Uniwersytet Mikołaja Kopernika

Wydział Fizyki, Astronomii i Informatyki Stosowanej

Maciej Drążek
nr albumu : 212581

Praca inżynierska
na kierunku Fizyka Techniczna

Aplety komputerowe w zakresie elektromagnetyzmu

Opiekun pracy
prof. zw. dr hab. inż. Grzegorz Karwasz

Toruń, 2010

Pracę przyjmuję i akceptuję

……..
data i podpis opiekuna pracy

Potwierdzam złożenie pracy dyplomowej

……
data i podpis pracownika dziekanatu

Dziękuję Panu prof. dr hab. Grzegorzowi Karwaszowi za okazywaną pomoc i cierpliwość w trakcie pisania niniejszej pracy. Dziękuję również moim bliskim za udzielone mi wsparcie.

UMK zastrzega sobie prawo własności niniejszej pracy inżynierskiej w celu udostępniania dla potrzeb działalności naukowo-badawczej lub dydaktycznej

Spis treści

		Strona
1.	**Wstęp**……..	**5**
2.	**Elektrostatyka**……	**8**
2.1.	Ładunek elektryczny…….……	8
2.2.	Prawo Coulomba……	8
2.3.	Pole elektryczne……	9
2.4.	Prawo Gaussa……	12
3.	**Prąd elektryczny**……	**15**
4.	**Magnetyzm**……	**17**
4.1.	Pole magnetyczne, siła Lorenza i indukcja pola magnetycznego…….	17
4.2.	Prawo Ampere'a……	19
4.3.	Prawo Biota-Savarta……..	22
5.	**Indukcja**…….	**24**
5.1.	Prawo indukcji Faraday'a…….	24
5.2.	Reguła Lenz'a……	24
5.3.	Zjawisko indukcji……..	26
6.	**Interfejs i implementacja apletów**……	**29**
6.1.	Informacje wstępne……	29
6.2.	Implementacja doświadczenia Oersteda……	30
6.3.	Implementacja doświadczenia z elektroskopem……	37
6.4.	Implementacja doświadczenia z solenoidem……	38
6.5.	Implementacja doświadczenia z ramką z prądem…….	46
6.6.	Implementacja doświadczenia Ampere'a…….	48
7.	**Podsumowanie** ……..	**57**
8.	**Literatura** ……..	**58**

1. Wstęp

Technologia Flash jest obecnie jedną z najbardziej popularnych technologii do tworzenia animacji, reklam, stron internetowych, gier oraz wielu innych. Jest to w pełni multimedialna platforma początkowo tworzona przez firmę Macromedia, obecnie rozwijana przez firmę Adobe. Tworząc animacje w technologii Flash możemy operować na grafice wektorowej, rastrowej, wykorzystywać strumienie audio i wideo oraz wykorzystywać wewnętrzny język programowania, zwany ActionScript aby tworzyć złożone i interaktywne animacje. Animacje wykonane tą technologią są zapisywane do specjalnego formatu o rozszerzeniu „.swf" i mogą być odtwarzane na stronach internetowych za pomocą specjalnej „wtyczki" instalowanej w przeglądarkach. Odbywa się to w ten sposób, że na dysk użytkownika ściągana jest animacja z odległego serwera. Po ściągnięciu takiej animacji, odtwarzanie jej odbywa się na lokalnym sprzęcie użytkownika wykorzystując zasoby sprzętowe komputera użytkownika. Za ojców tej technologii uważa się Jonathana Gay'a, Chraliego Jacksona i Michellea Welsha. Stworzyli oni mała firmę zwaną FutureWave Software, która stworzyła program SmartSketch do tworzenia grafiki wektorowej. Podczas rozwoju aplikacji pracownicy FutureSoftware zdali sobie z potencjału, jaki będzie towarzyszył zorientowanym wektorowo animacjom internetowym. Dodali więc do swojego programu opcję animacji poklatkowej i wydali program pod nazwą FutureSplash Animator. Program ten został wykorzystany przy kilku wielkich komercjalnych projektach takich jak elementy witryn Microsoft Network, oficjalnej strony internetowej serialu The Simpsons. W 1996 FutureWave została wykupiona przez firmę Macromedia. Program został wydany pod nową nazwą Macromedia Flash. Firma Macromedia dodała do kolejnych wersji swojego programu nowe funkcje m.in.: bibliotekę obiektów, przeźroczystość, obiekty typu MovieClip.

W 2000 roku, do wersji 5 programu dodano nowe rewolucyjne technologie takie jak wsparcie dla plików XML i dla znaczników HTML w tekście dynamicznie ładowanym. Najważniejszą innowacją było wprowadzenie języka programowania ActionScript. Jest to język skryptowy oparty o standard ECMAScript. Język ten umożliwia programowanie obiektowe i strukturalne. Dzięki wprowadzeniu tego języka do technologii Flash możemy tworzyć nie tylko interaktywne animacje ale również gry i aplikacje internetowe. Wersja siódma Macromedia Flash przyniosła ze sobą nową, drugą wersję języka ActionScript. Dodano również bibliotekę efektów osi czasu, które umożliwiają szybkie dodawanie efektów do określonych obiektów. Wprowadzano również komponenty do obsługi danych. Wersja siódma została wydana pod nazwą Macromedia Flash 2004. Ostanią wersją Flash'a wydaną pod szyldem Macromedia była wersja ósma. Ulepszono istniejące już funkcje, dodano nowe efekty, filtry. Znacznie poszerzono wachlarz właściwości i możliwości animacji. Dodano zawansowaną obsługę antyaliasingu. Macromedia w 2005 roku, zostało przejęte przez firmę Adobe.

W 2007 r. została wydana wersja 9 Flash'a nazwana Adobe Flash CS3. Dodano do niej wsparcie dla trzeciej wersji języka ActionScript. Firma Adobe wprowadziła do swojego programu możliwość integracji z innymi swoimi produktami takimi jak Adobe Photoshop, Adobe Ilustrator itp. W najnowszej wersji oprogramowania wydanej pod nazwą Adobe Flash CS4 dodano możliwość tworzenia i manipulowania podstawowymi obiektami trójwymiarowymi. Dodano również nowe funkcje do języka ActionScript w wersji trzeciej.

Przy tworzeniu pracy dyplomowej używano alternatywnego oprogramowania jakim jest SWiSH Max w wersji 2. Program ten jest prostszy i szybszy w tworzeniu animacji w technologii Flash. SWiSH nie wspiera kilku możliwości Adobe Flash takich jak ActionScript w wersji trzeciej, jednakże do zastosowania w animacjach okazuje się być wręcz wzorowym narzędziem. Zawiera rozbudowaną bazę efektów jakie możemy przypisać obiektom. Posiada

również bardzo dużą liczbę gotowych komponentów do wykorzystania w animacjach. Język skryptowy wykorzystywany w SWiSH jest zgodny z wersją drugą ActionScript.

Celem tej pracy dyplomowej było stworzenie zestawu kilku animacji komputerowych z zakresu elektromagnetyzmu. Animacje te powinny być interaktywne i przedstawione w efektowny sposób aby przyciągać i zainteresować użytkownika. Interaktywność tych animacji nie powinna być ograniczona do zespołu przycisków umożliwiających odtworzenie lub zatrzymanie animacji. Użytkownik musi mieć wrażenie, jakby miał do czynienia z prawdziwym układem doświadczalnym, na który ma wpływ np. poprzez zmianę szybkości przesuwania magnesu lub zamykania odpowiednich obwodów elektrycznych. Takie podejście zainteresuje go i sprawi że będzie chciał zagłębić się w dane zagadnienie. Rozwój kolorowej telewizji, multimediów, komputerów i Internetu spowodował że większość informacji trafia do odbiorców w postaci grafiki, filmów i różnego rodzaju animacji. Aby zainteresować człowieka, siedzącego przed monitorem komputera, treść jaką mu przekażemy musi być podana w efektowny, przejrzysty i treściwy sposób. Wypisanie setki zawiłych wzorów i dowodów sprawi że człowiek, który używa apletu zacznie się nudzić i będzie wolał spędzić ten czas na poszukiwaniach ciekawiej przedstawionej informacji. Większość ludzi jest wzrokowcami, co oznacza, że lepiej zapamiętują informacje w postaci obrazów dlatego ładna grafika i animacje przykują wzrok oglądającego, a w połączeniu z tekstem stworzą ciekawy środek dydaktyczny, którego oglądanie powoduje że użytkownik coś z niego wyniesie.

Dobrze zaprojektowana grafika znacznie bardziej poprawi czytelność nie tylko samej animacji ale również przesłania jakie będzie jej towarzyszyło. Nadając obiektom animacji efekt trójwymiarowości, użytkownik będzie widział że np: jedne kable przechodzą za magnesem a drugie przed lub jeden wektor skierowany jest za płaszczyznę ekranu a drugi przed. Animacje trójwymiarowe zostały zrealizowane w programie o nazwie Swift3D (wersja sześć) firmy Electric Rain. Kolejnym ważnym elementem przy projektowaniu pracy dyplomowej było to aby aplety nie zawierały zbyt dużo tekstów. Chodziło o to aby pokazać doświadczenia związane z elektromagnetyzmem, a nie uczyć fizyki osobę, która będzie używała apletów. Opisy doświadczeń zamieszczonych w pracy dyplomowej zostały zminimalizowane pod względem stosowania wzorów i trudnych pojęć fizycznych. U dołu każdego apletu znajduje się lista odnośników, w których czytający znajdzie więcej informacji na temat danego zagadnienia. Samo stworzenia apletów mijałoby się z celem, gdyby nie powstał główny interfejs który połączył by je w jeden spójny aplet.

Przy tworzeniu tego interfejsu kierowano się jego przejrzystością, intuicyjnością jak również tym, aby był on efektowny. Przejrzystość przejawia się w tym, że podzielono główny interfejs na kilka części nagłówek, menu, część do której wczytywana będzie zawartość, oraz stopka. Oprócz podziału interfejsu na części dobrano barwy animacje w taki sposób aby nie dekoncentrowały użytkownika, a zarazem aby były przyjemne dla oka i efektowne. Intuicyjność interfejsu cechuje się tym że zastosowane ikony mają symbole odwzorowujące przeznaczenie. Aby ułatwić użytkownikowi przeglądanie apletów, wchodzących w skład pracy dyplomowej, dodano do każdego z nich pomoc, w której opisano jak zmieniać i ustawiać parametry animacji. Podczas projektowania apletów ważnym czynnikiem była możliwość ich modyfikacji i dostosowywania. Uzyskano to dzięki dynamicznemu wczytywaniu poszczególnych apletów po kliknięciu na odpowiedni przycisk. Treść wczytywana do apletów umieszczona jest w pliku tekstowym i za pomocą specjalnych skryptów wczytywana jest do animacji. Umożliwia to łatwą modyfikację treści jak również poprawianie ewentualnych błędów. Treść dynamicznie wczytywana do animacji może być odpowiednio formatowana za pomocą plików styli CSS, które również są wczytywane za pomocą skryptów i przypisywane odpowiednim polom tekstowym. Aplety zostały wyposażone w preloadery, czyli paski które informują użytkownika o postępie ładowania danego pliku. Jest to użyteczna funkcja gdy aplety są wczytywane z serwera na komputer

użytkownika, gdyż informują go że wszystko działa poprawnie i jak długo ma czekać na załadowanie animacji. Ostatnim i chyba najważniejszym celem, jaki przyświecał podczas tworzenia pracy dyplomowej była pełna zgodność animacji z obowiązującymi prawami fizyki. Konsultacje z Promotorem doprowadziły do tego że zgodność taką uzyskano w stu procentach.

Układ pracy jest następujący. Zaraz po wstępie znajdują się 4 rozdziały teoretyczne, prezentujące teorię, jaka została wykorzystana do tworzenia apletów. Rozdział drugi opisuje zagadnienia z elektrostatyki. Składa się z czterech podrozdziałów: „Ładunek elektryczny" – opisuje zagadnienia związane z ładunkami elektrycznymi, „Prawo Coulomba" – rozdział związany z zagadnieniem prawa Coulomba, „Pole elektryczne" – opowiada o polach elektrycznych, „Prawo Gaussa" – w tym rozdziale znajdują się wszelkie zagadnienia związane z prawem Gaussa. Trzeci rozdział zatytułowany „Prąd elektryczny" wyjaśnia zagadnienia związane z prądem elektrycznym np. gęstość prądu.

Czwarty rozdział „Magnetyzm" porusza tematykę magnetyzmu. Składa się z trzech podrozdziałów: „Pole magnetyczne, siła Lorenza i indukcja pola magnetycznego", „Prawo Ampere'a" oraz „Prawo Biota-Savarta". Ostatni rozdział teoretyczny zatytułowany „Indukcja" opisuje zagadnienia związane ze zjawiskiem indukcji. Składa się z trzech podrozdziałów : „Prawo indukcji Faraday'a", „Reguła Lenz'a" oraz „Zjawisko indukcji" – podrozdział ten pokazuje zjawisko indukcji w sensie ilościowym oraz wyjaśnia zjawisko samoindukcji. Pracę kończy rozdział podsumowujący.

Do pracy dołączono również płytę CD na której znajdują się pliki źródłowe jak również gotowe animacje.

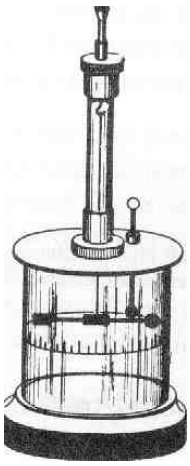
2. Elektrostatyka

2.1. Ładunek elektryczny

Ładunek elektryczny jest podstawową cechą materii. Cecha ta objawia się podczas oddziaływań pomiędzy naładowanymi ciałami. Rozróżniamy dwa rodzaje ładunków: ładunki dodatnie i ładunki ujemne. Ciała które posiadają podobne ilości ładunków ujemnych i dodatnich nazywamy ciałami elektrycznie obojętnymi. Ciała naładowane to takie, których ładunek jest niezrównoważony. Ładunek elektryczny ma postać kwantowaną tzn. jest wielokrotnością pewnej minimalnej wielkości. Wielkość tą nazywamy ładunkiem elementarnym i wynosi ona 1,60217653(14) * 10^{-19} C. Jak pokazują doświadczenia, materia składa się z trzech podstawowych cząstek elementarnych : protonów, elektronów i neutronów. Protony posiadają ładunek dodatni, elektrony ładunek ujemny, neutrony zaś są elektrycznie obojętne. Jednostką ładunku jest kulomb, czyli liczba ładunków jaka przechodzi przez przekrój poprzeczny przewodnika w czasie jednej sekundy.

2.2. Prawo Coulomba

Prawo Coulomba jest jednym z podstawowych praw rządzących w elektrostatyce. Mówi ono o wartościach sił oddziaływań pomiędzy naładowanymi elektrycznie ciałami. Zostało odkryte przez Charlesa Austina Coulomba w 1785 r. Coulomb był francuskim fizykiem urodzonym 14 czerwca 1736 roku w Angouelme. Zajmował się badaniem magnetyzmu, elektrostatyki i teorii maszyn prostych. Zmarł 23 sierpnia 1806 roku w Paryżu [1]. Coulomb do odkrycia swojego prawa wykorzystał wagę skręceń (Rys. 1.). Wykonując



Rys. 1. Waga skręceń [2].

doświadczenia Coulomb odkrył, że siła między dwoma ładunkami jest odwrotnie proporcjonalna do kwadratu odległości między tymi ładunkami, siła ta również jest proporcjonalna do iloczynu tych ładunków. Ostatecznie prawo to przyjmuje postać taką jak na wzorze (1). Symbole q_1 i q_2 to wartości oddziałujących między sobą. Symbol r wyraża

$$F = k\frac{|q_1||q_2|}{r^2} \qquad (1)$$

odległość pomiędzy oddziałującymi ładunkami. Stała k natomiast wynosi 8,99 $\cdot 10^9$ N · m^2/C^2 i wyraża się wzorem (2). Wielkość ε_0 we wzorze (2) nazywana jest przenikalnością elektryczną próżni i wynosi 8,85 $\cdot 10^{-12}$ $C^2/(N \cdot m^2)$. Wektorowo prawo Coulomba

$$k = \frac{1}{4\pi\varepsilon_0} \qquad (2)$$

przedstawione jest za pomocą wzoru (3), gdzie kierunek wektora siły zależy od kierunku wektora **r**, natomiast zwrot zależy o znaku oddziałujących ładunków.

$$\mathbf{F} = k\frac{q_1 q_2}{r^2}\frac{\mathbf{r}}{r} \qquad (3)$$

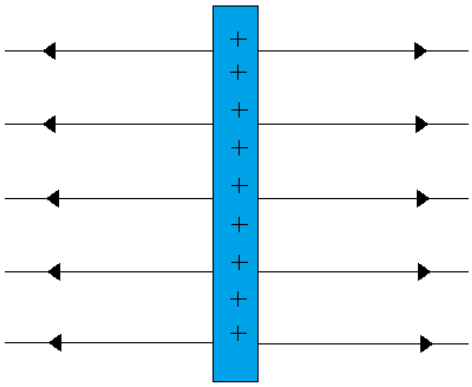
2.3 Pole elektryczne

Pole elektryczne jest to właściwość przestrzeni, która otacza ciała naładowane. W obszarze tym działają siły elektryczne. Aby znaleźć pole elektryczne w pobliżu pewnego punktu, musimy wprowadzić w pobliże tego punktu ładunek próbny q i zmierzyć siłę jaką pole elektryczne działa na ładunek. Wówczas natężenie takiego pola będziemy wyrażać wzorem (4). Jest to wielkość wektorowa, kierunek jest zgodny z kierunkiem siły F.

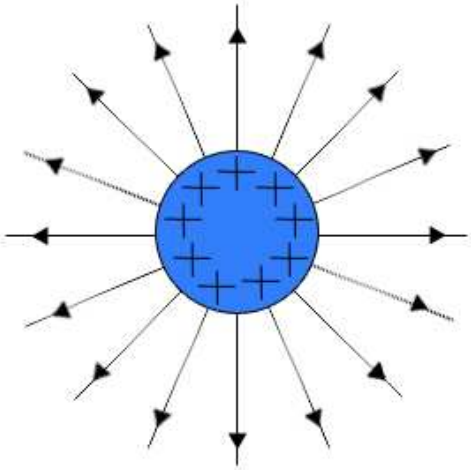
$$\mathbf{E} = \frac{\mathbf{F}}{q} \qquad (4)$$

Aby lepiej zobrazować pole elektryczne używamy pojęcia linii pola elektrycznego. Linia styczna w dowolnym punkcie do linii pola elektrycznego umożliwia wyznaczenie kierunku wektora pola elektrycznego **E**. Większa gęstość linii pola oznacza, że wartość natężenia pola jest większa niż tam gdzie gęstość tych linii jest mniejsza. Gdy umieścimy ładunek np. dodatni w polu elektrycznym płyty (Rys. 2.) to będzie on (w zależności od tego jakim ładunkiem będzie naładowana płyta) prostopadle przyciągany do płyty lub od niej odpychany. Pole takiej płyty jest polem jednorodnym, gdyż linie pola elektrycznego są rozłożone równomiernie.

Dodatnio naładowana kula będzie wytwarzać pole takie jak na rysunku 3. Jeżeli umieścimy ładunek dodatni będzie on odpychany wzdłuż przedłużenia promienia kuli. Pole takie zanika wraz z odległością od środka kuli.



Rys. 2. Linie pola elektrycznego dla nieskończenie naładowanej dodatnio płyty, ładunek próbny umieszczany w polu elektrycznym również jest dodatni.



Rys. 3. Linie pola dla dodatnio naładowanej kulki, ładunek próbny umieszczany w polu elektrycznym również jest dodatni.

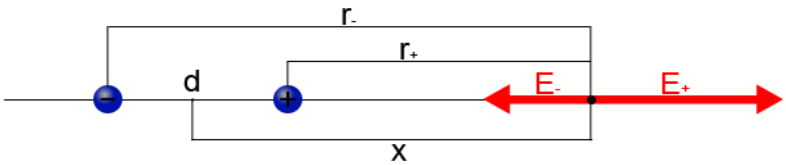
Pole elektryczne dla ładunku punktowego wyznaczamy poprzez umieszczenie ładunku próbnego (q_p) w odległości r od ładunku, dla którego chcemy wyznaczyć pole (q_z). Na początku należy wyznaczyć siłę elektrostatyczną jaka działa na ładunek próbny q_p. Wartość tej siły obliczamy ze wzoru (5). Następnie korzystając ze wzoru (4), dzielimy wartość siły

$$F = \frac{1}{4\pi\varepsilon_0}\frac{|q_p||q_z|}{r^2} \qquad (5)$$

elektrostatycznej przez wartość ładunku punktowego (wzór 6) i otrzymujemy wartość pola elektrycznego w pobliżu ładunku punktowego q_z.

$$E = \frac{F}{q_p} = \frac{1}{4\pi\varepsilon_0}\frac{|q_z|}{r^2} \qquad (6)$$

Dla dipola elektrycznego (przedstawionego na rysunku 4) czyli 2 ładunków różnoimiennych znajdujących się w odległości d, pole elektryczne poszukujemy w odległości x od środka dipola. Odległość ładunku ujemnego od punktu, w którym wyznaczamy pole wynosi r_- natomiast odległość ładunku dodatniego wynosi r_+. W punkcie, w którym wyznaczamy pole umieszczamy dodatni ładunek próbny. Wypadkowe pole elektryczne jest sumą pól elektrycznych wytworzonych przez poszczególne ładunki dipola. Zależność tą możemy przedstawić za pomocą wzoru 7 gdzie po przekształceniach otrzymamy wzór 8.



Rys. 4. Dipol elektryczny.

