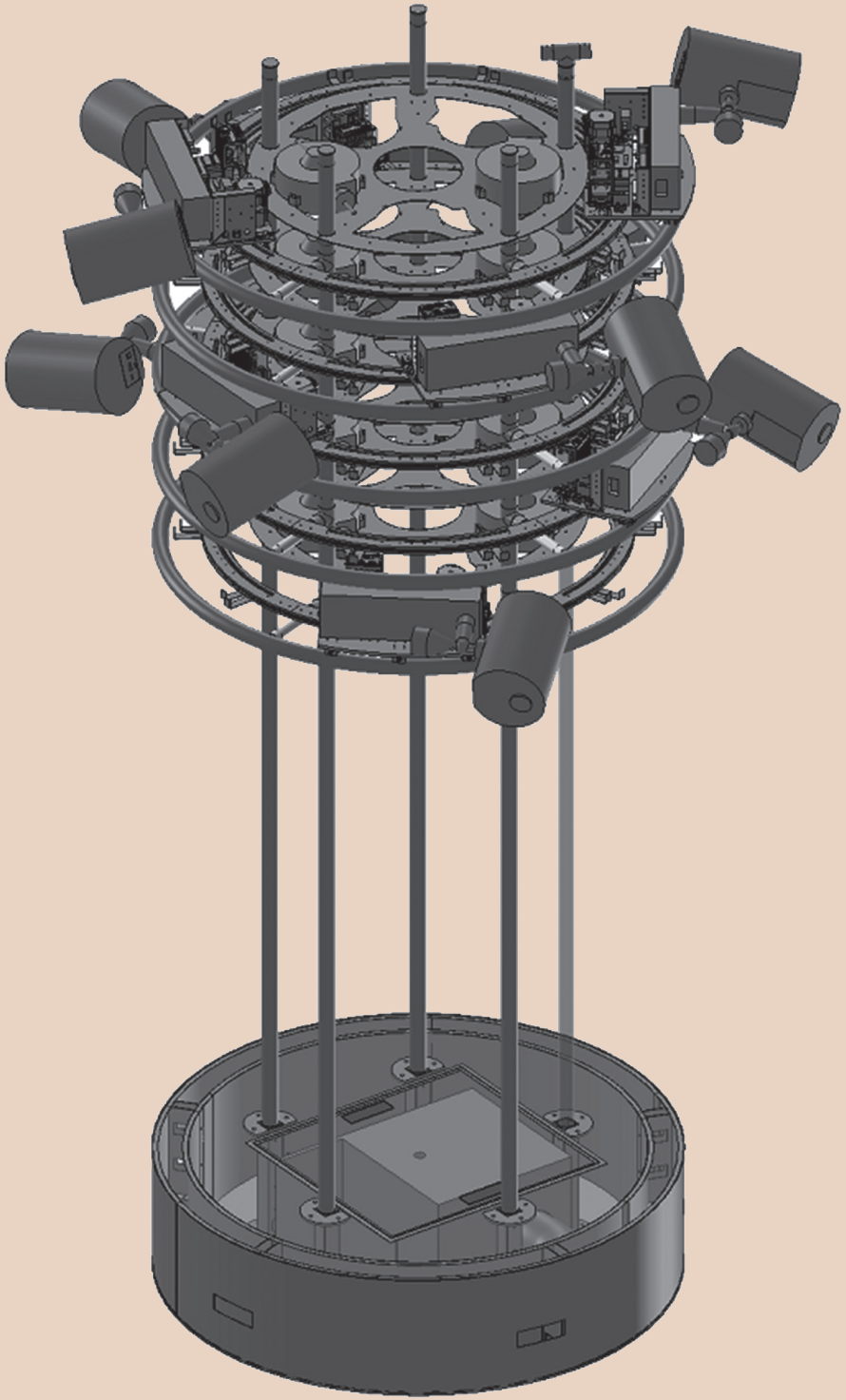
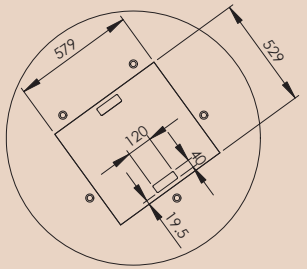
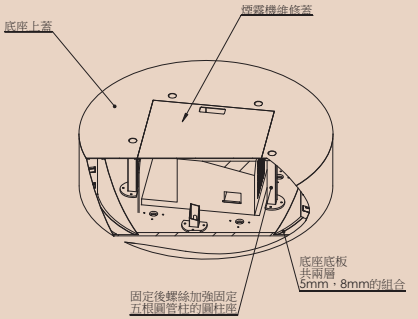
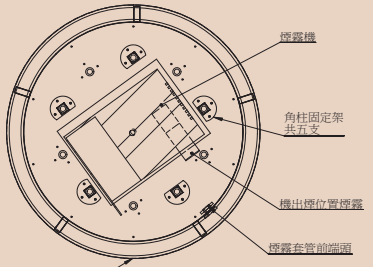
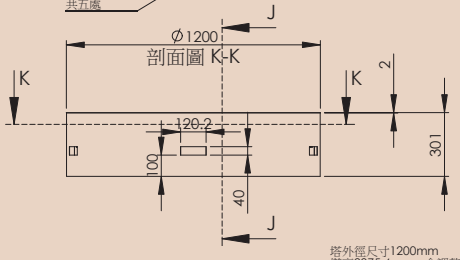
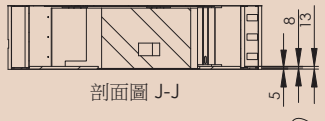




