae'])
39
40   history = model.fit(inputs[train],targets[train],epochs=epoch_per_fold, verbose=0)
41   # Append the record of training loss values and metric values and the index of the Kth-f
42   entire_history_kfold.append([history, fold_count])
43   scores= model.evaluate(inputs[test],targets[test], verbose=0)
44
45   # add the accuracy for each fold
46   accuracies_for_folds.append(scores[1])
47   # add the losses for each fold
48   losses_for_folds.append(scores[0])
49   # add the fold number
50   folds.append([inputs[train], targets[train]])
51
52   print('------------------------------------------------------------------------')
53   print(f'On fold {fold_count} ...')
54   print(f'Score for the fold {fold_count}: {model.metrics_names[0]} of {scores[0]}; {model
55   mae_per_fold.append(scores[1] * 100)
56   loss_per_fold.append(scores[0])
57   # Increement fold number counter
58   fold_count += 1
59
60 # Store the index of the training fold that the neural network model performed best with (
61 best_fold_index = np.argmin(mae_per_fold, axis=0) # add 1 as the index starts from 0
62
63 # Store the row-wise structure of the training datatset before K-fold Cross Validation (CV
64 training_dataset_before = x_train.shape[0]
65
66 # Update the network packet information and intrusion type category data, used for trainin
67 x_train = folds[best_fold_index][0]
68 y_train = folds[best_fold_index][1]
```

```
print
print
print("best accuracy for  fold {0} ".format(best_fold_index + 1, accuracies_for_folds[best
```

```
------------------------------------------------------------------------
On fold 1 ...
Score for the fold 1: loss of 79.61366271972656; mae of 624.7452735900879%
------------------------------------------------------------------------
On fold 2 ...
Score for the fold 2: loss of 0.35357069969177246; mae of 44.056934118270874%
------------------------------------------------------------------------
On fold 3 ...
Score for the fold 3: loss of 1.1710338592529297; mae of 68.90087127685547%
------------------------------------------------------------------------
On fold 4 ...
Score for the fold 4: loss of 0.37256237864494324; mae of 41.27769470214844%
------------------------------------------------------------------------
On fold 5 ...
Score for the fold 5: loss of 129.6458282470703; mae of 727.0722389221191%
best accuracy for  fold 4
```

2. Train the model on the best K-fold

We'll now train the new model on the x-train and y-train set by the K-fold cross validation.

```
1
2 # model with crossentropy loss and adam optimizer
3 model = sequential_model_builder()
4 # compile the  model
5 model.compile(optimizer='adam', loss='mse', metrics=['mae'])
6 history = model.fit(x_train,y_train,validation_data=(x_test,y_test),batch_size=64,epochs=1
```

```
Epoch 1/1647
10/10 [==============================] - 0s 14ms/step - loss: 2065.8127 - mae: 36.93
Epoch 2/1647
10/10 [==============================] - 0s 3ms/step - loss: 1756.3381 - mae: 33.799
Epoch 3/1647
10/10 [==============================] - 0s 4ms/step - loss: 1302.9323 - mae: 28.463
Epoch 4/1647
10/10 [==============================] - 0s 3ms/step - loss: 722.1385 - mae: 19.8152
Epoch 5/1647
10/10 [==============================] - 0s 3ms/step - loss: 384.9842 - mae: 15.4910
Epoch 6/1647
10/10 [==============================] - 0s 3ms/step - loss: 394.5277 - mae: 17.3032
Epoch 7/1647
10/10 [==============================] - 0s 4ms/step - loss: 365.0885 - mae: 14.9982
Epoch 8/1647
10/10 [==============================] - 0s 3ms/step - loss: 363.2054 - mae: 14.8976
Epoch 9/1647
10/10 [==============================] - 0s 3ms/step - loss: 357.3648 - mae: 15.4967
Epoch 10/1647
10/10 [==============================] - 0s 4ms/step - loss: 354.4474 - mae: 15.1531
Epoch 11/1647
```