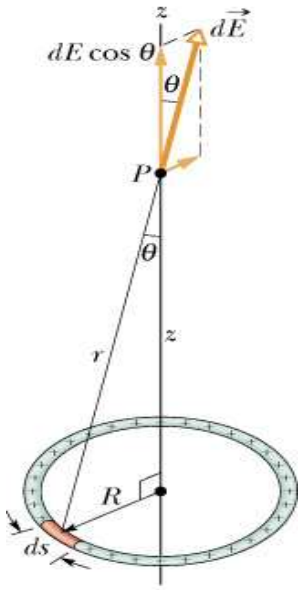
$$E = E_+ - E_- = \frac{1}{4\pi\varepsilon_0}\frac{q}{r_+^2} - \frac{1}{4\pi\varepsilon_0}\frac{q}{r_-^2} = \frac{1}{4\pi\varepsilon_0}\frac{q}{(x-\frac{1}{2}d)^2} - \frac{1}{4\pi\varepsilon_0}\frac{q}{(x+\frac{1}{2}d)^2} \tag{7}$$

$$E = \frac{q}{4\pi\varepsilon_0 x^2}\left[\left(1-\frac{d}{2x}\right)^{-2} - \left(1+\frac{d}{2x}\right)^{-2}\right] \tag{8}$$

Dla odległości x dużo większych od d otrzymujemy wzór (9).

$$E = \frac{qd}{2\pi\varepsilon_0 x^3} \tag{9}$$

Pierścień na rysunku 5 ma promień R i posiada liniową gęstość ładunku λ. Pierścień naładowany jest ładunkiem dodatnim. Szukamy pola elektrycznego w punkcie P. W celu znalezienia tego pola nie skorzystamy ze wzoru (6) gdyż mija się to z celem ponieważ



Rys. 5. Pierścień naładowany jednorodnie ładunkiem dodatnim [3].

pierścień z rysunku 5 nie jest ładunkiem punktowym. Jednakże zagadnienie to można rozwiązać w sposób taki, że dzielimy obiekt na bardzo małe części co sprowadzi nas do zagadnienia układu pojedynczych ładunków punktowych, gdzie będziemy mogli wykorzystać wzór (6). Następnie zsumujemy poszczególne ładunki punktowe i otrzymamy wynikowe pole elektryczne. Możemy napisać że :

$$dq = \lambda ds \tag{10}$$

Ładunek zadany wzorem (10) wytwarza cząstkowe pole wyrażone wzorem (11).

$$dE = \frac{1}{4\pi\varepsilon_0}\frac{dq}{r^2} = \frac{1}{4\pi\varepsilon_0}\frac{\lambda ds}{r^2} = \frac{1}{4\pi\varepsilon_0}\frac{\lambda ds}{(z^2+R^2)} \tag{11}$$

Jak widać z rysunku 5 punktowe ładunki pochodzące od pierścienia wytwarzają pole elektryczne, którego składowe prostopadłe do osi z się znoszą. Przyczynek do pola elektrycznego wnoszą tylko składowe równoległe do osi z, które mają ten sam kierunek i wartość. Wartość każdej takiej składowej wynosi dEcosΘ. Gdzie cosΘ można wyrazić wzorem (12). Ze wzoru (12) wynika zależność (13).

$$\cos\theta = \frac{z}{r} = \frac{z}{(z^2 + R^2)^{\frac{1}{2}}} \qquad (12)$$

$$dE\cos\theta = \frac{z\lambda}{4\pi\varepsilon_0 (z^2 + R^2)^{3/2}} ds \qquad (13)$$

Teraz należy wszystkie pola elektryczne wytwarzane przez „ładunki punktowe" dodać do siebie. Zrobimy to całkując poszczególne składowe równoległe pola elektrycznego wytwarzanego przez ładunki. Całkowanie odbywa się po zmiennej ds i obwodzie pierścienia czyli dla s od 0 do 2πR. Otrzymujemy wtedy wzór (14).

$$E = \int dE\cos\theta = \frac{z\lambda}{4\pi\varepsilon_0 \left(z^2 + R^2\right)^{2/3}} \int_0^{2\pi R} ds = \frac{z\lambda(2\pi R)}{4\pi\varepsilon_0 \left(z^2 + R^2\right)^{2/3}} = \frac{zq}{4\pi\varepsilon_0 \left(z^2 + R^2\right)^{2/3}} \qquad (14)$$

2.4 Prawo Gaussa

Strumień pola elektrycznego przepływającego przez powierzchnię S wyraża się wzorem (15). Jest to wielkość skalarna której jednostka to N·m^2/C. Powierzchnia po której całkujemy jest powierzchnią Gaussa, która to jest zamknięta. Powierzchnia ta dzielona jest na nieskończenie małe kawałeczki dS. Liczymy iloczyn skalarny d**S** z polem **E**, a następnie całkujemy po całej powierzchni.

$$\Phi = \oint \vec{E} \cdot d\vec{S} \qquad (15)$$

Prawo Gaussa, przedstawione na wzorze (16), jest określeniem związku pomiędzy strumieniem pola elektrycznego przechodzącego przez daną powierzchnię Gaussa (S) a całkowitym ładunkiem który znajduje się na tej powierzchni. Prawo Gaussa jest równoważne

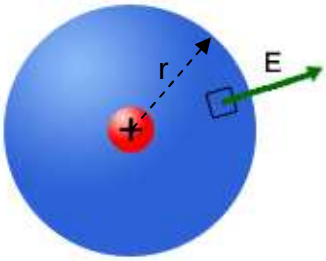
$$\varepsilon_0 \Phi = q_{cal} \qquad (16)$$

prawu Coulomba. Można to pokazać na przykładzie pewnej powierzchni kulistej (Rys.6.), w środku której znajduje się dodatni ładunek q. Natężenie pola jest skierowane na zewnątrz kuli i prostopadłe do infinitezymalnie małego wycinka powierzchni. Pole jest również równoległe do promienia kuli z czego wynika że kąt pomiędzy wektorem **E** i d**S** jest równy 0. Korzystając ze wzoru na prawo Gaussa otrzymamy zależność przedstawioną na wzorze (17).

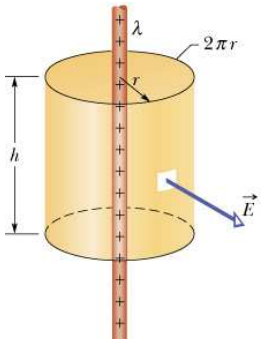
$$\varepsilon_0 \oint \mathbf{E} \cdot d\mathbf{S} = \varepsilon_0 \oint E \cdot dS = q \qquad (17)$$

Pole powierzchni sfery wynosi 4πr^2. Podstawiając to do wzoru i przekształcając otrzymamy wzór (18).

$$E = \frac{1}{4\pi\varepsilon_0} \frac{q}{r^2} \qquad (18)$$



Rys. 6. Kulista powierzchnia Gaussa otaczająca dodatni ładunek.



Rys. 7. Dodatnio naładowana cienka linia wraz z otaczającą ją powierzchnią Gaussa [3].

Jeżeli rozważymy nieskończenie długi przewodnik o kształcie walca, który jest naładowany jednorodnie dodatnim ładunkiem, to powierzchnia Gaussa będzie miała symetrię walcową (rys. 7). Kierunek wektora pola elektrycznego jest równoległy do promienia walca powierzchni Gaussa, zatem z tego wynika że strumień pola pochodzący od podstaw tego walca jest równy 0. Interesuje nas zatem tylko pole skierowane wzdłuż promienia walca, wyrażone wzorem (19). Człon 2πrh jest polem powierzchni walca. Walcowa powierzchnia Gaussa obejmuje ładunek o rozkładzie wyrażonym wzorem (20), podstawiając wzór (19) i (20) do wzoru (16) i przekształcając otrzymamy wzór (21) opisujący natężenie pola od naładowanej linii.

$$\Phi = E 2\pi r h \qquad (19)$$

$$q_{cal} = \lambda h \qquad (20)$$

$$E = \frac{\lambda}{2\pi\varepsilon_0 r} \qquad (21)$$

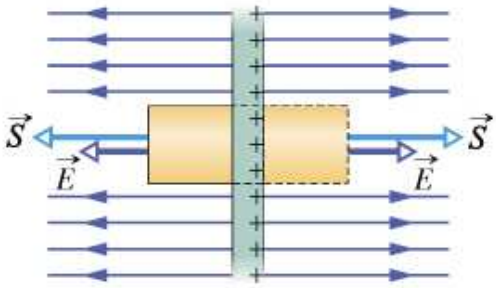
Kolejny przypadkiem, w którym możemy wykorzystać powierzchnię Gaussa w kształcie cylindra, jest nieskończona naładowana płaszczyzna (Rys. 8.). W tym wypadku mamy doczynienia z gęstością powierzchniową ładunku σ. Jak już wcześniej wspomnieliśmy powierzchnia Gaussa to cylinder umieszczony w płaszczyźnie tak, że przecina ją prostopadle. Strumień pola elektrycznego przechodzący przez powierzchnie boczną jest równy zero gdyż wektor pola elektrycznego i wektor powierzchni są ustawione do siebie prostopadle. Interesować nas będzie zatem strumień przechodzący przez denka walcowej powierzchni Gaussa (wzór (22)). Podstawiając wzór (15) do wzoru (22) otrzymamy wzór (23). Prawo Gaussa przyjmie teraz postać taką jak we wzorze (24), a po przekształceniach otrzymamy wzór (25) czyli pole elektryczne dla naładowanej płaszczyzny.

$$\varepsilon_0 \Phi = \sigma S \tag{22}$$

$$\varepsilon_0 \oint \mathbf{E} d\mathbf{S} = \sigma S \tag{23}$$

$$\varepsilon_0 (ES + ES) = \sigma S \tag{24}$$

$$E = \frac{\sigma}{2\varepsilon_0} \tag{25}$$



Rys. 8. Dodatnio naładowana płaszczyzna wraz z powierzchnią Gaussa [3].

3. Prąd elektryczny

W zwykłym przewodniku, elektrony poruszają się swobodnie i w nieuporządkowany sposób. Średnio tyle samo elektronów porusza się w lewo i tyle samo w prawo co daje całkowity bilans równy zero. Jeżeli do przewodnika zostanie podłączone źródło prądu, to ruch tych elektronów stanie się uporządkowany za sprawą działającego wewnątrz przewodnika pola elektrycznego. Ładunki będą się poruszać zgodnie z kierunkiem pola elektrycznego. Taki uporządkowany ruch nazywamy prądem. Prąd elektryczny możemy zdefiniować wzorem (26). Jednostką natężenia prądu u układzie SI jest Amper

$$I = \frac{dq}{dt} \qquad (26)$$

Wielkością związaną z prądem elektrycznym jest gęstość prądu elektrycznego **j**. Gęstość prądu jest wielkością wektorową. Jest używana do opisu prądu przepływającego w wybranym punkcie np: przewodnika. Gęstość prądu wiąże zależność (27), mówi ona że natężenie prądu I jest strumieniem gęstości prądu przez powierzchnię S. Jeżeli powierzchnia

$$I = \int \mathbf{j} d\mathbf{S} \qquad (27)$$

S, po której będziemy całkować będzie przekrojem poprzecznym przewodnika i gęstość prądu będzie stała na tej powierzchni, a jej kierunek prostopadły do powierzchni to wartość wektora gęstości prądu można zapisać za pomocą wzoru (28).

$$j = \frac{I}{S} \qquad (28)$$

Jeżeli nie przykładamy żadnego pola elektrycznego, to ruch ładunków w przewodniku jest przypadkowy (podobnie jak cząsteczek gazu). Średnia prędkość elektronów w takim ruchu wynosi zero. Dopiero gdy przyłożymy pole elektryczne do przewodnika, elektrony zaczną poruszać się z prędkością v_d. Wielkość v_d nosi nazwę prędkość dryfu. W łatwy sposób można wyznaczyć prędkość dryfu, jeżeli założymy ze jest ona stała dla wszystkich elektronów, oraz że gęstość prądu elektrycznego ma te same właściwości w całym polu przekroju przewodnika (S). Sumaryczny ładunek w przewodniku o długości L wynosi:

$$q = nSLe \qquad (29)$$

Natężenie prądu przepływającego przez przewodnik możemy przy pomocy wzoru (26) zapisać jako :

$$I = \frac{nSLe}{t} \qquad (30)$$

Czas znajdujący się w mianowniku równania (30) możemy wyznaczyć ze wzoru (31). Czynimy tak zakładając, że ładunki płyną wzdłuż przewodnika (o długości L) ze stałą prędkością v_d.

$$t = \frac{L}{v_d} \qquad (31)$$

Po podstawieniu równania (31) do równania (30) otrzymamy równanie (32) co po przekształceniach i po zastosowaniu równości (28) prowadzi do wzoru (33).

$$I = \frac{nSLe}{L/v_d} = nSev_d \qquad (32)$$

$$v_d = \frac{j}{ne} \qquad (33)$$

Jest to związek między gęstością prądu j a prędkością *unoszenia* v_d ładunków. O ile prędkość unoszenia łatwo pokazać za pomocą animacji, gęstość prądu nie jest łatwa do przestawienia. Jednakże wzór (33) ma ważne znaczenie na przykład do ilustracji prądu w półprzewodnikach, gdzie gęstość ładunków (nośników prądu) jest mniejsza niż w metalach.

4. Magnetyzm

4.1. Pole magnetyczne, siła Lorenza i indukcja pola magnetycznego

Naelektryzowane ciało w swoim otoczeniu wytwarzało pole elektryczne, czyli otoczenie w którym działały siły elektryczne. Magnesy oraz przewodniki z prądem wytwarzają pole magnetyczne, czyli obszar w którym działają siły magnetyczne. Pole magnetyczne jest wielkością wektorową. Podstawową wielkością określającą pole magnetyczne jest indukcja magnetyczna **B**.

Na poruszające się cząstki naładowane w polu magnetycznym działa określona siła zwana siłą Lorentza. Siła ta działa na ładunek próbny w taki sposób że odchyla tor jego ruchu. Zatem siła ta będzie działać prostopadle do wektora prędkości cząstki. Doświadczenia pokazują również że można znaleźć taki kierunek wektora prędkości cząstki, że na poruszający się ładunek będzie działać siła Lorentza równa zero. Kierunek ten wyznacza kierunek wektora indukcji pola magnetycznego **B**. Dla wszystkich kierunków wektora prędkości cząstki, różnych od kierunku wektora **B**, siła Lorenza jest proporcjonalna do wektora prędkości **v** oraz do sinusa kąta pomiędzy kierunkiem wektora **B** i wektora **v**. Siła Lorenza jest również proporcjonalna do ładunku próbnego q. Powyższe obserwacje możemy zapisać w postaci wzoru (34).

$$F_B = qvB\sin\phi \qquad (34)$$

Wektorowo równanie (34) możemy zapisać w postaci przestawionej we wzorze (35).

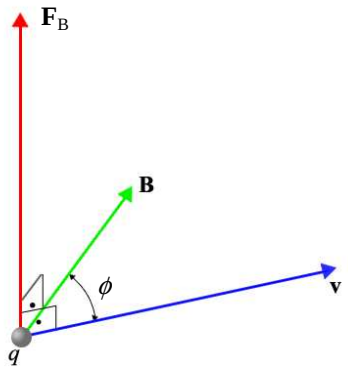
$$\mathbf{F}_B = q\mathbf{v}\times\mathbf{B} \qquad (35)$$

Z równania (35) że gdy prędkość cząstki jest równa zeru to siła Lorenza również jest równa zero. Kierunek wektora siły zależy od znaku ładunku próbnego. Jeżeli ϕ jest równe 90° to wzór (34) przyjmie postać (36) a po przekształceniach otrzymamy zależność (37) definiującą wartość bezwzględną wektora **B**.

$$F_B = qvB \qquad (36)$$

$$B = \frac{F_B}{qv} \qquad (37)$$

Na rysunku 9 przedstawiono graficznie zależności pomiędzy wektorami **B**, $\mathbf{F}_B$ i **v**.

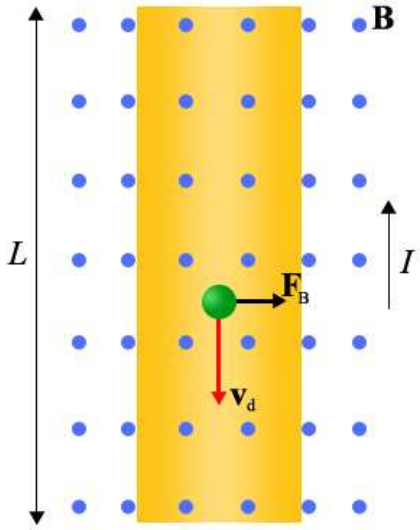


Rys. 9. Zależności pomiędzy wektorami **B**, $\mathbf{F}_B$ i **v**

Analogicznie do wzoru (15) strumień pola magnetycznego wyraża się wzorem (38).

$$\Phi_B = \oint \mathbf{B} \cdot d\mathbf{S} \qquad (38)$$

Na pojedynczy ładunek poruszający się w polu magnetycznym działa siła magnetyczna odchylająca tor jego ruchu. Skoro więc pojedynczy ładunek jest odchylany to również przewodnik z prądem, w którym płynie takich ładunku wiele, powinien być odchylany. Na rysunku 10 znajduje się obraz tego, co dzieje się w przewodniku.



Rys. 10. Obraz tego co dzieje się we wnętrzu przewodnika umieszczonego w polu magnetycznym

Elektrony poruszają się z prędkością unoszenia w dół przewodnika. Na rysunku zaznaczono tylko jeden taki elektron jednakże w przewodniku takich elektronów będzie więcej. Wektor indukcji pola magnetycznego B jest skierowany w naszą stronę. Dla każdego elektronu przewodnictwa siła magnetyczna będzie wyrażona wzorem (36). Ilość elektronów przewodnictwa przypadająca na jednostkę objętości wynosi n. Podstawiając we wzorze (36) za v_d wyrażenie ze wzoru (33) otrzymamy zależność (39). Wiemy że przewodnik, który ma

$$F = \frac{jB}{n} \qquad (39)$$

długość L, posiada *nSL* elektronów przewodzenia (S jest polem przekroju poprzecznego przewodnika). Zatem siła która działa na wszystkie elektrony przewodzenia wyraża się wzorem (40).

$$F_c = (nSL)F = (nSL)\frac{jB}{n} \qquad (40)$$

Dodatkowo korzystając ze wzoru (28) możemy przekształcić wzór (40) do wzoru (41), a w ogólności do wzoru (42) (gdzie L to wektor skierowany wzdłuż przewodnika o wartości bezwzględnej równej L).

$$F_c = ILB \qquad (41)$$

$$\mathbf{F}_c = I\mathbf{L} \times \mathbf{B} \qquad (42)$$

Cewka, przez którą płynie prąd umieszczona w polu magnetycznym będzie zachowywać się jak igła magnetyczna. Igła magnetyczna i magnes sztabkowy są przykładami dipola magnetycznego. Dipol magnetyczny to układ dwóch różnoimiennych biegunów magnetycznych umieszczonych blisko siebie. Igła kompasu umieszczona w polu magnetycznym zaczyna się obracać. Skoro igła kompasu jest dipolem to i cewka będzie się obracać. Skręcanie odbywa się pod wpływem momentu magnetycznego wyrażonego wzorem (43). N jest liczbą zwojów w ramce, I jest natężeniem prądu jaki płynie w ramce natomiast S jest polem powierzchni obejmowane przez każdy zwój cewki. Moment siły działający na taką cewkę wyrażony będzie wzorem (44). Co z kolei wektorowo możemy zapisać w postaci (45).

$$\mu = NIS \qquad (43)$$

$$M = \mu B \sin\theta \qquad (44)$$

$$\mathbf{M} = \boldsymbol{\mu} \times \mathbf{B} \qquad (45)$$

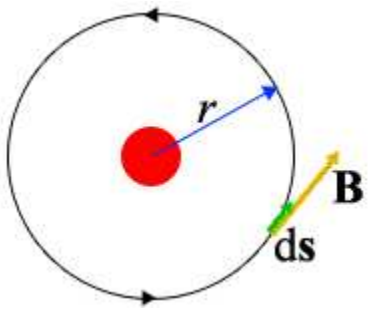
4.2. Prawo Ampere'a

Prawo Ampere'a (wzór (46)) opisuje nam zależność ilościową pomiędzy polem magnetycznym a natężeniem prądu. Prawo to zostało sformułowane tak jak na wzorze (46) przez Jamesa Clerka Maxwella.

$$\oint \mathbf{B} \cdot d\mathbf{s} = \mu_0 I \qquad (46)$$

Stała μ_0 jest przenikalnością magnetyczną próżni której wartość jest równa $4\pi \cdot 10^{-7}$ T · m/A.

Na rysunku 11 mamy do czynienia z długim, cylindrycznym przewodnikiem w którym płynie prąd. Aby obliczyć indukcje pola magnetycznego wykorzystując prawo Ampere'a



Rys. 11. Przewodnik cylindryczny (zaznaczony na czerwono) wraz z współśrodkowym konturem całkowania.

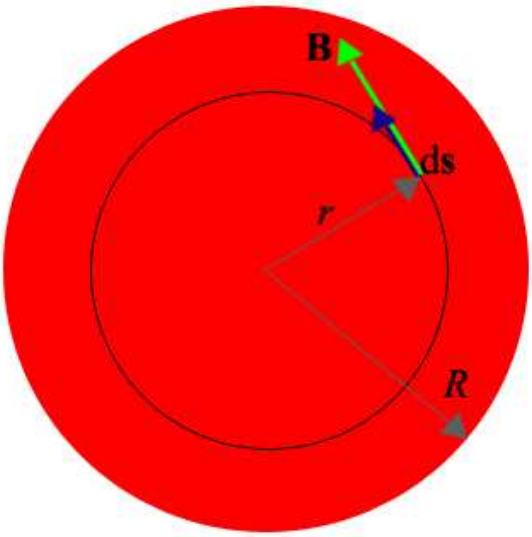
musimy otoczyć przewodnik zamkniętym konturem w kształcie okręgu, o środku w tym samym punkcie co środek przewodnika. Drogę całkowania dzielimy na bardzo małe elementy o długości d**s**, które są styczne do drogi całkowania. Kąt pomiędzy wektorem d**s** a wektorem indukcji pola magnetycznego **B** wynosi zero stopni. Następnie obliczamy całkę w postaci takiej jak na wzorze (47). Całkujemy po konturze w kształcie okręgu dlatego we wzorze (47) występuje czynnik $2\pi r$.

$$\oint \mathbf{B} \cdot d\mathbf{s} = \oint B \cos\theta ds = B \oint ds = B\, 2\pi r \qquad (47)$$

Podstawiając wynik całkowania do wzoru (46) i przekształcając otrzymamy równanie (48).

$$B = \frac{\mu_0 I}{2\pi r} \qquad (48)$$

Prawo Ampere'a można zastosować do sytuacji przedstawionej na rysunku 12. W tym przypadku obliczamy indukcję pola magnetycznego wewnątrz przewodnika o kształcie cylindrycznym, przez który płynie prąd. Prąd płynie w kierunku przed płaszczyznę kartki, podobnie jak w poprzedniej sytuacji.



