

Project Documentation

Smart Energy Token

This file describes all the components of the new implementation and their workings. Both hardware and software will be explained per component.

Gateway

Hardware

For the gateway we have used an Arduino UNO with an RF-transceiver connected to it. The UNO also contains an USB port which is used to send data to the PC. In the future the communication could also be done through RS232, but we did not have the necessary supplies to make this work for this implementation.

Software

For the gateway we have written an Arduino sketch which captures data from the buildings with the help of RF-transceivers. The captured data is converted into a custom data structure named TokenInfo which contains the id of the building and the amounts of connected energy innovations. The sketch file then sends this data over its serial port to the PC.

We've written a very simple python script that reads the data that gets sent from the Arduino over the serial connection. This data is then parsed and displayed in the terminal. Since we didn't have the source code that's on the TouchTable in Eindhoven, it's not been integrated into that project.

Building

Hardware

For the building we have chosen the ATtiny841 microcontroller because of multiple factors. First of all it's a very small processor so it can be easily implemented into the current 3D-printed buildings. Secondly, this controller is very cheap which makes it easy to expand the project in the future. Thirdly, this microcontroller contains 2 UART ports which is what we need in order to use the RF-transceiver and serial communication at the same time.

The building has the same RF-transceiver module as the gateway so that they can communicate with each other. The building also has a 4-pin output port with which tokens can connect. The building can then communicate with the tokens by using serial communication. The 4-pin connectors are connected to the VCC, GND, TX and RX pins of the microcontroller.

The building also has a battery pack as a power supply. The battery pack is connected to the ATtiny841 of the building but also to the 4-pin output port because it needs to power the

tokens. Unfortunately the electroengineering students that were a part of the project have failed to finish their work so we haven't been able to select a suitable battery yet.

Software

We also wrote an Arduino sketch for the building. It receives data from the tokens that are stacked on top of it through UART. In a loop it reads what is being sent through the UART connection. The token will just send what type it is, so a switch case is used to add or remove an innovation when a token is added or removed. The data is saved into a custom data structure named TokenInfo which contains the id of the building and the amounts of connected energy innovations. The building then sends the TokenInfo data over RF to the gateway.

Token

Hardware

Just like the building, the token also uses an ATtiny841 as its microcontroller. This microcontroller was chosen for the same reasons, the only difference being that the double UART ports aren't used for RF but only for serial. This way the token can communicate with the component beneath it and the component above it.

The token contains both a 4-pin input and a 4-pin output port. This is done so it can connect to a component on the bottom and another component can be connected to the token on the top. The 4-pin connectors are connected to the VCC, GND, TX and RX pins of the microcontroller.

Software

For the token we have written an Arduino sketch file. Because the ATtiny841 has two UART ports, it also has two serial objects in code (from the ATtinycore library): Serial and Serial1. In this sketch file the token is programmed to receive data through its Serial1 connection and it is programmed to write over its Serial connection.

The data it's sending and receiving is an array of integers. These integers represent different energy innovations of all the connected tokens. Each token has a constant variable called innovationType. This variable needs to be set according to the energy innovations it will be used as. The tokens are static so they can only be one innovation, the token can then be implemented in a 3D-printed object of the matching innovation type.