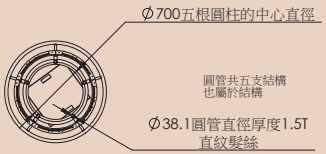


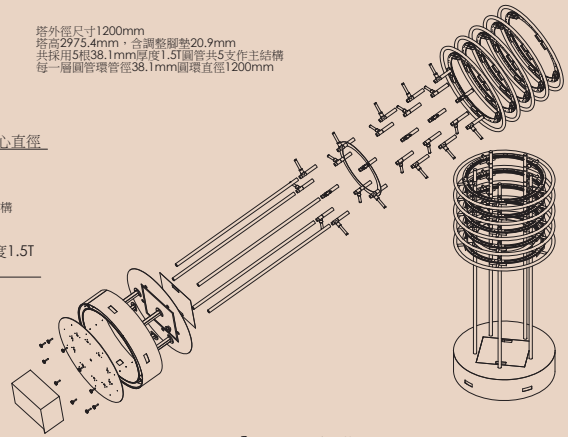


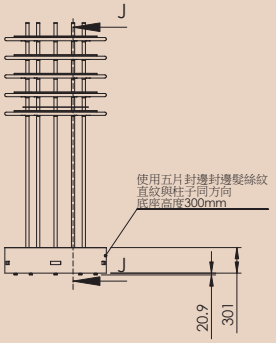


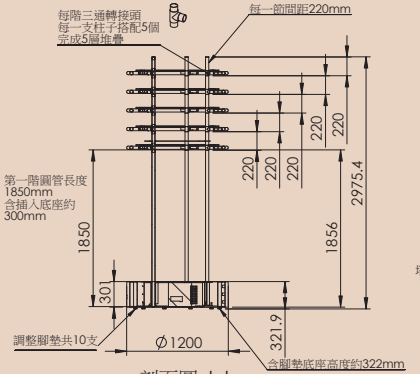


塔外徑尺寸1200mm
塔高2975.4mm，含調整腳墊20.9mm
共採用5根38.1mm厚度1.5T圓管共5支作主結構
每一層圓管環管徑38.1mm圓環直徑1200mm









塔總高2975.4mm

```
const mosca = require('mosca')
const Max = require('max-api');


var moscaSettings = {
    port: 1883,
};

var server = new mosca.Server(moscaSettings);
server.on('ready', setup);

server.on('clientConnected', function (client) {
    console.log('client connected', client.id);
});


// Sending data from mosca to clients
const handlers = {
    //file_path = path for playing video

    main_video: (file_path) => {
        publish_player('1-1', 'play-video', file_path);
    },

    side_video: (file_path) => {
        publish_player('1-1', 'play-video-loop', file_path);
    },

    stop_video: (file_path) => {
        publish_player('1-1', 'stop-video', file_path);
    },

    start_4k_video: () => {
        publish_4k_player('1', 'start');
        console.log('hello');
    },

    stop_4k_video: () => {
        publish_4k_player('1', 'stop');
        console.log('hello');
    }

};

Max.addHandlers(handlers);

//
function publish_player(id, mode, _file_path) {
    var message = {
        topic: `tower/mqtt-media-player/${id}/${mode}`,
        payload: `${_file_path}`, // or a Buffer
        qos: 0, // 0, 1, or 2
        retain: false // or true
    };
    server.publish(message, function () {
        console.log(message);
        //Max.post('fire video!!');
    });
}

function publish_4k_player(id, mode) {
    var message = {
        topic: `4k/4k-player/${id}/${mode}`,
        payload: "", // or a Buffer
        qos: 0, // 0, 1, or 2
        retain: false // or true
    };
    server.publish(message, function () {
        console.log(message);
        //Max.post('fire 4k video!!');
    });
}


// fired when a message is received
server.on('published', function (packet, client) {
    context = packet.payload.toString();
    console.log('Published', context);
    Max.outlet(context);
});

// fired when the mqtt server is ready
function setup() {
    console.log('Mosca server is up and running')
}
```