Rys. 12. Przekrój poprzeczny cylindrycznego przewodnika wraz z drogą całkowania (czarny okrąg).

W przypadku gdy r < R prawo Ampere'a zapiszemy za pomocą wzoru (49). Gdzie *I'* wyrażone jest wzorem (50). Całka z lewej strony wzoru (49) wynosi tyle co we wzorze (47).

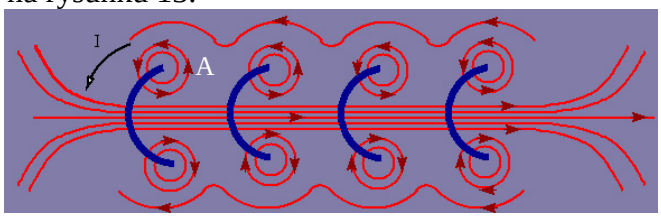
$$\oint \mathbf{B} d\mathbf{s} = \mu_0 I' \qquad (49)$$

$$I' = I \frac{\pi r^2}{\pi R^2} \qquad (50)$$

Podstawiając wszystkie wartości i przekształcając dochodzimy do wzoru (51)

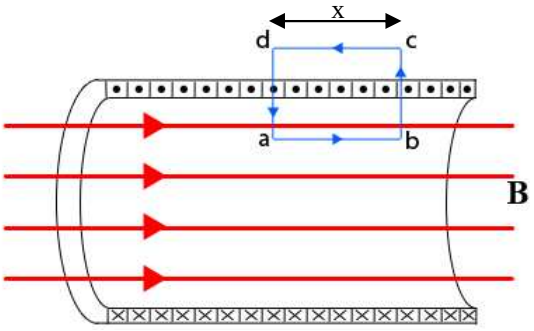
$$B = \left(\frac{\mu_0 I}{2\pi R^2} \right) r \qquad (51)$$

Prawo Ampere'a jest bardzo przydatne jeżeli rozważamy układ zwany solenoidem. Jest to długi drut, zwinięty w spiralę. Solenoid to praktycznie cewka która ma bardzo ciasno zwinięte zwoje. Gdy w solenoidzie płynie prąd pole magnetyczne wytworzone w solenoidzie wygląda podobnie jak na rysunku 13.



Rys. 13. Linie pola magnetycznego dla solenoidu z rozsuniętymi zwojami [4].

W obszarze w pobliżu literki A na rysunku 13 solenoid wytwarza pole magnetyczne podobne do zwykłego przewodnika prostoliniowego. Zwiększając odległość (w górę rysunku) od obszaru A, pole magnetyczne pola z górny części uzwojenia znoszą się częściowo z tymi pochodzącymi od dolnej części uzwojenia. Wewnątrz solenoidu pole magnetyczne jest w przybliżeniu równoległe. Model, który będziemy rozważać, to solenoid idealny (rys. 14.). Solenoid idealny ma zwoje bardzo ściśle ułożone, które mają przekrój kwadratowy. Jest on również nieskończenie długi. Zakładamy również, że pole indukcja pola magnetycznego jest równa zero poza obrębem solenoidu.



Rys. 14. Przekrój solenoidu idealnego wraz z konturem całkowania (zaznaczony na niebiesko).

Jeżeli zastosujemy prawo Ampere'a dla konturu całkowania takiego jak na rysunku 14, całkę ze wzoru (46) możemy rozpisać w postaci takiej jak na równości (52).

$$\oint \mathbf{B} d\mathbf{s} = \int_a^b \mathbf{B} \cdot d\mathbf{s} + \int_b^c \mathbf{B} \cdot d\mathbf{s} + \int_c^d \mathbf{B} \cdot d\mathbf{s} + \int_d^a \mathbf{B} \cdot d\mathbf{s} \qquad (52)$$

Całka w granicach od a do b wynosi Bx. Całki liczone w granicach od b do c i do d do a są równe zero gdyż wektor indukcji pola magnetycznego **B** jest prostopadły do wektora d**s**. Całka, która jest liczona w granicach c do d jest również równa zero, gdyż zakładaliśmy że indukcja pola magnetycznego poza obrębem solenoidu jest równa zero. Zatem możemy napisać równanie (53).

$$Bx = \mu_0 I' \qquad (53)$$

I' to natężenie prądu jakie występuje w obejmowanej, przez kontur całkowania, części solenoidu. Kontur ten obejmuje nx zwojów, zatem możemy napisać że całkowite natężenie prądu znajdujące się w obszarze obejmowanym przez kontur całkowania wyraża się wzorem (54).

$$I' = \mu_0 n x I \qquad (54)$$

Podstawiając równanie (54) do (53) i dzieląc przez x otrzymujemy równość (55).

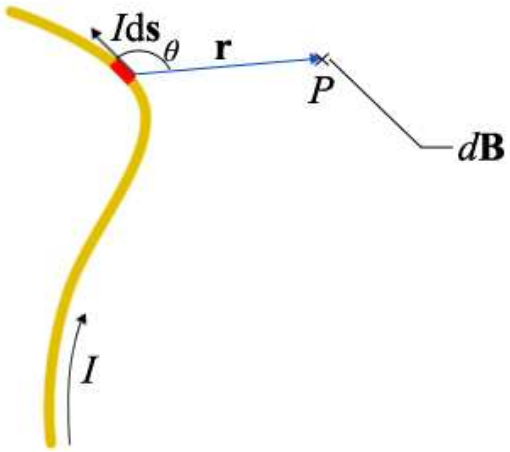
$$B = \mu_0 I n \qquad (55)$$

4.3. Prawo Biota-Savarta

Oprócz prawa Ampere'a istnieje również inne równanie, zwane prawem Biota-Savarta, które umożliwia obliczenie indukcji pola magnetycznego z dowolnego rozkładu prądu. Zwykle mamy do czynienia z sytuacjami w których rozkład prądów nie jest na tyle symetryczny aby skorzystać z prawa Ampere'a. Całka krzywoliniowa staję się wówczas trudna do obliczenia. Indukcję pola magnetycznego w punkcie P (rys. 15) będziemy wyznaczać poprzez dzielenie każdego prądu na małe kawałeczki a następnie dla tych małych kawałeczków będziemy obliczać przyczynki do indukcji pola magnetycznego dla każdego z nich, wprost z prawa Biota-Savarta. Kolejnym krokiem będzie wycałkowanie tych przyczynków dzięki czemu otrzymamy ostateczną postać wzoru na indukcję pola magnetycznego. Prawo Biota-Savarta opisuje równanie (56). Wartość wektora d**B** wygląda tak jak na wzorze (57).

$$d\mathbf{B} = \frac{\mu_0}{4\pi}\frac{Id\mathbf{s}\times\mathbf{r}}{r^3} \qquad (56)$$

$$dB = \frac{\mu_0}{4\pi}\frac{Ids\sin\theta}{r^2} \qquad (57)$$

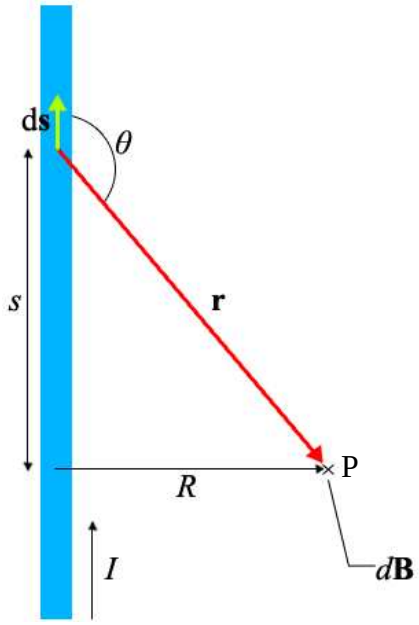


Rys. 15. Przewodnik o nieregularnym kształcie, przez który przepływa prąd, wytwarza w punkcie P pole magnetyczne o indukcji **B**.

Dzięki prawu Biota-Savarta możemy wyznaczyć indukcję pola magnetycznego pochodzącego od przewodnika prostoliniowego, przez który płynie prąd o natężeniu I (rys. 16.). Każdy element Ids przewodnika wytwarza w punkcie P wkład do całkowitej indukcji pola magnetycznego wyrażony wzorem (57). Całkowanie przyczynków indukcji pola magnetycznego dla górnej połowy przewodnika będzie się odbywało w granicach od zera do nieskończoności, gdyż mamy do czynienia z nieskończenie długim przewodnikiem. Dolna połowa przewodnika w wyniku całkowania będzie dawać te same wyniki dlatego całkę będziemy mnożyć przez dwa. Zatem możemy zapisać równanie (58). Z rysunku 16 widać że

$$B = 2\int_0^\infty dB = \frac{\mu_0 I}{2\pi}\int_0^\infty \frac{\sin\theta ds}{r^2} \qquad (58)$$

Spełnione są zależności (59) i (60). Po podstawieniu zależności (59) i (60) do całki w równaniu (58) i wyliczeniu jej otrzymamy równanie (61), które opisuje indukcje pola magnetycznego w punkcie P.



Rys. 16. Przewodnik prostoliniowy przez który płynie prąd wytwarza pole magnetyczne o indukcji **B** w punkcie P.

$$r = \sqrt{s^2 + R^2} \qquad (59)$$

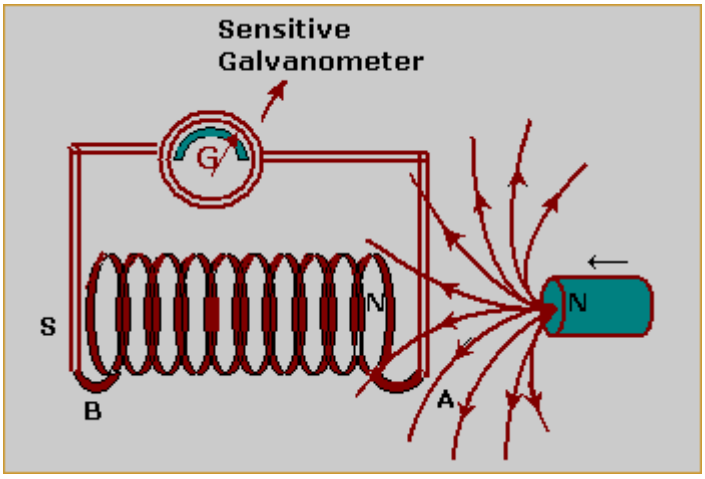
$$\sin\theta = \frac{R}{\sqrt{s^2 + R^2}} \qquad (60)$$

$$B = 2\int_0^\infty dB = \frac{\mu_0 I}{2\pi}\int_0^\infty \frac{Rds}{\left(s^2 + R^2\right)^{3/2}} = \frac{\mu_0 I}{2\pi}\left[\frac{s}{(s^2 + R^2)^{1/2}}\right]_0^\infty = \frac{\mu_0 I}{2\pi R} \qquad (61)$$

5. Indukcja

5.1. Prawo indukcji Faraday'a

Indukcja to wytwarzanie prądu elektrycznego za pomocą zmiennego w czasie pola magnetycznego. Pierwszym uczonym który doświadczalnie wykazał to stwierdzenie był Michael Faraday. Poniżej znajduje się rysunek (rys. 17.), który pokazuje przewodzące zwoje połączone z galwanometrem. Wskazówka galwanometru znajduje się w



Rys. 17. Wytwarzanie indukowanego prądu poprzez zbliżanie magnesu do układu zwojów [6].

w pozycji początkowej, ponieważ w obwodzie nie ma żadnego prądu. Gdy do zwojów będziemy zbliżać magnes wskazówka miernika zacznie się wychylać. Gdy magnes będzie odsuwany prąd zacznie płynąć w drugą stronę. Obserwacje pokazują że albo magnes albo zwoje muszą się poruszać w przeciwnym wypadku prąd nie popłynie. Zmiana bieguna na przeciwny powoduje, że podczas zbliżania magnesu prąd będzie płynął w odwrotnym kierunku niż poprzednio. Natężenie prądu jaki będzie płynął podczas zbliżania magnesu będzie tym większy im większa będzie prędkość zbliżania magnesu. Prąd, jaki wytwarza się w takich zwojach, nazywamy prądem indukowanym i jest on wywołany indukowaną siłą elektromotoryczną. Faraday ustalił że siła elektromotoryczna, jaka powstaje w obwodzie, jest zależna od szybkości zmian w czasie, strumienia pola magnetycznego (wyrażonego wzorem (38)). Możemy to zapisać za pomocą wzoru (62), gdzie N to liczba zwojów.

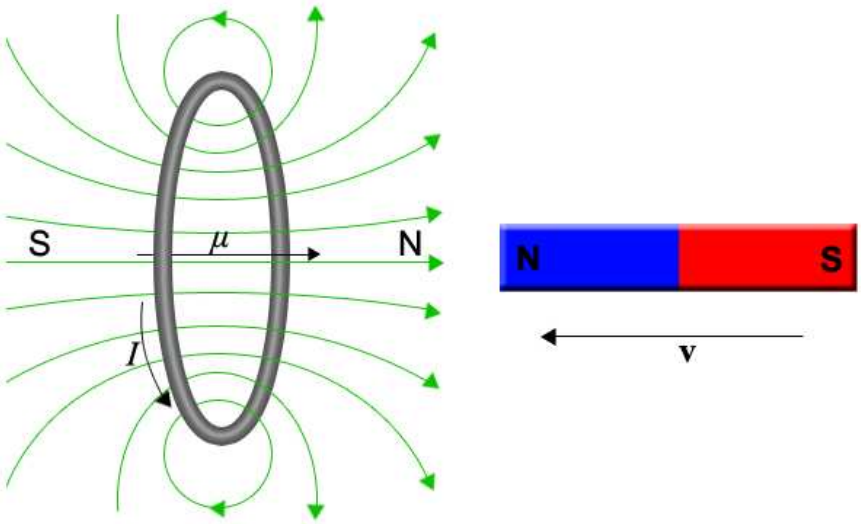
$$\mathcal{E} = -N\frac{d\Phi_B}{dt} \qquad (62)$$

Strumień magnetyczny możemy zmieniać w czasie poprzez np. rozciąganie, ściskanie, obracanie cewki lub poprzez zmianę wartości indukcji pola magnetycznego.

5.2. Reguła Lenz'a

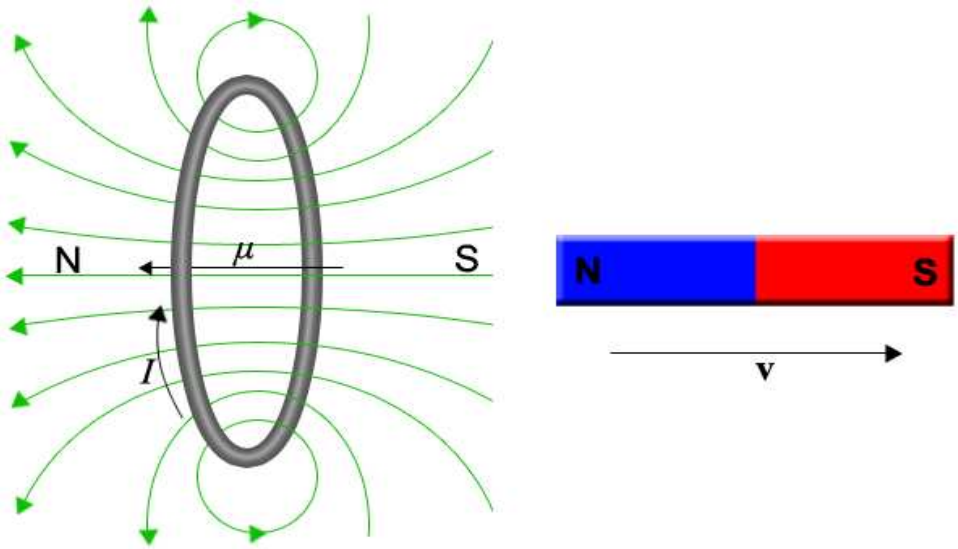
Aby wyznaczyć kierunek indukowanej siły elektromotorycznej przydatna będzie Reguła Lenz'a. Zgodnie z regułą Lenz'a prąd indukowany w obwodzie ma zawsze taki

kierunek że przeciwdziała zmianie strumienia pola magnetycznego. Zamknięta pętla taka jak na rysunku 18 zachowuje się jak dipol magnetyczny, gdyż indukuje się w niej prąd powstały w wyniku zbliżania do pętli magnesu. Pętla taka ma swoje bieguny magnetyczne pokazane na rysunku, oraz moment magnetyczny skierowany wzdłuż linii pola magnetycznego. Aby odpychać magnes (czyli przeciwdziałać zmianie strumienia pola magnetycznego) , który zbliża się do pętli, biegun północny dipola (pętli) musi być zwrócony w stronę wchodzącego magnesu. Wtedy zgodnie z regułą prawej dłoni, gdzie kciuk ma się pokrywać z kierunkiem wektora momentu magnetycznego, można wyznaczyć kierunek płynięcia prądu w pętli.



Rys. 18. Pętla przeciwdziała zmianie strumienia pola magnetycznego wytwarzanego przez zbliżający się magnes.

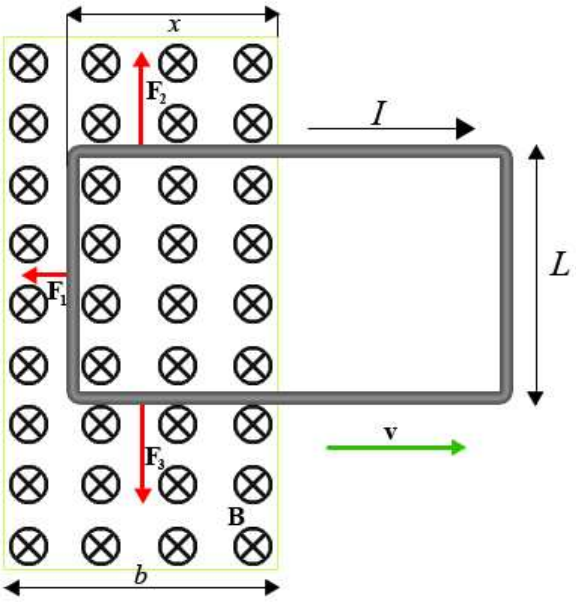
Jeżeli będziemy oddalać magnes tak jak na rysunku 19, sytuacja będzie wyglądała trochę inaczej. Podczas wyciągania dochodzi do zmiany strumienia magnetycznego. Aby jej przeciwdziałać musimy przyciągać wychodzący magnes. Jak wiemy odpychają się bieguny równoimienne natomiast odpychają się bieguny różnoimienne. Aby więc przyciągać wychodzący magnes (skierowany w stronę pętli biegunem północnym), dipol czyli pętla musi mieć biegun południowy zwrócony w stronę wychodzącego magnesu. Gdy zastosujemy regułę prawej dłoni, gdzie kciuk będzie się pokrywał z wektorem momentu magnetycznego to kierunek indukowanego prądu będzie taki jak na rysunku 19.



Rys. 19. Pętla przeciwdziała zmianie strumienia pola magnetycznego wytwarzanego przez oddalający się magnes.

5.3. Zjawisko indukcji

Rozważania ilościowe o indukcji oprzemy na przykładzie z rysunku 20. Mamy do czynienia z sytuacją, w której ramka częściowo umieszczona w polu magnetycznym skierowanym za płaszczyznę kartki, jest wyciągana. Wielkość x to długość ramki jaka jest „zanurzona" w polu magnetycznym. Strumień pola magnetycznego ulega zmianie w czasie gdyż zmienia się wielkość powierzchni ramki jaka „zanurzona" jest w polu magnetycznym.



Rys. 19. Pętla przeciwdziała zmianie strumienia pola magnetycznego wytwarzanego przez oddalający się magnes.

Strumień pola magnetycznego dla sytuacji z rysunku 19 można zapisać za pomocą zależności (63).

$$\Phi_B = BLx \qquad (63)$$

Wtedy indukowana siła elektromotoryczna będzie wyrażona wzorem (64).

$$\mathcal{E} = -\frac{d\Phi_B}{dt} = -\frac{d}{dt}(BLx) = BLv \qquad (64)$$

Wartość indukowanego prądu będzie opisana wzorem (65) (gdzie R jest oporem obwodu).

$$I = \frac{\varepsilon}{R} = \frac{BLv}{R} \qquad (65)$$

Na górny, dolny i lewy bok ramki działają siły F_2, F_3 i F_1. Warto zauważyć że siły F_2 i F_3 mają takie same wartości lecz przeciwne kierunki zatem będą się równoważyć. Pozostaje nam do rozważenia siła F_1, która będzie wyrażona za pomocą wzoru (42). Natomiast wartość siły wyraża się wzorem (41). Po podstawieniu wyrażenia (65) do wzoru (41) otrzymamy zależność przestawioną w postaci wzoru (66).

$$F_1 = \frac{B^2L^2v^2}{R} \qquad (66)$$

Indukowaną siłę elektromagnetyczna można otrzymać nie tylko poprzez ruch magnesów i zwojnic, ale także poprzez działanie na ciało pozostające w spoczynku zmiennym

w czasie polem magnetycznym. Umieszczając np. pierścień w zmienny polu magnetycznym sprawimy że dojdzie do zmiany strumienia magnetycznego, czyli wytworzy się indukowany prąd. Skoro płynie prąd to znaczy że występuje również pole elektryczne. Pole elektryczne pochodzące od prądu indukcji działa (podobnie jak pole elektryczne ładunków statycznych) na ładunek próbny z siła wyrażoną wzorem (67).

$$\mathbf{F} = \mathbf{E}q \qquad (67)$$

Wnioskiem płynącym z tego rozumowania jest to, iż pole magnetyczne, które zmienia się wraz z czasem, kreuje pole elektryczne. Doszliśmy w naszych rozważaniach do nowego sformułowania prawa Faraday'a. Na rysunku 20 znajduje się właśnie taka pętla, która jest umieszczona w zmiennym polu magnetycznym. W takiej pętli tworzy się indukowana siła elektryczna o wyrażona wzorem (62), gdzie N=1. Pola elektryczne wytworzone w ten sposób są współśrodkowymi okręgami. Jeżeli w takim układzie umieścimy ładunek próbny zacznie się on poruszać po okręgu, podczas każdego jednego pełnego obiegu będzie wykonywana praca równa iloczynowi siły elektromotorycznej i wartości ładunku punktowego. Wiemy że pracę możemy zapisać tak jak na wzorze (68).

$$\int \mathbf{F} \cdot d\mathbf{s} = 2\pi r E q \qquad (68)$$

Wiedząc że praca jest iloczynem siły elektromotorycznej i ładunku punktowego możemy wykorzystać tę własność we wzorze (68) i skrócić wartości ładunków punktowych otrzymamy wzór (69).

$$\mathcal{E} = 2\pi r E \qquad (69)$$

W przypadku ogólnym wzór na prace będzie wyglądać jak na wzorze (70).