```
10/10 [==============================] - 0s 3ms/step - loss: 354.8341 - mae: 14.9975
Epoch 12/1647
10/10 [==============================] - 0s 3ms/step - loss: 354.1934 - mae: 15.2962
Epoch 13/1647
10/10 [==============================] - 0s 3ms/step - loss: 352.3280 - mae: 15.0653
Epoch 14/1647
10/10 [==============================] - 0s 3ms/step - loss: 352.9427 - mae: 14.8669
Epoch 15/1647
10/10 [==============================] - 0s 3ms/step - loss: 356.3134 - mae: 15.4786
Epoch 16/1647
10/10 [==============================] - 0s 4ms/step - loss: 349.2812 - mae: 15.0251
Epoch 17/1647
10/10 [==============================] - 0s 4ms/step - loss: 351.1975 - mae: 14.6928
Epoch 18/1647
10/10 [==============================] - 0s 4ms/step - loss: 347.8564 - mae: 15.0230
Epoch 19/1647
10/10 [==============================] - 0s 3ms/step - loss: 349.1453 - mae: 15.2495
Epoch 20/1647
10/10 [==============================] - 0s 3ms/step - loss: 345.4914 - mae: 14.8748
Epoch 21/1647
10/10 [==============================] - 0s 4ms/step - loss: 347.0727 - mae: 14.5053
Epoch 22/1647
10/10 [==============================] - 0s 4ms/step - loss: 343.6832 - mae: 14.8291
Epoch 23/1647
10/10 [==============================] - 0s 3ms/step - loss: 343.9125 - mae: 14.9924
Epoch 24/1647
10/10 [==============================] - 0s 3ms/step - loss: 342.8815 - mae: 14.7388
Epoch 25/1647
10/10 [==============================] - 0s 4ms/step - loss: 341.9536 - mae: 14.8922
Epoch 26/1647
10/10 [==============================] - 0s 3ms/step - loss: 344.2408 - mae: 14.5214
Epoch 27/1647
10/10 [==============================] - 0s 4ms/step - loss: 339.8533 - mae: 14.8066
Epoch 28/1647
10/10 [==============================] - 0s 4ms/step - loss: 338.7524 - mae: 14.5913
Epoch 29/1647
```

3. Plot Metrics

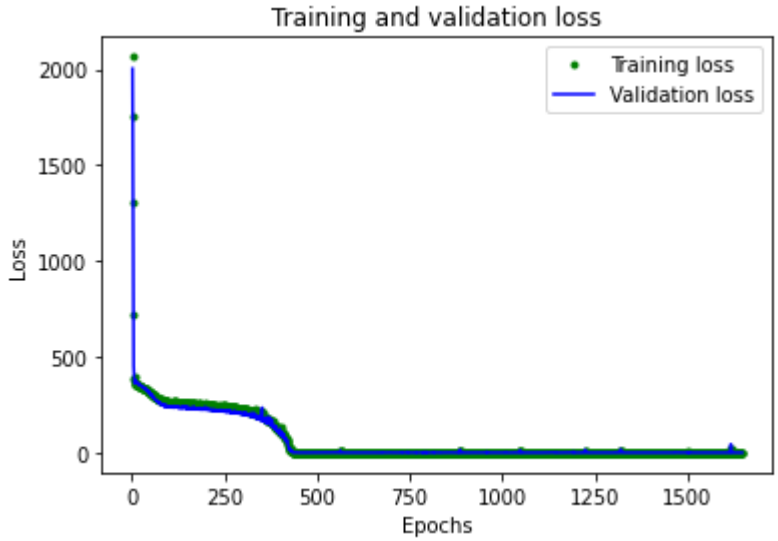
Each training epoch, the model prints out its loss and mean absolute error for training and validation. You can read this in the output above (note that your exact numbers may differ):

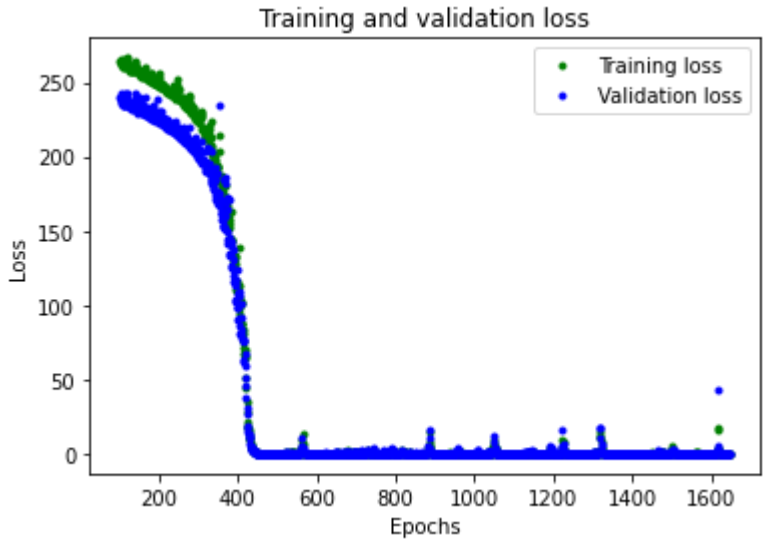
```
Epoch 500/500
600/600 [==============================] - 0s 51us/sample - loss: 0.0118 - mae: 0.0873 - val_loss: 0.
```

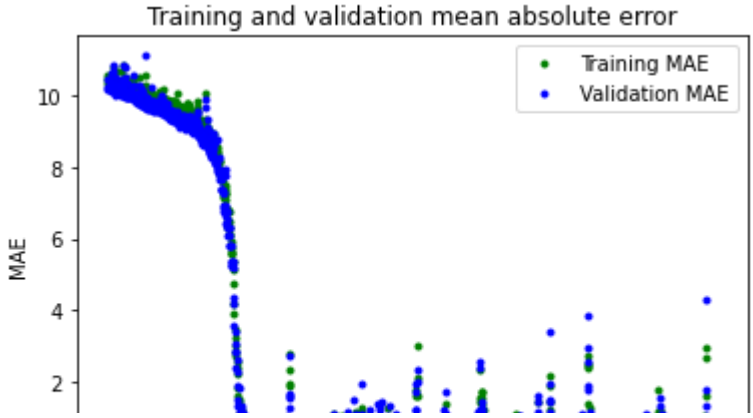
You can see that we've already got a huge improvement - validation loss has dropped from 0.15 to 0.01, and validation MAE has dropped from 0.33 to 0.08.

The following cell will print the same graphs we used to evaluate our original model, but showing our new training history:

```
# Draw a graph of the loss, which is the distance between
# the predicted and actual values during training and validation.
loss = history.history['loss']
val_loss = history.history['val_loss']

epochs = range(1, len(loss) + 1)

plt.plot(epochs, loss, 'g.', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

# Exclude the first few epochs so the graph is easier to read
SKIP = 100

plt.clf()

plt.plot(epochs[SKIP:], loss[SKIP:], 'g.', label='Training loss')
plt.plot(epochs[SKIP:], val_loss[SKIP:], 'b.', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()

plt.clf()

# Draw a graph of mean absolute error, which is another way of
# measuring the amount of error in the prediction.
mae = history.history['mae']
val_mae = history.history['val_mae']

plt.plot(epochs[SKIP:], mae[SKIP:], 'g.', label='Training MAE')
plt.plot(epochs[SKIP:], val_mae[SKIP:], 'b.', label='Validation MAE')
plt.title('Training and validation mean absolute error')
plt.xlabel('Epochs')
plt.ylabel('MAE')
plt.legend()
plt.show()
```







Great results! From these graphs, we can see several exciting things:

- Our network has reached its peak accuracy much more quickly (within 200 epochs instead of 500)
- The overall loss and MAE are much better than our previous network
- Metrics are better for validation than training, which means the network is not overfitting

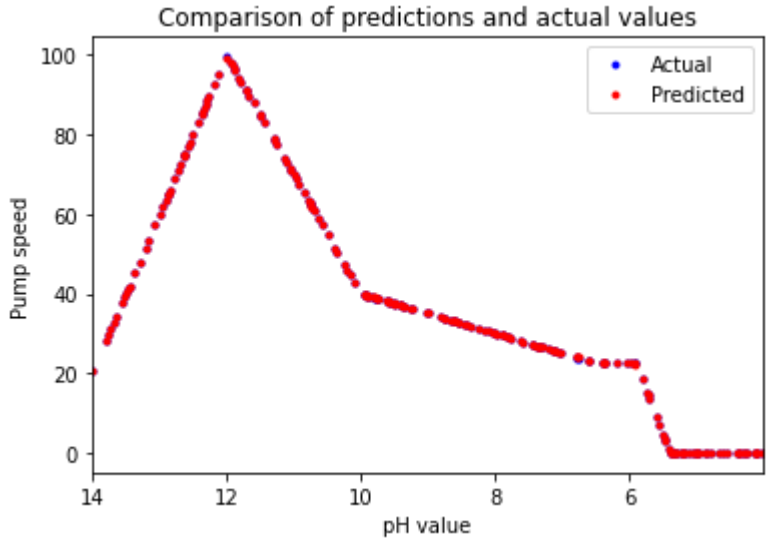
The reason the metrics for validation are better than those for training is that validation metrics are calculated at the end of each epoch, while training metrics are calculated throughout the epoch, so

validation happens on a model that has been trained slightly longer.

This all means our network seems to be performing well! To confirm, let's check its predictions against the test dataset we set aside earlier:

```
# Calculate and print the loss on our test dataset
loss = model.evaluate(x_test, y_test)

# Make predictions based on our test dataset
predictions = model.predict(x_test)

# Graph the predictions against the actual values
plt.clf()
plt.title('Comparison of predictions and actual values')
plt.plot(x_test, y_test, 'b.', label='Actual')
plt.plot(x_test, predictions, 'r.', label='Predicted')
plt.xlabel('pH value')
plt.ylabel('Pump speed')
plt.xlim(max(x_values), min(x_values))
plt.legend()
plt.show()
```

```
7/7 [==============================] - 0s 2ms/step - loss: 0.0030 - mae: 0.0285
```



Much better! The evaluation metrics we printed show that the model has a low loss and MAE on the test data, and the predictions line up visually with our data fairly well.

Generate a TensorFlow Lite Model

1. Generate Models with or without Quantization

We now have an acceptably accurate model. We'll use the TensorFlow Lite Converter to convert the model into a special, space-efficient format for use on memory-constrained devices.

Since this model is going to be deployed on a microcontroller, we want it to be as tiny as possible! One technique for reducing the size of models is called quantization while converting the model. It reduces the precision of the model's weights, and possibly the activations (output of each layer) as well, which saves memory, often without much impact on accuracy. Quantized models also run faster, since the calculations required are simpler.

Note: Currently, TFLite Converter produces TFlite models with float interfaces (input and output ops are always float). This is a blocker for users who require TFlite models with pure int8 or uint8 inputs/outputs. Refer to https://github.com/tensorflow/tensorflow/issues/38285

```
# Convert the model to the TensorFlow Lite format without quantization
converter = tf.lite.TFLiteConverter.from_keras_model(model)
model_no_quant_tflite = converter.convert()

# # Save the model to disk
open(MODEL_NO_QUANT_TFLITE, "wb").write(model_no_quant_tflite)

# Convert the model to the TensorFlow Lite format with quantization
def representative_dataset():
  for i in range(int(SAMPLES/2)):
    yield([x_train[i].reshape(1, 1)])
# Set the optimization flag.
converter.optimizations = [tf.lite.Optimize.DEFAULT]
# Enforce full-int8 quantization (except inputs/outputs which are always float)
converter.target_spec.supported_ops = [tf.lite.OpsSet.TFLITE_BUILTINS_INT8]
# Provide a representative dataset to ensure we quantize correctly.
converter.representative_dataset = representative_dataset
model_tflite = converter.convert()

# Save the model to disk
open(MODEL_TFLITE, "wb").write(model_tflite)
```

```
INFO:tensorflow:Assets written to: /tmp/tmp6uwtk3f1/assets
INFO:tensorflow:Assets written to: /tmp/tmp8r_2p0e4/assets
INFO:tensorflow:Assets written to: /tmp/tmp8r_2p0e4/assets
22352
```

2. Compare Model Sizes

```
import os
model_no_quant_size = os.path.getsize(MODEL_NO_QUANT_TFLITE)
print("Model is %d bytes" % model_no_quant_size)
model_size = os.path.getsize(MODEL_TFLITE)
print("Quantized model is %d bytes" % model_size)
difference = model_no_quant_size - model_size
```