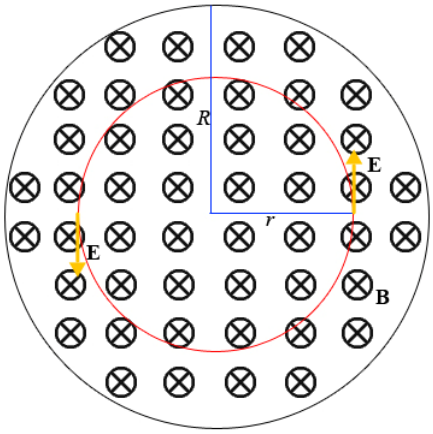
$$W = \oint \mathbf{F} \cdot d\mathbf{s} = q \oint \mathbf{E} \cdot d\mathbf{s} \qquad (70)$$

Czyli wzór na siłę elektromotoryczną wyrazić można za pomocą zależności (71).

$$\mathcal{E} = \oint \mathbf{E} \cdot d\mathbf{s} \qquad (71)$$

Co ostatecznie po podstawieniu wyrażenia (62) do (71) da nam zależność (72). Zakładamy że N=1.

$$\oint \mathbf{E} \cdot d\mathbf{s} = -\frac{d\Phi_B}{dt} \qquad (72)$$



Rys. 20. Pole magnetyczne stopniowo maleje do zera w miejscu R. Okrąg o promieniu r obejmuje strumień pola magnetycznego.

Cewka to nic innego jak zwoje wykonane z jakiegoś przewodnika nawinięte na walec lub pierścień. Podstawowym parametrem charakteryzującym cewki jest indukcyjność, czyli zdolność cewki do wytwarzania strumienia pola magnetycznego jeżeli przepuszczamy przez nią prąd. Wyrazić tą zależność możemy za pomocą równania (73).

$$L = \frac{N\phi_B}{I} \qquad (73)$$

Cewka jest szczególnym przypadkiem solenoidu. Skoro cewkę charakteryzuje indukcyjność tak samo jest z solenoidem. Jego indukcyjność na jednostkę długości l w pobliżu jego środka wyraża wzór (74), gdzie n jest liczbą zwojów na jednostkę długości a S polem przekroju poprzecznego.

$$\frac{L}{l} = \mu_o n^2 S \qquad (74)$$

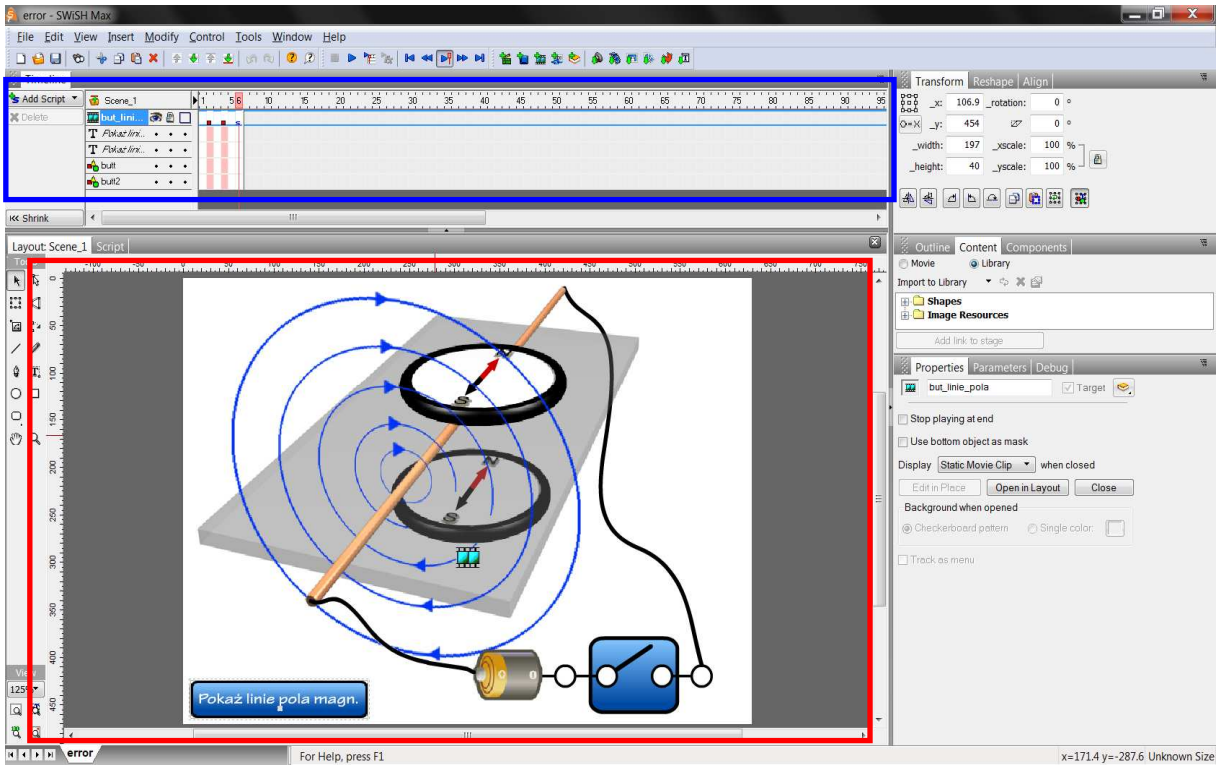
Gdy mamy do czynienia z dwiema cewkami, które znajdują się obok siebie i przez jedną przepuścimy prąd, to wytworzy on strumień magnetyczny, który będzie docierał również do drugiej. Teraz jeżeli zmienimy natężenie prądu w cewce pierwszej dojdzie do zmiany strumienia pola magnetycznego, a co jak wiemy z prawa Faradaya spowoduje pojawienie się siły elektromotorycznej w drugiej cewce. Takie zjawisko nazywane jest samoindukcją. Wzór definiujący siłę elektromotoryczną wytworzona w wyniku samoindukcji przedstawiony jest poniżej.

$$\mathcal{E} = -L\frac{dI}{dt} \qquad (75)$$

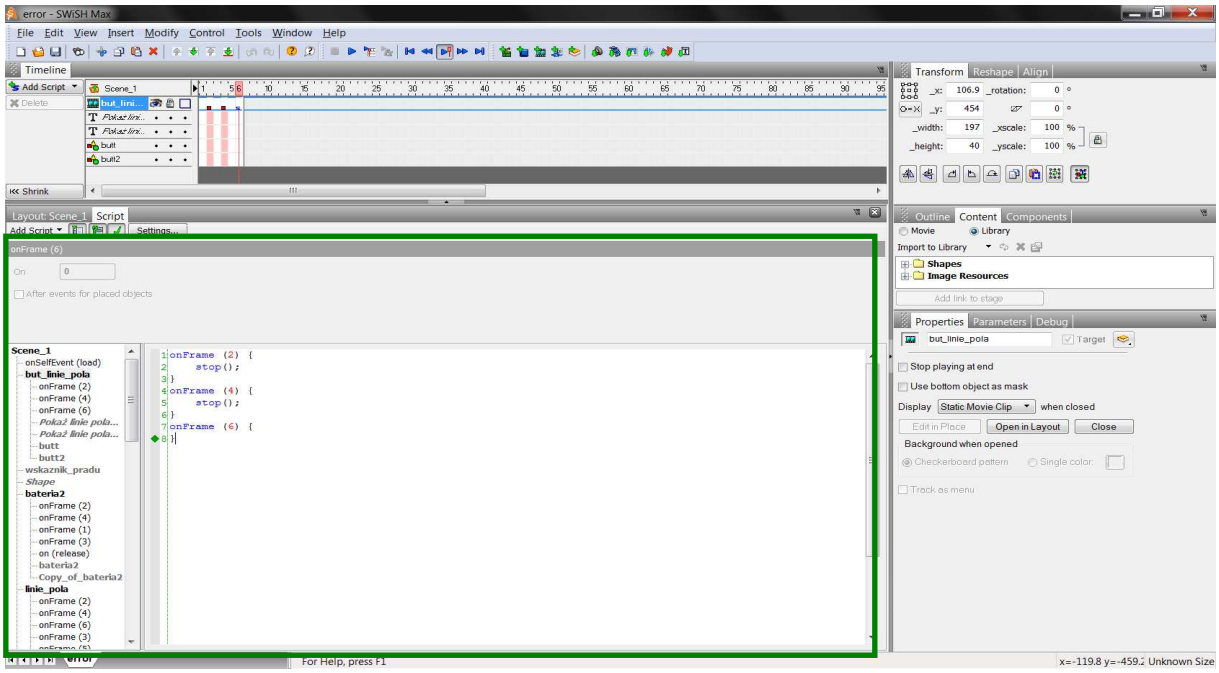
6. Interfejs i implementacja apletów

6.1. Informacje wstępne

W celu lepszego zobrazowania działania i implementacji apletów, na początku należy zapoznać się z interfejsem aplikacji SWiSH Max. Interfejs został pokazany na rysunku 21 i 22.



Rys. 21. Interfejs SWiSH z widocznym obszarem roboczym i linią czasową



Rys. 22. Interfejs SWiSH z widocznym panelem skryptów

Na rysunku 21 czerwony kolorem został zaznaczony obszar roboczy. W tym obszarze ustawiamy wszystkie elementy animacji. Elementami animacji mogą być tzw. Klipy filmowe – czyli specjalne obiekty, które mają osobną listwę czasową (są tak jakby animacjami w głównej animacji), dzięki nim tworzymy animacje z kilku niezależnych, pomniejszych animacji. Kolejnym rodzajem obiektów są kształty - czyli grafika wektorowa tworzona bezpośrednio w programie SWiSH, przyciski, oraz obiekty zdefiniowane przez użytkownika w innych aplikacjach np.: pliki grafiki rastrowej. Na niebiesko zaznaczono obszar listwy czasowej. W tym miejscu określamy zachowanie i właściwości obiektów z obszaru roboczego, w określonych klatkach animacji. Po lewej stronie znajdują się warstwy, czyli poziomy na których umieszczamy obiekty. Warstwa znajdująca się wyżej będzie przesłaniała te znajdujące się niżej.

Na rysunku 22 kolorem zielonym zaznaczono obszar, w którym dodaje się skrypty. Skrypt są powiązane z określonym zdarzeniem np.: OnFrame() – czyli zdarzenie wywoływane w momencie przejścia do określonej klatki animacji, OnPress() – zdarzanie wywoływane przy kliknięciu myszką. Po lewej stronie obszaru skryptów znajduje się lista obiektów, znajdujących się w obszarze roboczym. Obiektom tym można przyporządkować jakieś akcje, wystarczy kliknąć na dany obiekt i napisać dla niego skrypt.

6.2. Implementacja doświadczenie Oersteda

Na początku opiszemy implementacje apletu poświęconemu doświadczeniu Oersteda. Aplet składa się z dwóch klipów filmowych „aplet” oraz „opis”. Na początku omówimy klip „opis”, gdyż jego budowa jest prostsza. Układ warstw został przedstawiony na rysunku 23.

Rys. 23. Układ warstw klipu filmowego „opis”.

Pierwsze dwie górne warstwy (o nazwie Shape) to linie przerywane, poprawiające wygląd pola tekstowego. Warstwa o nazwie upb to górny przycisk służący do przewijania tekstu w dół. Wewnątrz klipu upb znajdują się grafiki, które są animowane po najechaniu na przycisk kursorem myszki. Animacja ta polega na cyklicznym pojawianiu się i zanikaniu odpowiedniego pliku graficznego. Oprócz tego, w klipie filmowym znajduje się również specjalna warstwa do której przyporządkowane są określone akcje.

```
onSelfEvent (enterFrame, includingFirstFrame) {
    if (this.bPress == "true") {
        _parent.PoleTekstu.wynik.scroll -= 1;
    }
}
```

Powyższy kod, jest fragmentem skryptu, przyporządkowanym do klipu filmowego. Zdarzenie `onSelfEvent(enterFrame, includingFirstFrame)` wywoływane jest dla każdej klatki

animacji. Zmienna `bPress` jest odpowiedzialna za wskazywanie czy przycisk został wciśnięty. Jeżeli tak (czyli jej wartość ustawiona jest na `"true"`), polu tekstowemu (`wynik`), poprzez właściwość `scroll`, nakazujemy przesunąć swoją zawartość o jedną linijkę w dół. Zmienna `bPress` ustawiana jest przy kliknięciu oraz przy puszczeniu przycisku myszy. Kod obsługujący zdarzenia kliknięcia i puszczenia przycisku myszy znajdziemy w klipie filmowym upb, po zaznaczeniu warstwy akcje.

```
on (press) {
    bPress = "true";
}
on (release) {
    bPress = "false";
}
```

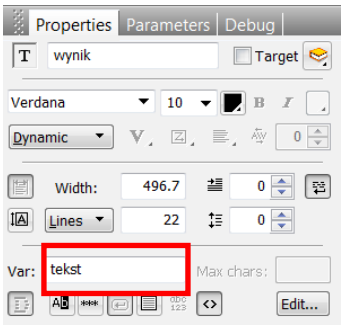
Pierwsze zdarzenie jest wywoływane w momencie naciśnięcia przycisku myszy. Ustawia ono wartość zmiennej `bPress` na `"true"`. Zdarzenie `on (release)`, wywoływane w momencie gdy zostanie puszczony przycisk myszy, ustawia zmienna `bPress` na `"false"`. Takie podejście zapewnia że gdy przycisk myszy będzie cały czas wciśnięty, zmienna `bPress` będzie ustawiona na `"true"` co w połączeniu ze zdarzenie wywoływanym w każdej ramce animacji, powodować będzie przewijanie tekstu aż do momentu puszczenia przycisku myszy. Rozwiązanie to jest bardzo wygodne dla użytkownika, gdyż nie musi ciągle klikać myszą na przyciski, w celu przewijania tekstu. Klip filmowy dwb, którego celem jest przewijanie tekstu do góry, został stworzony tą samą techniką. Również skrypty przyporządkowane do klipu dwb, oraz do warstwy akcje, znajdującej się wewnątrz niego, są praktycznie identyczne z tymi z klipu upb. Jedyną różnicą jest linijka `_parent.PoleTekstu.wynik.scroll -= 1`, która została zamieniona na `_parent.PoleTekstu.wynik.scroll += 1`. Aby cały mechanizm wczytywania tekstu działał należało dodać kod, który będzie wczytywał z określonych plików opis doświadczenia oraz plik arkuszy stylów CSS, który odpowiednio sformatuje tekst. Skrypt ten znajdziemy zaznaczając całą scenę (czyli warstwę o nazwie Scene_1). Kod ten wygląda następująco:

```
onFrame (1) {
      opis.PoleTekstu.loadVariables("opisy/oersted.txt");
      var format = new TextField.StyleSheet();
      var path = "opisy/styl.css";
      format.load(path);
      opis.PoleTekstu.wynik.styleSheet = format;
}
```

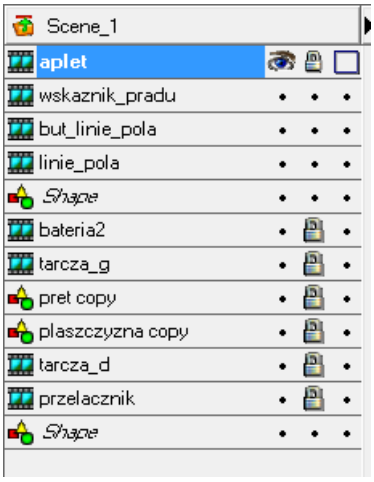
Fragment tego kodu zostanie wykonany w pierwszej klatce animacji (zdarzenie `onFrame(1)`). Kolejna linijka kodu odpowiedzialna jest za ładowanie tekstu z pliku oersted.txt znajdującego się w folderze opisy. Jeżeli otworzy się plik oersted.txt da się zauważyć że zaczyna się tekstem „&tekst=”. Linijka ta informuje aplet że w piku znajduje się zmienna o nazwie tekst której wartość jest tekstem opisu doświadczenia. Dzięki temu będziemy mogli wartość tej zmiennej przypisać do pola tekstowego. Uczynić to można klikając na pole tekstowe (warstwa wynik w klipie filmowy o nazwie PoleTekstu) a następnie w panelu Properties (rys. 24.) w polu Var wpisujemy nazwę tej zmiennej. Kolejna linijka odpowiedzialna jest za utworzenie zmiennej `format`, która reprezentuje styl formatowania tekstu (w tym wypadku będziemy formatować za pomocą arkusza styli CSS więc zostanie wywołana metoda `StyleSheet()`). Kolejna linijka tworzy zmienna reprezentującą ścieżkę do pliku z arkuszem stylów. Linijka niżej zawiera instrukcje która ładuje arkusz styli CSS do obiektu `format` z pliku `path`. Na samym końcu do właściwości `styleSheet` ,pola tekstowego wynik (znajdującego się w klipie filmowym o nazwie PoleTekstu, który to umieszczony jest w klipie

opis) przypisywana jest zawartość zmiennej `format`, która zawiera informacje o stylu formatowania tekstu. Warto nadmienić że wszystkie aplety z pracy dyplomowej posiadają tą samą budowę tzn. zawierają dwa główne klipy filmowe (aplet i opis). Klip opis we wszystkich apletach zbudowany jest tak samo, również jego skrypty są prawie identyczne, jedyną różnicą są zmiany w linijce, w której do zmiennej `path` przypisywana jest ścieżka do opisu danego doświadczenia. Mając to na uwadze, w kolejnych opisach implementacji danego apletu nie będą wytłumaczone metody projektowania klipu filmowego opis.

Oprócz klipu filmowego opis, w aplecie dotyczącym doświadczenia Oersteda, znajduje się również klip aplet. Jego struktura warstw została przedstawiona na rysunku 25.



Rys. 24. Panel Properties z zaznaczonym polem Var.



Rys. 25. Układ warstw dla klipu filmowego aplet

Grafiki kompasów, przewodnika i płaszczyzny zostały wykonane w programie do tworzenia grafiki trójwymiarowej Swift3D. Po wymodelowaniu, obiekty te, został poddane procesowi renderowania i zapisane do plików graficznych. Następnie pliki te zostały zaimportowane do programu, gdzie można było ich używać. Z klipem filmowym o nazwie aplet powiązane są poniższe zmienne.

```
onSelfEvent (load) {
    bateryjka = 1;
    wlaczniki=0;
    czy_linie_pola = 0;

}
```

Zmienna `bateryjka` opisuje położenie baterii, jeżeli przyjmuje wartość 1 zwrócona jest biegunem dodatnim w lewą stronę, jeżeli natomiast przyjmuje wartość 0 sytuacja jest odwrotna. Zmienna `bateryjka`, jeżeli przyjmuje wartość 0 to znaczy że włącznik prądu jest wyłączony, natomiast jeżeli przyjmuje wartość 1 to mamy do czynienia z sytuacja gdy obwód

jest zamknięty. Ostatnia zmienna określa nam czy linie pola magnetycznego powinny być włączone (wartość 1) lub wyłączone (wartość 0).

Kolejnym ciekawy przypadkiem implementacyjnym jest klip but_linie_pola. Jest to przycisk dzięki któremu można włączyć pokazywanie linii pola magnetycznego, wokół przewodnika. Kod obsługujący akcje przedstawiono poniżej.

```
16. on (release){
17.     if(czy_linie_pola == 0){
18.         czy_linie_pola=1;
19.         but_linie_pola.gotoAndPlay("pressed");
20.         if(wlaczniki==1 && bateryjka == 1){
21.             linie_pola.gotoAndPlay("prawa");
22.         }else if(wlaczniki==1 &&bateryjka == 0){
23.            linie_pola.gotoAndPlay("lewa");
24.         }
25.     }else if(czy_linie_pola == 1){
26.         czy_linie_pola=0;
27.         but_linie_pola.gotoAndPlay(1);
28.         linie_pola.gotoAndPlay(1);
29.     }
30. }
```

Powyższy skrypt jest wykonywany w momencie naciśnięcia myszą w obszarze klipu but_linie_pola. Na początku sprawdzamy czy linie pola są wyłączone, jeżeli tak to zmienną `czy_linie_pola` ustawiamy na wartość 1 (czyli linie pola są teraz włączone). W kolejnej linijce przechodzimy do klatki oznaczonej etykietą `czy_linie_pola`, w klipie filmowym but_linie_pola. Jeżeli wykonamy tą linijkę to kolor przycisku zmieni się na zielonym, co będzie sugerowało użytkownikowi że linie pola są włączone. W linijce niżej sprawdzany jest warunek czy bateria ustawiona jest biegunem dodatnim w lewo oraz czy włącznik jest włączony, jeżeli tak to w klipie linie_pola przechodzimy do klatki oznaczone etykietą . Przejście do tej klatki oznacza że pokazane zostaną linie pola magnetycznego ze strzałkami zwróconymi zgodnie z ruchem wskazówek zegara. Jeżeli warunek w linijce 20 nie jest spełniony sprawdzany jest warunek w linijce 22, czyli czy obwód jest zamknięty oraz czy bateria jest zwrócona biegunem ujemnym w lewą stronę. Jeżeli warunek ten jest spełniony przechodzimy do klatki klipu filmowego linie_pola (oznaczonej etykietą `lewa`), w której to linie pola są pokazywane ze strzałkami zwróconym przeciwnie do ruchu wskazówek zegara. Jeśli linie pola są włączone (warunek w linijce 25) to po kliknięciu włącznika myszką należy je wyłączyć. Zatem ustawiamy zmienna `czy_linie_pola` na wartość 0 (linijka 26), następnie w linijce 27, przycisk but_linie_pola powraca do swojego pierwotnego koloru. W linijce 28 ostatecznie chowamy linie pola magnetycznego.

Położenie baterii można zmieniać. Bateria może być zwrócona w lewą stronę dodatnio lub ujemną elektrodą, co ma wpływ na kierunek przepływu prądu. Funkcje te są realizowane w klipie filmowy bateria2. Poniżej przedstawiono skrypty przypisane do tego klipu.

```
14. on (release) {
15.
16.     if (bateryjka==1) {
17.         bateria2.gotoAndPlay("minus");
18.         bateryjka=0;
19.         if(wlaczniki==1){
20.             tarcza_g.gotoAndPlay("powrot_plus");
21.             tarcza_d.gotoAndPlay("powrot_plus");
22.             wskaznik_pradu.gotoAndPlay("dol");
23.         }
24.     }else{
25.         bateria2.gotoAndPlay("plus");
26.         bateryjka=1;
```

```
27.             if(wlaczniki==1){
28.                 tarcza_g.gotoAndPlay("powrot_minus");
29.                 tarcza_d.gotoAndPlay("powrot_minus");
30.                 wskaznik_pradu.gotoAndPlay("gora");
31.             }
32.         }
33.         if(bateryjka == 1 && wlaczniki == 1 && czy_linie_pola==1){
34.             linie_pola.gotoAndPlay("prawa");
35.         }else if (bateryjka == 0 && wlaczniki == 1 && czy_linie_pola==1){
36.             linie_pola.gotoAndPlay("lewa");
37.         }
38.     }
```

Jak widać zdarzenie to również jest wywoływane gdy naciśnięty zostanie przycisk myszy, gdy kursor będzie znajdował się w obszarze baterii. Jeżeli bateria dodatnią elektrodą jest ustawiona w lewą stronę (linijka 16), ustawiamy baterię w przeciwnym kierunku (linijka 17). Aby to zrobić przechodzimy w klipie bateria2 do klatki oznaczonej etykietą `minus`. W tej klatce bateria jest odwracana lewym biegunem w lewą stronę. Z doświadczeń wykonanych w realnym świecie wiadomo że zmieniając polaryzację baterii przy zamkniętym obwodzie spowoduje się obrót wskazówki kompasu w przeciwnym kierunku oraz zmianę kierunku płynięcia prądu. Sytuację tę wymodelowane również w tym aplecie. Jest ona opisana za pomocą linijek od 19 do 23. Jeżeli włącznik jest ustawiony na 1 (linijka 19) to uruchomiona zostaje animacja zaczynająca się w klatce o etykiecie `powrot_plus`, w klipie filmowy tarcza_g. Klip tarcza_g jest kompas górny znajdujący się nad przewodnikiem z prądem. Przechodząc do klatki oznaczonej etykietą , uruchamiamy animację, która obróci wskazówkę kompasu górnego z położenia, w którym północny biegun znajduje się po prawej stronie, do położenia, w którym wskazówka północna będzie znajdować się po lewej stronie. W kolejnej linijce, o numerze 21, obracana jest wskazówka kompasu dolnego, znajdującego się pod płaszczyzną. Odtwarzamy w klipie filmowym kompas_d klatkę o etykiecie `powrot_plus`. Przejście do tej klatki spowoduje obrót wskazówki kompasu z położenia, w którym biegun północny wskazówki kompasu górnego zwrócony jest w lewą stronę, do położenia, w którym biegun północny skierowany będzie w prawą stronę. Następnie wykonywana będzie linijka 22. Powoduje ona uruchomienie animacji, która zaczyna się w klatce o etykiecie `dol`. Animacja ta znajdująca się w klipie filmowym o nazwie wskaźnik_pradu, spowoduje że strzałka wskazująca kierunek prądu zmieni swój kierunek i będzie teraz skierowana w dół. Jeżeli warunek w linijce 16 nie jest spełniony (czyli bateria zwrócona jest elektrodą ujemną w lewą stronę) zmieniamy kierunek położenia ujemnej elektrody na prawy (linijka 25). Skrypt wykonywany w tej linijce powoduje przejście w klipie filmowym bateria2 do klatki o etykiecie `plus`. W tej klatce położenie baterii jest zmieniane na takie że bateria zwrócona będzie dodatnią elektrodą w lewą stronę. Wartość zmiennej `bateryjka`, ustawiana jest na 1 co będzie oznaczało że bateria jest położona w przeciwnym kierunki niż była do tej pory.

Powtarza się wcześniejszy scenariusz. Trzeba odwrócić kierunki wskazówek kompasów górnego i dolnego na kierunki przeciwne do dotychczasowych. Również kierunek prądu musi ulec zmianie na przeciwny. Odbywa się to w linijkach od 28 do 30. Należy również zmienić kierunek pola magnetycznego wokół przewodnika. W linijce 33 znajduje się warunek mówiący że jeżeli bateria jest zwrócona biegunem dodatnim w lewą stronę (czyli zmienna `bateryjka` ma wartość 1) oraz jeżeli włącznik jest włączony (czyli zmienna `wlaczniki` ma wartość 1) oraz jeżeli użytkownik chce wyświetlać linie pola (czyli zmienna `czy_linie_pola` ma wartość 1) to należy wyświetlić linie pola ze strzałkami zwróconymi w kierunku zgodnym z ruchem wskazówek zegara (linijka 34). Jeżeli warunek jest niespełniony to sprawdzana jest sytuacja w której bateria jest zwrócona ujemną elektrodą w lewą stronę (czyli zmienna przyjmuje wartość 0) oraz czy włącznik jest zamknięty a także czy użytkownik chce wyświetlić linie pola. Jeżeli warunek jest spełniony to pokazywane są linie pola

magnetycznego ze strzałkami zwróconymi w kierunku przeciwnym do ruchu wskazówek zegara (linijka 36).

Ostatnim wartym uwagi oraz najważniejszym elementem apletu jest skrypt obsługujący przełącznik zamykający i otwierający obwód elektryczny. Skrypt przedstawiony poniżej wykonywany jest w momencie naciśnięcia lewym przyciskiem myszy na przełącznik.

```
13.  on (release) {
14.      if(wlaczniki==1){
15.          przelacznik.gotoAndPlay("off");
16.           linie_pola.gotoAndPlay(1);
17.           wskaznik_pradu.gotoAndPlay(1);
18.          wlaczniki=0;
19.          if(bateryjka==1){
20.            tarcza_g.gotoAndPlay("powrot_plus");
21.            tarcza_d.gotoAndPlay("powrot_plus");
22.          }else if(bateryjka==0){
23.             tarcza_g.gotoAndPlay("powrot_minus");
24.             tarcza_d.gotoAndPlay("powrot_minus");}
25.      }else{
26.         przelacznik.gotoAndPlay("on");
27.        wlaczniki=1;
28.
29.        if(bateryjka==1){
30.            tarcza_g.gotoAndPlay("obrot_plus");
31.            tarcza_d.gotoAndPlay("obrot_plus");
32.            wskaznik_pradu.gotoAndPlay("gora");
33.        }else if(bateryjka==0){
34.            wskaznik_pradu.gotoAndPlay("dol");
35.             tarcza_g.gotoAndPlay("obrot_minus");
36.             tarcza_d.gotoAndPlay("obrot_minus")
37.        }
38.      }
39.      if(bateryjka == 1 && wlaczniki == 1 && czy_linie_pola==1){
40.          linie_pola.gotoAndPlay("prawa");
41.      }else if (bateryjka == 0 && wlaczniki == 1 && czy_linie_pola==1){
42.          linie_pola.gotoAndPlay("lewa");
43.      }
44.  }
```

Skrypt ten przypisany jest do klipu filmowego o nazwie przelacznik. W linijce 14 sprawdzana jest wartość zmiennej `wlaczniki`. Jeżeli jest ustawiona na to znaczy że włącznik jest włączony, należy go wyłączyć, wskazówki kompasów winny być ustawione w pozycji początkowej, natomiast linie pola należy wyłączyć gdyż przewodnik nie wytwarza pola magnetycznego. Czynności te są realizowane w linijkach o numerach od 15 do 24. Na początku w klipie przelacznik przechodzimy do klatki numer jeden. Powoduje to uruchomienie animacji przełącznika, która zmienia jego tło na niebieskie, sygnalizując tym samym że obwód jest otwarty. W linijce o numerze 16 przechodzimy w klipie linie_pola do klatki numer jeden, czyli wyłączamy linie pola magnetycznego. Linijka 17 jest odpowiedzialna za wyłączenie strzałki pokazującej kierunek płynięcia prądu. Linijki od 19 do 24 są odpowiedzialne za przywrócenie pierwotnego kierunku igieł magnetycznych kompasu górnego i dolnego. Jeżeli zmienna `bateryjka` ma wartość jeden (linijka 19), czyli bateria jest zwrócona elektrodą dodatnią w lewą stronę to uruchamiana jest animacja powrotu wskazówki do pozycji początkowej dla kompasu górnego (linijka 20) oraz dolnego (linijka 21). W tym wypadku wskazówka kompasu górnego (umieszczony jest on w klipie tarcza_g) była wychylona biegunem północnym w prawą stronę i powraca do położenia w którym wskazówka biegunem północnym będzie zwrócona do góry. Wskazówka kompasu dolnego

(klip tarcza_d) zwrócona była biegunem północnym w lewo co znaczy że teraz powróci do stanu początkowego, w którym będzie zwrócona biegunem północnym w górę. Jeżeli bateria jest zwrócona elektrodą dodatnią w prawą stronę to zmienna przyjmuje wartość zero (linijka 22) to również uruchomiona zostanie animacja powrotu wskazówek do położenia początkowego. Tym razem wskazówka kompasu górnego powróci z położenia, w którym biegun północny był zwrócony w lewą stronę, do położenie w którym będzie zwrócony w górę (linijka 23). Wskazówka kompasu dolnego powróci z położenia, w którym jej biegun północny był zwrócony w prawo, do położenia w którym będzie on zwrócony do góry (linijka 24).

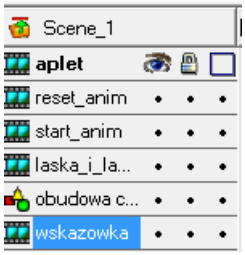
Wcześniejsze rozważania były oparte o przypadek w którym wyłączaliśmy przełącznik. Teraz rozważymy przypadek w którym przełącznik jest włączany (czyli zmienna przyjmuje wartość `wlaczniki` zero). Przypadek ten jest obsługiwany przez skrypt w linijkach od 26 do 36. Na samym początku w klipie przelacznik odtwarzamy klatkę o etykiecie `on`. Powoduje to że przełącznik będzie miał kolor zielony co sugeruje że obwód jest włączony. Kolejna linijka powoduje przypisanie do zmiennej `wlaczniki` wartości jeden, dzięki czemu poszczególne klipy apletu będą informowane o fakcie włączenia przełącznika. Podobnie jak w przypadku w którym otwieraliśmy obwód, tym razem również musimy zmienić położenie wskazówek kompasów oraz pokazać kierunek płynięcia prądu. W tym wypadku będziemy poruszać wskazówki z położenia początkowego, w którym są one zwrócone biegunem północnym do góry. Gdy bateria jest zwrócona elektrodą dodatnia w lewą stronę (zmienna przyjmuje wartość jeden) wskazówka kompasu górnego będzie obracać się z położenia początkowego do położenia, w którym jej biegun północny będzie zwrócony w prawo (linijka 30). Wskazówka kompasu dolnego obróci się natomiast z położenia początkowego do położenia w którym biegun północny będzie zwrócony w lewą stronę (linijka 31). Strzałka wskazująca kierunek prądu będzie w tym przypadku skierowana do góry (linijka 32). Jeżeli warunek w linijce 29 nie jest spełniony to zmienna ma wartość zero. Oznacza to że bateria jest zwrócona biegunem dodatnim w prawą stronę, wtedy wykonywany jest kod zawarty w linijkach 34, 35 i 36. Najpierw strzałka wskaźnika przepływu prądu zmienia swój kierunek na dolny (linijka 34). Kod w linijce 35 spowoduje że wskazówka kompasu górnego obróci się z kierunku początkowego do kierunku w którym jej biegun północny będzie skierowany w lewo, natomiast linijka 36 spowoduje obrót wskazówki kompasu dolnego do położenia, w którym jej biegun północny będzie zwrócony w prawo.

Pozostaje do rozwiązania kwestia wyświetlania linii pola magnetycznego. Jeżeli użytkownik wcisnął przycisk, który jest odpowiedzialny za ewentualne pokazywanie linii sił pola i dodatkowo jeżeli płynie prąd oraz bateria zwrócona jest elektrodą dodatnią w lewą stronę, to strzałki linii sił pola magnetycznego muszą być skierowane zgodnie z ruchem wskazówek zegara. Warunek ten jest rozważany w linijce 39. Sprawdza się czy zmienna `wlaczniki` ma wartość jeden oraz czy zmienna `bateryjka` ma wartość jeden. Sprawdzana jest również wartość zmiennej `czy_linie_pola`, jeżeli wynosi jeden to znaczy że użytkownik życzy sobie wyświetlać linie pola magnetycznego. Jeżeli wszystkie te warunki w klipie filmowym linie_pola odtwarzana jest klatka o etykiecie `prawa`, która jest odpowiedzialna za wyświetlenie linii pola ze strzałkami zwróconymi zgodnie z ruchem wskazówek zegara. Natomiast, jeżeli płynie prąd (czyli zmienna `wlaczniki` ma wartość jeden) oraz użytkownik życzy sobie oglądać linie pola magnetycznego (zmienna `czy_linie_pola` ustawiona na jeden) a także bateria jest zwrócona elektrodą dodatnią w prawą stronę (zmienna `bateryjka` ustawiona na zero) to odtwarzana jest klatka w której strzałki na liniach pola magnetycznego są zwrócone przeciwnie do ruchu wskazówek zegara (linijka 42).

6.3. Implementacja doświadczenia z elektroskopem

Aplet dotyczący doświadczenia z elektroskopem jest bardzo prosty. Zbudowany jest praktycznie z samych animacji i nie posiada żadnych skomplikowanych skryptów. Zbudowany jest, tak jak pozostałe aplety, z dwóch klipów filmowych o nazwach opis i aplet. Implementacja klipu filmowego została opisana w poprzednim rozdziale. Jedyną różnicą jest linijka w której znajduje się definicja zmienne określającej ścieżkę do pliki. Wygląda ona następująco : `opis.PoleTekstu.loadVariables("opisy/elektroskop.txt")`.

Struktura warstw klipu aplety została przedstawiona na rysunku 26.



Rys. 26. Układ warstw dla klipu filmowego aplet w doświadczeniu z elektroskopem.

Na samej górze znajduje się klip filmowy reset_anim, który jest przyciskiem wprowadzającym animacje do stanu początkowego. Poniżej znajduje się klip start_anim, zawierający przycisk startujący animację. Na kolejnej warstwie umieszczony jest klip filmowy laska_i_ladunki. Ten klip odpowiedzialny jest za animacje przesuwających się ładunków oraz za animację zbliżającej się laski ebonitowej. Na kolejnej warstwie znajduje się plik graficzny, który przedstawia obudowę elektroskopu. Ostatnia warstwa zawiera klip filmowy wskazowka, który jest animacją wskazówki elektroskopu. Obudowa elektroskopu oraz jego wskazówka zostały wykonane w programie do obróbki grafiki trójwymiarowej o nazwie Swift3D. Wyeksportowano te modele do plików graficznych i dodano do biblioteko obiektów w programie SWiSH.

Przycisk start_anim jest odpowiedzialny za uruchomienie odtwarzania animacji. Do tego przycisku przyporządkowane są trzy zdarzenia. Kod zawarty w pierwszym zdarzeniu uruchamiany jest w momencie zwolnienia naciśniętego, lewego przycisku myszy.

```
1.  on (release) {
2.
3.     _root.start_anim.gotoAndPlay(3);
4.     laska_i_ladunki.gotoAndPlay("start");
5.      wskazowka.gotoAndPlay("start");
6.
7.  }
```

Po wywołaniu tego zdarzenia przez użytkownika na początku w klipie start_anim odtwarzamy klatkę numer trzy. W tej klatce przycisk zmienia swój kolor na zielony. W kolejne linijce uruchamiana jest animacja przybliżania się laski do elektroskopu, oraz animacja ruchu ładunków elektrycznych. Linijka numer 5 jest odpowiedzialna za uruchomienie animacji obrotu wskazówki. Zdarzenie wywoływane w momencie najechania na przycisk kursorem myszki przedstawione jest poniżej.

```
1.  on (rollOver) {
2.     _root.start_anim.gotoAndPlay(5);
3.  }
```

W po wywołaniu tego zdarzenia następuję zmiana tła przycisku na zielony, aby użytkownik wiedział że właśnie najechał na przycisk. Ostatnie zdarzenie `on (rollout)`, wywoływane

jest w momencie zjechania kursora myszy z przycisku. Powoduje ono że przycisk wraca do pierwotnego stanu.

6.4. Implementacja doświadczenia z solenoidem

Jak poprzednie aplety również ten posiada dwa główne klipy filmowe opis i aplet. Klip opis został zbudowany w ten sam sposób jak w aplecie z doświadczenie Oersteda. Jedyną różnica jest linijka która zawiera ścieżkę do pliku z opisem doświadczenia. Wygląda ona następująco : `opis.PoleTekstu.loadVariables("opisy/solenoid.txt")`.

Klip filmowy aplet posiada układ warstw taki jak na rysunku 27.

Rys. 27. Układ warstw dla klipu filmowego aplet w doświadczeniu z solenoidem.

Na samej górze znajduje się klip bit_linie_pola – animacja przedstawiająca przycisk służący do włączania lub wyłączania linii pola magnetycznego. Klip strzałki prądu zawiera animację pokazującą lub chowającą strzałki wskazujące przepływ prądu w każdym z przewodników. Warstwy zderzak_amperomierza i zderzak_koncowy są przeźroczystymi prostokątami. Potrzebne są one do uruchamiania animacji obrotu wskazówki amperomierza. Na kolejnej warstwie umieszczony jest klip lider, który zawiera komponent do ustalania prędkości wsuwania magnesu. Klip amperomierz zawiera animacje amperomierza. Warstwa solenoid solenoid2a_front_small.png jest obrazkiem reprezentującym solenoid. Solenoid został wykonany w programie Swift3D a następnie wyeksportowany do pliku graficznego. Solenoid został podzielony na dwie części (przednią i tylną) po to, aby nadać animacji efekt trójwymiarowości podczas wkładania magnesu. Przednia część solenoidu przysłania magnes, natomiast tylna część solenoidu jest przysłaniana przez magnes. Obrazek z tylną częścią solenoidu został umieszczony na warstwie solenoid2a_back_small.png. W klipie filmowym magnes znajduje się animacja magnesu. Reszta warstw zawiera kształty wykonane w programie SWiSH. Przedstawiają one kable oraz połączenia między nimi.

Omawianie implementacji apletu omówimy zaczynając od pokazania zmiennych jakie zostały zdeklarowane i zdefiniowane dla apletu w klipie aplet. Kod tworzący wszystkie niezbędne zmienne został pokazany na listingu poniżej.

```
1.  onSelfEvent (load) {
2.      predkosc_magnesu = 10;
3.      max_kat = 0;
4.      biegun = 0;
5.
6.      czy_linie_pola = 0;
7.
8.  }
```

Kod zdarzenia `onSelfEvent (load)` wywoływany jest w momencie, w którym klip aplet zostaje załadowany. Tworzone jest wówczas pięć zmiennych. Zmienna `predkosc_magnesu` jest prędkością wsuwania magnesu, zostaje zainicjowana wartością 10. Zmienna `max_kat` będzie maksymalnym katem do jakiego może wychylić się wskazówka amperomierza. Zmienna `biegun` wskazuje w jakim położeniu aktualnie znajduje się magnes. Wartość 0 określa położenie, w którym magnes ma biegun południowy zwrócony w lewą stronę, natomiast zmienna będzie przyjmować wartość 1 jeżeli biegun północny będzie zwrócony w lewą stronę. Ostatnią zmienna jest `czy_linie_pola`. Jeżeli jej wartość jest 0 to znaczy że użytkownika nie chce żeby wyświetlane były linie pola magnetycznego wokół magnesu.

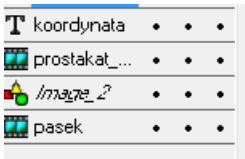
Klikając na klip filmowy but_linie_pola użytkownik uruchomi kod odpowiedzialny za włączenie lub wyłączenie linii pola magnetycznego. Kod ten przedstawiony jest poniżej.

```
16. on (release){
17.     if(czy_linie_pola == 0){
18.         czy_linie_pola=1;
19.         but_linie_pola.gotoAndPlay("pressed");
20.         if(biegun == 0){
21.             magnes.linie_pola.gotoAndPlay("lewo");
22.         }else if(biegun == 1){
23.            magnes.linie_pola.gotoAndPlay("prawo");
24.         }
25.     }else if(czy_linie_pola == 1){
26.         czy_linie_pola=0;
27.         but_linie_pola.gotoAndPlay(1);
28.         magnes.linie_pola.gotoAndPlay(1);
29.     }
30. }
```

Na początku sprawdzana jest wartość zmiennej `czy_linie_pola`. Jeżeli wynosi ona zero, czyli linie pola magnetycznego są wyłączone, to wykonywany jest kod pomiędzy linijkami 18 a 24. W linijce 18 zmienna będzie przyjmować wartość 1 co będzie znaczyło że użytkownik chce pokazywać linie pola. W klipie filmowym but_linie_pola odtwarzamy klatkę oznaczoną etykietą `pressed`. W klatce tej kolor przycisku zmienia się na zielony co sugeruje że linie pola magnetycznego są włączone. W następnej linii sprawdzane jest ustawienie magnesu, jeżeli zmienna `biegun` przyjmuje wartość 0, to magnes zwrócony jest biegunem południowym w lewą stronę. Uruchamiana jest wtedy klatka oznaczona etykietą `lewo`, która znajduje się w klipie filmowym linie_pola, który to z kolei znajduje się w klipie magnes. Odtworzenie tej klatki spowoduje pojawienie się linii pola magnetycznego, wokół magnesu, które będą miały strzałki zwrócone w lewą stronę. Jeżeli poprzedni warunek nie jest spełniony to sprawdzany jest dodatkowy warunek w linii 22. Jeżeli jest on spełniony to odtwarzana jest klatka oznaczona etykietą `prawo`, która znajduje się w klipie linie_pola. Spowoduje to włączenie linii pola magnetycznego ze strzałkami zwróconymi w prawo. W przypadku gdy warunek z linijki 17 nie jest spełniony, to sprawdzany jest warunek w linii 25. Warunek ten sprawdza czy wartość zmiennej wynosi 1. Jeżeli tak oznacza to że linie pola magnetycznego są włączone i użytkownik chce je teraz wyłączyć. Zmieniamy wówczas wartość zmiennej na 0, co oznaczać będzie że linie pola magnetycznego są wyłączone. W kolejnej linijce przycisk zostaje przywrócony do stanu początkowego poprzez zmianę jego

koloru na niebieski, dzięki czemu użytkownik wie że linie pola są wyłączone. W linijce 29, w klipie filmowym linie_pola, wyłączamy linie pola magnetycznego poprzez odtworzenie klatki oznaczonej numerem jeden.