```
7 print("Difference is %d bytes" % difference)
```

```
Model is 73652 bytes
Quantized model is 22352 bytes
Difference is 51300 bytes
```

Our quantized model is only 224 bytes smaller than the original version, which only a tiny reduction in size! At around 2.5 kilobytes, this model is already so small that the weights make up only a small fraction of the overall size, meaning quantization has little effect.

More complex models have many more weights, meaning the space saving from quantization will be much higher, approaching 4x for most sophisticated models.

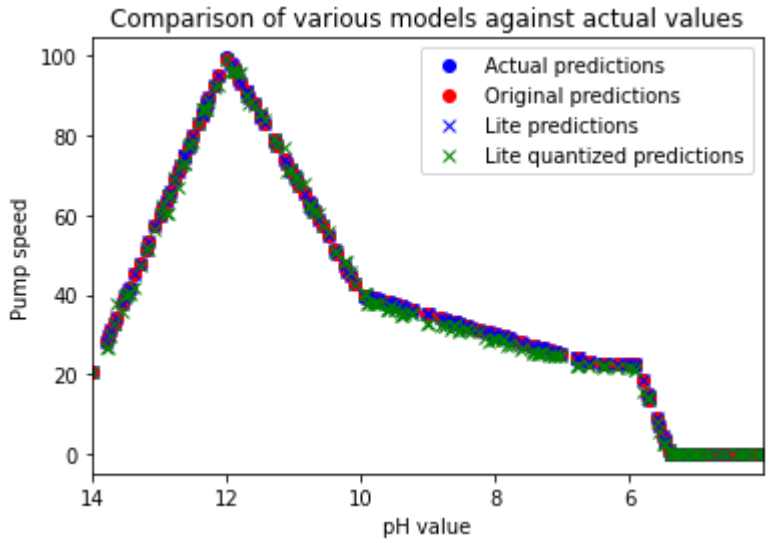
Regardless, our quantized model will take less time to execute than the original version, which is important on a tiny microcontroller!

3. Test the Models

To prove these models are still accurate after conversion and quantization, we'll use both of them to make predictions and compare these against our test results:

```
 1 # Instantiate an interpreter for each model
 2 model_no_quant = tf.lite.Interpreter(MODEL_NO_QUANT_TFLITE)
 3 model = tf.lite.Interpreter(MODEL_TFLITE)
 4
 5 # Allocate memory for each model
 6 model_no_quant.allocate_tensors()
 7 model.allocate_tensors()
 8
 9 # Get the input and output tensors so we can feed in values and get the results
10 model_no_quant_input = model_no_quant.tensor(model_no_quant.get_input_details()[0]["index"
11 model_no_quant_output = model_no_quant.tensor(model_no_quant.get_output_details()[0]["inde
12 model_input = model.tensor(model.get_input_details()[0]["index"])
13 model_output = model.tensor(model.get_output_details()[0]["index"])
14
15 # Create arrays to store the results
16 model_no_quant_predictions = np.empty(x_test.size)
17 model_predictions = np.empty(x_test.size)
18
19 # Run each model's interpreter for each value and store the results in arrays
20 for i in range(x_test.size):
21   model_no_quant_input().fill(x_test[i])
22   model_no_quant.invoke()
23   model_no_quant_predictions[i] = model_no_quant_output()[0]
24
25   model_input().fill(x_test[i])
26   model.invoke()
```

```
  model_predictions[i] = model_output()[0]

# See how they line up with the data
plt.clf()
plt.title('Comparison of various models against actual values')
plt.plot(x_test, y_test, 'bo', label='Actual predictions')
plt.plot(x_test, predictions, 'ro', label='Original predictions')
plt.plot(x_test, model_no_quant_predictions, 'bx', label='Lite predictions')
plt.plot(x_test, model_predictions, 'gx', label='Lite quantized predictions')
plt.xlabel('pH value')
plt.ylabel('Pump speed')
plt.xlim(max(x_values), min(x_values))
plt.legend()
plt.show()
```



We can see from the graph that the predictions for the original model, the converted model, and the quantized model are all close enough to be indistinguishable. This means that our quantized model is ready to use!

Generate a TensorFlow Lite for Microcontrollers Model

Convert the TensorFlow Lite quantized model into a C source file that can be loaded by TensorFlow Lite for Microcontrollers.

```
# Install xxd if it is not available
!apt-get update && apt-get -qq install xxd
# Convert to a C source file
!xxd -i {MODEL_TFLITE} > {MODEL_TFLITE_MICRO}
# Update variable names
REPLACE_TEXT = MODEL_TFLITE.replace('/', '_').replace('.', '_')
!sed -i 's/'{REPLACE_TEXT}'/g_model/g' {MODEL_TFLITE_MICRO}
```

```
Hit:1 http://ppa.launchpad.net/c2d4u.team/c2d4u4.0+/ubuntu bionic InRelease
Hit:2 http://archive.ubuntu.com/ubuntu bionic InRelease
Hit:3 https://cloud.r-project.org/bin/linux/ubuntu bionic-cran40/ InRelease
Get:4 http://archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Hit:5 http://ppa.launchpad.net/cran/libgit2/ubuntu bionic InRelease
Hit:6 http://ppa.launchpad.net/deadsnakes/ppa/ubuntu bionic InRelease
Get:7 http://archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Hit:8 http://ppa.launchpad.net/graphics-drivers/ppa/ubuntu bionic InRelease
Get:9 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Ign:10 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64  InRe
Ign:11 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x
Hit:12 https://developer.download.nvidia.com/compute/cuda/repos/ubuntu1804/x86_64  Relea
Hit:13 https://developer.download.nvidia.com/compute/machine-learning/repos/ubuntu1804/x
Fetched 252 kB in 2s (166 kB/s)
Reading package lists... Done
```

Deploy to a Microcontroller

Follow the instructions in the hello_world README.md for TensorFlow Lite for MicroControllers to deploy this model on a specific microcontroller.

Reference Model: If you have not modified this notebook, you can follow the instructions as is, to deploy the model. Refer to the `hello_world/train/models` directory to access the models generated in this notebook.

New Model: If you have generated a new model, then update the values assigned to the variables defined in `hello_world/model.cc` with values displayed after running the following cell.

```
1 # Print the C source file
2 !cat {MODEL_TFLITE_MICRO}
```

```
unsigned char g_model[] = {
  0x1c, 0x00, 0x00, 0x00, 0x54, 0x46, 0x4c, 0x33, 0x14, 0x00, 0x20, 0x00,
  0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x14, 0x00, 0x00, 0x00,
  0x18, 0x00, 0x1c, 0x00, 0x14, 0x00, 0x00, 0x00, 0x03, 0x00, 0x00, 0x00,
  0x18, 0x00, 0x00, 0x00, 0x24, 0x00, 0x00, 0x00, 0xdc, 0x00, 0x00, 0x00,
  0x24, 0x00, 0x00, 0x00, 0x84, 0x00, 0x00, 0x00, 0x7c, 0x00, 0x00, 0x00,
  0x03, 0x00, 0x00, 0x00, 0xa4, 0x07, 0x00, 0x00, 0xac, 0x06, 0x00, 0x00,
  0xbc, 0x01, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0xdc, 0x00, 0x00, 0x00,
  0x17, 0x00, 0x00, 0x00, 0xf4, 0x56, 0x00, 0x00, 0xf0, 0x56, 0x00, 0x00,
  0xf0, 0x55, 0x00, 0x00, 0x3c, 0x55, 0x00, 0x00, 0xc4, 0x52, 0x00, 0x00,
  0x34, 0x51, 0x00, 0x00, 0xbc, 0x30, 0x00, 0x00, 0x2c, 0x2e, 0x00, 0x00,
  0xb4, 0x0d, 0x00, 0x00, 0x24, 0x0c, 0x00, 0x00, 0xac, 0x09, 0x00, 0x00,
  0xfc, 0x08, 0x00, 0x00, 0x7c, 0x08, 0x00, 0x00, 0xe8, 0x07, 0x00, 0x00,
  0xbc, 0x56, 0x00, 0x00, 0xb8, 0x56, 0x00, 0x00, 0xb4, 0x56, 0x00, 0x00,
  0xb0, 0x56, 0x00, 0x00, 0xac, 0x56, 0x00, 0x00, 0xa8, 0x56, 0x00, 0x00,
  0xa4, 0x56, 0x00, 0x00, 0xa0, 0x56, 0x00, 0x00, 0x3c, 0x00, 0x00, 0x00,
  0x00, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00,
  0x08, 0x00, 0x0c, 0x00, 0x04, 0x00, 0x08, 0x00, 0x08, 0x00, 0x00, 0x00,
  0x08, 0x00, 0x00, 0x00, 0x16, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
  0x6d, 0x69, 0x6e, 0x5f, 0x72, 0x75, 0x6e, 0x74, 0x69, 0x6d, 0x65, 0x5f,
```