Przyjrzymy się teraz bliżej klipowi slider. Układ warstw został przedstawiony na rysunku 28.



Rys. 27. Układ warstw dla klipu filmowego aplet w doświadczeniu z solenoidem.

W klipie filmowym prostakat_slider znajduje się grafika o prostokątnym kształcie, która przesuwa się po pasku, dzięki czemu możemy dobrać prędkość magnesu. Na warstwie pasek znajduje się prostokąt o przeźroczystym kolorze, który określa granice w jakich może się poruszać prostokąt służący do zmiany prędkości wsuwania magnesu. Warstwa Image_2 reprezentuje plik graficzny, który jest tłem dla paska po którym będzie się poruszał prostokąt do ustalania prędkości magnesu. Do klipu filmowego slider, przyporządkowany jest kod przedstawiony poniżej, który jest wykonywany w każdej klatce animacji.

```
1.   onSelfEvent (enterFrame) {
2.     koordynata.text = prostakat_slider._x-pasek._x;
3.     _parent.predkosc_magnesu = prostakat_slider._x-pasek._x;
4.     if(_parent.biegun==0){
5.         _parent.max_kat=((prostakat_slider._x-pasek._x)/1000)*360*1.3;
6.     }else{
7.         _parent.max_kat=-((prostakat_slider._x-pasek._x)/1000)*360*1.3;
8.     }
9.     koordynata._x = prostakat_slider._x-10;
10.    koordynata._y = prostakat_slider._y-15;
11.  }
```

Na samym początku do pola tekstowego koordynata wpisywany jest wynik odejmowania współrzędnej x klipu filmowego prostakat_slider ze współrzędną x paska, po którym porusza się klip filmowy prostakat_slider. Odejmowanie to da nam wartość współrzędnej x klipu filmowego prostakat_slider względem współrzędnej x początku klipu pasek. Wartość ta nadawana jest również zmiennej `predkosc_magnesu` (linijka 3). W kolejnej linii sprawdzane jest, na podstawie zmiennej `biegun`, czy biegun magnesu południowy magnesu zwrócony jest w prawą stronę. Jeżeli tak to zmienna `max_kat` otrzyma wartość taką jak w linijce 5. Wartość ta była dobierana metodą prób i błędów. Jeżeli warunek z linijki 4 nie jest spełniony to wykonywany jest kod w linijce 7. Przypisuje on zmiennej `max_kat` taka samą wartość jak w linijce 5 tylko że z minusem. Ostatnie dwie linijki zmieniają współrzędne odpowiednio x i y pola tekstowego koordynata, na współrzędne x i y prostokąta który przesuwamy. Współrzędne te są pomniejszane o 15. Sprawia to wrażenie że pole tekstowe porusza się wraz przesuwanym prostokątem.

Za przesuwanie prostokąta i tym samym ustalanie szybkości z jaką będzie przesuwał się magnes odpowiada kod umieszczony w klipie filmowym prostakat_slider. Klip ten reaguje na trzy zdarzenia. Pierwsze z nich `onSelfEvent (load)`, wywoływane jest w momencie ładowania klipu filmowego prostakat_slider. Kod wywoływany w odpowiedzi na zdarzenie składa się z jednej linijki: `usehandcursor = false`. Linijka ta odpowiedzialna jest za to aby kursor przyjmował kształt strzałki. Drugim zdarzeniem na jakie reaguje klip filmowy jest zdarzenie `onSelfEvent (press)`, wywoływane w momencie naciśnięcia przycisku myszy. Kod tego zdarzenia pokazany jest na listingu poniżej.

```
4.   onSelfEvent (press) {
5.       startDragUnlocked(_parent.pasek._x, _parent.pasek._x +
6.               _parent.pasek._width-1, this._y, this._y);
7.   }
```

Jak widać w odpowiedzi na to zdarzenie wywoływana jest funkcja `startDragUnlocked`. Zadaniem tej funkcji jest uruchomienie trybu „przeciągnij i upuść” dla klipu filmowego prostakat_slider. Funkcja ta posiada cztery argumenty określające w jakim zakresie współrzędnych możemy przeciągać dany obiekt. Pierwszy i drugi parametr określają odpowiednio lewy i prawy kraniec obszaru w jakim będziemy przesuwać prostokąt. Trzeci i czwarty argument opisują górny i dolny kraniec obszaru przeciągania. Przeciąganie odbywa się tak długo aż nie zostanie wywołana funkcja `stopDrag`, lub gdy inny obiekt animacji wejdzie w tryb „przeciągnij i upuść”. Metoda stopDrag, niezbędna do wyjścia z trybu przeciągania wywoływana jest w klipie prostakat_slider w odpowiedzi na zdarzenie `onSelfEvent (release,releaseOutside)`. Zdarzenie to jest wywoływane w momencie puszczenia lewego przycisku myszy w obszarze klipu lub poza nim. Kod tego zdarzenie został pokazany poniżej.

```
8.   onSelfEvent (release,releaseOutside) {
9.       stopDrag();
10.      _parent._parent.magnes.gotoAndPlay(3);
11.  }
```

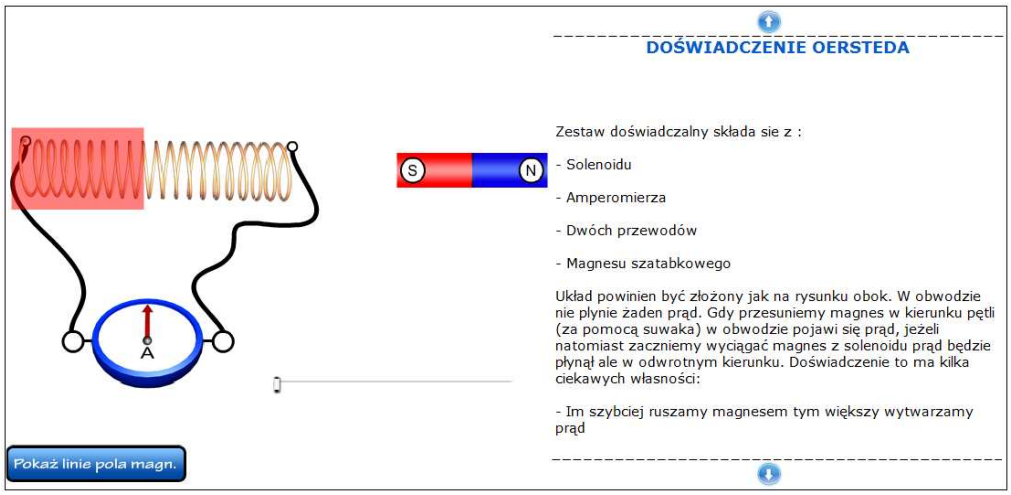
W linijce 9 wywoływana jest funkcja kończąca tryb przeciągania. W kolejnej linijce jest odtwarzana animacja poruszającego się magnesu. Magnes ten będzie się poruszał z prędkością określoną za pomocą prostokąta, którego przesuwaliśmy.

Za animację przesuwającego się magnesu odpowiedzialny jest skrypt umieszczony w klipie filmowym magnes. Skrypt ten jest dość złożony, dlatego zostanie przedstawiony i omówiony w kilku mniejszych częściach.

```
4.   onFrame (4) {
5.         if(this._x+magnes_wlasciwy._x+(magnes_wlasciwy._width/2)<=
6.         _parent.zderzak_koncowy._x){
7.           gotoAndPlay(7);
8.           if(_parent.biegun==0){
9.               _parent.strzalki_pradu.gotoAndPlay("gora_dol");
10.          }else{
11.              _parent.strzalki_pradu.gotoAndPlay("dol_gora");
12.          }
13.        }
14.      }else{
15.        if(this._x+magnes_wlasciwy._x<=_parent.zderzak_amperomierza._x)
16.        {
17.               if(_parent.biegun==0){
18.                    _parent.amperomierz.gotoAndPlay(3);
19.                    _parent.strzalki_pradu.gotoAndPlay("dol_gora");
20.               }else{
21.                   _parent.amperomierz.gotoAndPlay(11);
22.                   _parent.strzalki_pradu.gotoAndPlay("gora_dol");
23.               }
24.         }
25.         magnes_wlasciwy._x=magnes_wlasciwy._x-
26.              (_parent.predkosc_magnesu/10);
27.         linie_pola._x=linie_pola._x-(_parent.predkosc_magnesu/10);
28.      }
29.
30.  }
```

```
31. onFrame (5) {
32.     gotoAndPlay(4);
33. }
```

Zdarzenia w tym skrypcie tworzą swego rodzaju pętlę. Podczas odtwarzania animacji, w klatce numer cztery wykonywany jest kod skryptu. Po wykonaniu wszystkich instrukcji następuje przejście do klatki numer pięć. W tej klatce z kolei następuję przejście do klatki numer cztery i tak w kółko. W linijce sprawdzany jest warunek czy magnes nachodzi na przeźroczysty kwadrat o nazwie zderzak_koncowy. Kwadrat ten umieszczony jest na lewym końcu solenoidu (rys. 28.).

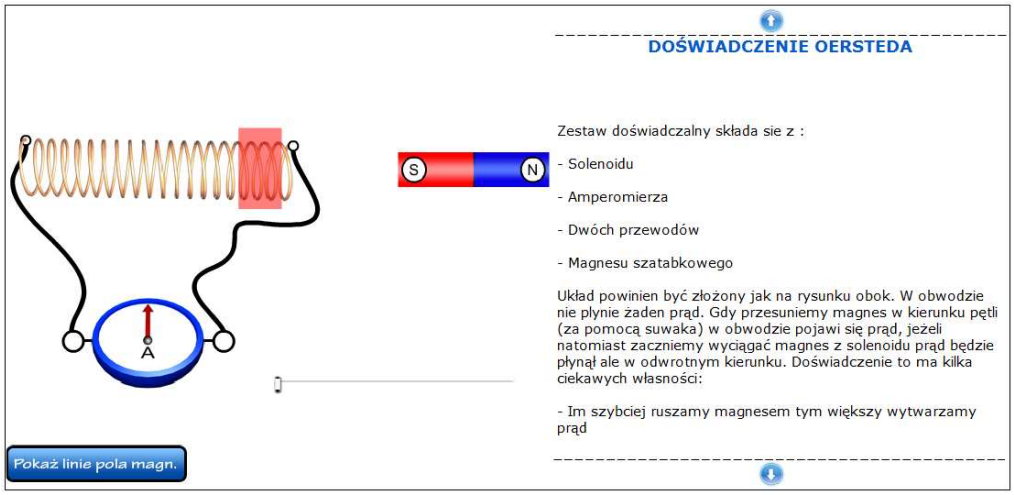


Rys. 28. Prostokąt zderzak_koncowy, którego normalnie nie widać, a na rysunku zaznaczony czerwonym przeźroczystym prostokątem

Prostokąt ten jest potrzebny aby wykryć moment, w którym magnes wszedł maksymalnie do solenoidu i za chwilę będzie wychodził. Należy wtedy odwrócić wskazówkę amperomierza do odwrotnego niż do tej pory wychylenia maksymalnego (`max_kat`) i uruchomić animacje obrotu do położenia, w którym będzie wskazywała do góry. W warunku w linijce 5 i 6 sprawdzane jest czy współrzędna x lewego końca magnesu (`this._x+magnes_wlasciwy._x+(magnes_wlasciwy._width/2)`) nachodzi na prawy kraniec przeźroczystego prostokąta (`_parent.zderzak_koncowy._x`). Jeżeli warunek jest spełniony to wychodzimy z pętli i odtwarzamy klatkę siódmą, po której znajduje się animacja obrotu wskazówki, o której opowiemy za chwilę. Oprócz uruchomienia animacji obrotu wskazówki musimy również pokazać odpowiednie wskaźniki przepływu prądu. Jeżeli zmienna `biegun` ma wartość 0 (linijka 8), czyli magnes jest zwrócony biegunem południowym w lewą stronę, to odtwarzamy w klipie filmowym strzalki_pradu klatkę o etykiecie `gora_dol`. Spowoduje to pokazanie dwóch strzałek przy lewym i prawym przewodzie amperomierza. Prawa strzałka będzie zwrócona do dołu natomiast lewa do góry (linijka 9). Jeżeli `biegun` ma wartość 1 (czyli magnes zwrócony jest biegunem północnym w lewą stronę) to zostaną pokazane strzałki zwrócone przeciwnie niż w poprzednim przypadku. Odtworzona zostanie klatka o etykiecie `dol_gora` w klipie filmowym strzalki_pradu (linijka 11). Jeżeli lewy koniec magnesu nie nachodzi na prawy kraniec kwadratu zderzak_koncowy, czyli niespełniony jest warunek w linijkach 5 i 6, to wykonywany jest kod pomiędzy linijkami 15 a 27.

W pierwszej linijce następuje sprawdzenie czy lewy koniec magnesu nachodzi na przeźroczysty prostokąt zderzak_amperomierza. Kształt ten jest umieszczony po lewej stronie prawego krańca solenoidu. Rysunek 29 dokładnie przedstawia ten prostokąt, oraz jego umiejscowienie w animacji. Warunek ten odwzorowuje sytuację, w której magnes dopiero co zostaje włożony do solenoidu. Wówczas musi zostać obrócona wskazówka z pozycji, w której

wskazuje ona w górę do pozycji maksymalnego wychylenia (`max_kat`) w lewą lub prawą stronę.



Rys. 29. Prostokąt zderzak_amperomierza, którego normalnie nie widać, a na rysunku zaznaczony czerwonym przeźroczystym prostokątem

Jeżeli warunek w linijce 15 jest spełniony sprawdzany jest kolejny, czy zmienna `biegun` ma wartość 0. Warunek ten sprawdza czy kierunek bieguna południowego magnesu jest zwrócony w lewą stronę. Jeżeli warunek jest spełniony to w klipie filmowym amperomierz przechodzimy do klatki numer trzy. Odtwarzana jest wówczas animacja obrotu wskazówki amperomierza z położenia początkowego, do położenia w którym odchyli się ona w prawą stronę o kąt zdefiniowany w zmiennej `max_kat`. Pokazane zostaną również strzałki wskazujące przepływ prądu. Lewa będzie wskazywać w dół natomiast prawa do góry. Jeżeli warunek w linijce 17 nie jest spełniony, czyli biegun północny zwrócony jest w lewą stronę, to odtwarzamy animację, która obraca wskazówkę amperomierza z położenia początkowego do położenia, w którym obróci się ona w lewą stronę o kąt zdefiniowany w zmiennej `max_kat`. Oprócz tego pokazane zostaną również strzałki symbolizujące kierunek przepływu prądu. Lewa strzałka zwrócona będzie ku górze natomiast prawa ku dołowi. W linijce 25 i 26 zmieniane jest położenie magnesu (który znajduje się w klipie magnes_wlasciwy), co w połączeniu ze zmianą klatek spowoduje animacje ruchu. W linijce 27 znajduje się kod, który przesuwa linie pola magnetycznego wraz z magnesem. Drugim fragmentem skryptu przypisanego do klipu magnes jest poniższy listing.

```
37. nFrame (9) {
38.     gotoAndPlay(8);
39. }
40. onFrame (8) {
41.
42.     if(_parent.biegun==0){
43.         _parent.amperomierz.aktualny_kat=
44.                 -_parent.amperomierz.aktualny_kat;
45.         _parent.amperomierz.gotoAndPlay(7);
46.     }
47.     else
48.     {
49.         _parent.amperomierz.aktualny_kat=
50.                 -_parent.amperomierz.aktualny_kat;
51.         _parent.amperomierz.gotoAndPlay(15);
52.     }
53.     if(this._x+magnes_wlasciwy._x>=_parent.zderzak_amperomierza._x){
54.         _parent.strzalki_pradu.gotoAndPlay(1);
55.     }
```

```
56.     if(this._x+magnes_wlasciwy._x>=this._x){
57.         gotoAndPlay(1);
58.     }
59.     else{
60.         magnes_wlasciwy._x=magnes_wlasciwy._x
61.             +(_parent.predkosc_magnesu/10);
62.         linie_pola._x=linie_pola._x+(_parent.predkosc_magnesu/10);
63.     }
64. }
```

W tym wypadku również została stworzona pętla. Odtwarzając klatkę numer osiem, wykonywany jest skrypt z linijek od 41 do 63. Następnie przechodzimy do klatki dziewiątej. Tam wracamy do klatki osiem itd. Najbardziej interesuje nas fragment kodu pomiędzy linijkami 41 a 63. Jest to kod uruchamiany w wyniku przetworzenia linijki 7 skryptu z poprzednio rozważanego fragmentu. Na początku sprawdzane jest ustawienie magnesu. Jeżeli zwrócony jest on biegunem południowym w lewą stronę, czyli zmienna `biegun` ma wartość 0 to zmienna `aktualny_kat` zdefiniowana w klipie filmowym amperomierz przyjmie swoją odwrotną wartość. Po uruchomieniu poprzednich skryptów zmienna ta ma wartość `max_kat`. Teraz będziemy animować wyciąganie magnesu z solenoidu, dlatego kąt ten należy odwrócić, gdyż prąd przy wyciąganiu magnesu zmienia swój kierunek na przeciwny. Należy również uruchomić animację powrotu wskazówki z kierunku w którym jest wychylona w lewo o kąt `max_kat`, do kierunku, w którym będzie wskazywała pionowo w górę. Jeżeli zmienna `biegun` będzie miała wartość 1 to zmienna `aktualny_kat` zmieni swoja aktualna wartość na przeciwną (linijka 49 i 50) oraz uruchomiona zostanie animacja (linijka 51) powrotu wskazówki amperomierza z kierunku, w którym jest ona wychylona w prawo o kat `max_kat` do kierunku, w którym będzie wskazywała pionowo w górę. Jeżeli magnes zostanie całkowicie wyciągnięty z solenoidu, prąd przestaje płynąć. Sytuacja ta została opisana za pomocą linijek 53, 54 i 55. W linijce 53 sprawdzamy czy współrzędna x lewego krańca magnesu jest większa od współrzędnej x prawego końca kwadratu zderzak_amperomierza. Jeżeli warunek jest spełniony, to wyłączane są strzałki wskazujące kierunek płynięcia prądu (linijka 54).

Ważne jest też aby magnes powrócił do początkowego położenia oraz aby podczas tego powrotu nie wyjechał poza obszar animacji. Kod odpowiedzialny za to zadanie został umieszczony w linijkach od 56 do 63. W linijce 56 sprawdzane jest aktualne położenie magnesu. Jeżeli jest ono większe lub równe położeniu początkowemu to animacja jest zatrzymywana i magnes powraca do położenia początkowego (linijka 57). Jeżeli warunek nie jest spełniony to do aktualnego położenia dodawana jest prędkość poruszania, co sprawia że magnes się porusza (linijki 60 i 61). Linijka 62 odpowiedzialna jest za przemieszczanie linii pola magnetycznego wraz z magnesem.

Ostatnim elementem implementacyjnym wartym omówienia jest klip filmowy amperomierz. Po załadowaniu klipu tworzona jest zmienna `aktualny_kat`, która zawiera wartość aktualnego wychylenia wskazówki amperomierza. Skrypt przypisany do tego klipu również zostanie omówiony w kilku fragmentach. Poniżej został przedstawiony pierwszy fragment.

```
7.  onFrame (4) {
8.      if(aktualny_kat<=_parent.max_kat){
9.          aktualny_kat+=((_parent.predkosc_magnesu)/50);
10.         wskazowka._rotation = aktualny_kat;
11.     }else{
12.         gotoAndPlay(1);
13.     }
14. }
15. onFrame (5) {
16.     gotoAndPlay(4);
17. }
```

W tym fragmencie również została zastosowana pętla, podobna do wcześniej rozważanych. Przejście do klatki cztery powoduje wykonanie zawartego w niej skryptu. Następnie przechodzimy do klatki pięć, gdzie wracamy z powrotem do klatki cztery. Linijki od 8 do 13 realizują przypadek, w którym magnes jest wsuwany do solenoidu biegunie południowym zwróconym w lewą stronę. Podczas wsuwania wskazówka amperomierza będzie się obracać z pozycji, w której wskazywała pionowo w górę do pozycji, w której będzie obrócona w prawą stronę o kąt zdefiniowany w zmiennej `max_kat`. Jeżeli zatem aktualny kat (reprezentowany przez zmienną `aktualny_kat`) jest mniejszy lub równy maksymalnemu dozwolonemu w danym momencie wychyleniu (linijka 8) to zwiększamy aktualny kąt o wartość prędkości podzieloną przez 50. Następnie obracamy wskazówkę o zaktualizowany kąt (linijka 10). Jeżeli natomiast aktualny kąt jest większy od maksymalnego dozwolonego w danym momencie kąta to wychodzimy z pętli, przechodząc do klatki numer jeden. Kolejnym scenariuszem jaki musimy rozważyć jest wkładanie magnesu do solenoidu biegunem północnym zwróconym w lewą stronę. Scenariusz ten realizowany jest za pomocą poniższego kodu.

```
41. onFrame (12) {
42.     if(aktualny_kat>=_parent.max_kat){
43.         aktualny_kat-=((_parent.predkosc_magnesu)/50);
44.         wskazowka._rotation = aktualny_kat;
45.     }else{
46.         gotoAndPlay(1);
47.     }
48. }
49. onFrame (13) {
50.     gotoAndPlay(12);
51. }
```

W tym wypadku musimy obrócić wskazówkę z położenia, w którym wskazuje do góry do położenia, w którym będzie wychylona w lewą stronę. Tym razem `max_kat` będzie ujemny. W linijce 42 sprawdzana jest wartość aktualnego wychylenia. Jeżeli jest ona większa od maksymalnego dozwolonego wychylenia to zmniejszamy wartość aktualnego wychylenia o prędkość podzieloną przez 50 i zaktualizowaną wartość wychylenia przypisujemy do kąta wychylenia wskazówki amperomierza. Jeżeli aktualny kąt okaże się mniejszy od maksymalnego dozwolonego wychylenia, czyli warunek w linijce 42 nie będzie spełniony, następuję wyjście z pętli poprzez instrukcje zawartą w linijce 46. Kolejnym rozważanym przypadkiem będzie wyciąganie magnesu z solenoidu gdy magnes ten będzie zwrócony biegunem południowym w lewą stronę. Kod obsługujący ten przypadek został przedstawiony na listingu poniżej.

```
21. onFrame (8) {
22.     if(aktualny_kat<0){
23.         aktualny_kat+=((_parent.predkosc_magnesu)/50);
24.         wskazowka._rotation = aktualny_kat;
25.     }else{
26.         gotoAndPlay(10);
27.     }
28. }
29. onFrame (9) {
30.     gotoAndPlay(8);
31. }
```

Tym razem pętla obejmuje klatki 8 i 9. Jeżeli wyciągamy magnes to na samym początku wskazówka amperomierza jest wychylona w lewą stronę o kąt równy minusowej wartości

max_kat. Musi powrócić do stanu początkowego, czyli musi wskazywać pionowo w górę. Sprawdzamy zatem czy aktualna wartość kąta wychylenia jest mniejsza od 0. Jeżeli ten warunek jest spełniony dodajemy do aktualnego wychylenia prędkość poruszania się magnesu podzieloną przez 50. Zaktualizowaną wartość tego kąta przypisujemy kątowi wychylenia wskazówki. Jeżeli okaże się że aktualna wartość kąta wychylenia jest większa od 0 to pętla jest przerywana za pomocą polecenia z linijki 26. Ostatnim już problemem jest obrót wskazówki podczas wyciągania magnesu z solenoidu, gdy magnes ten ma biegun północny zwrócony w lewą stronę. Problem ten rozwiązuje fragment skryptu, który przedstawiony jest poniżej.

```
52. onFrame (16) {
53.     if(aktualny_kat>0){
54.         aktualny_kat-=((_parent.predkosc_magnesu)/50);
55.         wskazowka._rotation = aktualny_kat;
56.     }else{
57.         gotoAndPlay(18);
58.     }
59. }
60. onFrame (17) {
61.     gotoAndPlay(16);
62. }
```

Jest podobny do poprzednich fragmentów. Również składa się z pętli. W tym przypadku wskazówka amperomierza na początku wychylona jest w prawo o kąt równy max_kat. Wskazówka musi powrócić do stanu początkowego, czyli jej kąt wychylenia ma wynosić 0. W linijce 53 sprawdzana jest wartość kąta aktualnego wychylenia. Jeżeli jest ona większa od 0 to od tego wychylenia odejmujemy prędkość przesuwania się magnesu podzieloną przez 50 (linijka 54) i tak zaktualizowana wartość przypisujemy do kąta wychylenia wskazówki amperomierza (linijka 55). Jeżeli warunek w linijce 53 nie jest spełniony to wychodzimy z pętli instrukcją gotoAndPlay(18).

6.5. Implementacja doświadczenia z ramką z prądem

Klip filmowy opis w tym aplecie mam identyczną budowę jak w poprzednio omawianych apletach. Różni się jedynie linijką, która jest odpowiedzialna za wczytywanie pliku tekstowego z opisem doświadczenia. Linijka ta wygląda następująco: opis.PoleTekstu.loadVariables("opisy/ramka_z_pradem.txt").

Układ warstw w klipie filmowym aplet wygląda tak jak na rysunku 30.

Rys. 30. Układ warstw w klipie filmowym aplet, w aplecie z doświadczeniem z ramką z prądem

Na samej górze znajduje się warstwa z przyciskiem do włączania lub wyłączania linii pola magnetycznego. Warstwa gorne_linie_pola zawiera klip filmowy z liniami pola magnetycznego biegnącymi od bieguna północnego magnesu do południowego. Klip nazywa się gorne_linie_pola magnetycznego, gdyż w celu nadania animacji efektu trójwymiarowości, znajdują się one nad wszelkimi innymi obiektami, przesłaniając je. Kolejne dwie warstwy zwierają tekst na biegunach magnesu. Kolejne cztery warstwy to białe koła, o czarnych obwódkach, znajdujące się na biegunach magnesu oraz przy uchwytach ramki z prądem. Klip filmowy przełącznik zawiera klip filmowy z przełącznikiem, którego zadaniem jest zamknięcie lub otwarcie obwodu elektrycznego.

Kolejna warstwa zawiera plik graficzny reprezentujący baterię. Klip filmowy ramka, znajdujący się o jedną warstwę niżej, jest animacją obracającej się o dziewięćdziesiąt stopni ramki. Ramka w całości została wymodelowana w programie do tworzenia grafiki trójwymiarowej. Nałożono na nią teksturę miedzi, oraz dodano efekt połysku i oświetlenie. Następnie na ramce dokonano animacji obrotu. Kolejnym etapem było wyeksportowanie poszczególnych klatek animacji do plików graficznych. Pliki te zostały z kolei zaimportowane do programu SWiSH i użyte w aplecie. Na kolejnej warstwie znajduje się obiekt przedstawiający przewód biegnący od dodatniego bieguna baterii do jednego z uchwytów ramki. Na warstwie dolne_linie_pola umieszczony jest klip filmowy z liniami pola magnetycznego, które znajdują się pod ramką z prądem i są przez nią przesłaniane. Dwie warstwy poniżej zawierają obiekty przedstawiające kolejno przewód łączący ujemy biegun baterii z przełącznikiem oraz przełącznik z uchwytem ramki. Kolejne warstwy zawierają grafiki przedstawiające bieguny magnesu.

Gdy klip filmowy aplet zostanie załadowany, tworzone są w nim wówczas dwie zmienne : `wlaczniki` – domyślnie ustawiona na 0 symbolizuje czy obwód jest otwarty (0) czy zamknięty (1) oraz zmienna `linie_pola` – ustawiona na jeden oznacza że użytkownik życzy sobie wyświetlać linie pola magnetycznego.

Klip filmowy but_linie_pola reaguje na dwa zdarzenia. Pierwsze zdarzenie, wywoływane w wyniku najechania na przycisk kursorem myszy (`on (rollOver)`) powoduje że wykonywany będzie poniższy kod.

```
14. on (rollOver) {
15.     but_linie_pola.gotoAndPlay(3);
16. }
```

Wykonanie tego kodu spowoduje że odtworzona zostanie klatka numer trzy w klipie filmowym but_linie_pola. Klatka ta zawiera animację, która powoduje zmianę stanu przycisku. Kolejne zdarzanie obsługiwane przez klip to puszczenie wciśniętego lewego przycisku myszy, w obszarze klipu (`on (release)`). Kod wykonywany w momencie zajścia tego zdarzenia przedstawiony jest poniżej.

```
1.  on (release) {
2.      if(linie_pola==0){
3.         gorne_linie_pola.gotoAndPlay("pokaz");
4.          dolne_linie_pola.gotoAndPlay("pokaz");
5.          but_linie_pola.gotoAndPlay(5);
6.          linie_pola=1;
7.      }else{
8.          gorne_linie_pola.gotoAndPlay("schowaj");
9.          dolne_linie_pola.gotoAndPlay("schowaj");
10.         but_linie_pola.gotoAndPlay(1);
11.         linie_pola=0;
12.     }
13. }
```

Na samy początku sprawdzamy czy zmienna `linie_pola` jest równa 0 (czyli czy włącznik jest wyłączony). Jeżeli tak, to włączamy górne linie pola magnetycznego (linijka 3), włączamy dolne linie pola magnetycznego (linijka 3), zmieniamy kolo przycisku but_linie_pola na zielony (linijka 5) aby zasygnalizować użytkownikowi że linie pola są włączone. Zmieniamy również wartość zmiennej na jeden co sygnalizuje reszcie skryptów, że użytkownik życzy sobie pokazywać linie pola magnetycznego. Jeżeli warunek w linijce 2 nie jest spełniony czyli linie pola są włączone, to wyłączamy górne i dolne linie pola magnetycznego (linijki 8 i 9), zmieniamy kolor przycisku na niebieski oraz ustawiamy zmienna `linie_pola` na 0.

Najważniejszym elementem całego apletu jest klip filmowy o nazwie przelacznik. Kod jaki jest wykonywany po jego kliknięciu wygląda następująco.

```
14.  on (release) {
15.      if(wlaczniki==0){
16.          przelacznik.gotoAndPlay("on");
17.         wlaczniki=1;
18.          ramka.gotoAndPlay("start");
19.      }else{
20.          przelacznik.gotoAndPlay("off");
21.          wlaczniki=0;
22.          ramka.gotoAndPlay(1);
23.
24.
25.      }
26.  }
```

Na samym początku skryptu sprawdzana jest wartość zmiennej `wlaczniki`. Jeżeli wynosi ona zero, czyli przełącznik jest wyłączono to odtwarzana jest klatka o etykiecie `on` w klipie filmowym przelacznik. Animacja ta zmieni kolor przełącznika na zielony co będzie sugerowało że jest on włączony. Następnie zmienną `wlaczniki` ustawiamy na jeden aby wszystkie skrypty wiedziały że przełącznik jest włączony. W linijce 17 uruchamiana jest animacja obrotu ramki o dziewięćdziesiąt stopni. Jeżeli zaś warunek w linijce 15 nie jest spełniony czyli przełącznik został już wcześniej włączony, to należy go wyłączyć (linijka 20). Wówczas przełącznik powrotem powróci do swojego domyślnego koloru – niebieskiego. Zmienna `wlaczniki` musi teraz przyjąć wartość 0, sugerującą że włącznik jest wyłączony. Ostatnim krokiem jest przywrócenie początkowego położenia ramki, realizowane jest ono w linijce 22.

6.6. Implementacja doświadczenia Ampere'a

Aplet z doświadczenie Ampere'a jest ostatnim apletem, którego implementacje będziemy omawiać. Jak poprzednie aplety, tak i ten zbudowany jest z dwóch głównych klipów o nazwach aplet i opis. Budowa klipu opis jest podobna do pozostałych, dlatego nie będzie omawiana szczegółowo. Linijka kodu, która ładuje plik z opisem doświadczenia przedstawia się następująco: `opis.PoleTekstu.loadVariables("opisy/amper.txt")`.

Ciekawiej pod względem implementacyjnym przedstawia się klip filmowy aplet. Układ warstw w tym klipie wygląda tak jak na rysunku 31. Pierwsza warstwa od góry zawiera klip filmowy z przyciskiem, który jest odpowiedzialny za włączanie lub wyłączanie linii pola magnetycznego wokół przewodników. Kolejne dwie warstwy zawierają klipy strzalki_pradu_prawa i strzalki_pradu_lewa, które zawierają strzałki wskazujące kierunek płynięcia prądu odpowiednio w prawym i lewym przewodniku. Kolejne cztery warstwy zawierają białe koła z czarnymi obwódkami, które są umieszczone na obu końcach

przewodników. Warstwy przelacznikl i przelacznikp są klipami filmowymi odwzorowującymi przełączniki zamykające odpowiednio lewy i prawy obwód elektryczny.

Rys. 31. Układ warstw w klipie filmowym aplet, w aplecie z doświadczeniem Ampere'a

Kolejna warstwa zawiera klip filmowy z wszelkimi animacjami związanymi z przewodnikami, w których będzie płynął prąd. Kolejne pięć warstw jest obiektami, które reprezentują kable łączące przewodniki z przełącznikami oraz z bateriami. Omawianie zagadnienie implementacyjnych zaczniemy od najprostszych klipów.

Jako pierwszy omówimy but_linie_pola. Z klipem tym związany jest skrypt, który wywoływany jest w odpowiedzi na zdarzenie kliknięcia myszą w obszarze przycisku. Skrypt ten przedstawiony jest poniżej.

```
16. on (release){
17.     if(czy_linie_pola == 0){
18.         czy_linie_pola=1;
19.         but_linie_pola.gotoAndPlay("pressed");
20.         if(wlacznikil==1 && baterial==0){
21.             przewodnik_lewy.linie_pola_bliz.gotoAndPlay("lewa");
22.             przewodnik_lewy.linie_pola_dal.gotoAndPlay("pokaz");
23.         }else if(wlacznikil==1 && baterial == 1){
24.             przewodnik_lewy.linie_pola_bliz.gotoAndPlay("prawa");
25.             przewodnik_lewy.linie_pola_dal.gotoAndPlay("pokaz");
26.         }
27.         if(wlacznikip==1 && bateriap==0){
28.             przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay("lewa");
29.             przewodnik_lewy.linie_pola_dal_p.gotoAndPlay("pokaz");
30.         }else if(wlacznikip == 1 && bateriap==1){
31.             przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay("prawa");
32.             przewodnik_lewy.linie_pola_dal_p.gotoAndPlay("pokaz");
33.         }
34.     }else if(czy_linie_pola == 1){
35.         czy_linie_pola=0;
36.         but_linie_pola.gotoAndPlay(1);
37.         przewodnik_lewy.linie_pola_bliz.gotoAndPlay(1);
38.         przewodnik_lewy.linie_pola_dal.gotoAndPlay(1);
39.         przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay(1);
40.         przewodnik_lewy.linie_pola_dal_p.gotoAndPlay(1);
41.     }
42. }
```

Na początku sprawdzana jest wartość zmiennej `czy_linie_pola`. Zmienna ta przyjmuje wartość jeden gdy linii pola magnetycznego są włączone, zero zaś gdy linie pola magnetycznego są wyłączone. Jeżeli wartość tej zmiennej wynosi zero (linijka 17) to znaczy że linie pola magnetycznego są wyłączone a użytkownik chce je włączyć. W linijce 18 zmieniamy wartość zmiennej `czy_linie_pola` na jeden dzięki czemu wszystkie skrypty będą wiedziały że linie pola, magnetycznego są włączone. Linie pola magnetycznego występują w przewodniku tylko gdy płynie w nim prąd, a ich kierunek uzależniony jest od kierunku płynięcia prądu. Kierunek płynięcia prądu jest zależny od ustawienia baterii. Ustawienie baterii wyznaczają dwie zmienne `baterial`, `bateriap`. Pierwsza zmienna określa położenie baterii dla lewego obwodu zaś druga dla prawego. Jeżeli bateria lewa lub prawa ustawiona jest biegunem dodatnim do góry to zmienne `baterial` i `bateriap` odpowiednio będą przyjmować wartości jeden. Jeżeli zaś bateria lewa lub prawa będzie ustawiona biegunem ujemny do góry to zmienne będą `baterial` i `bateriap` przyjmować wartość zero. Jeżeli w przewodniku płynie prąd to znaczy że obwód musi być zamknięty a żeby obwód był zamknięty to włącznik musi być włączony.

Z każdym włącznikiem, dla lewego i prawego obwodu, związana jest zmienna, która przechowuje jego stan. Jeżeli przełącznik lewy jest włączony, to zmienna `wlacznikil` przyjmuje wartość jeden, jeżeli zaś jest wyłączony to zmienna `wlacznikil` przyjmuje wartość zero. Podobnie jest z prawy przełącznikiem, jeżeli jest włączony to zmienna `wlacznikip` przyjmuje wartość jeden, jeżeli jest wyłączony zmienna `wlacznikip` przyjmuje wartość zero. Mając to na uwadze linie pola magnetycznego będziemy włączać jeżeli wartości zmiennych `wlacznikil` i `wlacznikip` będą ustawione na jeden, co sprawdzane jest we wszystkich warunkach w kodzie z linijek od 18 do 33. Dodatkowo sprawdzane są wartości zmiennych `baterial` i `bateriap`. Jeżeli bateria lewego obwodu znajduje się w pozycji, w której biegun ujemny zwrócony jest do góry (linijka 20) to linie pola wokół lewego przewodnika będą miały kierunek zgodny z ruchem wskazówek zegara (linijka 21 i 22). Jeżeli zaś bateria będzie miała biegun dodatni zwrócony do góry (linijka 23) to linie pola magnetycznego wokół lewego przewodnika będą miały kierunek przeciwny do ruchu wskazówek zegara. Podobne warunki będziemy sprawdzać dla prawego przewodnika, jeżeli bateria znajduje się w pozycji, w której biegun ujemny zwrócony jest do góry (linijka 27) to pokazane zostaną linie pola magnetycznego wokół prawego przewodnika ze strzałkami zwróconymi w kierunku zgodnym z ruchem wskazówek zegara (linijka 28 i 29). Jeżeli zaś bateria przy prawym przewodniku jest zwrócona biegunem dodatnim do góry (linijka 30) to pokazać należy linie pola magnetycznego wokół prawego przewodnika ze strzałkami zwróconymi przeciwnie do ruchu wskazówek zegara (linijka 31 i 32).

Jeżeli przełącznik but_linie_pola włączony a użytkownik wcisnął go drugi raz, to znaczy że życzy sobie wyłączyć linie pola magnetycznego. Sprawdzamy zatem czy zmienna ma wartość jeden (linijka 34). Jeżeli tak to zmieniamy wartość zmiennej na zero po to aby wszystkie skrypty wiedziały że teraz linie pola magnetycznego są wyłączone. Następnie przywracamy kolor przycisku na niebieski, aby użytkownik wiedział że teraz linie pola magnetycznego są wyłączone. Następnie odtwarzamy klatki, w których linie pola magnetycznego znikną. Dla lewego przewodnika animacje te są odtwarzane za pomocą skryptów w linijkach 37 i 38, zaś dla prawego przewodnika za pomocą poleceń w linijkach 39 i 40. Animacje linii pola magnetycznego są dla przewodnika lewego są zawarte w dwóch klipach linie_pola_bliz i linie_pola_dal, które znajdują się w klipie filmowym przewodnik_lewy. Natomiast dla prawego przewodnika te animacje znajdują się w klipach linie_pola_bliz_p i linie_pola_dal_p, które również znajdują się w klipie przewodnik_lewy. Każde z nich zostały umieszczone w dwóch klipach po to aby nadać efekt trójwymiarowości.

Klipy linie_pola_bliz i linie_pola_bliz_p przesłaniają odpowiednio lewy i prawy przewodnik, zaś pewne fragmenty klipów linie_pola_dal oraz linie_pola_dal_p są przesłaniane przez lewy o prawy przewodnik.

Kolejnym, wartym opisania jest przypadek implementacji przełączników, które zamykają i otwierają obwód elektryczny. Na początku zostanie opisany sposób implementacji lewego przełącznika, a następnie prawego. Lewy przełącznik znajduje się w klipie przelacznikl. Kod skryptu, wywoływanego w momencie wciśnięcia lewego przycisku myszy w obszarze przełącznik, został przedstawiony na poniższym listingu.

```
14.  on (release) {
15.      if(wlacznikil==0){
16.          przelacznikl.gotoAndPlay("on");
17.          if(baterial==1){
18.              strzalki_pradu_lewa.gotoAndPlay("dol");
19.              if(czy_linie_pola==1){
20.                   przewodnik_lewy.linie_pola_bliz.gotoAndPlay("prawa");
21.                  przewodnik_lewy.linie_pola_dal.gotoAndPlay("pokaz");
22.              }
23.          }else{
24.              strzalki_pradu_lewa.gotoAndPlay("gora");
25.              if(czy_linie_pola==1){
26.                   przewodnik_lewy.linie_pola_bliz.gotoAndPlay("lewa");
27.                  przewodnik_lewy.linie_pola_dal.gotoAndPlay("pokaz");
28.              }
29.          }
30.          if(wlacznikip==1){
31.              if(bateriap==baterial){
32.                  przewodnik_lewy.gotoAndPlay("wygiecie_prawo");
33.              }else if(bateriap!=baterial){
34.                  przewodnik_lewy.gotoAndPlay("wygiecie_lewo");
35.              }
36.          }
37.          wlacznikil=1;
38.      }else{
39.          przelacznikl.gotoAndPlay("off");
40.          przewodnik_lewy.linie_pola_bliz.gotoAndPlay(1);
41.          przewodnik_lewy.linie_pola_dal.gotoAndPlay(1);
42.          if(wlacznikip==1){
43.              if(bateriap==baterial){
44.                  przewodnik_lewy.gotoAndPlay("wygiecie_prawo_powrot");
45.              }else if(bateriap!=baterial){
46.                  przewodnik_lewy.gotoAndPlay("wygiecie_lewo_powrot");
47.              }
48.          }
49.          strzalki_pradu_lewa.gotoAndPlay(1);
50.          wlacznikil=0;
51.      }
52.  }
```

Obwód może zostać włączony wyłącznie wtedy gdy wcześniej był wyłączony. Dlatego na samym początku sprawdzane jest czy wartość zmiennej `wlacznikil` jest ustawiona na zero. Jeżeli tak jest to na początku odtwarzamy klatkę o etykiecie `on`, która znajduje się w aktualnie omawianym klipie. Odtworzenie tej klatki spowoduje że tło przełącznika zmieni się na kolor zielony co będzie sugerowało użytkownikowi apletu, że przełącznik jest włączony. Następnie sprawdzane jest położenie lewej baterii. Jeżeli biegun dodatni zwrócony jest do góry (warunek w linijce 17), to strzałka symbolizująca przepływ prądu w lewym przewodniku musi zostać pokazana z grotem zwróconym w dół. Wiemy że po włączeniu przepływu prądu wytwarza się pole magnetyczne. Użytkownik może chcieć (ale nie musi) wyświetlić linie tego pola magnetycznego. Dlatego dodatkowo w linijce 19 sprawdzana jest wartość zmiennej

czy_linie_pola i jeżeli wynosi ona jeden to dla lewego przewodnika pokazujemy linie pola magnetycznego ze strzałkami zwróconymi zgodnie z ruchem wskazówek zegara. Zadanie to realizują linijki 20 i 21, których zadaniem jest odtworzenie klatek o etykietach odpowiednio prawa i pokaz. Linijka 20 spowoduje że pokazane zostaną strzałki oraz fragmenty linii pola magnetycznego które są umieszczone „nad” lewym przewodnikiem, zaś linijka 21 spowoduje że będą pokazywane fragmenty linii pola magnetycznego, które są umieszczone „pod” lewym przewodnikiem. Jeżeli lewa bateria będzie zwrócona ujemny biegunem w górę to strzałka kierunku przepływy prądu w lewym przewodniku będzie zwrócona w górę (linijka 24) oraz zostaną pokazane linie pola magnetycznego wokół lewego przewodnika, zwrócone w kierunku przeciwnym do ruchu wskazówek zegara (linijka 26 i 27) jeżeli tylko zmienna czy_linie_pola będzie miała wartość jeden (linijka 25).