```
0x76, 0x65, 0x72, 0x73, 0x69, 0x6f, 0x6e, 0x00, 0xa2, 0xaa, 0xff, 0xff,
0x04, 0x00, 0x00, 0x00, 0x10, 0x00, 0x00, 0x00, 0x31, 0x2e, 0x31, 0x34,
0x2e, 0x30, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x0f, 0x00, 0x00, 0x00, 0x4d, 0x4c, 0x49, 0x52, 0x20, 0x43, 0x6f, 0x6e,
0x76, 0x65, 0x72, 0x74, 0x65, 0x64, 0x2e, 0x00, 0x00, 0x00, 0x0e, 0x00,
0x18, 0x00, 0x04, 0x00, 0x08, 0x00, 0x0c, 0x00, 0x10, 0x00, 0x14, 0x00,
0x0e, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x68, 0x00, 0x00, 0x00,
0x6c, 0x00, 0x00, 0x00, 0x70, 0x00, 0x00, 0x00, 0x90, 0x00, 0x00, 0x00,
0x15, 0x00, 0x00, 0x00, 0xb8, 0x55, 0x00, 0x00, 0x28, 0x55, 0x00, 0x00,
0x7c, 0x54, 0x00, 0x00, 0xe4, 0x53, 0x00, 0x00, 0x54, 0x51, 0x00, 0x00,
0xdc, 0x4f, 0x00, 0x00, 0x4c, 0x2f, 0x00, 0x00, 0xd4, 0x2c, 0x00, 0x00,
0x44, 0x0c, 0x00, 0x00, 0xcc, 0x0a, 0x00, 0x00, 0x3c, 0x08, 0x00, 0x00,
0xa4, 0x07, 0x00, 0x00, 0x0c, 0x07, 0x00, 0x00, 0x8c, 0x06, 0x00, 0x00,
0x80, 0x05, 0x00, 0x00, 0x6c, 0x04, 0x00, 0x00, 0x70, 0x03, 0x00, 0x00,
0x84, 0x02, 0x00, 0x00, 0x98, 0x01, 0x00, 0x00, 0xf8, 0x00, 0x00, 0x00,
0x7c, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00,
0x0c, 0x06, 0x00, 0x00, 0xfc, 0x04, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00,
0x04, 0x03, 0x00, 0x00, 0x18, 0x02, 0x00, 0x00, 0x2c, 0x01, 0x00, 0x00,
0x90, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x6d, 0x61, 0x69, 0x6e, 0x00, 0x00, 0x0a, 0x00, 0x10, 0x00, 0x04, 0x00,
0x08, 0x00, 0x0c, 0x00, 0x0a, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x10, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x14, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x2c, 0xfa, 0xff, 0xff, 0x06, 0x00, 0x00, 0x00, 0x00, 0x00, 0x00, 0x06,
0x02, 0x00, 0x00, 0x00, 0x28, 0xab, 0xff, 0xff, 0x14, 0x00, 0x00, 0x00,
0x15, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00, 0x30, 0x00, 0x00, 0x00,
0x20, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x08, 0x00, 0x00, 0x00, 0x49, 0x64, 0x65, 0x6e,
0x74, 0x69, 0x74, 0x79, 0x00, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0xff, 0xff, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0x0c, 0xab, 0xff, 0xff,
0xae, 0xfc, 0xff, 0xff, 0x00, 0x00, 0x00, 0x08, 0x01, 0x00, 0x00, 0x00,
0x18, 0x00, 0x00, 0x00, 0x0c, 0x00, 0x00, 0x00, 0x04, 0x00, 0x00, 0x00,
0x28, 0xab, 0xff, 0xff, 0x01, 0x00, 0x00, 0x00, 0x13, 0x00, 0x00, 0x00,
0x03, 0x00, 0x00, 0x00, 0x12, 0x00, 0x00, 0x00, 0x0b, 0x00, 0x00, 0x00,
0x0c, 0x00, 0x00, 0x00, 0x98, 0xfa, 0xff, 0xff, 0x00, 0x00, 0x00, 0x09,
0x14, 0x00, 0x00, 0x00, 0x14, 0x00, 0x00, 0x00, 0x18, 0x00, 0x00, 0x00,
0x30, 0x00, 0x00, 0x00, 0x20, 0x00, 0x00, 0x00, 0x02, 0x00, 0x00, 0x00,
0x01, 0x00, 0x00, 0x00, 0x01, 0x00, 0x00, 0x00, 0x09, 0x00, 0x00, 0x00,
0x40, 0x64, 0x65, 0x6e, 0x74, 0x69, 0x74, 0x79, 0x21, 0x00, 0x00, 0x00,
```