Na końcu rozważamy dodatkowo sytuację, w której prawy obwód już jest zamknięty i wtedy włączony zostaje lewy. W takiej sytuacji oba przewodniki powinny wygiąć się do środka (jeżeli obie baterie są jednocześnie zwrócone biegunem dodatnim w górę lub w dół) lub wygiąć się na zewnątrz (jeżeli obie baterie mają biegun dodatnie zwrócone w różne strony). Scenariusz ten realizowany jest za pomocą skryptu w linijkach od 30 do 35. W linijce sprawdzamy, analizując wartość zmiennej wlacznikip, czy włącznik prawy jest włączony. Jeżeli jest włączony (czyli zmienna wlacznikip ma wartość jeden), sprawdzamy czy elektrody dodatnie obu baterii są zwrócone w tę samą stronę, poprzez porównanie czy zmienne są sobie równe baterial i bateriap. Jeżeli warunek z linijki 31 jest spełniony to przechodzimy w klipie filmowym przewodnik_lewy do klatki o etykiecie wygiecie_prawo co spowoduje że linie reprezentujące przewodniki wygną się do środka. Jeżeli warunek równości zmiennych nie będzie spełniony to zostanie odtworzona klatka o etykiecie wygiecie_lewo, w której zostanie uruchomiona animacja wygięcia przewodników na zewnątrz. Na samym końcu zostanie przetworzona linijka 37, która sprawi że zmienna wlacznikil będzie miała wartość jeden dzięki czemu wszelkie skrypty będą posiadały informacje o stanie włącznika.

Na razie rozważyliśmy sytuację kiedy użytkownik włączył przełącznik, teraz przyszedł czas na zapoznanie się ze scenariuszem, w którym przełącznik ten zostaje wyłączony. Fragment skryptu realizujący ten przypadek został zawarty pomiędzy linijkami 39 i 50. Na samym początku uruchamiamy klatkę oznaczoną etykietą off w bieżącym klipie filmowym. Akcja ta spowoduje że przełącznik zmieni swój kolor z zielonego na niebieski. Kolejne dwie linijki są odpowiedzialne za wyłączenie linii pola magnetycznego wokół lewego przewodnika. W tym wypadku również musimy sprawdzić czy prawy obwód elektryczny nie jest zamknięty. Jeżeli jest zamknięty, czyli zmienna wlacznikip będzie miała wartość jeden (linijka 42) to należy dodatkowo sprawdzić czy obie baterie nie są zwrócone elektrodami dodatnimi lub ujemnymi w tę samą stronę. Jeśli warunek w linijce 43 będzie spełniony (czyli baterie są zwrócone tak samo) to należy uruchomić animację powrotu przewodników ze stanu, w którym są one wygięte do wewnątrz do stanu kiedy nie były wyprostowane. Realizowane jest to poprzez odtworzenie klatki o etykiecie wygiecie_prawo_powrot w klipie filmowym przewodnik_lewy (linijka 44). Kiedy równość zmiennych z warunku w linijce 43 nie jest spełniona uruchamiana jest animacja z klatki o etykiecie wygiecie_lewo_powrot, która spowoduje że przewodniki powrócą do stanu kiedy były wyprostowane ze stanu kiedy były wygięte do zewnątrz. Linijka 49 odpowiedzialna jest za wyłączenie strzałki pokazującej przepływ prądu w lewym przewodniku. Kolejna linijka aktualizuje wartość zmiennej na zero wlacznikil.

Prawy przełącznik, zaimplementowany w klipie filmowym przelacznikp, zbudowany jest w prawie identyczny sposób jak przełącznik lewy dlatego jego konstrukcja zostanie omówiona na zasadzie uwidocznienia różnić i podobieństw pomiędzy poprzednio omawianym klipem. Identycznie jak poprzednio, przełącznik prawy również reaguje na

zdarzenie naciśnięcia w jego obszarze lewym przyciskiem myszy. Pierwsze różnice można dostrzec już w kilku początkowych linijkach skryptu, który jest przyporządkowany do klipu filmowego przelacznikp. Skrypt ten został przedstawiony na poniższym listingu.

```
14.  on (release) {
15.      if(wlacznikip==0){
16.          przelacznikp.gotoAndPlay("on");
17.          wlacznikip=1;
18.          if(bateriap==1){
19.              strzalki_pradu_prawa.gotoAndPlay("dol");
20.              if(czy_linie_pola==1){
21.                   przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay("prawa");
22.                  przewodnik_lewy.linie_pola_dal_p.gotoAndPlay("pokaz");
23.              }
24.          }else{
25.              strzalki_pradu_prawa.gotoAndPlay("gora");
26.              if(czy_linie_pola==1){
27.                   przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay("lewa");
28.                  przewodnik_lewy.linie_pola_dal_p.gotoAndPlay("pokaz");
29.              }
30.          }
31.          if(wlacznikil==1){
32.              if(bateriap==baterial){
33.                  przewodnik_lewy.gotoAndPlay("wygiecie_prawo");
34.              }else if(bateriap!=baterial){
35.                  przewodnik_lewy.gotoAndPlay("wygiecie_lewo");
36.              }
37.          }
38.      }else{
39.          przelacznikp.gotoAndPlay("off");
40.          przewodnik_lewy.linie_pola_bliz_p.gotoAndPlay(1);
41.          przewodnik_lewy.linie_pola_dal_p.gotoAndPlay(1);
42.          if(wlacznikil==1){
43.              if(bateriap==baterial){
44.                  przewodnik_lewy.gotoAndPlay("wygiecie_prawo_powrot");
45.              }else if(bateriap!=baterial){
46.                  przewodnik_lewy.gotoAndPlay("wygiecie_lewo_powrot");
47.              }
48.          }
49.          strzalki_pradu_prawa.gotoAndPlay(1);
50.          wlacznikip=0;
51.      }
52.  }
```

W tym wypadku sprawdzamy stan przełącznika prawego, logiczne jest wiec, że będziemy sprawdzać wartość zmiennej `wlacznikip`. W kolejnej linijce modyfikujemy stan przełącznika prawego, dlatego uruchamiamy klatkę o etykiecie `on` w klipie filmowym przelcznikp a nie jak poprzednio przelacznikl. Podobnie jak w przypadku lewego przełącznika, tak i teraz sprawdzamy położenie baterii badając wartość zmiennej reprezentującej prawą baterię (`bateriap`). Jeżeli wynosi ona jeden w klipie filmowym strzalki_pradu_prawa uruchamiamy animację pokazującą strzałkę, wskazująca przepływ prądu w prawym przewodniku, która jest skierowana w dół. Linie pola które umieszczone są w klipach linie_pola_bliz_p oraz linie_pola_dal_p, będą miały kierunek zgodny z ruchem wskazówek zegara. Jeżeli wartość zmiennej wynosi zero strzałka wskazująca kierunek płynięcia prądu będzie zwrócona ku górze (linijka 25) zaś linie pola magnetycznego będą skierowane przeciwnie do ruchu wskazówek zegara (linie 27 i 28). Kod zawarty w linijkach od 31 do 36 ma to samo znaczenie co poprzednio różni się jedynie tym że sprawdzana jest wartość zmiennej `wlacznikil`, gdyż sprawdzamy stan przełącznika w obwodzie przeciwnym do aktualnie badanego, czyli lewym.

Gdy wartość zmiennej `wlacznikip` wynosi jeden wykonywany jest kod zawarty pomiędzy linijkami od 39 do 50. Na początku, tak jak w klipie filmowym przelacznikl, aktualizujemy kolor przełącznika prawego na niebieski. W dwóch następnych linijkach wyłączamy linie pola magnetycznego wokół prawego przewodnika. Kod pomiędzy linijkami 42 a 48 ma to samo znacznie jak w przypadku przełącznika lewego. Sprawdzamy tylko stan innego obiektu, mianowicie przełącznika lewego, za pomocą zmiennej `wlacznikil`. Na samym końcu aktualizujemy wyłączamy strzałki prądu, wskazujące kierunek jego przepływu w prawym przewodniku oraz aktualizujemy wartość zmiennej `wlacznikip` na aktualny stan prawego przełącznika.

W klipie filmowym przewodnik lewy znajdują się dwa klipy, których implementacja jest dość ciekawa. Klipy nazwane zostały bateria_lewa i bateria_prawa. Reprezentują one baterie dla lewego i prawego obwodu. Na początku rozważmy kod źródłowy skryptu, który będzie wykonywany w momencie naciśnięcia lewego przycisku myszy na baterii w pobliżu lewego przewodnika. Kod ten został przedstawiony poniżej.

```
14. on (release) {
15.    if (_parent.baterial==1) {
16.         bateria_lewa.gotoAndPlay("minus");
17.          _parent.baterial=0;
18.          if(_parent.wlacznikil==1){
19.              _parent.strzalki_pradu_lewa.gotoAndPlay("gora");
20.              if(_parent.czy_linie_pola==1){
21.                  linie_pola_bliz.gotoAndPlay("lewa");
22.                 linie_pola_dal.gotoAndPlay("pokaz");
23.             }
24.          }
25.         if(_parent.wlacznikil==1 && _parent.wlacznikip==1){
26.             if(_parent.baterial==_parent.bateriap){
27.                 this.gotoAndPlay("zlaczenie");
28.             }else{
29.                 this.gotoAndPlay("rozlaczenie");
30.             }
31.         }
32.     }else{
33.         bateria_lewa.gotoAndPlay("plus");
34.        _parent.baterial=1;
35.          if(_parent.wlacznikil==1){
36.              _parent.strzalki_pradu_lewa.gotoAndPlay("dol");
37.              if(_parent.czy_linie_pola==1){
38.                  linie_pola_bliz.gotoAndPlay("prawa");
39.                 linie_pola_dal.gotoAndPlay("pokaz");
40.             }
41.          }
42.        if(_parent.wlacznikil==1 && _parent.wlacznikip==1){
43.             if(_parent.baterial==_parent.bateriap){
44.                 this.gotoAndPlay("zlaczenie");
45.             }else{
46.                 this.gotoAndPlay("rozlaczenie");
47.             }
48.         }
49.     }
50. }
```

Pierwszy scenariusz jaki może się odegrać po kliknięciu myszą na baterię wygląda następująco. Jeżeli bateria ustawiona jest elektrodą dodatnia do góry to zostanie odwrócona i elektroda ujemna będzie zwrócona do góry. Dodatkowo jeżeli obwód w lewym przewodniku jest zamknięty to kierunek strzałki wskazującej przepływ prądu zostanie zamieniony z dolnego na górny a jeżeli użytkownik włączył linie pola magnetycznego to ich kierunek

zostanie zamieniony na przeciwny do ruchu wskazówek zegara. Dodatkowo jeżeli prawy obwód jest włączony i lewy również to musimy wygiąć przewodniki na zewnątrz jeżeli ustawienia obu baterii będą różne lub do wewnątrz jeżeli położenia baterii będą takie same. Przypadek ten realizowany jest przez kod znajdujący się pomiędzy linijkami 15 a 31. Na samym początku sprawdzany jest warunek czy zmienna `baterial` ma wartość jeden, czyli czy bateria znajduje się w położeniu biegunem dodatnim zwróconym w górę. Jeżeli warunek jest spełniony odtwarzamy w klipie bateria_lewa klatkę o etykiecie `minus`. Spowoduje to że grafika reprezentująca baterię zostanie obrócona o 180 stopni. Kod w linijce 17 spowoduje że zmienna `baterial` będzie miała teraz wartość zero dzięki czemu wszystkie skrypty będą posiadały aktualną informację o położeniu baterii. Linijka 18 zawiera warunek sprawdzający na podstawie wartości zmiennej, czy lewy przełącznik jest włączony. Jeżeli warunek ten jest spełniony, to należy na początku odwrócić kierunek strzałki wskazującej prąd płynący w lewym przewodniku. Akcja ta jest wykonywana poprzez odtworzenie w klipie filmowym sztralki_pradu_lewa klatki o etykiecie `gora` (linijka 19). Po tym zabiegu strzałka będzie zwrócona do góry.

Następnie sprawdzamy, czy użytkownik włączył opcję wyświetlania linii pola magnetycznego. Jeżeli to zrobił to zmienna `czy_linie_pola` musi przyjmować wartość jeden, odtwarzamy wówczas w klipie filmowym linie_pola_bliz klatkę o etykiecie `lewa`, oraz klatkę o etykiecie `pokaz` w klipie linie_pola_dal, które to spowodują że strzałki na liniach pola magnetycznego zmienią swój kierunek na odwrotny do ruchy wskazówek zegara. Linijka 25 zawiera kod przyrównujący zmienne `wlacznikil` i `wlacznikip` do jedynki. Jeżeli warunek ten jest spełniony to znaczy że oba przełączniki są włączone. Należy wówczas sprawdzić równość zmiennych `baterial` i `bateriap` względem siebie. Jeżeli równość ta jest spełniona, to znaczy że baterie mają elektrody zwrócone w tym samym kierunku, musimy wtedy odtworzyć animację która wygnie przełączniki do wewnątrz. Animacja ta zaczyna się w klatce oznaczonej etykietą `zlaczenie`, która znajduje się w klipie filmowym przewodnik_lewy a linijka ją uruchamiająca oznaczona jest numerem 27. Jeżeli warunek z linijki 26 nie jest spełniony, czyli baterie są w różnych pozycjach, to uruchamiana jest animacja wyginająca przewodniki na zewnątrz. Rozpoczyna się ona w klatce o etykiecie `rozlaczenie` w klipie filmowym przewodnik_lewy. Linijkę, która jest odpowiedzialna za to zdarzenie oznaczono numerkiem 29.

Drugi przypadek, zawarty w linijkach od 32 do 50 omawianego skryptu, odtwarzany jest wówczas gdy bateria ma elektrodę dodatnią zwróconą ku dołowi. Należy wówczas odtworzyć animację, zaczynającą się klatce o etykiecie `plus` w klipie filmowym bateria_lewa, która spowoduje obrót baterii o 180 stopni (linijka 33). Musimy również zaktualizować zmienną aby inne skrypty posiadały informację o aktualnym położeniu baterii (linijka 34). Następnie sprawdzamy czy lewy przełącznik zamyka obwód elektryczny. Sprawdzamy zatem wartość zmiennej `wlacznikil`, jeżeli wynosi ona jeden (czyli przełącznik jest włączony a wiec obwód jest zamknięty), to zmieniamy położenie strzałki, pokazującej przepływ prądu w lewym przewodniku na takie, w którym będzie wskazywała w dół. Animacja zamiany kierunku wskazywania strzałki uruchamiana jest w linijce 36, poprzez przejście do klatki oznaczonej etykieta `dol` w klipie filmowym o nazwie strzałki_pradu_lewa. Dalsze trzy linijki mają to samo znaczenie jak w poprzednio omawianym scenariuszu, jedyną różnicą jest klatka uruchamiana w celu zmienny kierunku linii pola magnetycznego. Odtwarzana jest klatka o etykiecie `prawa` gdyż linie pola tym razem muszą być zwrócone zgodnie z ruchem wskazówek zegara. Linijki od 42 do 49 są identyczne jak w poprzednio omawianym przydatku, identyczne jest również ich znaczenie. Odpowiedzialne są za odtworzenie animacji wygięcia przewodników do środka lub na zewnątrz w zależności od ustawienia obu baterii.

Oprócz lewej baterii istnieje również prawa, dla prawego obwodu. Znajduje się ona w klipie filmowym o nazwie bateria_prawa. Poprzednio klip filmowy z lewą baterią reagował

na przyciśnięcie lewego przycisku myszy. Podobnie jest również w klipie filmowym, który odpowiada na zdarzenie `on (release)`. Wówczas wywoływany jest kod prawie identyczny jak ten wykonywany w klipie bateria_lewa. Różnice występują jedynie w kilku linijkach i sprowadzają się jedynie do odtwarzania klatek w innych klipach filmowych. Mając to na uwadze przedstawimy jedynie fragmenty które się różnią. Pierwszy fragment to ten przedstawiony poniżej.

```
14.  on (release) {
15.     if (_parent.bateriap==1) {
16.          bateria_prawa.gotoAndPlay("minus");
17.           _parent.bateriap=0;
18.           if(_parent.wlacznikip==1){
19.               _parent.strzalki_pradu_prawa.gotoAndPlay("gora");
20.               if(_parent.czy_linie_pola==1){
21.                   linie_pola_bliz_p.gotoAndPlay("lewa");
22.                  linie_pola_dal_p.gotoAndPlay("pokaz");
23.              }
24.           }
```

Kod z powyższego listingu działa podobnie i ma to samo znaczenie, jak fragment skryptu pomiędzy linijkami 14 i 24 z poprzedniego kodu źródłowego. Tym razem sprawdzamy wartość zmiennej `bateriap` a nie `baterial`. Kolejna różnica polega na tym że tym razem aktualizujemy baterię prawą, dlatego operujemy na klipie filmowym bateria_prawa i na zmiennej `bateriap`. Dodatkowo w linijce 18 sprawdzany jest stan przełącznika prawego zamiast lewego. Uruchamiamy, w odróżnieniu od poprzednio rozważanej baterii, animację zmiany kierunku strzałki wokół prawego przewodnika oraz animacje zmiany kierunku linii pola magnetycznego wokół prawego przewodnika. Kolejny fragment skryptu, w którym występują różnice w porównaniu ze skryptem omawianym przy opisywaniu implementacji klipu bateria_lewa, został przedstawiony poniżej.

```
34.          bateria_prawa.gotoAndPlay("plus");
35.          _parent.bateriap=1;
36.          if(_parent.wlacznikip==1){
37.               _parent.strzalki_pradu_prawa.gotoAndPlay("dol");
38.               if(_parent.czy_linie_pola==1){
39.                   linie_pola_bliz_p.gotoAndPlay("prawa");
40.                  linie_pola_dal_p.gotoAndPlay("pokaz");
41.              }
42.           }
```

W linijce 34 będziemy tym razem działać na klipie filmowym bateria_prawa zamiast na klipie bateria_lewa. Podobną regułę można zaobserwować w kolejnej linijce gdzie nie będziemy działać jak poprzednio na zmiennej `baterial` ale na zmiennej `bateriap`. Również ewentualnym zmianom ulegają klipy filmowe ze strzałką wskazującą przepływ prądu w prawym obwodzie oraz klipy reprezentujące linie pola magnetycznego wokół prawego przewodnika. Poprzednio zmiany te dotyczyły lewego obwodu i przewodnika.

7. Podsumowanie

Tematem pracy było stworzenie apletów zawierających doświadczenia z zakresu elektromagnetyzmu. Aplety te miały w prosty sposób tłumaczyć zjawiska fizyczne towarzyszące tym doświadczeniom. Staraliśmy się, aby były możliwe jak najbardziej interaktywne po to żeby zachęcić użytkownika do czynnego uczestnictwa w doświadczeniach. Takie podejście powoduje że użytkownik będzie zainteresowany danym zagadnieniem i z większą chęcią będzie chciał je zgłębić. Staraliśmy się również, stworzyć animacje, które będą przejrzyste i efektowne.

Grafika została zaprojektowana z dbałością o szczegóły. Nadano jej również efekt trój wymiarowości aby użytkownik miał lepszą orientację jak poszczególne obiekty rozmieszczone są w przestrzeni. Opisy doświadczeń zostały skonstruowane w taki sposób że umożliwiają samodzielne stworzenie prezentowanych doświadczeń w warunkach domowych. Podczas tworzenia interfejsu kierowaliśmy się tym aby był on jak najbardziej intuicyjny i przejrzysty. Każdy przycisk posiada nazwę, która odwzorowuje jego przeznaczenie. Również grafika na niektórych przyciskach została dobrana w taki sposób, aby jednoznacznie identyfikować jego funkcję.

Zaopatrując aplety w pomoc kierowaliśmy się dobrem osoby, która będzie wykorzystywała aplety. Dzięki temu rozwiązaniu użytkownik w razie jakichkolwiek wątpliwości będzie mógł zasięgnąć wiedzy, która umożliwi mu wybrnięcie z problemu związanego z obsługa animacji. Staraliśmy się również zapewnić możliwość modyfikacji apletów poprzez dynamicznie wczytywaną zawartość. Wszystkie aplety są wczytywane w głównym interfejsie z zewnętrznych plików co umożliwi w przyszłości dodawanie nowych doświadczeń. Takie podejście zapewnia możliwość niezależnego osadzenia poszczególnych animacji na stronie WWW.

Opisy są również wczytywane z zewnętrznych plików, co umożliwia ich łatwą modyfikację i uzupełnianie. Teksty opisów są formatowane są za pomocą pliku z arkuszami styli CSS dzięki czemu z łatwością zapanujemy nad sposobem ich wyświetlania. Najważniejszym czynnikiem jednak było stworzenie animacji, które będą zgodne z obowiązującymi prawami fizyki.

Niniejsza praca nie zawiera opisu implementacji głównego interfejsu, gdyż jest on tylko dodatkiem do całości i powodowałby niepotrzebne zagmatwanie pracy. Na płycie CD znajdują się pliki źródłowe, które w razie potrzeby można podejrzeć w celu zapoznania się z implementacją interfejsu. Pracę inżynierską można w przyszłości zmodyfikować poprzez dodanie do niej nowych animacji z doświadczeniami. Łatwo można również przetłumaczyć aplety na inne języki dzięki czemu staną się bardzo ciekawym i użytecznym środkiem dydaktycznym.

8. Literatura

[1] http://pl.wikipedia.org/wiki/Charles_Coulomb - Strona Wikipedi o Charlesie Coulombie
[2] http://pl.wikipedia.org/wiki/Waga_skr%C4%99ce%C5%84 – Strona Wikipedi o Wadze Skręceń
[3] D. Halliday, R. Resnick, J. Walker, Podstawy Fizyki 3, Wydawnictwo Naukowe PWN, Warszawa, 2003
[4] http://ptsg.eecs.berkeley.edu/~helens/edu/sol_page/solenoid.html - strona stworzona na potrzeby edukacji młodzieży autorstwa dr Helen Smith
[5] D. Halliday, R. Resnick, Fizyka 2, Wydawnictwo Naukowe PWN, Warszawa, 2001
[6] http://www.tutorvista.com/content/physics/physics-iv/electromagnetic-induction/experiment-faraday-and-henry.php
[7] P. Hammond, Od Gilberta do Einsteina. Krótka historia elektromagnetyzmu, Wydawnictwo Politechniki Częstochowskiej, Częstochowa, 1999.